Universidade de Évora - Instituto de Investigação e Formação Avançada

Programa de Doutoramento em Informática

Tese de Doutoramento

Promoting Understandability in Consumer Health Information Seach

Hua Yang

Orientador(es) | Teresa Gonçalves

Évora 2019

Universidade de Évora - Instituto de Investigação e Formação Avançada

Programa de Doutoramento em Informática

Tese de Doutoramento

Promoting Understandability in Consumer Health Information Seach

Hua Yang

Orientador(es) | Teresa Gonçalves

Évora 2019

A tese de doutoramento foi objeto de apreciação e discussão pública pelo seguinte júri nomeado pelo Diretor da Instituto de Investigação e Formação Avançada:

- Presidente | Paulo Quaresma (Universidade de Évora)
- Vogal | Fernando Manuel Marques Batista (ISCTE - Instituto Universitário de Lisboa)
- Vogal | Carla Alexandra Teixeira Lopes (Universidade do Porto - Faculdade de Engenharia)
- Vogal | Paulo Quaresma (Universidade de Évora)
- Vogal | Mário Jorge Costa Gaspar da Silva (Instituto Superior Técnico)
- Vogal-orientador | Teresa Gonçalves (Universidade de Évora)

Évora 2019

To my parents.

Acknowledgements

My deep gratitude goes first to my supervisor Teresa Gonçalves, for her guidance, enthusiasm, encouragement and patience. In the past few years, she has supported me in every way that she could to facilitate my Ph.D study; she believes in my research ability to achieve something great; she has given me the space to pursue my own research ideas; she has offered me her time and friendship generously whenever I needed them. She is not only my supervisor, but also friend.

I would like to give my thanks to Professor Paulo Quaresma, Salvodor Abreu, Irene Rodrigues, and Luís Rato, who advised me a lot in the early stages of my Ph.D. study. And my particular thanks go to Professor Paulo Quaresma who contributed greatly to the clarification of my research ideas and provided many valuable suggestions. My thanks also goes to Professor Pedro Salgueiro and Vasco Pedro who sought every opportunity to allocate clusters resources for me. Without their help, I was not able to carry out my experiments. My thanks also goes to Professor José Saias who gave help in allocating resources to facilitate my study. I would like to thank all the teachers working in the libraries of Colégio do Espírito Santo, University of Évora. My special thanks are to teacher Catarina Costa, her kindness and warmness make me feel at home.

I would like to thank my peers and colleagues at VISTA lab in the Department of Informatics. Their nice comments, wonderful suggestions and valuable tips filled in the gaps of my knowledge. Of my colleagues, my particular thanks go to Roy Bayot and Prakash Poudyal, who gave me great advice at the beginning of my Ph.D. study. My thanks also go to Madhu Agrawal, who made my life as a Ph.D. student enjoyable. I am also thankful to all the friends I've made in Évora.

Finally, I would not have made it through without the deep love and support from my whole family. Without the accompany of my husband and my son, I could not continue and finish my Ph.D. study. To my dear mum and dad, they have offered me great encouragements and these keeps me going on the way.

Contents

List of Figures

List of Tables

Acronyms

CHIR	Consumer Health Information Retrieval
CHIS	Consumer Health Information Search
CHV	Consumer Health Vocabulary
CUI	Concept Unique Identifier
CLI	ColemanLiau index
CBOW	Continuous bag-of-words model
cTAKES	clinical Text Analysis and Knowledge Extraction System
CLEF	Conference and Labs of the Evaluation Forum
FIRE	Forum for Information Retrieval Evaluation
GFI	Gunning fog index
HIR	Health Information Retrieval
HIS	Health Information Search
IR	Information Retrieval
LETOR	Learning to Rank
LSA	Latent Semantic analysis
LT	LETOR, Learning to Rank
MCM	Medical Concept Model
MeSH	Medical Subject Headings
MCI	Medical Concepts Identification
MAP	Mean Average Precision
NDCG	Normalized Discounted Cumulative Gain
NIH	National Institute of Health
NLP	Natural Language Processing
PRF	Pseudo Relevance Feedback
P@n	Precision at position n

QE	Query Expansion
QI	Query Independent
QD	Query Dependent
RBP	Rank Biased Precision
SGM	Continuous Skip-gram Model
SMOG	Simple Measure of Gobbledygook
TREC	Text REtrieval Conference
UMLS	Unified Medical Language System
uRBP	understandability Rank Biased Precision
uRBPgr	understandability Rank Biased Precision in graded gain
WE	Word Embedding

Abstract

Nowadays, in the area of Consumer Health Information Retrieval, techniques and methodologies are still far from being effective in answering complex health queries. One main challenge comes from the varying and limited medical knowledge background of consumers; the existing language gap between non-expert consumers and the complex medical resources confuses them. So, returning not only topically relevant but also understandable health information to the user is a significant and practical challenge in this area.

In this work, the main research goal is to study ways to promote understandability in Consumer Health Information Retrieval. To help reaching this goal, two research questions are issued: (i) how to bridge the existing language gap; (ii) how to return more understandable documents. Two modules are designed, each answering one research question. In the first module, a Medical Concept Model is proposed for use in health query processing; this model integrates Natural Language Processing techniques into state-of-the-art Information Retrieval. Moreover, aiming to integrate syntactic and semantic information, word embedding models are explored as query expansion resources. The second module is designed to learn understandability from past data; a two-stage learning to rank model is proposed with rank aggregation methods applied on single field-based ranking models.

These proposed modules are assessed on FIRE'2016 CHIS track data and CLEF'2016-2018 eHealth IR data collections. Extensive experimental comparisons with the state-of-the-art baselines on the considered data collections confirmed the effectiveness of the proposed approaches: regarding understandability relevance, the improvement is 11.5%, 9.3% and 16.3% in RBP, uRBP and uRBPgr evaluation metrics, respectively; in what concerns to topical relevance, the improvement is 7.8%, 16.4% and 7.6% in P@10, NDCG@10 and MAP evaluation metrics, respectively.

Keywords: Information Retrieval, Health, Consumer, Understandability, Query expansion, Learning to Rank.

Sumário

Promoção da Compreensibilidade na Pesquisa de Informação de Saúde pelo Consumidor

Atualmente as técnicas e metodologias utilizadas na área da Recuperação de Informação em Saúde estão ainda longe de serem efetivas na resposta às interrogações colocadas pelo consumidor. Um dos principais desafios é o variado e limitado conhecimento médico dos consumidores; a lacuna linguística entre os consumidores e os complexos recursos médicos confundem os consumidores não especializados. Assim, a disponibilização, não apenas de informação de saúde relevante, mas também compreensível, é um desafio significativo e prático nesta área.

Neste trabalho, o objetivo é estudar formas de promover a compreensibilidade na Recuperação de Informação em Saúde. Para tal, são são levantadas duas questões de investigação: (i) como diminuir as diferenças de linguagem existente entre consumidores e recursos médicos; (ii) como recuperar textos mais compreensíveis. São propostos dois módulos, cada um para responder a uma das questões. No primeiro módulo é proposto um Modelo de Conceitos Médicos para inclusão no processo da consulta de informação que integra técnicas de Processamento de Linguagem Natural na Recuperação de Informação. Mais ainda, com o objetivo de incorporar informação sintática e semântica, são também explorados modelos de *word embedding* na expansão de consultas. O segundo módulo é desenhado para aprender a compreensibilidade a partir de informação do passado; é proposto um modelo de *learning to rank* de duas etapas, com métodos de agregação aplicados sobre os modelos de ordenação criados com informação de campos específicos dos documentos.

Os módulos propostos são avaliados nas coleções CHIS do FIRE'2016 e eHealth do CLEF'2016-2018. Comparações experimentais extensivas realizadas com modelos atuais (*baselines*) confirmam a eficácia das abordagens propostas: relativamente à relevância da compreensibilidade, obtiveram-se melhorias de 11.5%, 9.3% e 16.3 % nas medidas de avaliação RBP, uRBP e uRBPgr, respectivamente; no que respeita à relevância dos tópicos recuperados, obtiveram-se melhorias de 7.8%, 16.4% e 7.6% nas medidas de avaliação P@10, NDCG@10 e MAP, respectivamente.

Palavras chave: Recuperação de Informação, Saúde, Consumidor, Compreensibilidade, Expansão de Interrogações, *Learning to Rank*.

Chapter 1

Introduction

Generally speaking, in the Computer Science research domain, Information Retrieval (IR) refers to the methodologies and technologies that seek for relevant information from a data collection regarding a user's information need.

Health Information Search (HIS) is a domain specific IR activity concerning the health area, which is usually known as Health Information Retrieval (HIR). Health information is of interest to different kinds of users. According to their varying medical knowledge background, these users can be categorized into two types (Goeuriot et al., 2016):

- Non-expert users: laypeople without strong medical knowledge background including patients and their families.
- Expert users: medical professionals including clinicians, physicians, medical examiners, general practitioners and expert practitioners such as surgeons and radiologists.

Consumer Health Information Retrieval (CHIR) is one specific research area in HIR which aims to search health information specifically for non-expert users.

The *Health Online 2013*, by *Pew Internet Project*, shows that 73% of US people use Internet, and 71% of them use Internet to search health information (Fox and Duggan, 2013). Consumers commonly use the World Wide Web as a source for health information, with general search engines being popularly employed for this goal. Some dedicated services are also available to meet consumers' need such as the *Health on the Net*[1] system.

[1] Available at https://www.hon.ch/en/.

Despite the popularity of consumer health search in daily activity and its topic interest in the IR research community, the development of search technologies remains challenging in the area of CHIR (Goeuriot et al., 2016). For example, access mechanisms for factual health information search have developed greatly and it is easy to get an answer to "what is gout?" or "what are the symptoms of gout?".

Nevertheless, for complex health searches which do not have a single definitive answer like "does daily aspirin therapy prevent heart attack?", it still remains indefinable. For this kind of searches, not a single answer, but answers from different viewpoints should be presented, since a common user should get a balanced view of the different perspectives. Moreover, non-expert consumers have difficulty in understanding the answers to a complex query and the methodologies and techniques applied are still far from being effective in addressing such queries (Goeuriot et al., 2016; Yang and Gonçalves, 2017).

1.1 Motivation

Currently, the main measure considered when assessing an IR system is topicality relevance or, in other words, to what extent the searched information is topically relevant to a user's need. Typically, in modern IR systems, topically relevant contents are retrieved and ranked after issuing a query. Nonetheless, if the user that issued the query thinks the retrieved document is difficult to comprehend, even if this document is highly relevant, he tends to give up and move on to another one (Yilmaz et al., 2014).

Theoretically, a user regards the returned information as relevant if the information can, to some extend, satisfy the user's need. Relevance is a multiple-dimension concept: numerous factors may affect the user's judgment in his decision and the criteria can be a complex one (Cuadra and Katter, 1967; Saracevic, 1996, 2016). Topical relevance is only one of these factors which is typically and historically taken as the key measure in modern IR systems. Beyond topicality, the relevance of information is affected by other factors such as understandability, reliability, novelty and scope, for example. From those, understandability is an essential and significant factor which can be defined as "*...the extent to which the content of a retrieved document is perceived by the user as easy to read and understand.*" (Xu and Chen, 2006).

The goal of a written text is to serve as a communication line between the writer and the reader. What if the readability is beyond the reader's understandability? A hard to read text for a user means no joy and can't reach the goal of communication. In the area of CHIR, although all consumers are categorized as non-expert users, the individual's medical knowledge or, in other words, the individual's understandability regarding the same information, can differ greatly. Consumers with the same information need may have different choices when reading the retrieved content (Yang and Gonçalves, 2017).

Since consumers vary in their medical knowledge, their comprehension of the retrieved contents differ: an easy to read document for one consumer can be hard for another one. More specifically, in health related area, readability of written texts regarding appointments, medication and medication doses is very important for a reader; poor understandability of these texts is associated with poor health outcomes and may include increased mortality (Oliffe et al., 2017). Summarizing, a topically relevant document may help nothing to a consumer if the document is beyond his understandability level. Zuccon (2016) enumerated these ideas as:

1. A document is of no use to a user if it cannot be understood by a consumer, even if a document is topically relevant.
2. In a specific IR domain such as consumer health search, understandability is a main factor when assessing relevance beyond topic.

Similarly, Yang and Gonçalves (2017) elaborated that understandability of health documents means the relevant contents should be both comprehensive and useful to users. They discussed that topically relevant information may or may not be valuable to users and explained understandability within two dimensions: (i) **comprehension**, meaning that the relevant content is comprehended, and (ii) **usefulness**, meaning that the relevant content is useful.

In order to increase the access and utility of health-related information to general public, some organizations recommended a specific readability level for health information. The United Kingdom's Patient Information Forum (PIF) recommends that the readability of patient information material could be no higher than grade eight equivalent (Narwani et al., 2016) and the United States National Institute of Health (NIH) recommends that print materials for the public should use plain language with a target readability equivalent to the sixth grade level and no greater than eighth grade (Eltorai et al., 2014).

However, it is found that the public has health literacy skills lower than an eighth grade or equivalent (Kutner et al., 2006) and, on the other hand, the on-line medical websites provide information far from this level. A study that examined what 70 websites returned when performing the health query "congestive heart failure" on a popular search engine concluded that only 7.1% of the documents were at the recommended sixth-grade reading level when using one assessment tool and no website was at or under the sixth-grade reading level when using all five assessment tools (Kher et al., 2017). Another work found that no article abstract met the NIH readability target of sixth grade or below and only one was below the recommended ceiling of eighth grade equivalent (Hollada et al., 2017).

Given that the understandability of a medical document depends on both the readability of the document and the medical knowledge background of the consumer, we can not ask a user to comprehend all the relevant content, but we could try to provide understandable content according to the user's knowledge background. People with certain medical knowledge may prefer to read technical or professional content retrieved from professional websites or journal articles and people with limited medical knowledge may enjoy reading more popular content coming from blogs and forums. To the best of our knowledge, hardly no search system takes this into account; the ranking results are the same for the same information need independent of the user background. This means that for a user, the needed information can appear at the first one or two pages, while for other the needed content may be ranked at the tenth page.

This important need constitutes the motivation of this work. Returning not only **topically relevant but also understandable** health information to an individual is, in fact, a **significant and practical** challenge in the area of CHIR.

1.2 Research Goals

As retrieving understandable information beyond topically relevant one is a challenging task, the research goal of this work is **to promote understandability in Consumer Health Information Retrieval**. This goal could be further detailed as improving state-of-the-art methodologies and techniques used in IR, HIR and CHIR to retrieve both topically relevant and understandable health information to non-expert consumers. In order to achieve it, two factors of relevance will be taken into account: one is the traditional and classic topicality; another, and new one, is understandability.

To achieve this main goal, the research work is divided into two phases:

1. The first phase aims at **bridging the medical language gap** of the common user;
2. Based on the improved expressions of original queries (obtained from the first phase), the second phase aims at **learning understandability** from experience.

Next we'll detail each phase.

1.2.1 Bridging the Language Gap

One challenging issue related to the retrieval of relevant and understandable information comes from the varied medical background knowledge of consumers. Related research work has shown that the language gap between the consumers and the complex medical resources confuses non-experts (Kher et al., 2017; Hollada et al., 2017). If we want an IR system to retrieve understandable health documents to a consumer (or, from the user point of view, the consumer is able to interpret the retrieved health documents) what are the essential determining factors? Two obvious and significant ones are:

- The language vocabulary used in a document should be within the knowledge of the consumer.
- Certain medical knowledge background is needed.

Defining book reading levels and recommending corresponding books is an available amenity in education. Following this idea, one straightforward solution to solve this language gap issue could be to measure the reading level of a user and define the reading level of the documents; then, corresponding reading level documents are recommended to the user. However, defining the reading level of an on-line document and assessing the reading level of a consumer is not feasible nowadays, because: (i) the consumer personal information is usually private and is not available; (ii) defining the reading level of large medical documents, especially the on-line ones, is not practical with existing methodologies and techniques.

Another possible solution is finding a way to bridge the language gap between users and experts. So, the first research question for this work can be stated as:

> *How can the consumer's information need be expressed in more professional words and terms?*

Finding a way to solve this question will bridge the language gap between non-expert consumers and medical professionals.

1.2.2 Learning Understandability from Experience

Learning through experience is an important way for human learning. Although being a young research community, CHIR has grown fast and is attracting more focus. This increased interest generated an increasing number of available useful and valuable data either from related scientific research work but also from task campaigns. This leads to the second research question, that can be stated as:

Can we learn users' medical knowledge from past data?

Finding a way to solve this question will enable the promotion of more understandable documents and improve retrieval performance.

1.3 Proposed Approach

As presented in the previous sub-section, this research aims at promoting understandability in CHIR. To reach it, two sub-problems were defined, and related to each one, two research questions were raised. The proposed approach to answer each question follows.

How to bridge the language gap? The solution requires that the characteristics of the medical language should be fully taken into account when using an universal IR search engine to process health or medical queries. Since in the area of health information retrieval there's a language gap between non-expert consumers and professionals, one feasible approach to bridge this gap is to add new words to the queries proposed by consumers. These new words can be synonyms or related words to the original query terms, but will have the characteristic of being more professional and typically used by medical professionals. Also, it is natural that different terms and phrases contribute differently to a query and, as such, they will be processed differently and assigned different weights in query processing.

How to learn understandability from experience? Past data assessed by CHIR experts are valuable resources that can be used to promote understandability. Using the improved query expressions obtained from the first sub-task and past data, models for promoting understandability during retrieval will be trained; these models that will rank higher more understandable documents can then be applied to new data. Morevover, exploring potential features and their combination is a necessary and significant step for learning the best models.

1.4 Main Contributions

The main contribution in this research work can be stated as:

- Propose a Medical Concept Model (MCM). Typically, general-purpose query expansion techniques are used for processing health queries. However, these techniques don't fully take into account the specificity of the medical language and, as such, health queries won't be well processed. This issue is studied in this research and the MCM model is proposed to solve the problem. This model improves the state-of-art query expansion techniques for HIR and CHIR. Being useful to health queries, it can also be applied to other related tasks.

- Construct *loose phrases*. Considering the lack of medical knowledge, consumers may use only some words of a medical phrase. Based on modern IR weighing techniques, using exact matching will not enable the retrieval of some useful information, so loose phrases, aiming at building flexible expressions, are introduced and constructed. This processing is useful in bridging the language gap between consumers and medical experts and it is general enough to be applied to other IR research with similar language gap issues.

- Group explored features. An important step in any traditional machine learning application is feature engineering that aims at exploring the most useful features for the problem and using them to characterize the examples. Adding these new features to the original list leads to more time consuming learning phases take. In this work, rather than blindly mixing all the potential features into one feature list, features are grouped on the fields they derive from and a set of models are trained with each group. This approach can be easily generalized to other works, using different groupings besides the field-based one used in this work.

- Propose a two-stage LETOR model. Rank aggregation has shown to be effective in combining results obtained from different rankers. In this work, this method is combined with a Learning to Rank (LETOR) approach and a two-stage LETOR model is proposed: during the first stage, a set of LETOR models are learned each emphasizing in information taken from one specific field; then an aggregated model is constructed applying rank aggregation over the learned models. This two-stage model is not only useful in CHIR research, but can be easily generalized to other IR research areas.

1.5 Dissertation Overview

This dissertation can be divided in five parts. The first part is the introduction. The second part, that includes Chapter 2 and Chapter 3, introduces the background information, the state-of-the-art and related work in IR and HIR (including CHIR); then, the third part (Chapter 4) introduces the proposals for reaching the thesis goal and solve the research questions and the fourth part describes the experiments carried out and results obtained (Chapter 5 and Chapter 6). Finally, the fifth part presents the conclusions and the future work (Chapter 7). The chapter breakdown is as follows:

- In Chapter 2, the general idea of IR and the essential elements of an IR system are presented and then the understandability relevance is discussed. Next, state-of-the-art IR techniques are discussed, including IR weighting models, query expansion approach, learning to rank approach, rank aggregation methods and evaluation measures of an IR system. Then, significant resources and tools used in IR are introduced. Finally, related and significant literature is reviewed.

- Chapter 3 specifically reviews the literature in the area of HIR and CHIR. First, medical resources used in HIR and CHIR are discussed thoroughly; then, state-of-the-art techniques specially applied in these two areas are discussed, including query expansion using medical thesaurus and word embedding model. Finally, related literature in HIR and CHIR are reviewed.

- Chapter 4 presents the proposal. First, the system framework is presented; then, the following two sections present the proposal to answer the two research questions presented in Chapter 1, respectively. Referring to the first question, "How to bridge the language gap between a consumer and an expert?", a Medical Concept Model is proposed to further process queries and word embedding models are technically used as the query expansion resources. To answer the second question, "How to learn understandability from experience?", a two-stage learning to rank model is proposed. Methods and techniques used for each research question are thoroughly discussed.

- Chapter 5 and Chapter 6 present the experiments designed and carried out. In Chapter 5, the experimental setup for all the experiments is presented, including data collections, evaluation measures, experimental platform and general parameter settings. Then, the proposed models, methods and techniques for answering the first research question are tested on the corresponding data collections, the results are evaluated with the corresponding metrics, the observations are discussed and the conclusions are made based on the results obtained. Chapter 6 presents the experiments designed to assess the methods for answering the second research question, evaluates the results, discusses observations and elaborates conclusions.

- Chapter 7 concludes the dissertation, presenting conclusions along with the contributions of this research work and the discussion of future work.

Chapter 2

Information Retrieval

Information retrieval is the science of searching for information in the format of texts, images or sounds. Text information retrieval is one basic and important research area and there have been quite an abundant number of studies on it. This chapter reviews the state-of-the-art in text IR, including the basic and important IR concepts, state-of-the-art techniques, useful resources and related work.

2.1 A Classic Information Retrieval System

In the area of computer science, text information retrieval[1] can be defined as: "*...finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers).*" (Christopher et al., 2008).

Figure 2.1 basically presents how an IR system works. Briefly, the process of an IR activity can be explained as: first, an information need proposed by a user is issued to an IR system and the documents collection is provided; second, this IR system uses weighting algorithms to score the documents in the data collection according to its relevance to the query. The retrieved documents are typically ranked and returned to the user as a ranked list where the more relevant documents are ranker higher.

Next, we discuss the essential elements that an classic IR system is composed of. These elements and their connections are presented in Figure 2.2:

[1] We scope our discussion to text IR; and for easy to use, we omit *text* and use IR only in the latter parts.

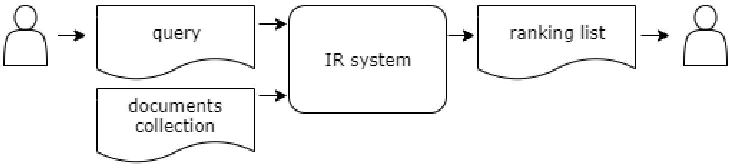


Figure 2.1: The basic flow of an IR system.

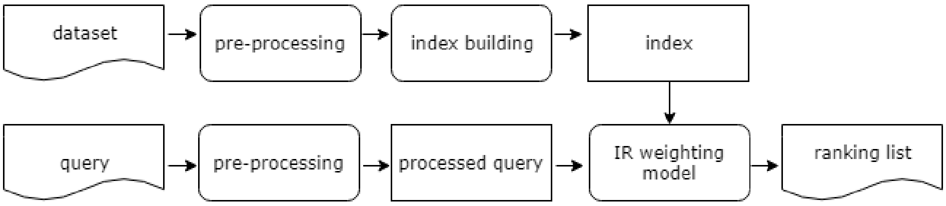


Figure 2.2: The essential elements of a classic IR system.

Input of an IR system. Typically, for an IR system, the input includes a query and a dataset. The query is the human user information need and is usually expressed in simple and short text. The dataset will be the source of information and usually contains a huge amount of documents.

Pre-processing. Usually, the documents included in the dataset and the original queries proposed by the users are pre-processed with various kinds of techniques. Two typical pre-processing approaches are tokenization and normalization (Christopher et al., 2008):

- Tokenization. The process of chopping a character sequence into pieces which are referred as tokens. Stop words can be identified and removed during this process.
- Normalization. The linguistic process of the tokens so that matches occur despite superficial differences in the character sequences of the tokens. Typical processing such as lower-casing characters and stemming words take place in this process.

Index Building. In modern IR, typically an inverted index is built over the dataset and serve as input to the IR system. An inverted index is an index data structure which consists of a list of all the unique words that appear in any document in the data collection; moreover, for each word, its location in each document or the frequency of appearance can also be stored (Christopher et al., 2008).

For example, as shown in Table 2.1, term *t1* appears in document *d1* at position 11 and 101, in *d2* at 1 and 22, and in *d3* at 7 and 37.

Table 2.1: An example of how an inverted index is structured.

Terms	**Doc1**	**Doc2**	**Doc3**	...
t1	11,101	1,22	7,37	...
t2	35,76	-	-	...
t3	-	2,200	10,217	...
...	...	...	...	...

When building an index, different indexing algorithms can be used, depending on the size of the data collection, if the collection is static or dynamic, direct and inverted index, hardware constraints and other factors.

Retrieval Model. A retrieval model takes the built index and the processed query as the input. It employs matching algorithms to measure how much a document is relevant to the given query: usually, the documents with higher scores are the ones more relevant to the query.

An IR system is often able to implement many different kinds of retrieval models. Each retrieval model adopts its own algorithm in calculating the similarity between a document and a query.

Output of an IR system. The output is a ranking list which contains a group of documents retrieved from the dataset and ranked according to the score each document achieves. It can be deemed that a retrieval process finishes at this stage.

Qrels file. Qrels file is usually a text formatted file and contains the relevance judgments of the query-document pairs. This file is used to evaluate the effectiveness of an IR system.

Evaluation of an IR system. To evaluate the effectiveness of an IR system in a standard way, these two items are used to perform the testing: (i) the input of an IR system which includes a data corpus and a group of queries; (ii) the output of an IR system which is the returned ranking list; (iii) qrels file. Different kinds of evaluation metrics can be adopted to measure the effectiveness of an IR system.

2.2 Relevance Judgments

An IR system can be evaluated in different dimensions, such as topicality, utility, usefulness, user satisfaction, understandability or reliability, etc. (Mizzaro, 1998). Usually, an traditional IR systems mainly takes into account the topically relevant. And in turn, relevant research work are mainly concentrating on improving an IR system performance with retrieving more relevant documents. Compared with improving topical relevance, understandability in IR and especially in CHIR attracts little attention.

2.2.1 Topical Relevance

Topicality relevance can be described as "... how well the topic of the information retrieved matches the topic of the request." (Harter, 1992).

Nowadays, the dominant evaluation model in IR evaluation area is a Cranfield paradigm. One of the major assumptions of the Cranfied paradigm is: the relevance can be approximated by topical similarity. This assumption assumes the relevance is only depending on topicality similarity between the queries proposed by users and the retrieved documents; and not taking into account the user proficiency (Voorhees, 2001).

Typically, the topicality relevance between the documents and the proposed queries are assessed by human experts or assessors. In the area of IR, relevance is equal to topicality relevance if not specified in literature.

2.2.2 Understandability Relevance

Given a document, its readability is related with the easiness that readers can understand the document. The Oxford dictionary explains readability as the quality of being easy or enjoyable to read.

Assessing the reading level of a written text is a prior and important work to provide or recommend a user texts equivalent to its reading level. Providing texts equivalent to a users' reading level can make the users enjoy the reading and the communication goal can be achieved in the end.

Computational readability assessment uses computer techniques to automatically assess the reading level of a given text, which is useful in various applications such as readability for second-Language learners, international language support, supporting readers with disabilities, computer-assisted educational learning systems, readability prediction for the Web, and so on (Collins-Thompson, 2014).

The popular used computational readability assessments include: Flesch reading ease, Flesch-Kincaid grade level, Gunning Fog index, SMOG and Coleman-Liau index (Kher et al., 2017; Badarudeen and Sabharwal, 2010). These measures are based on surface characteristics of a text such as sentence length and words length of syllables.

Flesch reading ease. The Flesch reading ease readability is used to indicate how difficult a passage in English is to understand: higher score indicates the document is easier to read; lower score is more difficult to read. The Flesch reading ease readability scores can be calculated with:

$$206.835 \text{ - } 1.015 \times \frac{\text{total words}}{\text{total sentence}} \text{ - } 84.6 \times \frac{\text{total syllables}}{\text{total words}}$$

Flesch-Kincaid grade level. Different from Flesch reading ease, Flesch-Kincaid grade level presents a score as a U.S. grade level. It is calculated with the following formula:

$$0.39 \times \frac{\text{total words}}{\text{total sentence}} + 11.8 \times \frac{\text{total syllables}}{\text{total words}} \text{ - } 15.59$$

Gunning fog index. Gunning fog index (GFI) be used to estimate the education year a person needs to understand the text. For example, an universal understanding needs an gunning fog index less than 8 (equal to 8th grade education level). The complete calculation is:

$$0.4 \times (\frac{\text{words}}{\text{sentences}} + 100 \times \frac{\text{complex words}}{\text{total words}})$$

SMOG. The SMOG (Simple Measure of Gobbledygook) measure is widely used to estimate the years of education needed to understand a piece of writing and particularly for checking health information (Hedman, 2008). SMOG score can be calculated with:

$$1.0430 \times \sqrt{\text{number of polysyllables} \times \frac{30}{\text{number of sentences}}} + 3.1291$$

Fitzsimmons *et al.* (Fitzsimmons et al., 2010) recommended that SMOG should be the preferred measure of readability when evaluating consumer-orientated health care material.

ColemanLiau index. ColemanLiau index (CLI) outputs approximates the U.S. grade level that is necessary to comprehend a document. It is calculated as:

$$\text{CLI} = 0.0588 \times L \text{ - } 0.296 \times S \text{ - } 15.8$$

where L is the average number of letters per 100 words and S is the average number of sentences per 100 words.

2.3 Techniques

To improve the retrieval performance of an IR system, various kinds of techniques have been researched and applied over a classic IR system.

Among these techniques, retrieval model, which is an essential and core part of an IR system, has always been an research focus in IR area; many different kinds of retrieval models have been proposed and applied. Besides retrieval model, another two important state-of-the-art techniques are query expansion and learning to rank. These state-of-the-art techniques have been widely used and shown their effectiveness in improving the performance in the area of information retrieval.

Next, we discuss these three kinds of state-of-the-art techniques applied in IR area in details.

2.3.1 Retrieval Models

Retrieval models are essential in IR and designed to realize these two functions (Christopher et al., 2008):

- Representation. A retrieval model defines how to represent a query and a document.
- Scoring. A retrieval models defines how to score or rank a document when matching a document to a query.

Classically, retrieval models are classified into four categories: Boolean Model, Vector Space Model, Probabilistic Model an Statistical Language Model. Besides these four classic ones, Field-based retrieval Model which uses field information of a document is also widely used nowadays. In Table 2.2, we list some example models for each category.

Table 2.2: Retrieval models.

Retrieval model	**Examples**
Boolean Model	Boolean model
Vector Space Model	TFIDF
Probabilistic Model	BM25, Binary Independence Model
Statistical Language Model	LM_Dirichlet, LM_Hiemstra
Field Based Retrieval Model	BM25F, PL2F

Among these different kinds of IR models, Boolean Model is usually used in a specialized area: the users are expert who have precise understanding of their needs. This model is not suitable for common users who normally are not capable of writing Boolean queries.

The other four kinds of models are widely and classically used in modern IR area. Next. we discuss these four models in details in the following sections.

Vector Space Model

Vector Space Model is a classic IR model which represents the contents of a query or a document in a vector space. Vector Space Model has proved to be robust and able to achieve good experimental results in practice.

TFIDF is the most popularly used Vector Space Model. Now we look at the calculation of TFIDF and we discuss how a TFIDF model is used to perform retrieval in an IR system.

TFIDF is the abbreviation for Term Frequency and Inverse Document Frequency.

Term frequency *tf* is the number of times a term t occurs in a document d and often denoted as $\mathrm{tf}_{\mathrm{t,d}}$. The value of *tf* can simply take use of the raw frequency of a term or other complicated calculations such as Boolean frequencies, logarithmic scaled frequency, or augmented frequency (Christopher et al., 2008).

Inverse document frequency *idf* is a factor to specify whether a term is common or rare across all documents, which aids to adjust if a term appears too frequently in a data corpus. *Idf* decreases the weight of terms that occur very frequently in the data corpus D while increases the weight of terms that occur rarely. Thus, the specialty of a term can be calculated as an inverse function of the number of documents in which the term occurs and often denoted as $\mathrm{idf}_{\mathrm{t,D}}$. Simply, the value of *idf* is calculated as: dividing the total number of documents by the number of documents containing this term;

then taking the logarithm of this division. The formula can be expressed as:

$$\mathrm{idf_{t,D}} = \log \frac{\mathrm{N}}{\mathrm{df}}$$

where N is the total number of documents in a data corpus D, and $\mathrm{df_t}$ is the number of documents containing term t.

TFIDF model takes into account the number of times a term appears in one document; and at the same time is offset by the frequency of the term in the data corpus. Basically, with TFIDF model, the weight of a term $\mathrm{w_{t,d}}$ is calculated using the combined scores of *tf* and *idf*:

$$\mathrm{w_{t,d}} = \mathrm{tf_{t,d}} \times \mathrm{idf_{t,D}}$$

As we can see from the formula, a term which has high term frequency in a document and low frequency in the whole data corpus can obtain a high weight. For example, if a term appears in too many documents of a data corpus, the ratio of the logarithm approaches 1, then *idf* is close to 0, and so $\mathrm{w_{t,d}}$ is closer to 0.

Here we only present the basic calculation of $\mathrm{w_{t,d}}$, TFIDF family contains a number of algorithms and weighting schemes which differ in their term weighting method and similarity measure.

Now we look at TFIDF retrieval model in IR. First, all documents in a data corpus and all queries are represented as vectors in a vector space. And typically, the vectors are generated like this: the value of each element inside these vectors represents the normalized term weight, which is the $\mathrm{w_{t,d}}$ values of this term.

Next, given a query, the score of a document *score(q,d)* is measured by the similarity between a query vector $\vec{v}(q)$ and the document vector $\vec{v}(d)$. A simple measure is to calculate the cosine similarity between the query vector and a document vector:

$$\mathrm{score(q,d)} = \vec{v}(q) \cdot \vec{v}(d)$$

Then, this resulting *score(q,d)* is used to rank a document for a query (Christopher et al., 2008).

Although simple, TFIDF has been an effective retrieval model in IR and plays an important role nowadays (Christopher et al., 2008).

Probabilistic Model

Probabilistic Model takes different principles from Boolean Model or Vector Space Model: if a document is relevant to a query, the similarity among the document and the query is computed as a probability. This model is widely used in Web search nowadays and a typical representative is BM25.

BM25 is based on probabilistic theory and probability ranking principle: if the retrieved documents are ranked decreasingly on their probability of relevance to a query, then the effectiveness of the system will be the best that is obtainable on the basis of those data (Manning et al., 2009).

Compared to TFIDF model, which mainly takes into account term frequency and inverse document frequency, BM25 model includes more factors in its weighting scheme, such as: document length, query length, term frequency in a query, term frequency in a document, relevance feedback and inverse document frequency (Christopher et al., 2008). BM25 also uses tuning parameters set up to ideally optimize performance on a development test collection. Several tuning parameters are available for this IR model. One parameter k_1 is used for adjusting the document term frequency scaling. Term frequency has a light weight if k_1 takes a larger value. Another parameter b is used for determining the normalization by document length, where b=1 means fully normalizing the term weight by the document length, and b=0 means no length normalization. Parameter k_3 is for scaling of the query. The tuning parameters should ideally be set to optimize performance on a development test collection. Experimental common values set k_1 and k_3 to 1.2 and b to 0.75 respectively.

BM25 retrieval model has been successfully used in IR area, shown its effectiveness in many search tasks like TREC[2], and been used to build baseline system by many groups as well (Voorhees et al., 2005).

Statistical Language Models

Different from TFIDF which is based on a Vector Space Model and similar to the Probabilistic Model, Statistical Language Models is based on probabilistic theory.

Statistical Language Model assumes that a document is relevant to a query if the document model is able likely to generate the query. When using a language model to retrieve relevant documents to a query, each document is regarded as a language model and the query as the output generated from the model. The retrieved documents are ranked based on the generation

[2] The Text REtrieval Conference https://trec.nist.gov/.

probabilities of the sample that produce the query. Higher probability means document is more relevant to the query (Manning et al., 1999).

An important issue concerning language models is smoothing, which is used for compensating data sparseness (Zhai and Lafferty, 2004). If some word included in the query does not appear in any documents, then a zero probability will be given. Smoothing is used to solve the problem and give some probability to words not appearing in the documents. From another point of view, we can say that smoothing adjusts the words weighting (Manning et al., 1999). Two common used methods are interpolation-based and Bayesian updating process. And a group of smoothing ways are available, for example, Dirichlet smoothing and Hiemstra smoothing are two widely used ones (D., 2009). Zhai and Lafferty (2017, 2004) studied the problem of language model smoothing in the context of information retrieval; they examined and evaluated several popular interpolation-based smoothing methods on several TREC task collections. Detailed empirical comparisons of different smoothing methods were presented in their work. Usually, language models differ from each based on the different smoothing methods adopted.

Statistical language models have been successfully used in IR area and have a long history. Ponte and Croft (1998) first experimented Language Modeling to information retrieval; Song and Croft (1999) proposed a general language model for information retrieval which was based on a range of data smoothing techniques and proved the effectiveness of these methods. Based on the statistics of 12 years' TREC experiments (Voorhees et al., 2005), language modelling techniques were usually found to be popular and effective in IR circles as they provide a theoretical justification for the weights assigned to terms in the weighting schemes.

Field Based Retrieval Models

Some new retrieval models are developed based on the classic IR models and the field based retrieval model is one of them.

Field based retrieval model is usually an derivative of a classic IR model and takes into account the several fields of a document, where both the occurrence of a term in a field and the occurrence frequency in that field are considered.

Fields refer to different parts of a document and are considered to be of different degrees of importance and length normalization. For example, Figure 2.3 presents that a web document can include title, body, H1[3] and other fields (Macdonald et al., 2013).

[3] H1 is a HTML element and represents the highest level of section headings.

```
<HTML>
   <HEAD>
        <TITLE>Consumer Health Information Search</TITLE>
   </HEAD>
   <BODY>
       <H1>What is health information search?</H1>
       <P>Queries can be proposed by a layperson or a professional.</P>
   </BODY>
</HTML>
```

Figure 2.3: Fields information of an example Web document.

A document has a large chance of being related to a query term, if this term occurs in the title. On the other way, if a query term occurs with low frequency in the body of a document, this document has a small chance of being related to the query. A document has a small chance of being related to a query term, if this query term occurs in the body with low frequency. Query terms occurs in the body of a document with low frequency (Macdonald et al., 2013).

One popularly used field based retrieval models is BM25F (Robertson et al., 2009) which is extended from BM25 retrieval model. Other field based retrieval model include PL2F (Lioma et al., 2006) which is from PL2 retrieval model (Amati, 2003).

2.3.2 Query Expansion

When conducting a search, users may use different words, which are known as synonyms, to refer to the same meaning. For example, Table 2.3 presents a group of words which share the same meaning of heart attack[4]: "A heart attack happens when blood flow to the heart suddenly becomes blocked."

Supposing a user issues a simple query "what is heart attack" to the searching system. Based on the theory of modern retrieval models, documents that only include the query term *heart* or *attack* or both terms can be retrieved as relevant to the issued query. However, the documents that do not include either of the query terms can also be related to this query. For example, these documents may use other words like *cardiac infarction* which is preferred by professionals (see the second column presented in Table 2.3).

This language gap between the common users and professionals can be solved using query expansion (QE) techniques.

Query expansion can be described as: "*a method for improving retrieval performance by supplementing an original query with additional terms. Expan-*

[4] Refer to https://medlineplus.gov/heartattack.html

Table 2.3: Synonyms for *heart attack*.

Preferred by common users	**Preferred by professionals**
attack hearts	cardiac infarction
attacking heart	cardiovascular stroke
heart attack	coronary attack
heart disease	disorder infarction myocardial
heart failure	heart attack
	myocardial infarct
	myocardial infarction
	myocardial infarcts
	myocardial necrosis
	syndrome myocardial infarction

sion can take place in the initial query formulation, the query reformulation stage of the online search, or both and can be performed manually, automatically, or interactively." (Efthimiadis, 1996).

Conceptually, the expanded query ($\mathrm{Q_e}$) is obtained after the query expansion phase by adding the new words ($\mathrm{Q_a}$) to the original query ($\mathrm{Q_o}$). This can be written as

$$\mathrm{Q_e} = \mathrm{Q_o} \cup \mathrm{Q_a}$$

The general process of using query expansion approach is presented in Figure 2.4: first, query terms are identified after pre-processing (tokenization and normalization); then, an expanding resource is used to find new words and the expanding techniques are applied; finally, an expanded query is built by adding these new words to the original one.

Different kinds of techniques and resources can be used to find words which are similar or related to a query term. And generally, QE techniques can be classified into two major classes depending on where the added words are derived from (Christopher et al., 2008):

- QE with relevance feedback. This method takes use of the retrieved results from the initially retrieval to refine the original query. This method is also known as the local method.
- QE with thesaurus. Synonymous words are expanded from a thesaurus. This method is also known as the global method.

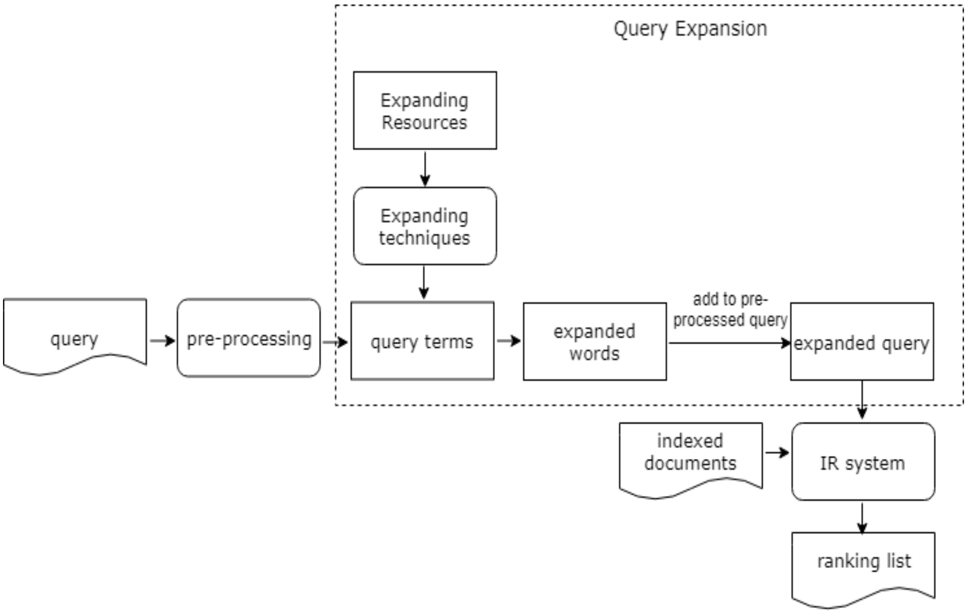


Figure 2.4: Query expansion approach.

Query Expansion with Relevance Feedback

As shown in Figure 2.5, the basic and general procedure of applying relevance feedback techniques in query expansion method can be summarized in five steps (Christopher et al., 2008).

```
Step 1. A user issues a query to a search system.
Step 2. This search system returns a set of retrieval results
        with an initial retrieval.
Step 3. The retrieved documents from the previous step are
        identified as relevant or irrelevant by human assessors
        or by IR weighing  models.
Step 4. Based on the feedback obtained in the previous step,
        the search system proposed a refined query which is
        deemed as a better representation of the information
        need.
Step 5. This refined query is then issued to the search system
        to perform another retrieval which is assumed to be an
        improved result compared to the initial retrieval.
```

Figure 2.5: Applying relevance feedback techniques in query expansion.

Step 1 and step 2 perform the initial retrieval with the original query. During step 3 and step 4, various relevance feedback techniques such as Rocchio algorithm (Rocchio, 1971), probabilistic relevance feedback and Pseudo Relevance Feedback (Christopher et al., 2008) have been proposed to determine what kind of words should be added to refine the original query. And finally with step 5, a new ranking list which is obtained using the expanded query is returned to the user.

Next we take the widely used Pseudo Relevance Feedback as an example and discuss its usage in refining the original query.

Pseudo Relevance Feedback. Pseudo relevance feedback (a.k.a. blind relevance feedback) is a way to improve retrieval performance without the user interaction. This method performs the initial retrieval and assumes that the top ranked documents are relevant.

Figure 2.6 presents how pseudo relevance feedback technique can be used in an IR model to satisfy the user more. Given a query q and the dataset, the retrieval system retrieved the dataset and returned an initial ranked list. Newly added terms are extracted from the top n documents in the initial list. An expanded query q' is generated, which includes the original query and the expanded terms. The search system retrieves the same dataset with the new generated query q' and an expansion-based list is produced.

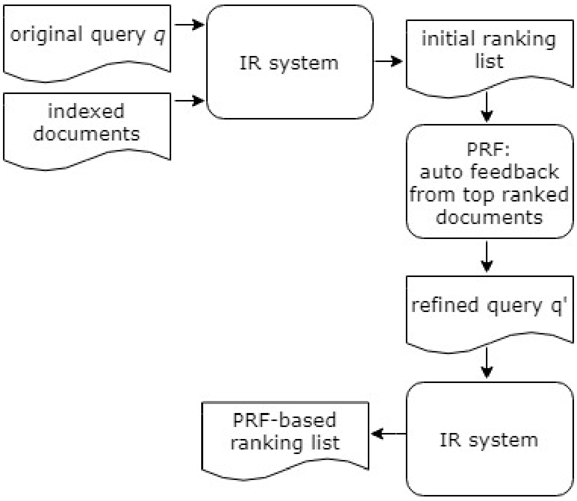


Figure 2.6: Pseudo relevance feedback.

Query Expansion with Thesaurus

Different from QE with relevance feedback, given a query term, its synonyms or related words can be automatically identified from the thesaurus. Many different ways have been explored to built a thesaurus and they can be summarized as (Christopher et al., 2008):

- Manually controlling and maintaining a thesaurus.
- Automatically building a thesaurus by analyzing a collection of documents such as using word co-occurrence statistics or shallow grammatical analysis.

2.3.3 Learning to Rank

In IR research area, machine learning techniques can be applied to improving ranking performance and this is known as Learning to Rank (LETOR) (Liu et al., 2009).

LETOR approach typically uses traditional supervised machine learning methods and trains a model with features extracted from documents and queries. Liu et al. (2009) presented a typical framework of learning to rank approach, as Figure 2.7 demonstrates[5].

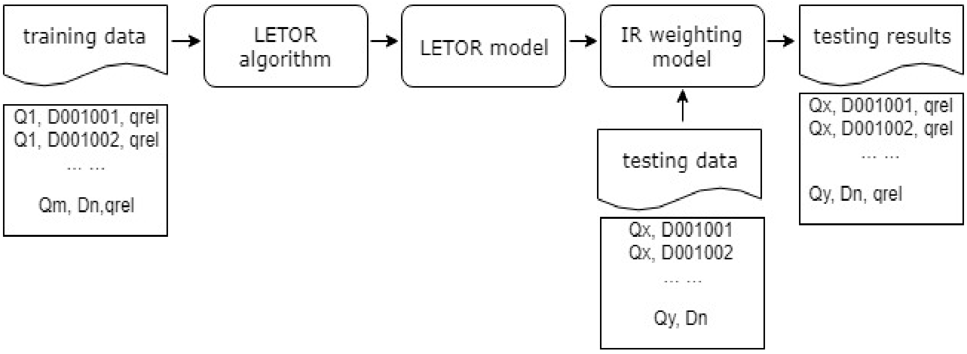


Figure 2.7: Learning to rank model.

LETOR Framework

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

[5] This figure is originated from Liu et al. (2009).

Training Data. Usually, with an enormous data set, not all the retrieved documents, but only the top documents from the ranking list are selected to be evaluated by human beings. The qrel file is the evaluation results containing the top documents and its judgement by humans, which can be used as training data when apply learning to rank techniques to learn a model.

Learning to Rank Algorithms. Various kinds of LETOR approaches have been proposed in learning a model. Typically, these approaches are classified into three categories: pointwise, pairwise and listwise approaches.

The pointwise approach takes into account a single document and train a classifier on this document. The classifier is then used to predict the relevance degree between the document and the query. The pairwise approach is used on a pair of documents and to find the optimal ordering for this document pair or the pairwise preference between each pair of documents. The listwise approach considers the entire list of documents and aims to find the optimal ordering for the whole list.

For each of these approaches, they can be further divided into sub-categories according to different machine learning technologies used. Table 2.4 lists some of the widely used algorithms according to each LETOR approach (Burges, 2010; Xu and Li, 2007; Cao et al., 2007).

Table 2.4: LETOR algorithms classification.

LETOR approach	**Example algorithm**
the pointwise approach	Random Forest,PRank,McRank
the pairwise approach	RankNet,RankBoost,RankSVM,MART
the listwise approach	LambdaRank,LambdaMART,ListMLE,AdaRank, ListNet

Testing Data. According to if the training and testing data are from the same collection or not, the testing process is categorized into two kinds. The first is when training and testing data are from the same collection. The provided queries are divided into training, validation and testing; the training queries together with the evaluated documents for the queries are used to train a learning to rank model, following the process mentioned in learning to rank module; the validation part is used for adjusting the learned model; and finally the testing queries are used for evaluating the model. The second kind of is when training and testing data are from different collection: the testing process is tested on a new collection different from the one that training or validation based on. The testing data includes new queries as well. Both these two kinds were researched in our work.

Features in Learning to Rank

Creating a feature list is an important task in applying LETOR approach. Traditionally, explored potential features are defined and blindly combined all together to create a feature list which is used to train a LETOR model afterwards. Vast amount of work has been researched into digging new features which is a costly job in Learning to Rank.

Depending on different application, various kinds of features can be extracted and used with machine learning techniques to obtain a learning to rank model.

Usually, depending on its dependency to the query, features are classified into two groups (Macdonald et al., 2013):

- Query independent (QI) features. Query independent features are independent of a query and are extracted from the documents only; for example, document length and document PageRank (Page et al., 1999) are categorized as QI features.
- Query dependent (QD) features. Query dependent features are typically extracted from query and document pairs; for example, the score obtained through a retrieval model can be used as the QD features.

In addition, depending on where a feature is extracted from, features can be classified according to the field of a document that it is originated from, such as the feature of Title, H1, Else, body or the whole document.

LETOR Benchmark Dataset. Microsoft LETOR Benchmark Dataset developed by Qin et al. (2010) and Qin and Liu (2013) contains a group of standard features which can be used for research on LETOR. Table 2.5 presents the 46 features available from LETOR 4.0 benchmark dataset (Qin and Liu, 2013).

Most of features (feature 1-40) can be extracted locally using different algorithms or retrieval models which include: tf, idf, document length, TFIDF retrieval model, document length, BM25 retrieval model, three language models LMIR.ABS, LMIR.DIR and LMIR.JM (Zhai and Lafferty, 2017). Also, six statistics of web-related features (feature 41-46) are also included.

Table 2.5: Features in LETOR 4.0 benchmark dataset.

Category	**Nr.**	**Feature name**	**Feature type**
term frequency	1	TF of Title	QD
	2	TF of anchor	QD
	3	TF of body	QD
	4	TF of URL	QD
	5	TF of whole document	QD
inverse document frequency	6	IDF of Title	QD
	7	IDF of anchor	QD
	8	IDF of body	QD
	9	IDF of URL	QD
	10	IDF of whole document	QD
TFIDF retrieval model	11	TFIDF of Title	QD
	12	TFIDF of anchor	QD
	13	TFIDF of body	QD
	14	TFIDF of URL	QD
	15	TFIDF of whole document	QD
Document length	16	Dl of Title	QI
	17	Dl of anchor	QI
	18	Dl of body	QI
	19	Dl of URL	QI
	20	Dl of whole document	QI
BM25 retrieval model	21	BM25 of Title	QD
	22	BM25 of anchor	QD
	23	BM25 of body	QD
	24	BM25 of URL	QD
	25	BM25 of whole document	QD
Language model with absolute discounting smoothing	26	LMIR.ABS of Title	QD
	27	LMIR.ABS of anchor	QD
	28	LMIR.ABS of body	QD
	29	LMIR.ABS of URL	QD
	30	LMIR.ABS of whole document	QD
Language model with Bayesian smoothing using Dirichlet prior	31	LMIR.DIR of Title	QD
	32	LMIR.DIR of anchor	QD
	33	LMIR.DIR of body	QD
	34	LMIR.DIR of URL	QD
	35	LMIR.DIR of whole document	QD
Language model with Jelinek-Mercer smoothing	36	LMIR.JM of Title	QD
	37	LMIR.JM of anchor	QD
	38	LMIR.JM of body	QD
	39	LMIR.JM of URL	QD
	40	LMIR.JM of whole document	QD
web-related statistics	41	PageRank	QI
	42	Inlink number	QI
	43	Outlink number	QI
	44	Number of slash in URL	QI
	45	Length of URL	QI
	46	Number of child page	QI

Format LETOR Feature File.

Prior to applying machine learning algorithms, features are extracted and detailed in a LETOR feature file (Liu et al., 2009), as shown in Figure 2.8.

```
                          LETOR feature file
#1: feature1
#2: feature2
#3: feature3
2  qid:001  1:0.279  2:0.067  3:0.056  #docid=001
1  qid:007  1:0.591  2:0.732  3:1.000  #docid=002
... ...
```

Figure 2.8: An example of format LETOR feature file.

The header part of a LETOR feature file is composed of features names. Then, every single line represents a retrieved document for a query: the first entry is the evaluation label from the qrel file, then the feature value, and finally the document ID. Also, when building a LETOR feature file, different features combinations were explored.

With a LETOR feature file built, LETOR algorithms can then be applied.

2.3.4 Rank Aggregation

Rank aggregation means combining the results available from multiple retrieval models and produces a single ranking list. Rank aggregation is also referred as data fusion (Vogt and Cottrell, 1998, 1999) or rank combination (Dwork et al., 2001) in some literature[6].

Vogt and Cottrell (1998, 1999) pointed out that aggregation allowed a significant reduction in the number of features and enumerated three beneficial effects of combining multiple models:

- The Skimming Effect. Skim means only top-ranked items are selected. Documents are represented by different retrieval methods and thus retrieve different relevant items. A combination model taking the top-ranked items from each of the retrieval approaches can increase recall as well as precision.

- The Chorus Effect. A number of retrieval approaches suggesting that an item is relevant to a query provide stronger evidence for relevance than that of a single approach. A combination model can explore this

[6] We use rank aggregation as the terminology in our work.

effect when ranking documents in the intersection of the retrieved lists higher.

- The Dark Horse Effect. Compared to other retrieval approaches, a retrieval approach may produce effective estimates of relevance for some documents.

The aggregation results can be a valued score or a sorting regarding to a document. Based on this, the techniques applied can be classified as score-based aggregation (Fox and Shaw, 1994; Lee, 1997; Vogt and Cottrell, 1999; Montague and Aslam, 2001; Manmatha et al., 2001; Xia et al., 2014; Kuzi et al., 2016) or sort-based aggregation (Dwork et al., 2001; Aslam and Montague, 2001; Deng et al., 2014):

- Score-based aggregation. A single score is computed by using an aggregation algorithm on all the scores achieved by each IR ranker; this computed single score is deemed as the final score and used to re-rank the documents.
- Sort-based aggregation. This kind of aggregation is applied when the scores are not available and only the ordering of the documents is known. It is also referred as rank-based aggregation in literature.

The classic and widely used scored-based aggregation method is a group of strategies which were first proposed by Fox and Shaw (1994). The representative sort-based aggregation methods include Borda's method, Footrule and Markov chain method (Dwork et al., 2001).

Sort-based rank aggregation is the basic kind of rank aggregation. Next, we discuss the classic one proposed by Fox and Shaw (1994).

Score-based Aggregation Strategies by Fox and Shaw

Early in 1993, Fox and Shaw (1994) presented their method for combining the similarity values from multiple retrieval runs. They investigated six combining strategies in their work: CombMAX, CombMIN, CombSUM, CombANZ, CombMNZ and CombMED. The definitions for these strategies are explained in Table 2.6. Although simple, these aggregation strategies showed their efficiency and are still popularly used by researchers in this area, being the classic methods for score-based aggregation.

Table 2.6: Aggregation methods.

Name	Aggregation Method
CombMAX	Maximum of Individual Similarities
CombMIN	Minimum of Individual Similarities
CombSUM	Sum of Individual Similarities
CombANZ	Sum of Individual Similarities / Number of Nonzero Similarities
CombMNZ	Sum of Individual Similarities × Number of Nonzero Similarities
CombMED	Median of Individual Similarities

2.4 Resources

This section discusses useful resources related to our research work, including resources for query expansion and the open-source IR platforms.

2.4.1 Resources for Query Expansion

Resources used for query expansion typically include manually controlled thesaurus and locally trained ones (Christopher et al., 2008).

Manually controlled thesaurus

Manually controlled or hand-crafted thesaurus is built by human editors and can contain concepts, groups of synonymous names for concepts and relationship between synonymous words or concepts. Manually controlled thesaurus can be used as the resource of query expansion. WordNet is such a thesaurus which is a large lexical database of English.

WordNet. WordNet[7] is a large and general-purpose lexical system built at Princeton University. WordNet's basic object is *synset* which is a set of synonyms. *Synsets* are organized by the lexical relations defined on them, which differ depending on part of speech. *Synsets* are constructed into a hierarchy and organized by the lexical relations defined on them. The lexical relations include antonym, homonym (is-a relation) and holonym (part-of relation) relations (Miller, 1995; Voorhees, 1994).

[7] https://wordnet.princeton.edu/

Word Embedding Model

Simply, word embeddings are one type of representations of corpus vocabulary, where words from the vocabulary are mapped to real-number vectors.

Word embeddings have been shown to be able to capture semantic as well as syntactic similarity of terms. Linking this characteristic to the aim of query expansion which expands an original query with similar or related terms; a well trained word embeddings model can be paralleled to a thesaurus and applied in the area of query expansion. Semantic similarities between words are shown to correspond to similarities between the learned words vectors (Mikolov et al., 2013b,a). Applying word embeddings into query expansion tasks demonstrates its potential application in the area of IR.

Different methods have been proposed and applied in training word embeddings and they are generally divided into two groups (Pennington et al., 2014): one is using latent semantic analysis (LSA) (Deerwester et al., 1990); the other is using context information (Mikolov et al., 2013b,a; Pennington et al., 2014). Two representative tools are Word2vec (Mikolov et al., 2013b,a) and GloVe (Pennington et al., 2014).

Word2vec models. Recently, Word2vec method (Mikolov et al., 2013b,a) which is able to map words into a high-quality and dense vectors has received a great deal of attention in many computer science areas and IR is an active area among them. Word2vec method employs a shallow and two-layers neural network to reconstruct linguistic contexts of corpus words. Word2vec method takes as its input a large corpus of text and outputs a high-dimensional vector space which typically are several hundred or even thousand dimensions. Each unique word in the corpus is assigned a corresponding vector in the space. And word that share common contexts in the corpus are closely positioned in the vector space. Two models are proposed in Word2vec method:

- Continuous bag-of-words model (CBOW). The CBOW model predicts one target word by given its context words. This model takes the average of the vectors mapped from the input context words, and then output the vector product of this averaged vector and the input hidden weight matrix.

- Continuous Skip-gram Model(SGM). The Skip-gram model can be seen as the opposite of the CBOW model, where the context words are predicted given a target word. This target word now is at the input layer, and the context words are on the output layer.

GloVe. The GloVe[8] is developed by Stanford University and an unsupervised learning algorithm for obtaining vector representations for words. Different from Word2vec which trains a model on the context words within a setting window, GloVe trains a model depending on the co-occurrence which is generated with the frequency of the words in a context (Pennington et al., 2014).

2.4.2 Open-source IR Platforms

A number of open source IR platforms are available to researches. Among them, three widely used ones are Terrier, Apache Lucene and Indri.

Terrier. Terrier[9] is described to be "a highly flexible, efficient, and effective open source search engine, readily deployable on large-scale collections of documents" (Ounis et al., 2006). It implements state-of-the-art indexing and retrieval functionality, and provides an ideal platform for the rapid development and evaluation of large-scale retrieval applications.

Apache Lucene. Apache Lucene[10] is an open source Java-based platform used to accomplish general search tasks (such as indexing and retrieval) and other search-related tasks as well (Białecki et al., 2012). Apache Lucene provides the ability of powerful and high-speed indexing over large data collections. This tool supports many query types such as phrase queries, proximity queries, wildcard queries, etc.

Indri. Indri[11] is a search engine developed by the Lemur project. Indri provides state-of-the-art text search and a rich structured query language for text collections of up to 50 million documents or 500 million documents (Strohman et al., 2005). Indri supports powerful query operators as well.

[8] The program and some pre-trained models are available from https://nlp.stanford.edu/projects/glove/.

[9] http://terrier.org/

[10] http://lucene.apache.org/

[11] https://www.lemurproject.org/indri.php

2.5 Evaluation

Classically and popularly adopted evaluation metrics related to our research work are discussed; the relevant evaluation tools are then introduced.

2.5.1 Evaluation Metrics

Generally, the evaluation metrics of an IR system can be divided as (Christopher et al., 2008):

- Ranked Retrieval Measures. Measures for evaluating ranked retrieval results which is the main measurement of modern IR system nowadays.
- Un-ranked Retrieval Measures. Measures for evaluating un-ranked retrieval results which is also known as set-based measure.

Ranked Retrieval Measures

Ranked information retrieval concerns retrieving documents from a huge data collection and users usually only pay attention to the top ranked documents returned by a search system. So it makes more sense to take into account only the top ranked results and evaluated an IR system at a given cut-off position.

P@n. Precision at position n (P@n) considers the precision at a given cut-off rank n. If r relevant documents have been retrieved at position n, P@n can be defined as (Craswell, 2009):

$$\mathrm{P@n} = \frac{\mathrm{r}}{\mathrm{n}}$$

NDCG@n. Normalized Discounted Cumulative Gain (NDCG) (Wang et al., 2013) is another popular used metrics for evaluating search results and is defined as:

$$\mathrm{NDCG@n} = \frac{\mathrm{DCG_n}}{\mathrm{IDCG_n}}$$

Where DCG (Discounted Cumulative Gain) is a gain accumulated over the results from the top to the bottom in a ranking list; highly relevant documents appearing lower in a ranking list should be penalized. And IDCG is ideal discounted cumulative gain.

MAP. For a set of queries, Mean Average Precision (MAP) is the mean of the average precision scores for each query q, where Q is the number of queries:

$$\mathrm{MAP} = \sum_{\mathrm{q=1}}^{\mathrm{Q}} \frac{\text{Average Precision}}{\mathrm{Q}}$$

RBP. Basically, RBP (Rank Biased Precision) (Zuccon, 2016) is defined as

$$\mathrm{RBP} = (1-\rho)\sum_{\mathrm{k=1}}^{\mathrm{K}} \rho^{\text{k-1}}\mathrm{r(k)}$$

The parameter ρ attempts to model user behaviour. The *r(k)* function is the standard RBP gain function: the value equals to 1 if the document at rank k is relevant and 0 if it is irrelevant.

uRBP. The formulation of understandability assessment is based on the Rank Biased Precision (RBP) and typically referred as uRBP, which is calculated as:

$$\mathrm{uRBP} = (1-\rho)\sum_{\mathrm{k=1}}^{\mathrm{K}} \rho^{\text{k-1}}\mathrm{r(k)u(k)}$$

where the *u(k)* function is a gain function for the readability dimension: the value is 1 if the document at rank k is understandable, and zero if not understandable; other values are equals to RBP.

uRBPgr. The uRBPgr measure is the graded version of uRBP and replaces *u(k)* with a graded gain (the usefulness of a document) in uRBP accordingly.

Typically, uRBP uses binary understandability assessments while uRBPgr uses graded understandability assessments (Zuccon, 2016; Palotti et al., 2015).

Un-ranked Retrieval Measures

For easy to understand, a contingency table is defined, as shown in Table 2.7: the predicted items are results obtained from an IR system and noted as retrieved and non-retrieved; the actual items means the document is actually relevant or irrelevant to a query.

Table 2.7: The contingency table for the evaluation of an IR system.

		Actual	
		relevant	irrelevant
Predict	retrieved	true positive (tp)	false positive (fp)
	non-retrieved	false negative (fn)	true negative (tn)

The two basic measures for un-ranked retrieval results evaluation are precision and recall (Christopher et al., 2008).

Precision. Precision (P) is the fraction of retrieved documents that are relevant to the query and can be defined as:

$$\text{precision} = \frac{\text{relevant documents retrieved}}{\text{retrieved documents}}$$

According to the contingency table (see Table 2.7), precision is calculated as:

$$\text{precision} = \frac{\text{tp}}{\text{tp} + \text{fp}}$$

Recall. Recall is the fraction of relevant documents that are retrieved:

$$\text{recall} = \frac{\text{relevant documents retrieved}}{\text{relevant documents}}$$

Using the contingency table, recall can be calculated as :

$$\text{recall} = \frac{\text{tp}}{\text{tp} + \text{fn}}$$

Accuracy. Accuracy is the fraction of an IR system classifications (relevant or irrelevant) that are correct which can be calculated using the contingency table as:

$$\text{accuracy} = \frac{\text{tp} + \text{tn}}{\text{tp} + f\,p + f\,n + \text{tn}}$$

F measure. F measure combines and trades off precision and recall. One classic F measure is F1 and defined as:

$$\mathrm{F1} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

2.5.2 Evaluation Tools

A number of standard evaluation tools are available to compute the evaluation metrics.

For topical relevance assessment, one classic and widely used is trec_eval[12], which is the standard tool used by the TREC community for evaluating an ad-hoc retrieval run, given the results file and a standard set of judged results.

For understandability assessment, one effective tool is ubire tool[13], which is an understandability-biased IR evaluation tool (Palotti et al., 2015).

2.6 Related Work Review

This section reviews the important past work concerning query expansion, learning-to-rank and rank aggregation.

The early work on QE relevance feedback using vector space models can be traced back to 1970s. Jones (1971) presented the early work on using term co-occurrence statistics to select query expansion terms. The classic Rocchio Algorithm was also proposed during that time (Christopher et al., 2008). This algorithm is based on relevance feedback and assumes that most users have a general idea of the relevance of the documents to the proposed queries.

Later, Voorhees (1994) applied QE by using manually selected words from WordNet and experimented on TREC data collection. Concepts of the synonym sets of WordNet were expanded using the links inside WordNet. The experimental results showed that this method presented little difference for complete queries; but significantly improved the retrieval performance on poorly built queries. The author concluded that this method had the potential to improve the retrieval performance since the original queries were usually not well detailed.

[12] https://trec.nist.gov/trec_eval/

[13] https://github.com/ielab/ubire

Recently, Carpineto and Romano (2012) conducted a detailed survey of AQE (Automatic Query Expansion) in the area of IR. They reviewed a number of recent approaches in using AQE and answer a group of questions related to AQE. In this paper, related experiments carried out on the classical benchmark data collections confirmed the effectiveness of the AQE techniques; obvious improvements were achieved and reported in averaged performance in the experiments. Based on the survey, the researchers concluded the effectiveness of applying AQE techniques in information retrieval area. They stated that AQE has the potential to overcome the difficulty of users in providing a more precise description of their information needs. A number of advantages in applying the AQE techniques in IR were concluded. Moreover, the shortcomings of the AQE techniques which needed to be improved in the future work were also listed out, such as: AQE implementation in an IR system, parameter setting automation and large queries executing and computing ability.

More recent works showed that applying word embeddings to query expansion is presented to be effective in information retrieval (Roy et al., 2016; Kuzi et al., 2016; Diaz et al., 2016; ALMasri et al., 2016).

Roy et al. (2016) proposed an Automatic Query Expansion employing word2vec methods. They assessed the effectiveness of the AQE method on TREC adhoc corpus and TREC web data collection. Their proposed AQE method was able to outperform the ones using the original queries, but underperformed feedback-based methods. Also, they found that feedback information did not affect the performance of the word2vec-based query expansion approaches.

Kuzi et al. (2016) proposed word embeddings based query expansion methods. They utilized trained word embeddings to select terms and used them in different ways: the selected terms were expanded to the original queries; the selected terms were integrated with the pseudo relevance feedback techniques.

Diaz et al. (2016) studied the use of word embeddings for query expansion. Globally trained word embeddings were compared to locally trained ones. Global word embeddings included: four GloVe embeddings trained on Wikipedia and Gigaword documents with different dimensions; one Word2vec embedding trained on Google News documents; a global embeddings trained with the entire corpus; a GloVe embedding trained on Common Crawl data. Local embeddings were trained with word2vec using one the three retrieval sources, respectively: trec12 corpus, robust corpus and ClueWeb2009 Category B web corpus. From their experiments, they concluded that locally trained word embeddings provided better similarity measures and outperformed globally trained ones significantly for query expansion.

Another similar work (ALMasri et al., 2016) compared the performance between deep learning based query expansion with pseudo-relevance feedback and mutual information; from the experiments the authors observed that neural network based models obtained a statistically significant improvement over the language models and other expansion models.

Despite the positive results obtained in related works, some researchers also observed that word vector based lexicon could reduce the retrieval performance (Roberts et al., 2016; Goodwin and Harabagiu, 2014).

LETOR approach has recently been proved to be effective in the area of IR. Evaluation information for a pair of document and query can be utilized to learn the model (Liu et al., 2009; Severyn and Moschitti, 2015; Cohen et al., 2018; Wang et al., 2018; Ai et al., 2018; Wang et al., 2016b).

Traditional learning to rank techniques typically depend on hand-crafted features for model training (Liu et al., 2009). In last few years, neural approaches have been applied in learning features directly from the data. These neural models show the potential ability to learn higher level features for the ranking task and capture new relationships not available by hand crafted features (Severyn and Moschitti, 2015; Cohen et al., 2018).

Traditional learning to rank techniques usually depend on manual judgments when training a model. In last years, another trend are online learning to rank approach which has also attracted attention in the information retrieval research area. Online learning to rank gather information from the implicit user feedback like clicks, and can directly take use of the returned search results (Wang et al., 2018; Ai et al., 2018; Wang et al., 2016b).

In the area of IR, there is a long history of using aggregation techniques over different retrieval models, and rank aggregation techniques have been applied in different applications (Fox and Shaw, 1994; Lee, 1997; Vogt and Cottrell, 1999; Montague and Aslam, 2001; Manmatha et al., 2001; Abacha, 2016).

In the early years, rank aggregation techniques were mainly used to combine ranking results obtained from different retrieval models inside a search engine (Fox and Shaw, 1994; Lee, 1997; Vogt and Cottrell, 1999).

Following work by Fox and Shaw (1994), Lee (1997) also further observed that CombMNZ worked best among all these combination techniques, with CombSUM, and CombMIN and CombMAX performing the worst. Vogt and Cottrell (1999) thoroughly analyzed these methods by Fox and Shaw (1994) using a linear combination model for information retrieval systems; in their work, the linear combination model combined results from multiple IR systems and a weighted sum of scores was used.

Later, rank aggregation techniques were extended and used for meta-search where ranking results are obtained from different search engines or meta-search engines (Montague and Aslam, 2001; Manmatha et al., 2001; Abacha, 2016).

Montague and Aslam (2001) stated they empirically improved the performance of well known CombMNZ and CombSUM meta-search algorithms. They improved these strategies with using more statistics than max and min in the normalization scheme.

Manmatha et al. (2001) used model of score distributions in combining results from various search engines to produce a meta-search engine.

Abacha (2016) used QE and rank-based result fusion for TREC 2016 Clinical Decision Support (CDS) track. The team used CombSUM method to combine the ranking results obtained from three IR weighting models BM25, TFIDF and In_expB2 and this method scored top 10 among all the participants.

More recently, some research work also showed the effectiveness of rank aggregation in other applications (Xia et al., 2014; Kuzi et al., 2016).

Work by Xia et al. (2014) adopted CombSUM CombMNZ and CombANZ in their work for cross-language bug localization, where their methods combined the top-100 files from each ranked lists into one list.

Kuzi et al. (2016) proposed to use combination strategies on fusion-based term scoring. Resulting term lists were fused and CombSUM, CombMNZ and CombMAX were mainly used in their work. These techniques were used for words selecting and applied on word embedding based query expansion application. The experiments tested on TREC dataset proved the effectiveness of using these techniques in word scoring and selecting.

Chapter 3

Health Information Retrieval

As a crossed research area, health information retrieval roots and develops from the techniques concerning both information retrieval and health area. There have been quite an abundant number of studies in the area of HIR and CHIR. This chapter first reviews the state-of-the-art in HIR and CHIR which includes: the important concepts used, state-of-the-art techniques, useful resources and related work. Then we discuss the concepts and related work of understandability as well as its use in CHIR.

3.1 Medical Resources for Query Expansion

A number of controlled medical thesaurus have been generated for health-related research, such as Medical Subject Headings, SNOMED, ICD-9, ICD-10, HL7, Consumer Health Vocabulary, etc. The following sections discuss three widely used ones in CHIR: Medical Subject Headings, Consumer Health Vocabulary and Unified Medical Language System.

3.1.1 Medical Subject Headings

Medical Subject Headings (MeSH) vocabulary is a controlled vocabulary and popularly used in HIR area. MeSH vocabulary was first created in the 1960s and is annually updated by National Library of Medicine (NLM). MeSH is originally used for the purpose of indexing and cataloging biomedical literature. For example, MEDLINE/PubMed[1] database takes use of MeSH to index the articles. MeSH is also used in assisting the searchers with subject search when searching in a biomedical database.

[1] https://www.nlm.nih.gov/bsd/pmresources.html

MeSH Structure

MeSH Vocabulary includes four types of terms: Headings, Subheadings, Supplementary Concept Records and Publication Types[2].

The description and examples of each term type are presented in Table 3.1. The commonly used alternative name in literature is also introduced, for example, *MeSH Headings* is also referred as *main Headings* or *Descriptors*.

Table 3.1: MeSH term types.

Term type	**Description**	**Examples**
MeSH headings (main Headings or Descriptors)	Biomedical concepts	Body Weight, Heart, Dental Cavity Preparation
Subheadings (Qualifiers)	A specific aspect of a concept	adverse effects, diagnosis, therapy
Supplementary Concept Records	Substance terms, protocols and rare disease terms	cordycepin, MOPP protocol, Snyder Robinson syndrome

Publication Types
(Publication Characteristics)
	Type of publication
	being indexed
		Publication Formats:
		Lectures;
		Study Characteristics:
		Clinical Trial;
		Publication Components:
		English Abstract;

Next, we discuss the two mostly used terms types in HIR research: MeSH Headings and MeSH Subheadings.

MeSH Headings. All MeSH Headings are organized in a hierarchical structure with 16 main branches[3] or main category of biomedical concepts, as shown in Table 3.2. And in turn, each main branch has many sub-branches.

Each MeSH Heading has a position in the this hierarchy structure. Specially, some terms may appear in more than one branch of the hierarchy. For example and as presented in Figure 3.1, term *Asthma* is categorized under one main branch *Disease*, appears in two second-level sub-branches *Respiratory Tract Diseases* and *Immune System Diseases*, and located in four different third-level sub-branches[4].

[2] https://www.nlm.nih.gov/mesh/meshhome.html
[3] https://meshb.nlm.nih.gov/treeView
[4] This example is performed using MeSH 2019.

```
Diseases
   Respiratory Tract Diseases
      Bronchial Diseases
         Asthma
            Asthma, Aspirin-Induced
            Asthma, Exercise-Induced
            Asthma, Occupational
            Status Asthmaticus

Diseases
   Respiratory Tract Diseases
      Lung Diseases
         Lung Diseases, Obstructive
            Asthma

Diseases
   Respiratory Tract Diseases
      Respiratory Hypersensitivity
         Alveolitis, Extrinsic Allergic
            Aspergillosis, Allergic Bronchopulmonary
               Asthma
                  Asthma, Aspirin-Induced
                  Asthma, Exercise-Induced
                  Asthma, Occupational
                  Status Asthmaticus

Diseases
   Immune System Diseases
      Hypersensitivity
         Hypersensitivity, Immediate
            Respiratory Hypersensitivity
               Alveolitis, Extrinsic Allergic
                  Aspergillosis, Allergic Bronchopulmonary
                     Asthma
                        Asthma, Exercise-Induced
                        Asthma, Occupational
                        Status Asthmaticus
```

Figure 3.1: Term *Asthma* organized in MeSH hierarchy.

Table 3.2: Main branches of MeSH Headings.

Nr.	**Category**
A	Anatomy
B	Organisms
C	Diseases
D	Chemicals and Drugs
E	Analytical, Diagnostic and Therapeutic Techniques and Equipment
F	Psychiatry and Psychology
G	Phenomena and Processes
H	Disciplines and Occupations
I	Anthropology, Education, Sociology and Social Phenomena
J	Technology, Industry, Agriculture
K	Humanities
L	Information Science
M	Named Groups
N	Health Care
V	Publication Characteristics
Z	Geographicals

This hierarchy in MeSH makes available a search of a broader term to include its narrower terms in all branches automatically. For instance, a search of *Asthma* would automatically find its four narrower terms: *Asthma, Aspirin-Induced*; *Asthma, Exercise-Induced*; *Asthma, Occupational* and *Status Asthmaticus.*

MeSH Subheadings. All MeSH Subheadings are categorized in a logical hierarchy structure and the main or the first-level category[5] is shown in Table 3.3. We can see that MeSH Subheadings are categorized according to the logic aspect of a concept.

Concepts Relationship in MeSH

Till now, we have discussed the main concepts and structures included in the MeSH vocabulary, now we look at the relationships among concepts.

In MeSH, synonymous terms are clustered into a concept; concepts closely related to each other in meaning are associated by a MeSH Heading record.

[5] This table presents the 2019 MeSH Subheadings category and is generated based on information available from https://www.nlm.nih.gov/mesh/subhierarchy.html.

Table 3.3: MeSH Subheadings Categorization.

Nr.	**MeSH Subheadings Category**	**Nr.**	**MeSH Subheadings Category**
1	analysis	14	therapy
2	anatomy & histology	15	classification
3	chemistry	16	drug effects
4	diagnosis	17	education
5	etiology	18	ethics
6	organization & administration	19	history
7	standards	20	injuries
8	supply & distribution	21	instrumentation
9	trends	22	methods
10	pharmacology	23	pathogenicity
11	physiology	24	psychology
12	statistics & numerical data	25	radiation effects
13	therapeutic use	26	veterinary

Possible relationships between concepts are *preferred term*, *related*, *narrower-than* and *broader-than*; the *narrower-than* relationship is more common than *broader-than* or *related* relationship in MeSH vocabulary (Darmoni et al., 2012).

Table 3.4 presents all the MeSH concepts related to the concept *Abortion Induced*. This medical concept *Abortion Induced* has several synonymous terms including *Abortion*, *Induced* and *Induced Abortion*; some *broader-than* concepts like *Fertility Control*; *narrower-than* concepts like *Abortion Saline-Solution*; and *related* concepts like *Abortion Rate*.

3.1.2 Consumer Health Vocabulary

Some consumer health vocabularies have been built to facilitate the increasing online health search by consumers.

Personal Health Terminology (PHT) developed by Intelligent Medical Objects[6] is a commercial resource which is able to translate the most common terms in structured ICD-9 codes to consumer friendly synonyms. Terms in the PHT can be mapped to more than one ICD-9 code which is designed to have one preferred clinician term and one preferred patient term. Cross-mappings are available from the PHT terms to other medical vocabularies (Zielstorff, 2003).

[6] http://www.e-imo.com

Table 3.4: Concepts Relationship in MeSH.

Concepts	**Concept Name**	**Synonyms**
concept	Abortion Induced	Abortion Induced Induced Abortion
broader-than concepts	Fertility Control	
narrower-than concepts	Abortion, Saline-Solution Abortion,Rivanol Abortion, Soap-Solution Abortion, Drug-Induced Embryotomy	
related concepts	Abortion Rate Previous Abortion Anti-Abortion Groups Abortion Techniques Abortion Failure	

The open-access and collaborative (OAC) consumer health vocabulary (CHV)[7] is a non-commercial resource and co-produced by University of Utah, NLM and several other academic groups[8]. The aim of OAC CHV is to help bridging the communication gap, particular between consumers and informatics applications (Zeng and Tse, 2006). Different from MeSH which is mainly composed of technical medical terms, OAC CHV is designed to aid the needs of consumer health applications and includes in more consumer friendly terms used by consumers, such as jargon, slang, ambiguous and misspelled words. OAC CHV enables the translation of technical terms used by health professionals to informal and common words used by common consumers (Keselman et al., 2008). Table 3.5 presents the semantic types defined in OAC CHV[9].

3.1.3 Unified Medical Language System

Within different medical vocabularies, the concepts sharing the same meaning are organized and named differently. It is not easy to distribute useful information among different application systems which use different medical

[7] http://consumerhealthvocab.chpc.utah.edu/CHVwiki/

[8] https://www.nlm.nih.gov/research/umls/sourcereleasedocs/current/CHV/

[9] This table is generated based on https://www.nlm.nih.gov/research/umls/sourcereleasedocs/current/CHV/stats.html.

Table 3.5: The semantic types in OAC CHV.

Nr.	Semantic Type
1	Pharmacologic Substance
2	Organic Chemical
3	Disease or Syndrome
4	Finding
5	Therapeutic or Preventive Procedure
6	Amino Acid, Peptide, or Protein
7	Body Part, Organ, or Organ Component
8	Neoplastic Process
9	Medical Device
10	Sign or Symptom
11	Injury or Poisoning
12	Laboratory Procedure
13	Biologically Active Substance
14	Pathologic Function
15	Plant
16	Diagnostic Procedure
17	Intellectual Product
18	Manufactured Object
19	Qualitative Concept
21	Mental or Behavioral Dysfunction

vocabularies. The Unified Medical Language System (UMLS)[10] has been proposed to tackle the problem (Humphreys et al., 1998):

- UMLS is a system which can bring together many health and biomedical vocabularies and standards.
- UMLS is composed of a set of files and applications which can make more efficient the inter-operation of the different applications in health area.
- UMLS makes available more effective retrieval of machine readable information.

The UMLS has a variety of applications in the health area such as medical language translation, electronic health records and classification concerning health information.

[10] https://www.nlm.nih.gov/research/umls/

The UMLS includes three knowledge sources: (i) Metathesaurus which contains Terms and codes from a group of medical vocabularies; (ii) Semantic Network which defines semantic types and their relationships; (iii) SPECIALIST Lexicon and Lexical Tools which is composed of a set of Natural language processing tools for health information.

UMLS Metathesaurus

UMLS Metathesaurus is a very large vocabulary database which contains information about health and biomedical related concepts, various names of these concepts, and the relationships among these concepts (Bodenreider, 2004).

UMLS Metathesaurus is built from nearly 200 different resources, including health-related thesauri, biomedical classification, public health statistics, indexed biomedical literature, clinical information, health service research and so on. In some way, UMLS Metathesaurus is a fusion of these various medical resources. The two medical vocabulary MeSH and OAC CHV introduced in previous sections are as well included in the UMLS Metathesaurus.

UMLS Metathesaurus is generated by the other two knowledge sources from the UMLS (Semantic Network and Lexical Tools). The three key functions of the UMLS Metathesaurus can be summarized as (Hiemstra, 2009):

- Grouping synonymous terms from various medical vocabularies into concepts.
- Identifying relationship between concepts while preserving the meanings, term names and relationships from each source vocabulary.
- Categorizing concepts with defined semantic types and structuring them in a tree hierarchy.

In the UMLS Metathesaurus, synonyms sharing the same meaning refer to a single concept. In other words, a concept can have many different names. The various source medical vocabularies which are incorporated by the UMLS Metathesaurus may use different names for a concept; moreover, even in the same medical vocabulary, the concept can have different names.

In the UMLS Metathesaurus, Concept Unique Identifier (CUI) is used to associate all the different concept names which sharing the same meaning but in different formats.

Term identifier CUI. CUI is an important terminology in the UMLS Metathesaurus which can solve this problem. Terms or different names originated from various medical vocabularies and referring to the same concept are clustered into a concept; in turn, each concept is assigned a CUI beginning with the capital letter C and followed by seven numbers.

As shown in Table 3.6, the concept *Hypertensive disease* contains a set of synonyms and is assigned with a unique CUI: C0020538; its synonyms from different vocabularies are organized under this same CUI[11]. In other words, CUI C0020538 links all the words related to the concept *Hypertensive disease* from various sources in the UMLS Metathesaurus.

When searching with a query including *high blood pressure*, the UMLS Metathesaurus can be used to find its synonyms or related terms from different source vocabularies, such as *Hypertension* from source MeSH and *systemic hypertension* from source OAC CHV (Aronson, 2006).

Table 3.6: Associating the concept *Hypertensive disease* with a CUI code.

Vocabulary	**Vocabulary Code**	**Atoms**
MeSH	D006973 (MSH) Hypertension	Blood Pressure, High [A26603831/MSH] Blood Pressures, High [A6954576/MSH] High Blood Pressure [A6955926/MSH/] High Blood Pressures [A6954738/MSH] Hypertension [A0070978/MSH]
OAC CHV	0000006443 (CHV) high blood pressure disorder	high blood pressure [A18684846/CHV] systemic hypertension [A18554948/CHV] hyperpiesis [A18554947/CHV] hypertension [A18592030/CHV] hypertensive disease [A18684848/CHV] hypertensive disorder [A18573396/CHV] vascular hypertension [A18629187/CHV] high blood pressure disorder [A18684847/CHV] hypertensive vascular disease [A18592031/CHV]

AUI, SUI and LUI. Besides CUIs, several other types of term identifiers used in the UMLS Metathesaurus are: AUIs which is used to represent

[11] Part of the atoms from MeSH and OAC CHV source vocabularies are listed here. Atoms identified with other codes or from other source vocabularies are not presented in this table.

synonyms in the source vocabulary and these synonyms are referred as *atoms*; SUIs which are used for representing lexical variants in source vocabularies; LUIs which are for the normalized name of strings;

Semantic type identifier TUI. Besides these term identifiers, another important identifier is Type Unique Identifier (TUI) which is used of identify the semantic type. Table 3.7 presents parts of the semantic types available in UMLS[12].

Table 3.7: The semantic type in UMLS.

Abbr.	Type Unique Identifier (TUI)	Full Semantic Type Name
aapp	T116	Amino Acid, Peptide, or Protein
acab	T020	Acquired Abnormality
acty	T052	Activity
aggp	T100	Age Group
amas	T087	Amino Acid Sequence
amph	T011	Amphibian
anab	T190	Anatomical Abnormality
anim	T008	Animal
anst	T017	Anatomical Structure
antb	T195	Antibiotic
arch	T194	Archaeon

3.1.4 Word Embedding Models Trained with Health Data

Table 3.8 lists some of the publicly available word embeddings models trained on health data.

Model $\text{WE}_{\text{bioasq}}$ is trained by BioASQ[13], which organizes online biomedical challenges. This model is trained using Word2vec tools and using English abstracts of biomedical articles from PubMed as the training data (Pavlopoulos et al., 2014). Models $\text{WE}_{\text{trec-cbow}}$ and $\text{WE}_{\text{trec-skip}}$[14] are trained with TREC Medical Records Track collection, with each using a different training model from Word2vec (Zuccon et al., 2015).

[12] The full list is available from https://metamap.nlm.nih.gov/Docs/SemanticTypes_2018AB.txt.

[13] Available from http://bioasq.org/.

[14] Available from http://zuccon.net/ntlm.html.

Table 3.8: Pre-trained word embedding models using biomedical data.

	$\textbf{WE}_{\textbf{bioasq}}$	$\textbf{WE}_{\textbf{trec-cbow}}$	$\textbf{WE}_{\textbf{trec-skip}}$
Number of Models	1	20	20
Training Tool	Word2vec	Word2vec CBOW	Word2vec Skipgram
Window Size	5	[5,10]	[5,10]
Dimension	200	[100,200,1000]	[100,200,1000]
Training Data	English abstracts of PubMed biomedical articles (10,876,004)	TREC Medical Records Track (2011-2012)	TREC Medical Records Track (2011-2012)
Distinct Words	1,701,632	-	-

3.2 Techniques and Tools

The issue of the medical vocabulary gap between lay queries and expert expressions in CHIR can prevent lay users from finding relevant information (Yang and Gonçalves, 2017) and may arise other problems like too much concerns about common symptomatology (White and Horvitz, 2009).

To overcome the difficulty that arises from this language gap, different methods have been proposed. One widely adopted and effective method is using query expansion techniques, which is often an effective way to retrieve more relevant results and improve retrieval performance (Christopher et al., 2008; Carpineto and Romano, 2012).

A wide range of query expansion techniques have been applied in HIR and can be classified into two main threads depending on the resources used: one is using existing controlled medical thesaurus to find synonymous and related terms which serve as expanding terms for the query; the other one is using the pre-trained word embeddings model as the expanding resource.

The following sections discuss state-of-the-art techniques on using query expansion techniques in the area of HIR.

3.2.1 Query Expansion with Medical Thesaurus

As discussed, the Concept Unique Identifier CUI is used to link various expressions which represent the same meaning for a UMLS Metathesaurus concept. In turn, we can use the CUI to find desired expansion words.

In the UMLS Metathesaurus, one CUI is associated with one or more atoms; but one atom is associated with only one CUI, In a source vocabulary, one vocabulary code is associated with one or more atoms; but one atom is associated with only one vocabulary code[15].

When using the UMLS Metathesaurus to find expansion words to a query term, one typical usage is to find words which are from the same source vocabulary. And the processing steps are as follows: (i) a CUI identification tool is used to find one or more CUIs mapped from a query term; (ii) with a CUI code, all the atoms associated with this CUI can be found; (iii) the vocabulary code of one atom is also identified; (iv) related atoms can be found given a vocabulary code since one vocabulary code is associated with one or more atoms. In this way, various words which express the same meaning and from the same source vocabulary can be identified. These words are then used to expand the original query.

We can use the same techniques to find expansion words from different source vocabulary in the UMLS Metathesaurus.

As presented in Table 3.6, query terms *high blood pressure* is mapped to CUI C0020538. Using this CUI code, we are able to find a number of synonyms or related words for *high blood pressure*.

3.2.2 Query Expansion with Word Embeddings

These trained word embeddings have been explored and used in various kinds of applications and query expansion is one of them. Since semantic similarities between words are shown to correspond to similarities between the learned words vectors (Mikolov et al., 2013b,a), well trained word embedding model can be paralleled to a thesaurus and applied in the area of query expansion.

In recent years, training word embeddings on large medical and health corpus have also been researched in the area of HIR. These word embeddings models can integrate existing medical knowledge; related words found with these models can be used for query expansion in HIR and CHIR.

[15] https://documentation.uts.nlm.nih.gov/rest/atoms/

As presented in Table 3.9, a group of words are listed as related to the word *headaches*[16] in a pre-trained word embeddings model. In this example, the similarity between *headaches* and the identified word is measured by cosine distance, where the higher score indicating that the word is more associated with *headaches*.

Table 3.9: Words related to *headaches* in a trained word embedding model.

Word	**Cosine distance**
urinary	0.846022
hypertension	0.836082
diarrhea	0.835776
sweating	0.83254
constipation	0.821217
diarrhoea	0.819985
arthritis	0.81948
vomiting	0.818145
gastric	0.817944
drowsiness	0.815294
...	...

The typical usage is paralleling such a pre-trained word embeddings model to a medical thesaurus and employing it as a resource to find related terms. The expansion process is similar to the way adopted when applying a medical thesaurus. Concerning words choosing, the top ranks words are usually regarded as more related or similar to the query term.

3.2.3 Medical Concepts Identification

Medical entity recognition and extraction is an important step in query processing. We note this application as medical concepts identification (MCI) in our document. Based on ideas from Abacha and Zweigenbaum (2011), the MCI process mainly includes these three steps:

- Identifying and delimiting medical entities from medical or health related documents.
- Identifying the semantic category of these medical entities.

[16] The result is obtained using a pre-trained word embeddings tool which is available from https://code.google.com/archive/p/word2vec/.

- Mapping these identified medical entities or concepts to the UMLS Metathesaurus CUI codes.

Since health or medical text has unique characteristics that distinguish it from other literature in the general domain, specific Natural Language Processing (NLP) approaches or tools are required for the MCI application. MCI are typically accomplished by tools using NLP or machine learning techniques (Abacha and Zweigenbaum, 2011).

A number of such tools exist and two classic and widely used ones are MetaMap (Aronson and Lang, 2010) and cTAKES (Savova et al., 2010).

MetaMap

MetaMap[17] is a program developed by the NLM and aims to map biomedical text to the concepts in the UMLS Metathesaurus. The tool was built taking use of a hybrid technique including natural language processing, computational linguistic and knowledge intensive approach (Aronson, 2001).

MetaMap takes use of NLP techniques and the general processes of analyzing a biomedical text can be summarized as: (i) tokenization, sentence boundary determination and abbreviation identification; (ii) part-of-speech tagging; (iii) input words lookup in the SPECIALIST lexicon; (iv) phrases and their lexical heads identification by the parser available in SPECIALIST; (v) variants of all phrase words are determined; (vi) how well the candidate identifications from the UMLS Metathesaurus match the input text are computed and represented with scores. Besides these general processed taken by MetaMap, other optional process like Word Sense Disambiguation is also available.

MetaMap tool is a configurable program and includes a group of options when mapping the biomedical text to the UMLS Metathesaurus:

- Data options where different source vocabulary version and data model are choose-able.
- Output options where the output format generated by MetaMap are customized to cater to different application needs.
- Processing options which control the algorithmic computations to be performed by MetaMap, such as using Word Sense Disambiguation techniques or not.

[17] https://metamap.nlm.nih.gov/

Table 3.10 presents an example of using MetaMap to map a biomedical text *headaches caused by too much blood or “high blood pressure”* into the UMLS Metathesaurus[18]. In this example, five phrases are identified from the short text: *headaches*, *caused by too*, *or*[19], *blood* and *high blood pressure*.

In Table 3.11, we present the processing results for the identified phrase *high blood pressure*; the other three phrases get their results in the same format[20]. In this example, MetaMap identified 19 effective Metathesaurus candidates for phrase *high blood pressure*. Further on, the best two candidates which received a score of 1000 each form the top-scoring mapping (Aronson and Lang, 2010). We can include more mapping terms by changing the threshold of the score for mapping.

Table 3.12 gives an example of the first candidate for phrase *high blood pressure*. For every candidate returned from the MetaMap tool, the score indicating how well this candidate match the phrase *high blood pressure*, CUI number, the meaning of the concept, source vocabulary and abbreviated semantic type are displayed.

cTAKES

cTAKES (Apache clinical Text Analysis and Knowledge Extraction System) is a natural language processing system to extract information from electronic medical record and clinical text. Similar to MetaMap, cTAKES is also able to map biomedical text into the UMLS Metathesaurus. cTAKES was built using rule-based approaches and machine learning techniques (Savova et al., 2010).

cTAKES is able to deal with a biomedical text with these processing (Savova et al., 2010): (i) sentence boundary detecting; (ii) tokenization; (iii) normalization; (iv) part-of-speech tagging; (v) shallow parsing; (vi) named entity recognition.

We use cTAKES to process the example text *headaches caused by too much blood "or" high blood pressure* and the direct result returned by cTAKES is presented in Figure 3.2. As we can see, six UMLS Metathesaurus concepts are found by cTAKES; their semantic name as well as the CUIs are identified[21].

[18] Different options in MetaMap can have different output. We here choose showing candidates numbers, mappings of the highest score, semantic types and CUIs.

[19] This example is not processed and without stop words removing.

[20] We here choose showing candidates numbers, mappings of the highest score, semantic types, source vocabulary and CUIs.

[21] The first line under each tokenized term identifies the part-of-speech tag, such as *NN* means the term is a noun.

Table 3.10: Example of MetaMap biomedical text processing.

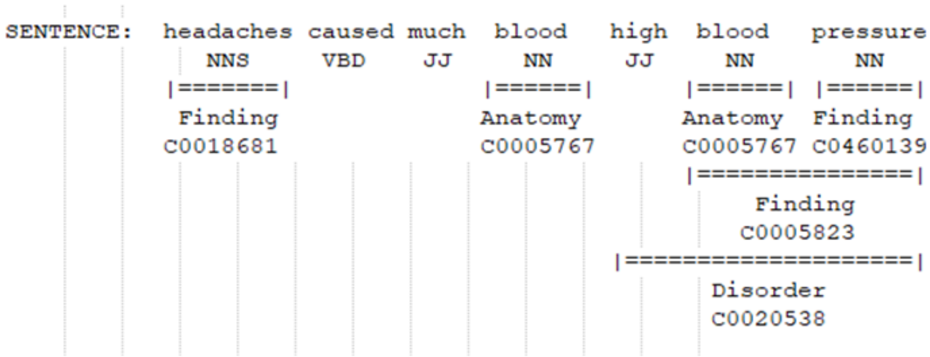
```
Processing text: headaches caused by too much blood or “high blood pressure”
Phrase: headaches
1. Meta Mapping (1000):
1000 C0018681:Headaches (Headache) [sosy]
Phrase: caused by too
1. Meta Mapping (756):
756 C0015127:cause (Etiology aspects) [ftcn]
2. Meta Mapping (756):
756 C1524003:Cause (Science of Etiology) [cnce]
Phrase: much blood
1. Meta Mapping (1000):
1000 C0005767:Blood [bdsu]
2. Meta Mapping (1000):
1000 C0005768:BLOOD (In Blood) [bdsu]
3. Meta Mapping (1000):
1000 C0229664:BLOOD (peripheral blood) [bdsu]
Phrase: or
Phrase: “high blood pressure”
1. Meta Mapping (1000):
1000 C0020538:Blood Pressure High (Hypertensive disease) [dsyn]
2. Meta Mapping (1000):
1000 C2926615:High blood pressure (Ever told by doctor or nurse that you have high blood pressure:Finding:
Point in time:^Patient:Ordinal) [clna]
```

Table 3.11: Example of MetaMap output for *high blood pressure*.

```
Meta Candidates (19)
1000 C0020538 Blood Pressure, High (Hypertensive disease {AOD,CCS,CCS_10,CHV,COSTAR,CSP,CST,etc.}) [dsyn]
1000 C2926615 High blood pressure (Ever told by doctor or nurse that you have high blood pressure:Finding:Point in time:^Patient:Ordinal
              {LNC,MTH,NLMSubSyn}) [clna]
901 C0005823 BLOOD PRESSURE (Blood Pressure {AOD,CHV,CSP,LCH,LCH_NW,LNC,MEDLINEPLUS,MSH,etc.}) [orgf]
901 C1271104 Blood Pressure (Blood pressure finding {MTH,NCI,NLMSubSyn,SNOMEDCT_US}) [fndg]
901 C1272641 Blood pressure (Systemic arterial pressure {CHV,MSH,MTH,NLMSubSyn,SNOMEDCT_US}) [fndg]
827 C0005767 Blood {AOD,CHV,CSP,FMA,HL7V2.5,etc.} [bdsu]
827 C0005768 BLOOD (In Blood {MSH,MTH}) [bdsu]
827 C0033095 Pressure (Pressure- physical agent {AOD,CHV,LCH,LCH_NW,LNC,MSH,MTH,NCI,NCI_FDA,SNOMEDCT_US}) [phpr]
827 C0205250 High {CHV,LNC,MTH,NCI,SNMI,SNOMEDCT_US} [qlco]
827 C0229664 BLOOD (peripheral blood {CHV,MTH,NCI,NCI_CDISC,NCI_NCI-GLOSS,SNM,SNMI,SNOMEDCT_US}) [bdsu]
827 C0460139 Pressure (Pressure (finding) {CHV,LNC,MTH,SNM,SNMI,SNOMEDCT_US}) [fndg]
827 C1299351 High (Abnormally high {CHV,HL7V3.0,MTH,SNOMEDCT_US}) [qlco]
827 C1306345 Pressure (Pressure - action {MTH,SNOMEDCT_US}) [ftcn]
827 C2700149 HIGH (Value Above Reference Range {MTH,NCI,NCI_CDISC}) [inpr]
827 C3887512 high (high - ActExposureLevelCode {HL7V3.0,MTH}) [idcn]
827 C3889660 High (High Mitosis-Karyorrhexis Index {MTH,NCI}) [qnco]
827 C4284008 Pressure (Pressure (property) {MTH,SNOMEDCT_US}) [qnco]
827 C4321237 High (High Level {MTH,NCI}) [fndg]
827 C4522209 High (IPSS Risk Category High {MTH,NCI}) [inpr]
Mappings
Meta Mapping (1000)
1000 C0020538 Blood Pressure, High (Hypertensive disease {AOD,CCS,CCS_10,CHV,COSTAR,CSP,CST,etc.}) [dsyn]
Meta Mapping (1000)
1000 C2926615 High blood pressure (Ever told by doctor or nurse that you have high blood pressure:Finding:Point in time:^Patient:Ordinal
              {LNC,MTH,NLMSubSyn}) [clna]
```

Table 3.12: One candidate of phrase *high blood pressure*.

Item	**Meaning**
1000	candidate score
C0020538	CUI
Blood Pressure, High	candidate name
Hypertensive disease	concept meaning
{AOD,CCS,CCS_10,CHV,COSTAR,CSP,CST,etc.}	source vocabulary
[dsyn]	short semantic type: T047 Disease or Syndrome



```
SENTENCE:  headaches caused much  blood   high  blood   pressure
              NNS      VBD    JJ    NN     JJ    NN       NN
           |=======|             |======|       |======| |======|
            Finding              Anatomy        Anatomy  Finding
           C0018681              C0005767       C0005767 C0460139
                                                |===============|
                                                     Finding
                                                     C0005823
                                          |====================|
                                                  Disorder
                                                  C0020538
```

Figure 3.2: Example of medical concept identification with cTAKES.

To fully and easily understand the mapping between the biomedical text and the UMLS Metathesaurus from the example, we list all the mappings in Table 3.13: the category information of the concepts[22] and their corresponding CUI codes[23] are listed out. For example, term *headaches* is identified as a semantic type of *Finding* and with a CUI code C0018681.

3.3 Related Work Review

This section reviews the important past work concerning applying query expansion and learning to rank approaches in HIR and CHIR. The prior work regarding to the understandability is also reviewed.

Regarding to the specific domain health IR, early in 1996, Srinivasan (1996)

[22] For category information, different MCI tools can adopt different names and classification, and this category is from cTAKES

[23] Medical concept *blood* is identified twice with same CUI code and we merge them into one in the table.

Table 3.13: Medical concepts identification.

Medical Concept	Category	CUI code
headaches	Finding	C0018681
blood	Anatomy	C0005767
pressure	Finding	C0460139
blood pressure	Finding	C0005823
high blood pressure	Disorder	C0020538

had experimented the query expansion approach on a MEDLINE data collection. Three QE strategies were tested to find suitable MeSH terms. The experimental results showed all three QE methods were able to improve the baseline. Among them, the combination QE approach presented the best of performance and achieved 17% improvement over the baseline; the relevance feedback achieved similar results and an improvement of 16.4% over the baseline; the thesaurus achieved 9.9% improvement. The paper proved that using expanded queries had a significant improvement over the un-expanded queries; moreover, using QE approach and adding MeSH terms was efficient to improve retrieval performance in HIR.

Many following work showed as well that using a controlled vocabulary to expand a query obviously improve the effectiveness of the retrieval system (Aronson and Rindflesch, 1997; Zhu and Carterette, 2012a,b; Song et al., 2015; Lopes and Ribeiro, 2016). This techniques is popularly used in HIR tasks (Palotti et al., 2015; Zuccon et al., 2016).

A more recent overview of CLEF eHealth IR task 2015 by Palotti et al. (2015) demonstrated that query expansion techniques played an important role in improving search effectiveness: the researchers stated that query expansion was found to often improve results when comparing different methods employed by the CLEF 2015 eHealth IR task participating teams and, the best result on that task was achieved using a query expansion method.

Query expansion with UMLS Metathesaurus is one prime method in consumer health search and has been shown to be effective in improving search effectiveness in this area (Aronson and Rindflesch, 1997; Lopes and Ribeiro, 2016; Zuccon et al., 2016).

Aronson and Rindflesch (1997) used MetaMap tool to find associated UMLS Metathesaurus concepts and expand them to the original queries. They compared their work to the relevance feedback QE approach experimented by Srinivasan (1996). The experimental results showed that UMLS Metathesaurus based QE approach achieved similar results the relevance feedback approach by Srinivasan. They also further pointed out that the combined techniques (UMLS Metathesaurus with relevance feedback) could achieve better performance.

Lopes and Ribeiro (2016) stated they implemented several query expansion strategies using various term resources and various techniques to select the terms to expand the original query. For term selection, they reported using Wikipedia articles and UMLS meta-thesaurus definitions as external resources; pseudo relevance feedback was also reported to be used as a query expansion technique. To re-rank the documents retrieved with expanded queries they used readability metrics.

Literature showed that compared with other kinds of query expansion techniques, MeSH was more effective and preferred. Related works (Zhu and Carterette, 2012a,b; Song et al., 2015) which used MeSH as the expansion resource were proved to obviously improve the effectiveness of the retrieval system.

Zhu and Carterette (2012b,a) showed that expanding queries with related terms improved retrieval performance significantly for medical records search; they used MeSH for query expansion. Song et al. (2015) explored an unique expansion approach based on mined results from Google. They first issued a query to Google and maintained the top 10 returned results; next, they used MeSH to match and find the medical terms; those terms were then added to the original query.

The overview paper of the 2015 CLEF eHealth IR task (Palotti et al., 2015) also demonstrated that most participant teams used query expansion techniques and MeSH was the most popularly used and preferred expansion resource.

Besides the widely used MeSH vocabulary, using OAC CHV to perform query expansion also showed to be effective in consumer health information search (Lopes and Ribeiro, 2016).

Despite amounts of papers have presented the effectiveness of using query expansion techniques with domain specific thesaurus, some paper have also observed disappointed results.

Lu et al. (2009) compared the different works concerning query expansion in the domain of biomedical text retrieval and pointed out that the results had been mixed. They pointed out that some papers showed that the techniques could result in improved retrieval performance, while on the other hand, other papers showed contradictory reports of using query expansion.

Other research works (Voorhees and Hersh, 2012; Shen and Nie, 2015) observed that expanding queries with synonyms improved performance for certain instances; it was not always effective of using thesaurus based query expansion techniques and the results could be mixed.

Voorhees and Hersh (2012) pointed out that top performing groups each used some sort of vocabulary normalization device specific to the medical domain, supporting the hypothesis that language use within electronic health records was sufficiently different from general use to warrant domain specific processing. However, he also pointed out that such devices must be used carefully as multiple groups also demonstrated that aggressive use harms baseline performance.

It was shown in some work (Darmoni et al., 2012; Shen and Nie, 2015) that UMLS concepts might or might not improve the performance of the medical information retrieval. Balaneshin-Kordan et al. (2015) proposed that only the concepts belonged to some specific kinds of semantic type could be included in the expanded query. And another paper also used these techniques, but did not clearly report the assessment results and the effectiveness about these techniques (Lopes and Ribeiro, 2016). Although the work (Song et al., 2015) obtained the highest effectiveness among the submissions for CLEF 2015 eHealth IR task (Palotti et al., 2015). However, this unique technique was based on a commercial search engine and not easy or appropriate to be generalized for a system.

Applying pre-trained word embeddings as a query expansion resource have been proved to be efficient in CHIR (Wang et al., 2015; Oh and Jung, 2016; Budaher et al., 2016; De Vine et al., 2014).

Wang et al. (2015) constructed a medical concept space for medical synonyms extraction. The model was trained on manually extracted medical knowledge and a corpus with Word2vec tools. The training corpus incorporated a set of Wikipedia articles, MEDLINE abstracts, as well as the source from around 20 medical journals and books. They concluded that their proposed model outperforms the baseline approaches by a large margin on a dataset with more than one million term pairs. Although they did not explore the use of query expansion, their work could be regarded as a pre-work for query expansion; they aimed to extract medical synonyms which could be used to find query term synonyms and expand the the query.

Oh and Jung (2016) constructed a word vector model from medical Wikipedia with word2vec tools aiming to use the model to properly understand the information need of a query. They used word vectors in two different ways: they explored the model to compute the relevance scores between a query and a document and to find more related words adding them to the original query.

Budaher et al. (2016) researched the effectiveness of word embeddings for query expansion in the health domain. They experimented on two different training corpora using the normalized cosine value to measure the similarity between two words aiming to find the k-most similar words of a specific word; then the expanded query is obtained adding them to the original query. No clear effectiveness results are reported for techniques explored in the above two works.

De Vine et al. (2014) empirically demonstrated that a word embeddings model built with two medical corpora outperforms a number of state-of-the-art benchmarks for medical semantic similarity.

Besides using UMLS Metathesaurus and pre-trained word embeddings, a number of works also explored the use of some other query expansion techniques (Luo and Tang, 2008; Zeng et al., 2006; Shen and Nie, 2015; Soldaini et al., 2016; Seung-Hyeon Jo, 2016).

A search system iMed (Luo and Tang, 2008) provided two suggestions of expanding medical phrases to assist users in refining their queries, the collection of crawled web pages, and the query.

Shen and Nie (2015) proposed a method of query expansion with mutual information. Two concepts were considered to be related if they co-occurred frequently. They found related concepts using concept co-occurrences. The original query was expanded by the top mutual information concepts.

Recently, Soldaini et al. (2016) presented a query clarification approach aiming at improving medical IR by laypeople. Query clarification was an another form of expansion, where the most appropriate expression or the most similar expert expression is added to the query. They showed that users are more satisfied with the results using this approach.

Seung-Hyeon Jo (2016) proposed a query expansion method by building a clinical semantic knowledge. It was built by using the medical terms obtained from UMLS as well as Wikipedia documents. During the query expansion process, associated terms were selected from the knowledge. They carried out their approaches on TREC Clinical Decision Support track 2015 and the proposed methods achieved 0.2327 and 0.3033 in the inferred NDCG on Task A and Task B, respectively. However, comparison to the baselines or other teams were not clearly reported in this paper.

During the past decade, learning to rank technique has shown its effectiveness in solving the ranking problem in IR area. Recently, learning to rank has also gain great interest of researchers in health IR area (Roberts et al., 2015; Palotti et al., 2015; Zuccon et al., 2016).

Next, we specifically reviewed all the works that apply learning to rank techniques in CLEF eHealth IR task. We reviewed all the approaches that use learning to rank techniques in their participating for these tasks from 2015 to 2018 (Song et al., 2015; Thuma et al., 2015; Palotti et al., 2015, 2016; Wang et al., 2016a; Soldaini and Goharian, 2017; Scells et al., 2017; Palotti and Rekabsaz, 2018).

In 2015 CLEF eHealth IR task, two teams explored learning to rank techniques in their work: one explored combing scores and ranks from BM25, PL2 and BB2 into a six-dimensional vector (Song et al., 2015), the results show that this method under-performed query expansion techniques explored by the same team; another team investigated learning to rank along a Markov Random Fields approach (Thuma et al., 2015), no clear results was reported in the paper. Palotti et al. (2015) concluded that learning to rank techniques did not work as well as query expansion techniques.

Later, Palotti et al. (2016) researched on the effectiveness of a learning to rank method which exploited retrieval features as well as readability features. This was a continued research work on CLEF 2015 data collection. They used standard weighting models features for learning topical relevance, and additional features based on readability measures and medical lexical aspects to learn understandability; they concluded that the combination of retrieval features and readability features improved search engine results.

In 2016 CLEF eHealth IR task, one group proposed a learning-to-rank algorithm to re-rank the result (Wang et al., 2016a), no clear result evaluation was reported in the paper. Soldaini and Goharian (2017) proposed a combination of statistical and semantic features to train a learning to rank model, and their methods were tested on CLEF eHealth 2016 dataset; the results showed that their approach outperforms the best baseline approach by 26.6%.

In 2017 CLEF eHealth IR task, Scells et al. (2017) proved that the use of the PICO-based feature within learning to rank provides improvements over the use of baseline features alone.

In 2018 CLEF eHealth IR task, Palotti and Rekabsaz (2018) performed a personalized retrieval in a learning to rank setting and concluded that using learning to rank technique did not always increase the baseline.

From the above discussion, we can see that although learning to rank methods have shown its success in information retrieval, their performance in health IR is not quite clear and more research work is needed to prove its usefulness in this specific IR area.

Recently, some work investigated the use of understandability to improve search performance.

Collins-Thompson et al. (2011) argued that web search engines were not efficient in addressing childrens information need. They pointed out that many important factors might affect the relevance of the retrieved results; however, only topicality relevance was typically considered. They further stated that reading level was valuable for the information search. Models and algorithms were proposed to address the problem: consumers reading proficiency were estimated; reading level of the retrieved results were estimated; documents were re-ranked based on the reading difference between the consumers and the retrieved documents. Their proposals were evaluated on a large amount of Web queries and the logs were analysis. Their findings proved that providing consumers with personalized search results at their reading levels was meaningful.

Yilmaz et al. (2014) proposed a user model taking into account topicality relevance, user dwell time, user judging time, etc. The model showed that the judgments of the document utility relied on the efforts paid by the users. Readability of a document was one of the factors and was highly important. They argued that these effort factors including readability features should be considered as the assessing behaviours.

Palotti et al. (2016) experimented on improving topical ranking performance as well as document understandability in consumer health search with learning to rank techniques. In their work, they defined four groups of readability features: the first group used the traditional formulas including Coleman Liau, Dale-Chall Score, Flesch Kincaid Grade, Gunning Fog and SMOG; the second group uses the surface measures such as the number of characters, words and sentences; the third group used general vocabulary related features such as stop words information; and the fourth one concerns the medical vocabulary related features such as consumer health vocabulary information.

3.4 HIR Campaigns

CLEF eHealth Challenge

CLEF eHealth[24] is an evaluation challenge in the medical and biomedical domain, with the goal to provide researchers with datasets and evaluation frameworks. Since 2013, CLEF eHealth has been running annual evaluation campaigns in these domains: information retrieval, information extraction (IE) and Information management.

[24] https://sites.google.com/site/clefehealth/

TREC PM/CDS Track

TREC[25] (Text REtrieval Conference) Precision Medicine and Clinical Decision Support Track[26]is a biomedical challenges organized in TREC.

The focus of the PM (Precision Medicine) task (held from 2017-2018) is to provide useful precision medicine-related information to clinicians treating cancer patients. The focus of the CDS (Clinical Decision Support) Track (held from 2014 to 2016) is the retrieval of biomedical articles relevant for answering generic clinical questions about medical records.

BioASQ Challenges

The BioASQ challenge organizes biomedical semantic challenges which comprises the following two tasks: one is on large-scale online biomedical semantic indexing and the participants are asked to classify new PubMed documents, before PubMed curators annotate them manually; the other task is on biomedical semantic question answering (QA).

[25] https://trec.nist.gov/

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

Chapter 4

A Compound System for Consumer Health Information Retrieval

As discussed in Chapter 1, the main research goal is to improve the performance in retrieving both relevant and understandable documents for a consumer in the scenario of Consumer Health Information Retrieval. And in turn, two research questions are proposed. In Chapter 2 and 3, the related state-of-the-art techniques in IR, HIR and CHIR are discussed; related work is reviewed. In this chapter, the overall proposal is presented; two modules are proposed with each addresses one research question proposed in Chapter 1. Each module is presented in details; corresponding techniques and their improvements over the state-of-the-art are thoroughly discussed.

4.1 System Architecture

This research work aims to improve state-of-the-art techniques in information retrieval and apply them to consumer health information search. The overall architecture of the system is presented in Figure 4.1[1]. Queries are first preprocessed and further processed with Module.1 which is proposed to solve the first research question. Using the improved queries obtained from Module.1 and the indexed documents as the input, a ranking list is produced. Then in Module.2, which is proposed to solve the second research question, past

[1] This figure is presented as a complete IR system for the integrity. Same modules shared with Figure 2.2 are not detailed illustrated in texts in this section, readers are recommended to refer back to section 2.1 for these modules.

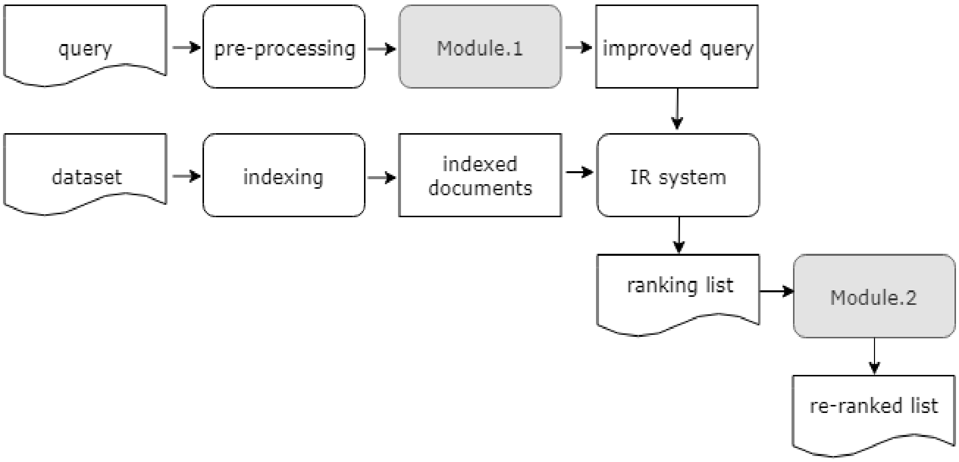


Figure 4.1: Overall system architecture.

data are used to improve the performance in retrieving more understandable documents to users and the re-ranked list is obtained.

4.2 Bridging the Language Gap

Module.1 is proposed to answer the first research question proposed in Chapter 1: how to bridge the language gap between non-expert consumers and medical professionals? The methods applied within Module.1 can be generally explained as: integrating Natural Language Processing techniques into the state-of-the-art information retrieval approaches.

Medical language has its own specificities and this characteristic should be paid enough attention when using an universal IR system to process health queries. On one hand, when searching for health related information online, a consumer without or with limited medical knowledge background usually presents his information need with lay words. On the other hand, the health texts are more professional texts which contain medical terminology or concepts.

As discussed in previous chapters, modern information retrieval algorithms are based on term matching. The basic theory is to check whether or not a query term is present within a document. What happens when the same concept is expressed using different terms? For example, the consumer query term “heart attack” may not appear within a document, but instead, “cardiopathy” does, which is a more professional term referring to the same concept. Based on the theory of the IR retrieval model, this related document will not be retrieved since it does not contain the term “heart attack”.

To solve the problem caused by the language gap, one straightforward solution may be to measure the reading level of both the users and the documents; then recommending suitable reading level documents to the users. However, this solution is not currently feasible in CHIR area, because:

- The consumers’ personal information are usually private and not available.
- Defining the reading level of large medical documents, especially the on-line ones, is not practical with existing methodologies and techniques.

Query expansion, an approach which can add new and related words to a query term, provides a feasible way to this research question.

So, our solution in answering the first research question is proposed as: applying query expansion approach to bridge the language gap.

The typical process when applying the query expansion approach in HIR or CHIR is as follows: step 1, a query is pre-processed; step 2, the medical terms are identified with a medical identification tool; step 3, expanding resources are used to find new words to the identified query terms; step 4, the new words are added to the query and a new query is built.

In this work, we improve the state-of-the-art query expansion approach by advancing its processing procedures and integrating the NLP techniques into it.

Different from previous works, we propose a Medical Concept Model (MCM) to further processing a query before using an expansion resource to find new words to query terms. And for newly added words, we take extra processing on the selected words rather than simply adding them into the query. Moreover, locally trained word embedding models serving as the query expansion resources are also researched in Module.1. Classically and traditionally, an existing medical thesaurus such as the UMLS Metathesaurus is used as the resource for expanding new words to the query terms. Recently, a word embedding model trained over a collection of documents can also be applied to serve as a resource for query expansion.

In the following sections, we first discuss the proposed MCM model and then the trained word embedding models as the query expansion resources in CHIR.

4.2.1 A Medical Concept Model

Considering the characteristic of medical language, we assume that:

1. Each query term does not contribute equally to a query when searching relevant documents.
2. Phrases are more effective than single, separate terms when finding relevant documents.

Based on these two assumptions, we propose a Medical Concept Model which aims to improve the state-of-the-art techniques adopted in processing a query.

Terms of the MCM model

The terms used in the MCM model are introduced and defined in this section, including medical phrase concept C_p, medical term concept C_t, loose phrase, weights and the ineffective type list.

Medical phrase concept. A medical phrase concept is a phrase which is identified as a UMLS medical concept by the medical concept identification tool (such as MetaMap or cTAKES) from an original query. A medical phrase concept consists of no less than two terms.

We denote a medical phrase concept as C_p, which is composed of n terms:

$$(\mathrm{t_1}, \mathrm{t_2}, ..., \mathrm{t_n})(n >= 2)$$

Medical term concept. A medical term concept is a single term identified as a UMLS medical concept and is noted as C_t.

Loose phrase. The human language demonstrates its diversity and flexibility by the possibility of expressing the same idea in different ways. Taking this into account, an undemanding phrase is introduced and denoted as *loose phrase.* In a loose phrase, a maximum number of words between two terms is allowed.

Based on this, a medical phrase concept $\mathrm{C_p}$ can be reconstructed into a loose phrase allowing a maximum number of words within its' two terms inside. The reconstructed medical phrase is noted as:

$$(\mathrm{t}_1, \mathrm{t}_{12}^1, \mathrm{t}_{12}^2, ..., \mathrm{t}_{12}^m, \mathrm{t}_2, \mathrm{t}_{23}^1, ..., \mathrm{t}_{(n-1)n}^m, \mathrm{t}_\mathrm{n})(\mathrm{n} >= 2; \mathrm{m} >= 1)$$

where, m is the number of terms allowed between two terms in a medical phrase concept.

Weights. Since we assume that each query term or phrase doesn't contribute equally, we propose to increase the weights to the terms or phrases which are supposed to contribute more to a query. We note the weights assigned to a term and a phrase as w_t and w_p, respectively.

Ineffective type list. As discussed in previous chapters, the UMLS concepts can be classified as different types. For example, 16 categories are defined in the MeSH Headings and 21 in the OAC CHV. In practice, regarding to different data collections and task requirements, not all types are equally useful. We define an ineffective type list which includes the types which are deemed to be no help in a task; all concepts identified with these types are to be discarded. For example, in a typical consumer health search task, "Procedure" or "Finding" types usually are not useful for consumers in seeking advice on symptoms, diagnosis or treatments and may affect the retrieval performance.

Query processing using the MCM model

Next, we discuss the query processing using the MCM model. This MCM model improves the state-of-the art query expansion techniques. The general process of using this model can be divided in three stages: medical concepts classification, medical concept processing; and new query built.

Medical concepts classification. In a query, a single term or a phrase can potentially express a medical concept and existing medical concepts identification tools can be used to identify such concepts in a plain text.

The processing taken during this stage is presented in Figure 4.2. First, the medical concepts identified by the medical concept identification tool are kept and non-identified terms are discarded. Then, for all the identified concepts, the decision to discard or maintain them was taken. An ineffective type list is used. If a medical concept is classified as one of the type in the list, this concept is discarded. Finally, as noted above in assumption two, a phrase concept and a term concept contribute differently, so a further classification in one of these two types of concepts was done. This was done in order to apply different techniques in the following procedures.

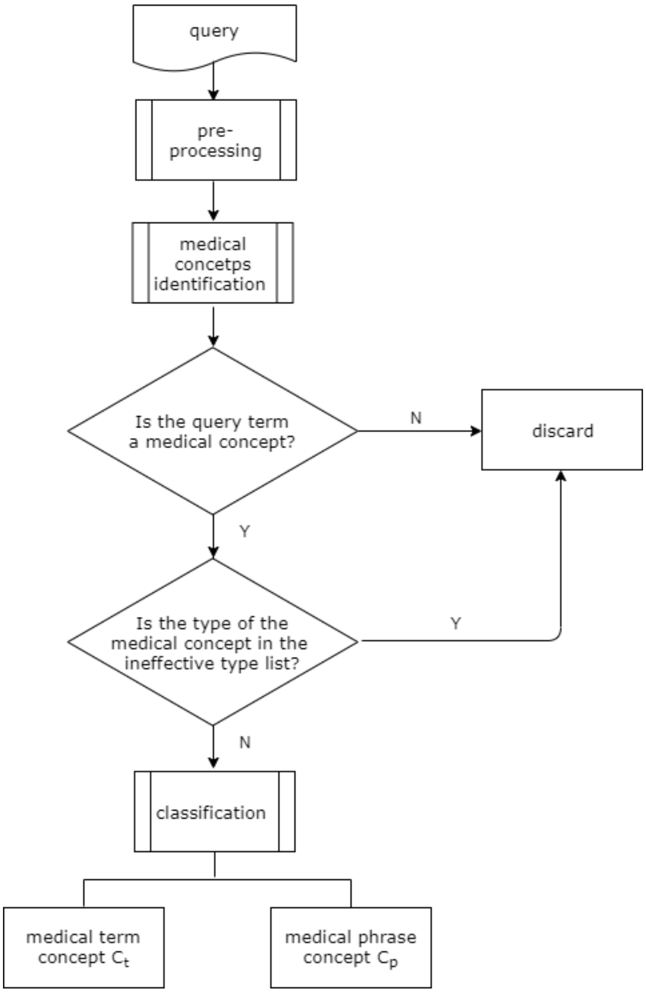

query
pre-processing
medical concetps identification
Is the query term a medical concept?
N
discard
Y
Is the type of the medical concept in the ineffective type list?
Y
N
classification
medical term concept C_t
medical phrase concept C_p

Figure 4.2: Medical Concept Model: medical concepts classification.

Medical Concepts Processing. The classified medical concepts $\mathrm{C_t}$ and $\mathrm{C_p}$ obtained from the first stage are further processed using different techniques, respectively.

Medical Term Concepts. The processing of medical term concepts $\mathrm{C_t}$ mainly include two techniques. One is assigning extra weight to a $\mathrm{C_t}$. As noted in assumption one, query terms do not contribute equally to a query. Thus, for a term identified as a concept, its contribution is deemed to be higher and an extra weight is assigned to that term. We note an weighted term concept as $\mathrm{w_tC_t}$.

Another technique is using query expansion resources to find related words to a $\mathrm{C_t}$ and we note such a word as:

$$\mathrm{C_tE_i(i = 1,...,n)}$$

where, n is the number of words expanded from term $\mathrm{C_t}$[2].

Medical Phrase Concepts. Three techniques are mainly used on medical phrase concepts $\mathrm{C_p}$. As we do with term concepts, one technique is assigning extra weight to a $\mathrm{C_p}$. We note an weighted phrase concept as $\mathrm{w_pC_p}$.

Similarly, the second techniques used is query expansion and we note an added word from a phrase concept $\mathrm{C_p}$ as:

$$\mathrm{C_pE_i(i = 1,...,n)}$$

The third techniques we use on phrase concept is reconstructing a loose phrase based on a $\mathrm{C_p}$, which can be noted as $\mathrm{C_pL(i)}$, where, i is the maximum number of words allowed inside a loose phrase.

New query built. Finally, the processed concepts and expanded words are added to the original query. Ideally, an expanded query can be like this if we include all words in:

$$\{\mathrm{Q_o, C_t, C_p, w_tC_t, C_tE_i, w_pC_p, C_pE_i, C_pL(i)}\}$$

where, $\mathrm{Q_o}$ is the original query.

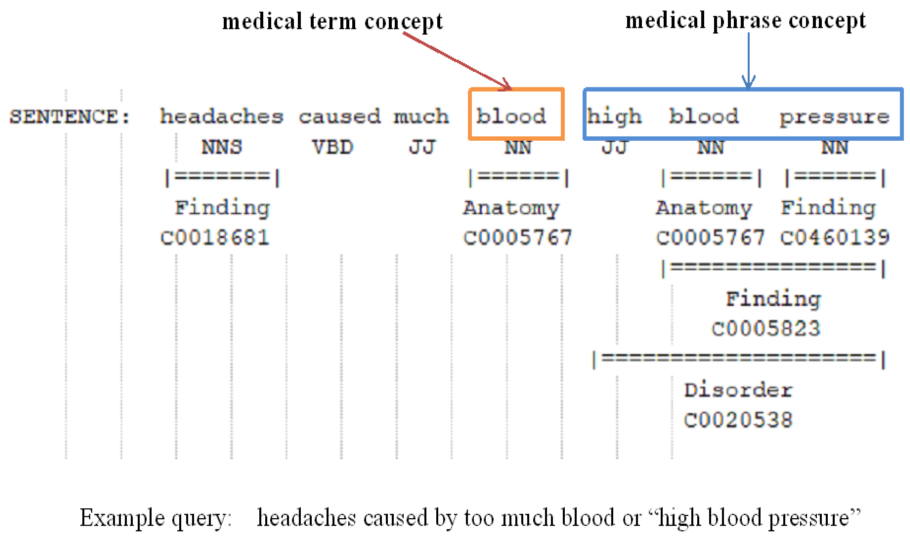
In practice, we may define different strategies on expanded words chose regarding to the specific usage concerning different data collection. For example, a phrase concept $\mathrm{C_p}$ is usually assumed to definitely contribute to the query and in this case we can compulsorily process $\mathrm{C_p}$ as a must check item during the retrieval process.

A Concrete Example of the MCM model

To better understand the techniques adopted in the medical concept model, Figure 4.3 demonstrates a concrete example.

In this example, a query is issued by a consumer: headaches caused by too much blood or "high blood pressure". Following the medical concept model

[2] The number of added words from a terms is determined by the pre-set threshold.



```
SENTENCE:  headaches  caused  much  blood   high  blood    pressure
              NNS       VBD    JJ     NN     JJ     NN        NN
           |=======|                |======|     |======|  |======|
            Finding                 Anatomy      Anatomy   Finding
           C0018681                 C0005767     C0005767  C0460139
                                                 |===============|
                                                      Finding
                                                      C0005823
                                            |====================|
                                                  Disorder
                                                  C0020538
```

Example query: headaches caused by too much blood or “high blood pressure”

Figure 4.3: A concrete example of Medical Concept Model.

proposed, first the query is pre-processed where the stop words (by, too, or) and the quotation marks are removed. Then a medical NLP tool is used to identify the concepts inside the query. “Finding” concepts are discarded; “blood” is identified as a medical term concept and “high blood pressure” as a medical phrase concept. Finally, term concept “blood” is assigned an extra weight and phrase concept “high blood pressure” is reorganized into a loose phrase.

Improved usage of the UMLS Metathesaurus

In the area of health information retrieval, a traditional and popular way is to use the controlled and manually maintained UMLS Metathesaurus.

When applying the UMLS Metathesaurus to find new words to a query term, we propose to built a local dictionary of CUIs and their synonyms. The expanding and selection strategies are explained in the pseudo code displayed in Figure 4.4.

To fully understand the idea, we apply it on an example query “what causes strong headaches at base of skull, stops with blood donation”.

```
                    Algorithms for the improved UMLS query expansion
1. Identify the CUIs for the corresponding medical phrase concepts
   or term concepts
2. If the concept overlaps with others, preserve the concept with
   the largest length and discard others
3. Remove duplicate CUIs
4. Use the CUIs to find synonymous from the UMLS Metathesaurus
5. Build a local dictionary for the CUIs and its synonymous
6. Optimize the local dictionary
7. Use this local dictionary to add new words to the query
8. Assign extra weight to synonyms expanded from a phrase concept
in comparison to the synonyms expanded from a term concept
```

Figure 4.4: Improved techniques using UMLS as a query expansion resource.

The expanded query is shown in Figure 4.5. First, we use the medical concepts model to process the query, then, the procedures presented in Figure 4.4 is taken and the UMLS Metathesaurus is used as the expanding resource.

```
Original query:
what causes strong headaches at base of skull, stops with blood
donation

Expanded query:
(using Medical Concept Model and UMLS thesaurus)
causes strong headaches base skull stops blood donation
"base skull"~m blood^$w_t$ cranium cranial skull skulls skulling^$w_p$

Parameters:
$m$: maximum number of words allowed in a loose phrase
$w_t$: weight assigned to a medical term concept
$w_p$: weight assigned to words expanded from medical phrase concept
$w_t$<$w_p$
```

Figure 4.5: An example using improved UMLS query expansion techniques.

4.2.2 Word Embeddings as the Query Expansion Resource

Medical terms are identified and then classified as term concepts or phrase concepts using the techniques proposed in the MCM Model. The following step is to find related words to these concepts and then add the new words to the original query. Apparently different yet semantically related words can refer to the same concept. Serving as a part of query expansion process, how to find effective expanding words plays an important role.

The quality and quantity of the added words can be determined by two factors:

- The expansion resource or vocabulary;
- The techniques used in the expanding ways such as word choosing.

These two factors will affect what kinds of words are to be found as related and be expanded to an original query, which in the end will cause the matching with a document.

In recent years, word vector representations have attracted a lot of attentions. In a word vector representations model, similar or related in meaning words tend to be represented by similar vectors, and such a model can be used as a resource to identify synonyms or related words for a query term to be used on query expansion techniques. Recently, word vector representation models trained with neural network has been researched and employed in the area of CHIR. Locally trained word vectors have been shown to carry semantic meanings. This finding can be used to identify related terms and therefore provide a feasible way to perform query expansion.

Besides applying the state-of-the art UMLS Metathesaurus as the expansion resource, in this work, we also explore the usage of word embedding models to expand new words to query terms in the area of CHIR.

Our next sections discuss the improved techniques proposed on using the word embedding model as the expanding resources in details.

Word Embedding Model Training. Some pre-trained word embedding models are available for HIR research, such as the BioASQ word2vec tool, $\mathrm{WE}_{\text{trec-cbow}}$ and $\mathrm{WE}_{\text{trec-skip}}$ tools discussed in previous Chapter 3.

In this work, we propose to train word embedding models that can be specially applied in the area of CHIR. As shown in Figure 4.6, two models are proposed to be trained: one using the Wikipedia data and the other one using the medical data from PubMed.

These models trained on different resources can then be used to assess how much the training resource affects the expanding words and, in turn, the retrieval performance in the scenario of consumer health search.

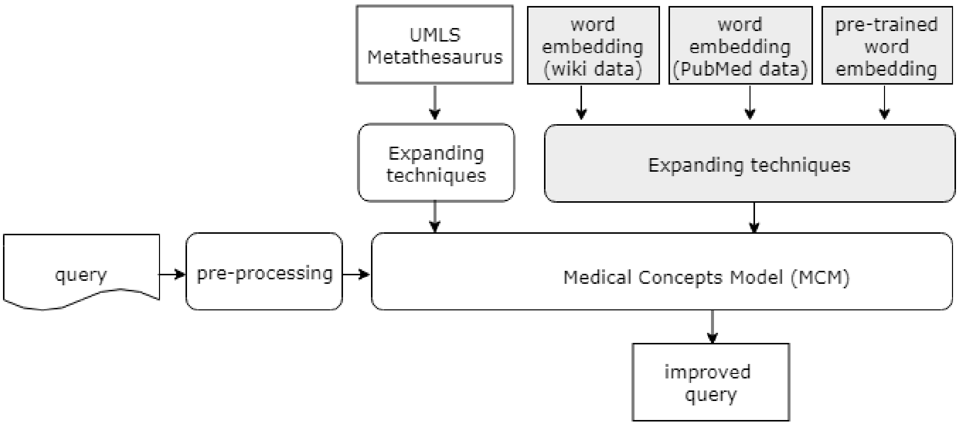

UMLS Metathesaurus
word embedding (wiki data)
word embedding (PubMed data)
pre-trained word embedding
Expanding techniques
Expanding techniques
query
pre-processing
Medical Concepts Model (MCM)
improved query

Figure 4.6: Word embedding model as the query expansion resource.

Word Embedding Model Use. The common way of exploring a trained word embedding model is to find synonyms or related words to a term. For an identified medical term concept $\mathrm{C_t}$ or a phrase concept $\mathrm{C_p}$, its expansion words can be noted as:

$$\mathrm{CE_i}(i = 1, ..., n)$$

where n is the number of words expanded from a concept C.

Then, the added words can be expanded to the original query.

Besides using the word embedding model to find related words, this research work also explores in finding the most related terms inside a query. The most related terms in a query will more likely reflect the user's need compared to other terms.

The process is as follows: first, we process to return the top-N most related terms in a query by comparing the vector similarity between terms; next, the most related terms are reconstructed into a *loose phrase* where a number of interval words within the phrases is allowed; finally, this loose phrase is added to the original query and usually regarded as a must check item during the retrieval process.

Supposing we get a return of two-most related terms: t_i, t_j. Then the loose phrase built from this pair can be noted as:

$$(\mathrm{t_i}, \mathrm{t}_{ij}^1, \mathrm{t}_{ij}^2, ..., \mathrm{t}_{ij}^m, \mathrm{t_j})(\mathrm{i,j} = 1,...,\mathrm{n}; \mathrm{m} >= 1))$$

where n is the number of terms in a query and m is the maximum of words allowed between two terms.

4.2.3 Improved IR Model for Bridging the Language Gap

Based on the above ideas, the improved system model is proposed to solve the first research question, as presented in Figure 4.7[3]. The processing of the queries using this improved model is as follows[4]: (i) the original query is pre-processed; (ii) the medical terms are identified and classified as term or phrase concepts; (iii) new words are expanded from the query expansion resources; (iv) different processing are taken on term concepts and phrase concepts, respectively; (v) the expanded query is presented to the retrieval platform for the first round of retrieval that, in turn, using a retrieval model returns an initial ranked list of documents; (vi) the classic pseudo relevance feedback method is used to do the second round of retrieval and an improved ranking list is obtained.

4.3 Learning Understandability from Experience

Module.2 is proposed to answer the second research question proposed in Chapter 1: how to learn understandability from experience? As discussed in previous chapters, during the past years, LETOR approach, which applies machine learning techniques on ranking problems, has been successfully used to improve retrieval performance in the area of CHIR. Meanwhile, valuable data including queries, data corpus and worthy assessments by human beings are available from the CHIR area and can be well used. Thus, our solution in solving the second research question is proposed as: in order to improve the performance of understandability ranking in CHIR, we take use of the past data and train LETOR models on them; these trained models are then used to re-rank the documents.

When applying LETOR techniques, creating a feature list is an important task. Traditionally, explored potential features are defined and blindly combined all together to create a feature list which is used to train a LETOR model afterwards. Vast amount of work has been researched on digging new features which is a costly job in Learning to Rank.

Compared with this research line aiming to finding new features, there has been little work on studying how to take good use of the explored features, for instance, the classification of features based on field information.

[3] Modules presented with grey background are the main contributions in improving the state-of-the-art techniques.

[4] This figure is presented as a complete IR system for the integrity. Same modules shared with Figure 2.2 are not detailed illustrated in texts in this section, readers are recommended to refer back to section 2.1 for these modules.

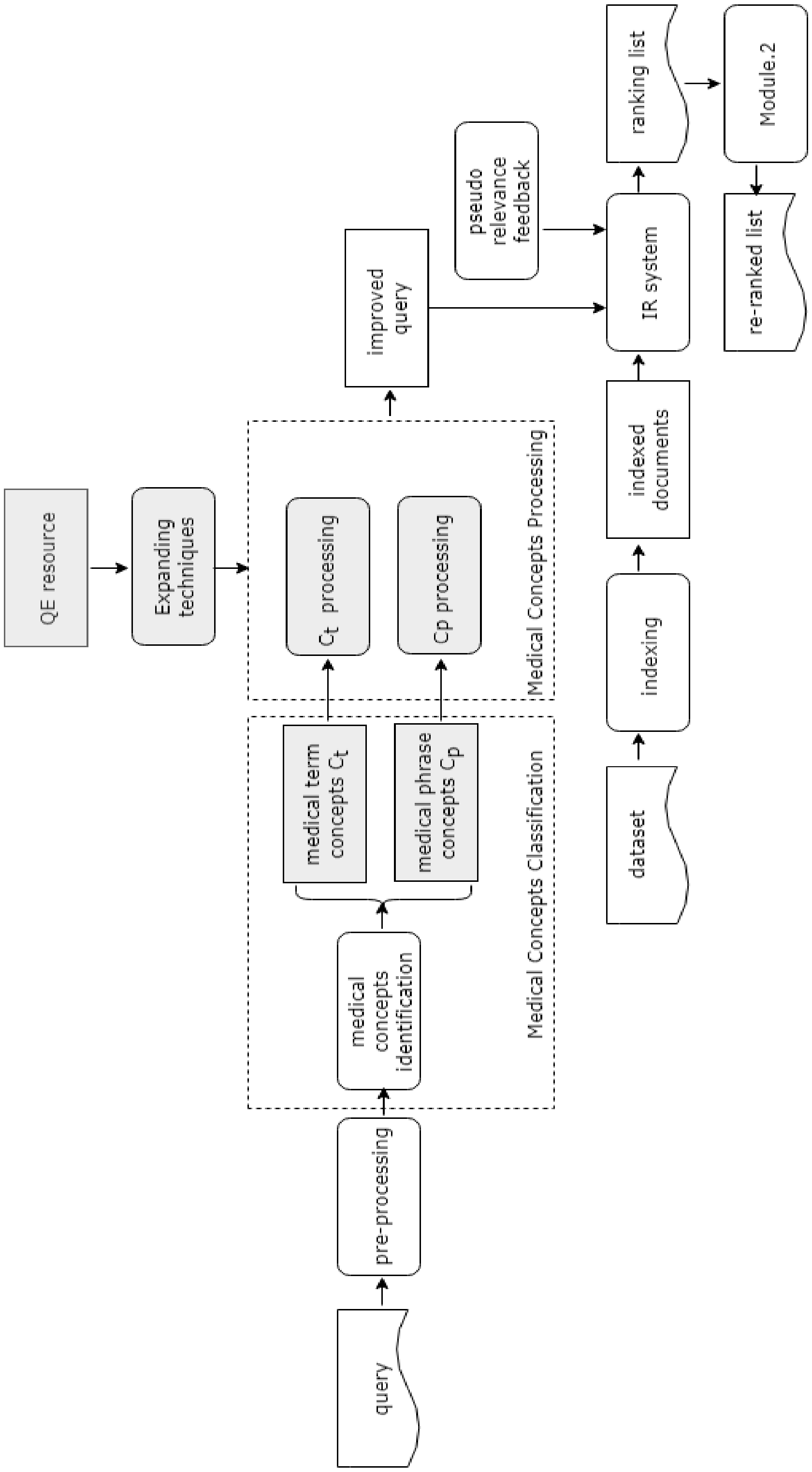


Figure 4.7: An improved system model for bridging the language gap.

In this work, we study this challenge by improving state-of-the-art LETOR techniques in the area of CHIR. We propose to use rank aggregation methods on field-based LETOR models. The methods applied in Module.2 can be generally explained as: introducing single field-based LETOR models and applying rank aggregation methods on these field-based LETOR models.

Since a document field (e.g.: title, H1 or body) contributes differently to this document, we assume that:

- Blind combination of features extracted from different fields into one single feature list will make the features blur.
- An aggregated model over a set of pre-trained field-based models is more effective than a model trained with features which are blindly joined together from different fields.

Based on the above ideas, a two stage LETOR framework is proposed, as presented in Figure 4.8[5]:

- During the first stage, the defined features are grouped and a set of field-based LETOR models are generated with each model trained using one group of single field features. Meanwhile, following the traditional way, one LETOR model is trained using all blindly joined features together.
- During the second stage, the scores obtained from the pre-trained field-based models are aggregated employing various aggregation methods.

4.3.1 Field-based LETOR models

In this work, LETOR techniques are explored to research on learning understandability from experience. Documents are annotated with labels referring to relevance or understandability regarding to a query. These evaluation labels can then be used to learn a ranking model.

Inspired by work from Macdonald et al. (2013) and following a similar feature exploring way as presented in LETOR Benchmark dataset (Qin and Liu, 2013), we propose:

- Extract features from different fields of a document and group the featured based on the field they derive from.

[5] Same modules shared with Figure 2.2 are not detailed illustrated in texts in this section, readers are recommended to refer back to section 2.1 for the same modules.

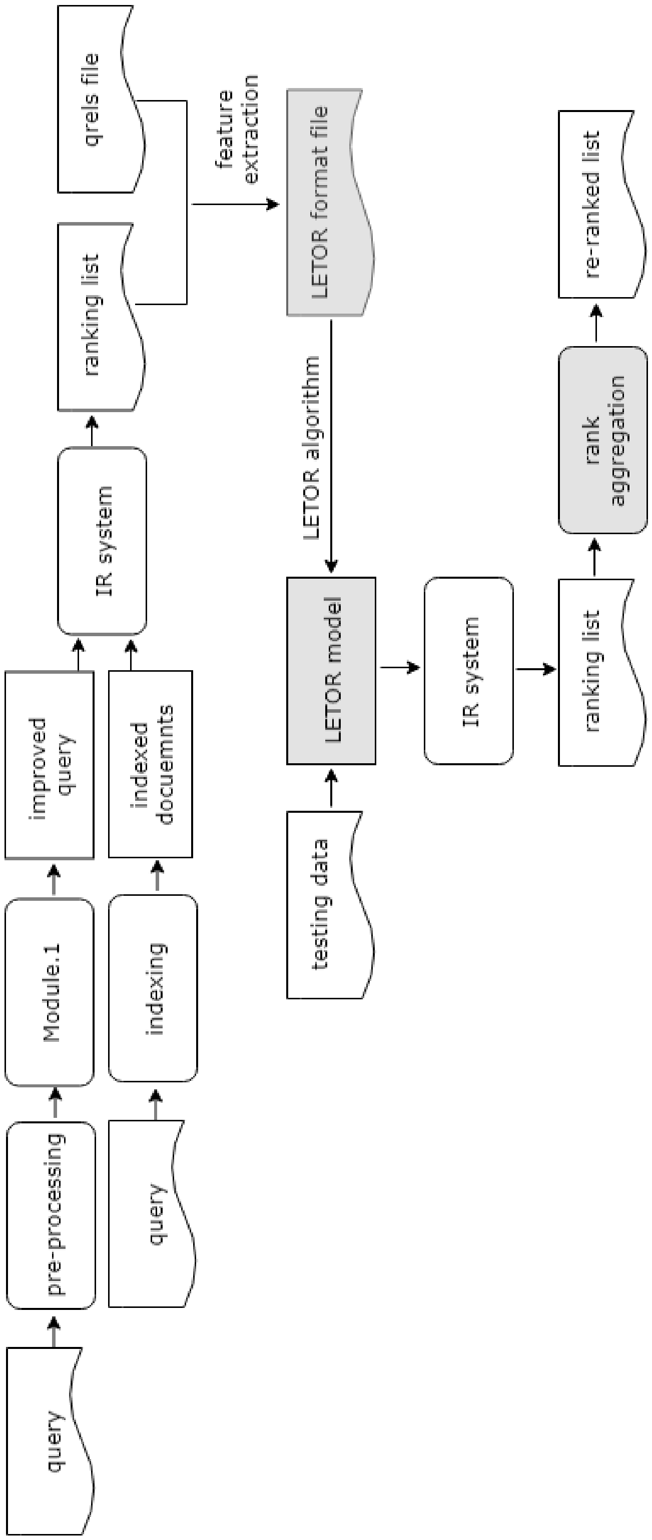


Figure 4.8: Rank aggregation on field-based LETOR models.

Table 4.1: Fields of a HTML document used to extract LETOR features.

Field	**Description**
H1-H6	Section headings at different levels; H1 is the highest-level heading and H6 is the lowest-level.
Title	A document title.
Header	Defines a header for a document or section.
Meta	Meta data of a document such as author, publication date, keywords etc.
Anchor	Anchor a URL to some text on a web page.
Body	Body content of a document.
Else	Not define in any field
Whole	The contents of full document.

- Train a set of LETOR models with each model using one single-field features.

Supposing a query q is composed of n terms:

$$\{\mathrm{t_1, t_2, ..., t_n}\}$$

The field of a document f contains these fields:

$$\mathrm{f_1, f_2, ..., f_m}$$

And we note the the score of document d regarding to this query q as:

$$\mathrm{score(q,d)}$$

If only one single field is taken into account when retrieval with an IR weighting model, the *score(q,d)* can be defined as :

$$\mathrm{score(q,d)} = \sum_{t=1}^{n} \mathrm{w_{t,d}(f)}$$

where $\mathrm{w_{t,d}(f)}$ is the query term t weighted in the field f of the document d.

As shown in Table 4.1, we list some essential fields of a HTML format document that we can take use of to extract field-based features.

As shown in Figure 4.9, the process of learning a field-based LETOR model is as follows: first, a number of IR weighting models are used to extract

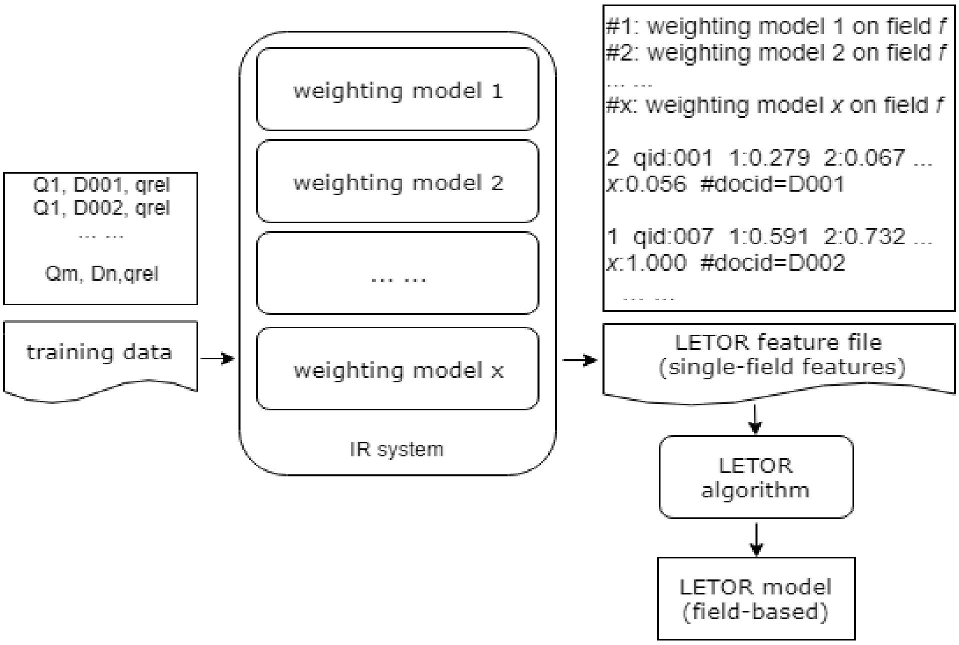


Figure 4.9: Train a field-based LETOR model with single-field features.

features only from one single field f; then, a format LETOR feature file which is composed of these single-field features are used to train a field-based LETOR model.

Likewise, suppose we define m (from f_1 to f_m) fields during the indexing process, then we can train a number of m field-based LETOR models with each learned from f_1 to f_m field features, respectively.

4.3.2 Rank Aggregation on Field-based LETOR Models

During the first phase, a set of field-based LETOR model are trained with each using one group of single field features. Next, we apply rank aggregation methods on these trained models.

For a query q and a document d, the score of document d regarding to a query q with a field-based LETOR model can be noted as:

$$\mathrm{score(q,d)_{f_m}}$$

where f_m represents a single-field that a model is trained on.

And we define rank aggregation algorithm as a function F[x], then the final score of a document d can be noted as :

$$\mathrm{score(q,d)} = \mathrm{F[score(q,d)_{f_1}, score(q,d)_{f_2}, ..., score(q,d)_{f_m}]}$$

where a number of m fields are used in training LETOR models.

Our idea can be concretely illustrated using Figure 4.10: first, each field-based LETOR model (using field features from f_1 to f_m) is applied to perform a retrieval in the IR system, and a ranking list is produced for each model; then, rank aggregation method is used to combine the results obtained from each field-based model, and a final ranking list is returned.

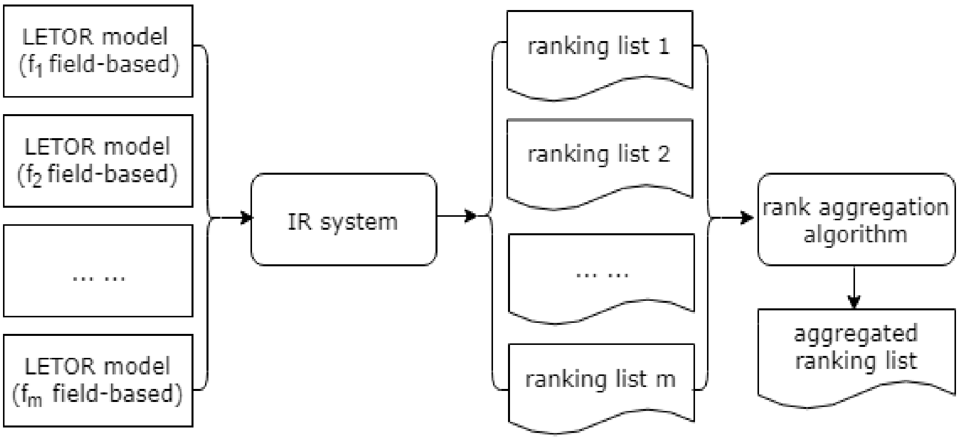

LETOR model (f1 field-based)
LETOR model (f2 field-based)
... ...
LETOR model (fm field-based)
IR system
ranking list 1
ranking list 2
... ...
ranking list m
rank aggregation algorithm
aggregated ranking list

Figure 4.10: Aggregation over field-based LETOR models.

4.4 Summary

In this chapter, we proposed our solutions to answer the two research questions proposed in Chapter 1. Our system is built over a classic IR system. A number of new modules were added and improvements were made compared to the state-of-the-art techniques. These improvements mainly concerned with query expansion and learning to rank techniques applied in the area of CHIR.

Concerning query expansion techniques, in Module.1, we proposed to integrate NLP techniques into the state-of-the-art query expansion approach. A medical concept model was proposed to process a query. The techniques of using the UMLS Metathesaurus as the expansion resources were improved.

Word embedding models were explored as the query expansion resource beyond classic UMLS Metathesaurus; they were proposed to be trained over different corpora and with neural network tools.

Concerning learning to rank techniques, in the Module.2, we proposed to employ rank aggregation method over the state-of-the-art LETOR techniques. First, field-based LETOR models were proposed to be trained; we proposed to classify learning features based on the different field they were derived from; a set of single-field based LETOR models were to be learned. Second, we proposed to use rank aggregation methods to combine the retrieval results obtained from each single-field based models.

Based on the methods proposed in this chapter, next two chapters present the experiments performed to test the validation of the proposals.

Chapter 5

Bridging the Language Gap

In the previous chapter, the proposed approaches are presented. Corresponding experiments are performed to test the validation of the proposals. This chapter first presents the experimental setup, then thoroughly discusses the experiments carried out in Module.1.

5.1 Experimental Setup

In this section, we present the experimental setup in our work. First, we talk about the four IR data collections used in our experiments: one from FIRE CHIS and three from CLEF eHealth; next, we introduce the pre-trained word embedding models used for query expansion and the evaluation metrics used. For CLEF eHealth IR data collection, we assess the proposal in both topical relevance and understandability assessment and corresponding metrics for each relevance judgement are introduced. For FIRE CHIS data collection, we choose evaluation metrics different from the ones used for the CLEF eHealth collections. Finally, the experimental platform to carry out our experiments is presented.

5.1.1 Data Collections

Health information retrieval concerns different areas, from biomedical literature retrieval for clinical cases to health related retrieval by general non-expert users. The experimental data collections should differ from one case to another. Our research work aims to improve understandability for consumer health information retrieval, and the ideal data collection for carrying the research work should include:

- Queries proposed by non-experts consumers.
- Documents retrieved should be medical or health related, but not scientific biomedical literature.
- High-quality assessments for topic-document pairs should be available and be done by medical experts.
- Large datasets are needed to cater to the information retrieval task.

However, it is not an easy job to obtain this kind of high quality data that satisfies all the requirements listed above, mainly because:

- Due to data privacy and being a very specific topic in health search area, not too much open shared data collections are available.
- The assessment of the topic-document pairs is very costly since assessments require that the assessors should have strong medical knowledge backgrounds which usually only experts or majors in medical area have.

With carefully selection, in total, four data collections were chosen for the understandability study of consumer health information search, being FIRE'2016 CHIS data collection and CLEF'2016, CLEF'2017 and CLEF'2018 eHealth IR data collections. The statistics of these four datasets are presented in Table 5.1.

FIRE'2016 CHIS Data Collection is shared by Sinha et al. (2016) for the Consumer Health Information Search track on FIRE'2016[1]; this task is specific to consumer health search. We participated in this task and were able to get access to the data collection.

CLEF eHealth[2] is an evaluation lab that has organized evaluation campaigns in the medical and biomedical domain since 2013 and IR task in one among them. CLEF eHealth IR task follows the TREC-style evaluation process and provides a shared and standard IR data collection which contains a data set, a query set and qrels files. High-quality data collection and evaluation frameworks are available to the campaign participants, so we are able to successfully test our approaches on these data collections.

FIRE'2016 CHIS Data Collection

FIRE'2016 CHIS task description is as follows (Sinha et al., 2016): *"Given a CHIS query, and a document/set of documents associated with that query,*

[1] https://sites.google.com/site/multiperspectivehealthqa/.

[2] https://sites.google.com/site/clefehealth/.

Table 5.1: Statistics of experimental data collections.

	FIRE'2016 CHIS	**CLEF'2016 eHealth IR**	**CLEF'2017 eHealth IR**	**CLEF'2018 eHealth IR**
Base queries	5	50	50	50
Variations per query	-	6	6	7
Total queries	5	300	300	350
Dataset	medical data by organizers	ClueWeb12 -B13	ClueWeb12 -B13	based on 1,903 domain acquired from CommonCrawl
Dataset size	1,537 documents	52 million web pages	52 million web pages	5,535,120 Web pages
Sub-dataset	-	-	-	1,653 domains
Qrels file (topic-doc pairs)	1,537	150,000	119,232	18,763

the task is to classify the sentences in the document as relevant to the query or not. The relevant sentences are those from that document, which are useful in providing the answer to the query. These relevant sentences need to be further classified as supporting the claim made in the query, or opposing the claim made in the query." The task can be divided into two sub-tasks: the first sub-task decides if the text is relevant or irrelevant; and then, the perspective label (support/oppose/neutral) is assigned to each piece of text. An example query and its retrieval results are illustrated in Figure 5.1.

FIRE'2016 CHIS data collection includes five consumer health queries and their corresponding datasets, which is presented in Table 5.2.

Although FIRE'2016 CHIS is not a big data collection, its task is specifically aiming at the non-expert consumer health information search; integral and high-quality assessments for all topic-document pairs are available. These is valuable information for implementing parts of our proposed research work and carry out corresponding experiments.

```
Query: "Are e-cigarettes safer than normal cigarettes?"

Retrieved sentence S1:
"Because some research has suggested that the levels
of most toxicants in vapor are lower than the levels
in smoke, e-cigarettes have been deemed to be safer
than regular cigarettes."
A) Relevant, B) Support

Retrieved sentence S2:
"David Peyton, a chemistry professor at Portland
State University who helped conduct the research,
says that the type of formaldehyde generated by
e-cigarettes could increase the likelihood it would
get deposited in the lung, leading to lung cancer."
A) Relevant, B) Oppose

Retrieved sentence S3:
"Harvey Simon, MD, Harvard Health Editor, expressed
concern that the nicotine amounts in e-cigarettes
can vary significantly."
A) Relevant, B) Neutral
```

Figure 5.1: An example in the FIRE'2016 CHIS task.

CLEF'2016 & CLEF'2017 eHealth IR Data Collections

CLEF'2016 and CLEF'2017 eHealth IR data collections include the same query set and corpus. Nonetheless, CLEF'2017 data collection increased the assessment pool with more topic-documents paired assessed (see Table 5.1). We combined their qrels file into one and merged these two data collections into one, which is noted as *CLEF'2016-2017* in later sections. All assessments of our experimental work is based on the combined data collections.

Table 5.2: FIRE'2016 CHIS data collection.

Nr.	Query	Dataset Size
Q1	Does sun exposure cause skin cancer?	341
Q2	Are e-cigarettes safer than normal cigarettes?	413
Q3	Can Hormone Replacement Therapy (HRT) cause cancer?	246
Q4	Can MMR vaccine lead to children developing autism?	259
Q5	Should I take vitamin C for common cold?	278

Query Set. The query set was issued by the general public and expresses their real health information needs. The generation of the query set can be generally divided into two stages: preferable posts which served as base queries were selected; query variations for each query were generated.

During the fist stage, 50 posts were selected and later used for the next step query creation stage. The task organizers collected posts available from a public health web forums AskDocs of Reddit7[3]. This forum functions similarly to other common forums but mainly for medical information. The general public is allowed to post a medical or health related case, to inquire about diagnosis and treatments as well as interact with others in the comments section. To get high quality queries, the organizers selectively chose the posts available from the forum taking into account the following guidelines:

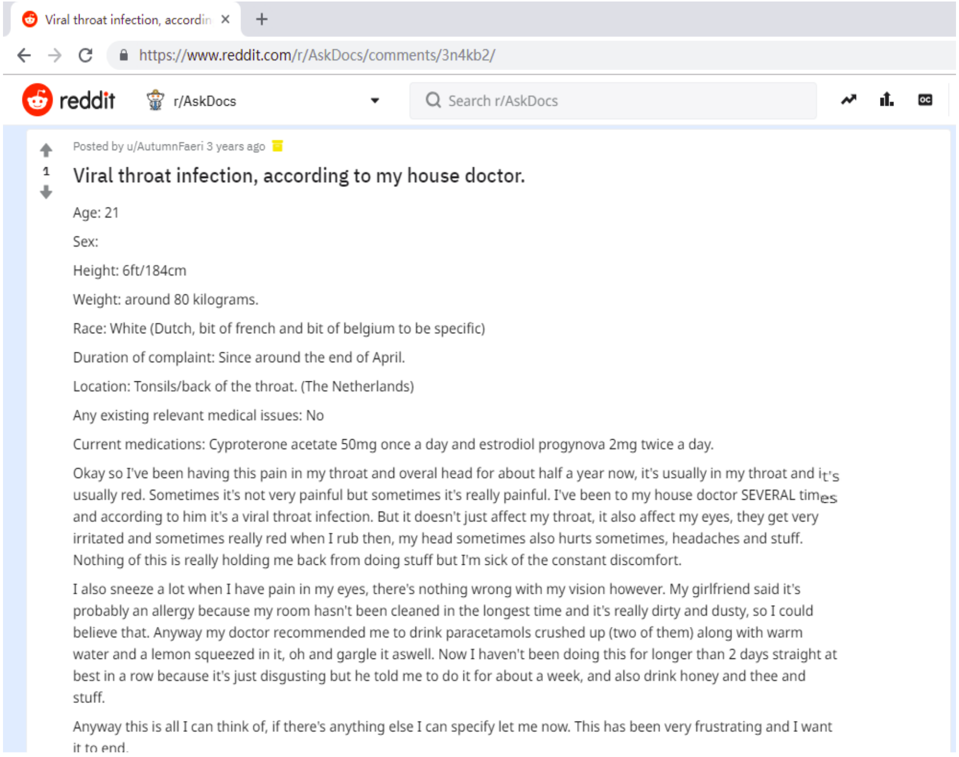
- Posts written in understandable and descriptive texts.
- Posts containing detailed patient information were preferred; the detailed information could be demographics information like age and gender, medical history or current medical condition.
- Posts with comments where users labelled according to their medical expertise were preferred.
- Posts where a main and single information need could be identified were preferred (aiming at gathering more queries on the same aspects).

An example original post with No.147 is presented in Figure 5.2.

During query creation phase, the following process was taken creating a total of 300 queries:

- Three medical experts and three non-expert users were selected to create query variations based on the 50 selected posts.
- Linux Aspell was used to correct spelling mistakes of the queries produced.
- Manual check for some special cases like the drug name.
- Punctuation marks were kept during pre-processing, since, for example, quotation marks can indicate proximity terms or phrasal terminology.

[3] https://www.reddit.com/r/AskDocs/


Viral throat infection, accordin ×
+
https://www.reddit.com/r/AskDocs/comments/3n4kb2/
reddit
r/AskDocs
Search r/AskDocs
1

Posted by u/AutumnFaeri 3 years ago

Viral throat infection, according to my house doctor.

Age: 21

Sex:

Height: 6ft/184cm

Weight: around 80 kilograms.

Race: White (Dutch, bit of french and bit of belgium to be specific)

Duration of complaint: Since around the end of April.

Location: Tonsils/back of the throat. (The Netherlands)

Any existing relevant medical issues: No

Current medications: Cyproterone acetate 50mg once a day and estrodiol progynova 2mg twice a day.

Okay so I've been having this pain in my throat and overal head for about half a year now, it's usually in my throat and it's usually red. Sometimes it's not very painful but sometimes it's really painful. I've been to my house doctor SEVERAL times and according to him it's a viral throat infection. But it doesn't just affect my throat, it also affect my eyes, they get very irritated and sometimes really red when I rub then, my head sometimes also hurts sometimes, headaches and stuff. Nothing of this is really holding me back from doing stuff but I'm sick of the constant discomfort.

I also sneeze a lot when I have pain in my eyes, there's nothing wrong with my vision however. My girlfriend said it's probably an allergy because my room hasn't been cleaned in the longest time and it's really dirty and dusty, so I could believe that. Anyway my doctor recommended me to drink paracetamols crushed up (two of them) along with warm water and a lemon squeezed in it, oh and gargle it aswell. Now I haven't been doing this for longer than 2 days straight at best in a row because it's just disgusting but he told me to do it for about a week, and also drink honey and thee and stuff.

Anyway this is all I can think of, if there's anything else I can specify let me now. This has been very frustrating and I want it to end.

Figure 5.2: CLEF'2016-2017 eHealth IR data: original post No.147

A concrete set of queries is presented in Figure 5.3. These queries were generated from No.147 post presented in Figure 5.2. The former three digits '101' of a query id identify the post number, meaning this group of queries were created based on the same post '101'. The latter three digits of a query id identify each individual query creator. '001', '002' and '003' identify non-experts and '004', '005' and '006' identify medical experts creators.

Dataset. CLEF'2016-2017 eHealth IR data collection includes ClueWeb12-B13[4] as the dataset which contains about 52 million web pages. ClueWeb12-B13 is a subset of ClueWeb12 which is a crawl of common Internet web pages[5].

Qrels File. The topic-document pairs were assessed by medical majors. Relevance between a topic and a document was graded as *highly relevant*, *somewhat relevant* and *not relevant*. Understandability of a document according to a topic was valued between 0 to 100, with 0 meaning the hardest

[4] http://lemurproject.org/clueweb12/

[5] ClueWeb12-B13 dataset is also the TREC 2013 "Category B" dataset.

```
<query>
        <id> 147001 </id>
        <title> throat infection sore throat irritated eyes treatment options
        </title>
        <url> https://www.reddit.com/r/AskDocs/comments/3n4kb2 </url>
</query>

<query>
        <id> 147002 </id>
        <title> constant sore throat causes and treatment </title>
        <url> https://www.reddit.com/r/AskDocs/comments/3n4kb2 </url>
</query>

<query>
        <id> 147003 </id>
        <title> head throat eye pain </title>
        <url> https://www.reddit.com/r/AskDocs/comments/3n4kb2 </url>
</query>

<query>
        <id> 147004 </id>
        <title> viral throat infection symptoms </title>
        <url> https://www.reddit.com/r/AskDocs/comments/3n4kb2 </url>
</query>

<query>
        <id> 147005 </id>
        <title> pain in throat and head and red eyes, viral throat infection?
        </title>
        <url> https://www.reddit.com/r/AskDocs/comments/3n4kb2 </url>
</query>

<query>
        <id> 147006 </id>
        <title> what pain in throat, irritated eyes, headaches - more than
        a viral infection? </title>
        <url> https://www.reddit.com/r/AskDocs/comments/3n4kb2 </url>
</query>
```

Figure 5.3: CLEF'2016-2017 eHealth IR data: queries from post No.147.

to understand and 100 the easiest. The assessment pool consisted of 100 runs from baselines and participants.

At the end, a total assessment of 150,000 and 119,232 topic-documents pairs were obtained for CLEF'2016 and CLEF'2017, respectively.

CLEF'2018 eHealth IR Data Collection

Query Set. CLEF'2018 eHealth IR data collection contains 50 base queries gathered from Health on the Net (HON) search engine[6]. These queries were issued by the general public and the selection constraints were as follows:

- A group of raw queries were first gathered from HON search engine over 6 months; then, medical domain experts manually selected the ones that were later used as the query set. To get rid of the possibility of using predetermined queries by web crawlers, only non-capitalized queries were taken into account.
- Queries only written in English were taken into account, with no complex medical terms and with more than 2 query terms.
- These 50 selected queries were not pre-processed and were considered as the base queries. Query variations generated from the base queries were produced by 3 non-experts and 3 experts users, using their own narrative each. No post-processing was applied to these query variations.

Similar to CLEF'2017 query set, all the queries were numbered using a 6 digits number. The former three digits identifies a topic number and the latter three digits identifies the generator. For each topic, identifier '001' represents the base query and '002', '003', '004', '005', '006', and '007' represents query variations from the base query. A concrete example of CLEF'2018 query is shown in Figure 5.4.

Dataset. CLEF'2018 eHealth IR dataset was obtained from the CommonCrawl[7]. The procedure of how it was produced is as follows:

- CLEF'2018 base query set were submitted to Microsoft Bing[8] repeatedly over a period of few weeks, and the URLs of the retrieved results were acquired.

[6] https://www.hon.ch/

[7] http://commoncrawl.org

[8] https://www.bing.com/

```
<query>
        <id> 151001 </id>
        <en> anemia diet therapy </en>
</query>

<query>
        <id> 151002 </id>
        <en> anemia change in diet </en>
</query>

<query>
        <id> 151003 </id>
        <en> diet for anemia </en>
</query>

<query>
        <id> 151004 </id>
        <en> anemia treat diet therapy changes </en>
</query>

<query>
        <id> 151005 </id>
        <en> anemia diet therapy </en>
</query>

<query>
        <id> 151006 </id>
        <en> anemia diet </en>
</query>

<query>
        <id> 151007 /id>
        <en> Diet for anemia </en>
</query>
```

Figure 5.4: CLEF'2018 eHealth IR data: a set of queries.

- The domains of the URLs were then selected and included in a list of websites. A group of reliable and known health websites were added to the list as well.
- The dataset was obtained from compiling web pages of these selected websites, using CommonCrawl for the acquisition.
- Pdf documents were excluded from the data acquired.

A total of 1,903 domains[9] were successfully requested. The full collection contained 5,535,120 Web pages (uncompressed size was about 480GB). In addition to the full collection, a subset was released by removing a few of non-strictly health-related websites; this subset contained 1,653 domains with a size about 294GB.

Qrels File. Following a similar assessment process as CLEF'2016 and CLEF'2017, CLEF'2018 Health IR data collection contains 18,763 topic-documents pairs.

5.1.2 Evaluation Metrics and Tools

When choosing the evaluation metrics, we mainly follow these two principles:

- All the official evaluation metrics used in task competitions are included.
- Other important and frequently used measure besides the official ones are included.

Since FIRE CHIS and CLEF eHealth data collections are distinguished from each other, we adopt different evaluation metrics for them. For FIRE CHIS data collection, we mainly follow the official evaluation metrics; for CLEF eHealth, all the experimental rankers are evaluated in terms of topic relevance as well as understandability relevance. Next sections discuss each in detail.

Evaluation Metrics of FIRE CHIS Data Collection

For FIRE'2016 CHIS data collection, the official evaluation measure is accuracy (Sinha et al., 2016). In this work, besides accuracy, we also use the

[9] https://github.com/CLEFeHealth/CLEFeHealth2018IRtask/blob/master/clef2018collection_listofdomains.txt

other three important evaluation metrics: F1, precision and recall calculated over the full dataset.

Topical Relevance Assessment of CLEF eHealth Data Collection

For CLEF eHealth data, in terms of topical relevance assessment, we include in the three most important and frequently used ones in information retrieval: precision at 10 (P@10), normalized discounted cumulative gain at 10 (NDCG@10) and MAP. We evaluate the IR rankers with assessment metrics at position 10 since users of online search engines are more likely to pay attention to the first page of retrieved results. P@10 and NDCG@10 are also the official assessments of the CLEF eHealth IR task.

Understandability Assessment of CLEF eHealth Data Collection

The formulation of understandability assessment is based on RBP. uRBP uses binary understandability assessments and uRBPgr uses graded understandability assessments (Zuccon, 2016; Palotti et al., 2015). In this work, we include in all these three measures in assessing understandability relevance.

Evaluation Tools

To compute these metrics discussed above, we use the standard evaluation tools available. For topical relevance assessment, we use trec_eval[10], which is the standard tool used by the TREC community for evaluating an ad-hoc retrieval run, given the results file and a standard set of judged results; for understandability assessment, we use ubire tool[11], which is a understandability-biased IR evaluation tool (Palotti et al., 2015). The assessment of experimental rankers on FIRE data collection uses scikit-learn[12] modules.

5.1.3 Experimental Platform

In this work, Terrier retrieval platform version 4.2 was used as the fundamental environment for us to carry out the experiments.

All queries were pre-processed by lower-casing characters, removing stop words and applying stemming with the Porter Stemmer. The default stop

[10] https://trec.nist.gov/trec_eval/

[11] https://github.com/ielab/ubire

[12] https://scikit-learn.org/stable/index.html

words list available in the Terrier platform was used. The Okapi BM25 retrieval model was mainly used when building a ranking model and all the parameters were set to default values, with b = 0.75 and k1 = 1.2 and k3=8, as recommended by Robertson et al. (1995). Besides BM25, in some experiments, we used other retrieval models including TFIDF, InL2 and language models.

Generally, all experiments performed follow the same experimental setup mentioned here. If an experiment takes a different setting, it will be explained in the corresponding place.

5.1.4 Summary

In this section, we present the experimental setup for building our experimental models and rankers. Four IR data collections were selected with each containing a set of standard and high quality query set, dataset and assessment files (qrels files). A group of evaluation metrics were carefully chosen: both the topical relevance and the understandability relevance were taken into account. The IR platform and the default settings used for building the experimental rankers were also presented. Based on these settings, next sections present the experiments which systematically test the proposals put forward in answering the first research question.

5.2 Medical Concept Model

To bridge the language gap between a medical expert and a non-expert consumer, we propose a Medical Concept Model (MCM) which is used to further process a query: we divide the query terms identified by the medical NLP tool into term concepts and phrase concepts; extra weights are then assigned accordingly.

To examine the usefulness of this MCM model, we design a set of experiments and carry out them on two CLEF eHealth IR data collections. The MCM model is first assessed with the CLEF'2016-2017 data and then re-assessed with the CLEF'2018 data.

We present the designed experiments and elaborate them in detail; we built a group of baselines using state-of-the-art techniques; and finally, we evaluate these experiments both in topical relevance and understandability assessment. At the end of this section, we make a conclusion over the observations from our experiment.

5.2.1 CLEF'2016-2017 Data Experiments

This section presents the assessments of the MCM model on CLEF'2016-2017 data. Corresponding experiments are designed; the evaluation on topical relevance and understandability are performed and the relevant results are presented, respectively; then, the overall results are discussed.

Experiments

The designed experiments are first carried out on the CLEF'2016-2017 data collection and three rankers are built, as shown in Table 5.3.

Table 5.3: MCM: rankers built on CLEF'2016-2017 data.

Ranker	**Methods Description**
BM25_MCM_PRF	BM25, MCM, with PRF technique
BM25_MCM$_{\text{umls}}$_PRF	BM25, MCM:UMLS QE, with PRF technique
BM25_MCM$_{\text{comb}}$_PRF	BM25, concepts processing&UMLS QE , with PRF technique

Ranker BM25_MCM_PRF. The first ranker is designed to verify the usefulness of using the MCM model independently (without using the UMLS Metathesaurus expansion) in the query expansion process: by employing the MCM model, the query terms are identified as medical term concepts or phrase concepts. Different techniques are then applied to them.

The process is as follows: first, the medical concept identification tool cTAKES is used to identify the medical concepts in the pre-processed queries; next, phrase concepts are reconstructed into loose phrases which are regarded as must check items during the retrieval process, and single term concepts are added with extra term weights. These processed terms and phrases are then added into the original query.

Two tuning-able parameters are: maximum interval words inside a phrase concept and weights assigned to a term concept, and for easy to use, we note them as $P_{maxinter}$ and $P_{termweight}$ respectively. Following the usual procedure of parameter tuning, we take use of the training and validation sets in seeking the best values for these two parameters. We vary $P_{maxinter}$ ranging from 1 to 3 and $P_{termweight}$ ranging from 1 to 2, with an incremental step of 0.1. The optimal values obtained were 2 and 1.5, respectively.

Ranker BM25_MCM$_{\text{umls}}$_PRF. Different from *BM25_MCM_PRF*, the second ranker is designed to using the UMLS Metathesaurus as the resource to expand new words when applying the MCM model to process the query.

First, cTAKES is used to identify the medical concepts; then, new words are expanded from the UMLS Metathesaurus regarding to the identified medical terms or phrases and we follow the procedure proposed in Chapter 4 to do word selections; next, the MCM model is applied to process the expanded new words respectively[13], different weights are assigned depending on whether the new words are expanded from a phrase concept or a single term concept; and finally, expanded new words with extra weights are added to the query.

The weights are tuned following the same procedure discussed in ranker *BM25_MCM_PRF*, and are set to 2 and 1.5 for expanded phrase and term, respectively.

Ranker BM25_MCM$_{\mathbf{comb}}$_PRF. The third ranker is built with combining the schemes used both in *BM25_MCM_PRF* and *BM25_MCM$_{umls}$ _PRF*: the processed concepts using methods adopted in the first ranker are expanded to the original query; the expanded and processed new words using methods adopted in the second ranker are added to the original query as well.

Intuitively, a concrete example of the processed queries for each experimental ranker is shown in Figure 5.5. This figure exhibits the processed query from original query No.140071[14] by applying the different techniques designed for each experimental ranker.

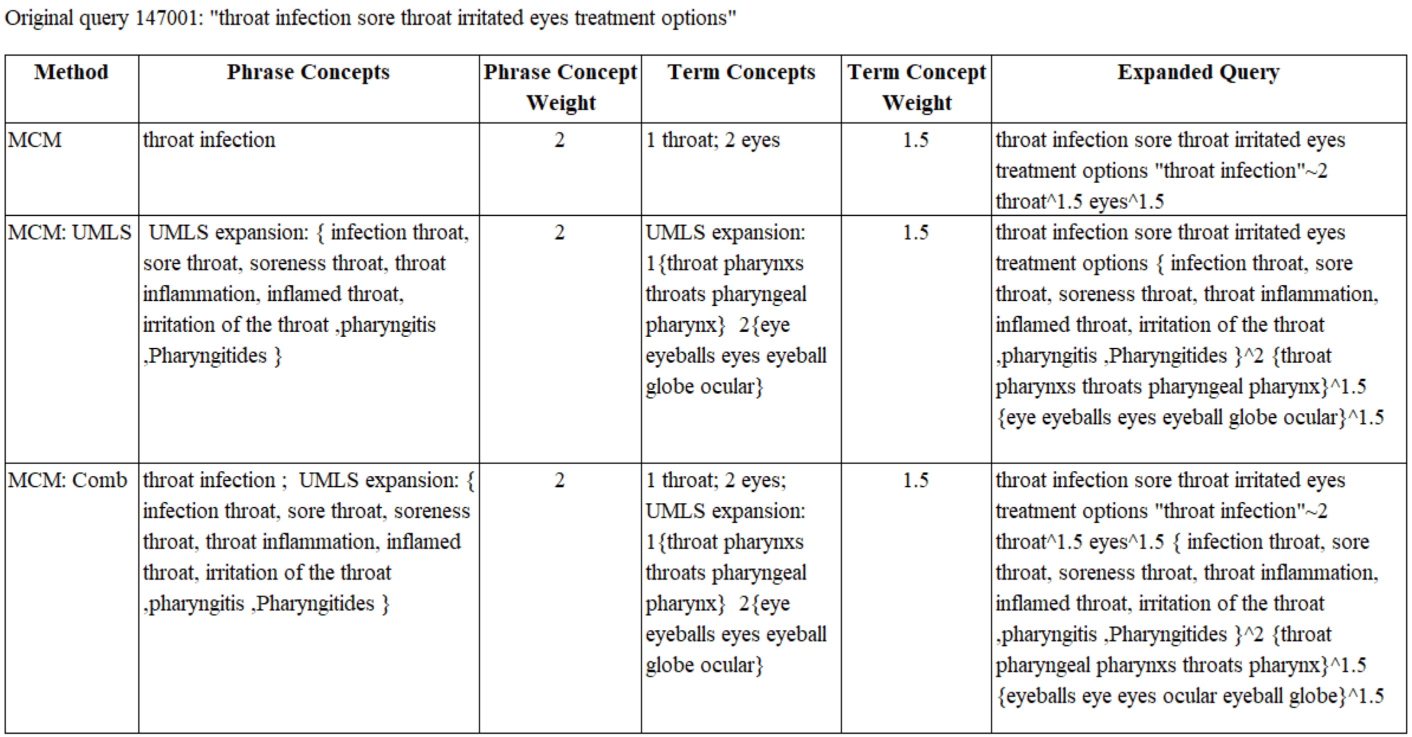
Topical Relevance Assessment

Baselines. For this experimental set carried out on CLEF'2016-2017 data, we totally built 6 topical relevance baselines employing state-of-the-art techniques.

These 6 topical relevance baselines were all built in the Terrier platform and used three classic retrieval models: TFIDF, BM25 and language model DirichletLM, with and without auto query expansion techniques. We used the pseudo relevance feedback as the auto query expansion and the top 10 terms from the top 3 documents in the retrieved ranked list were selected to expand the original query. For example, *baselineBM25_PRF* represents a

[13] Extra weight are assigned to newly added words and no loose phrases are constructed in the second ranker.

[14] Every query is assigned with a query ID in CLEF eHealth IR data collection.



Original query 147001: "throat infection sore throat irritated eyes treatment options"

Method	Phrase Concepts	Phrase Concept Weight	Term Concepts	Term Concept Weight	Expanded Query
MCM	throat infection	2	1 throat; 2 eyes	1.5	throat infection sore throat irritated eyes treatment options "throat infection"~2 throat^1.5 eyes^1.5
MCM: UMLS	UMLS expansion: { infection throat, sore throat, soreness throat, throat inflammation, inflamed throat, irritation of the throat ,pharyngitis ,Pharyngitides }	2	UMLS expansion: 1{throat pharynxs throats pharyngeal pharynx} 2{eye eyeballs eyes eyeball globe ocular}	1.5	throat infection sore throat irritated eyes treatment options { infection throat, sore throat, soreness throat, throat inflammation, inflamed throat, irritation of the throat ,pharyngitis ,Pharyngitides }^2 {throat pharynxs throats pharyngeal pharynx}^1.5 {eye eyeballs eyes eyeball globe ocular}^1.5
MCM: Comb	throat infection ; UMLS expansion: { infection throat, sore throat, soreness throat, throat inflammation, inflamed throat, irritation of the throat ,pharyngitis ,Pharyngitides }	2	1 throat; 2 eyes; UMLS expansion: 1{throat pharynxs throats pharyngeal pharynx} 2{eye eyeballs eyes eyeball globe ocular}	1.5	throat infection sore throat irritated eyes treatment options "throat infection"~2 throat^1.5 eyes^1.5 { infection throat, sore throat, soreness throat, throat inflammation, inflamed throat, irritation of the throat ,pharyngitis ,Pharyngitides }^2 {throat pharyngeal pharynxs throats pharynx}^1.5 {eyeballs eye eyes ocular eyeball globe}^1.5

Figure 5.5: Query processing using the Medical Concept Model.

baseline built in Terrier search engine, using BM25 retrieval model and with the PRF query expansion technique applied.

Results. Table 5.4 presents the topical relevance assessments for the experiments in assessing the usefulness of the MCM model and carried out on CLEF'2016-2017 eHealth IR data collection. As already mentioned, all rankers are evaluated with three evaluation metrics: P@10, NDCG@10 and MAP.

Table 5.4: MCM: rankers' topical relevance on CLEF'2016-2017 data.

Algorithm	**Ranker**	**P@10**	**NDCG@10**	**MAP**
Baselines	baselineTFIDF_PRF$_{\text{no}}$	0.2897	0.2335	0.0909
	baselineTFIDF_PRF	0.3147	0.2509	0.1147
	baselineBM25_PRF$_{\text{no}}$	0.2903	0.2346	0.0914
	baselineBM25_PRF	0.3167	0.2512	0.1149
	baselineDirichletLM_PRF$_{\text{no}}$	0.2507	0.2011	0.0733
	baselineDirichletLM_PRF	0.2297	0.1797	0.0632
MCM Model	BM25_MCM_PRF	0.3037*	0.2434*	0.1015
	BM25_MCM$_{\text{umls}}$_PRF	0.3017	0.2413	0.1079*
	BM25_MCM$_{\text{comb}}$_PRF	0.3020	0.2413	0.1025

$_=$: strongest baseline
* : the best score achieved by developed rankers

As it can be observed, all three rankers built using MCM model are able to outperform most baselines except the two strongest baselines in all three evaluation metrics. Although not able to surpass the top two baselines, their scores are close to them. Comparing the performance among the three rankers developed and looking at P@10 and NDCG@10, it can be observed that the best result is achieved with ranker *BM25_MCM_PRF*; considering MAP, the best score is achieved with ranker *BM25_MCM$_{umls}$_PRF*. Moreover, the combined scheme of ranker *BM25_MCM$_{comb}$_PRF* achieved modest results and no obvious improvement when compared with *BM25_MCM_PRF* and *BM25_MCM$_{umls}$_PRF*. Also, using UMLS expansion (*BM25_MCM$_{umls}$_PRF*) does not seem to improve results when compared to the one without using it (*BM25_MCM_PRF*).

Concluding, the ranker *BM25_MCM_PRF* performed the best in most cases, exceeding most baselines and approaching the strongest baseline.

Understandability Assessment

Baselines. For understandability assessment, we totally built 13 understandability baselines employing state-of-the-art IR techniques in the Terrier platform.

These baselines can be divided into two groups. One group includes the same 6 baselines built for topical relevance assessment: using TFIDF, BM25 and DirichletLM retrieval model, with and without the PRF technique.

The other group includes 7 baselines all built using BM25 retrieval model. Two of them used the scores with readability measure CLI and GFI each; the scores of these two readability measures represented the number of school years necessary to read the text being evaluated[15]. The other 5 understandability baselines were generated based on the spam rankings distributed with ClueWeb12[16]; for each of them, all documents that had a spam score smaller than a given threshold were removed and threshold values experimented were: 50, 60, 70, 80 and 90 (Zuccon et al., 2016).

Results. Table 5.5 presents the understandability assessment results for the experiments carried out on CLEF'2016-2017 eHealth IR data collection. As mentioned earlier, all the rankers are evaluated with the understandability metrics: RBP, uRBP and uRBPgr.

As it can be observed, all three rankers built with MCM model are able to outperform most baselines except the two strongest baselines in RBP evaluation metrics; although not able to surpass these top two baselines, their scores are close to them. Looking at uRBP metric, all three rankers outperform the strongest baseline with a large space. And when considering uRBPgr, the almost similar results are obtained: with ranker $BM25\_MCM\_PRF$ and $BM25\_MCM_{comb}\_PRF$ are able to outperform the strongest baseline with obvious improvements; ranker $BM25\_MCM_{umls}\_PRF$ are quite close to the two strongest baselines. More specifically, the best results are achieved with ranker $BM25\_MCM\_PRF$, and followed by $BM25\_MCM_{comb}\_PRF$ and then $BM25\_MCM_{umls}\_PRF$. Meanwhile, comparing $BM25\ \_MCM_{umls}\_PRF$ to $BM25\_MCM\_PRF$, we can see that using UMLS expansion techniques lower understandability scores; combining both approaches ($BM25\_MCM_{comb}\_PRF$) improves UMLS expansion ($BM25\_MCM_{umls}\_PRF$) but not the one without using it ($BM25\_MCM\_PRF$).

Summing up, the ranker $BM25\_MCM\_PRF$ using MCM model indepen-

[15] Readers can refer back to Chapter 2 for the details of CLI (Coleman-Liau Index) and GFI (Gunning Fox Index).

[16] http://www.mansci.uwaterloo.ca/ msmucker/cw12spam/

Table 5.5: MCM: rankers' understandability on CLEF'2016-2017 data.

Algorithm	**Ranker**	**RBP**	**uRBP**	**uRBPgr**
Baselines	baselineTFIDF_PRF$_{\text{no}}$	0.2961	0.1110	0.1175
	baselineTFIDF_PRF	0.3199	0.1222	0.1273
	baselineBM25_PRF$_{\text{no}}$	0.2972	0.1126	0.1189
	baselineBM25_PRF	0.3193	0.1220	0.1272
	baselineDirichletLM_PRF$_{\text{no}}$	0.2616	0.1106	0.1082
	baselineDirichletLM_PRF	0.2394	0.1082	0.1040
	baselineBM25CLI	0.0897	0.0170	0.0256
	baselineBM25GFI	0.0979	0.0213	0.0315
	baselineBM25spam50	0.2874	0.1022	0.1120
	baselineBM25spam60	0.2887	0.1054	0.1152
	baselineBM25spam70	0.2792	0.1007	0.1110
	baselineBM25spam80	0.2629	0.0890	0.1005
	baselineBM25spam90	0.2028	0.0660	0.0753
MCM Model	BM25_MCM_PRF	0.3117*	0.1340+*	0.1315+*
	BM25_MCM$_{\text{umls}}$_PRF	0.3067	0.1262+	0.1262
	BM25_MCM$_{\text{comb}}$_PRF	0.3106	0.1319+	0.1293+

= : strongest baseline
* : the best score achieved by developed rankers
$\underline{\underline{+}}$: better than the strongest baseline

dently (without using UMLS expansion) performed the best. It exceeds the strongest baseline and the other two experimental rankers with a large margin in both uRBP and uRBPgr metrics, and approaching the strongest baseline in RBP.

Overall Results

Based on the observations obtained from topical relevance and understandability assessment, we can conclude that our proposed Medical Concept Model is efficient in query processing and shown to be an effective solution in improve state-of-the art techniques in consumer health information retrieval. In particular, the ranker built using the MCM model independently (*BM25_MCM_PRF*) achieved the best result compared with the other two built rankers.

5.2.2 CLEF'2018 Data Experiments

Since the Medical Concept Model presents its usefulness in CLEF'2016-2017 eHealth IR data collection, it is meaningful and interesting to verify its scalability on a different data collection. This section discusses the experiments of the MCM model on CLEF'2018 eHealth IR Data Collection[17].

Experiments

Since the ranker using Medical Concept Model independently (*BM25_-MCM_PRF*) performed the best on the CLEF'2016-2017 data collection, we selectively test this method on CLEF'2018 data collection.

We use similar techniques taken in building this ranker on CLEF'2016-2017 data and apply the corresponding techniques CLEF'2018:

Similar techniques were taken and improved experiments were designed: a group of 6 rankers were built using three different retrieval models TFIDF, BM25 and InL2, with and without using the PRF technique, respectively. The built rankers and their methods are shown in Table 5.6.

Table 5.6: MCM: rankers built on CLEF'2018 data.

Ranker	Methods Description
TFIDF_MCM_PRF$_{\text{no}}$	TFIDF, MCM, no PRF technique
TFIDF_MCM_PRF	TFIDF, MCM, with PRF technique
BM25_MCM_PRF$_{\text{no}}$	BM25, MCM, no PRF technique
BM25_MCM_PRF	BM25, MCM, with PRF technique
InL2_MCM_PRF$_{\text{no}}$	InL2, MCM, no PRF technique
InL2_MCM_PRF	InL2, MCM, with PRF technique

Topical Relevance Assessment

Baselines. For the experiment in assessing the MCM model and carried out on CLEF'2018 task, we totally built 7 topical relevance baselines employing state-of-the-art IR techniques.

Six of them were generated following the same techniques used in CLEF'2016-2017 data and were built in the Terrier platform. The seventh baseline *base-*

[17] We use similar methods as we applied on the CLEF'2016-2017 eHealth IR data collection, so the same techniques are not repeated in this section and the readers are asked to refer to the previous section in need.

lineBing was obtained from task organizers and generated with Microsoft Bing application (Jimmy et al., 2018).

Results. Table 5.7 presents the topical relevance assessments of the experiment carried out on CLEF'2018 eHealth IR data collection.

Table 5.7: MCM: rankers' topical relevance on CLEF'2018 data.

Algorithm	**Ranker**	**P@10**	**NDCG@10**	**MAP**
Baselines	baselineTFIDF_PRF$_{\text{no}}$	0.7360	0.6292	0.2586
	baselineTFIDF_PRF	0.7200	0.6080	0.2526
	baselineBM25_PRF$_{\text{no}}$	0.7100	0.5919	0.2575
	baselineBM25_PRF	0.6900	0.5698	0.2471
	baselineDirichletLM_PRF$_{\text{no}}$	0.7120	0.6054	0.2752
	baselineDirichletLM_PRF	0.6520	0.5521	0.1455
	baselineBing	0.4940	0.4856	0.0185
MCM Model	TFIDF_MCM_PRF$_{\text{no}}$	0.7280	0.6247	0.2565
	TFIDF_MCM_PRF	0.7040	0.5889	0.2472
	BM25_MCM_PRF$_{\text{no}}$	0.6980	0.5885	0.2534
	BM25_MCM_PRF	0.6840	0.5664	0.2386
	InL2_MCM_PRF$_{\text{no}}$	0.7340*	0.6254*	0.2542*
	InL2_MCM_PRF	0.6920	0.5791	0.2380

= : strongest baselines
* : the best score achieved by developed rankers

As it can be observed, the best results were obtained with ranker *InL2_MCM_PRF$_{no}$*. All developed rankers were not able to outperform the strongest baseline (*TFIDF_PRF$_{no}$*), but rankers *TFIDF_MCM_PRF$_{no}$* and *InL2_MCM_PRF$_{no}$* were able to surpass most baselines and quite close to the strongest baseline. Also, we can see rankers built without using the PRF techniques achieved better performance than the ones using it.

Understandability Assessment

Baselines. For the understandability assessment, we totally built 8 understandability baselines employing state-of-the-art IR techniques.

Similar to CLEF'2016-2017 understandability baselines: 6 baselines were built using TFIDF, BM25 and DirichletLM retrieval model, with and without the PRF technique; the other two of them used the scores with readability measure CLI and GFI each.

Results. Table 5.8 presents the understandability assessment of the experiments performed on CLEF'2018 eHealth IR data collection. Similarly, rankers were evaluated using RBP, uRBP and uRBPgr.

Table 5.8: MCM: rankers' understandability on CLEF'2018 data.

Algorithm	**Ranker**	**RBP**	**uRBP**	**uRBPgr**
Baselines	baselineTFIDF_PRF$_{\text{no}}$	0.7297	0.7370	0.3170
	baselineTFIDF_PRF	0.7199	0.7348	0.3030
	baselineBM25_PRF$_{\text{no}}$	0.6987	0.7076	0.3030
	baselineBM25_PRF	0.6873	0.7006	0.2820
	baselineDirichletLM_PRF$_{\text{no}}$	0.7735	0.7241	0.3010
	baselineDirichletLM_PRF	0.6654	0.6818	0.2750
	baselineBM25CLI	0.6005	0.6126	0.2280
	baselineBM25GFI	0.5981	0.6030	0.2340
MCM Model	TFIDF_MCM_PRF$_{\text{no}}$	0.7273*	0.7515+*	0.3190+
	TFIDF_MCM_PRF	0.7012	0.7298	0.2990
	BM25_MCM_PRF$_{\text{no}}$	0.6967	0.7201	0.3040
	BM25_MCM_PRF	0.6746	0.7001	0.2790
	InL2_MCM_PRF$_{\text{no}}$	0.7242	0.7496+	0.3210+*
	InL2_MCM_PRF	0.6920	0.6797	0.2380

= : strongest baselines
* : the best score achieved by developed rankers
+ : better than the strongest baseline

From the results, we can see that rankers *TFIDF_MCM_PRF$_{no}$* and *InL2_-MCM_PRF$_{no}$* were able to outperform the strongest baseline with obvious improvements in uRBP and uRBPgr metrics. The best results was achieved by ranker *TFIDF_MCM_PRF$_{no}$* which showed the best performance in RBP and uRBP compared to other rankers. The best results in uRBPgr was achieved by *InL2_MCM_PRF$_{no}$*. Also, rankers built without using the PRF techniques achieved better performance than the ones using it.

Overall Results

When testing the MCM model on CLEF'2018 data collection, the built rankers were shown to be effective with topical relevance assessment and some rankers were able to surpass the strongest baseline with obvious improvements in understandability evaluation metrics. We can conclude that the proposed Medical Concept Model is a competitive method for query processing compared with other state-of-the-art techniques; and more specifically, using the MCM model is an effective solution in improving understandability in the area of CHIR.

5.2.3 Conclusions

To bridge the language gap between non-expert consumers and medical experts, a Medical Concept Model is proposed. In this section, we tested this MCM model's effectiveness with two sets of data collections. First, we carried out three groups of experiments on CLEF'2016-2017 eHealth IR data collections; then, based on the observations obtained from the results of these experiments, we selectively experimented the model with the best scores on CLEF'2018 eHealth IR Data Collection.

From the experimental results obtained on CLEF'2016-2017 eHealth IR data collections, we observed that all three rankers trained with MCM model outperformed most baselines and were very close to the top two strongest baselines. The results on CLEF'2018 eHealth IR Data Collection showed that our proposed MCM model exceeded the strongest baseline with an obvious improvement.

In what concerns understandability assessments, on all data collections, all the developed rankers were able to exceed the strongest baseline with great improvements and providing an effective solution to improve understandability in consumer health search.

In summary, this conclusion can be definitely drawn from our two sets of experiments: the Medical Concept Model proposed for consumer health search shown to be superior in almost all the cases producing very competitive results compared to the state-of-the-art techniques, for both topical relevance and promoting understandability.

5.3 Word Embeddings in QE

Query expansion approaches have been shown to be effective in health search. Using the UMLS Metathesaurus to expand the original query is a classic way in this area. In recent years, locally trained word embedding models have also been used as a resource of query expansion. Word vectors are shown to be able to capture both syntactic and semantic similarity between words that share the same context.

Based on these ideas, in this section, first, we train two word embedding models and carefully selected one pre-trained model; then, we tested the usefulness of these models on three different data collections: experiment set 1 is carried out on FIRE'2016 CHIS data collection; experiment set 2 is on CLEF'2016-2017 eHealth IR data collection; and experiment set 3 is on CLEF'2018 data collection.

For each experiment set: first, we display the designed experiments and explain them in detail; then, we build a group of baselines with state-of-the-art techniques; next, we display the results both in topical relevance and understandability assessment; finally, our observations from the results are discussed. At the end of this section, we make a conclusion over our three experiment sets.

5.3.1 Word Embedding Models

In this work, three word embedding models are used: model $\mathrm{WE}_{\mathrm{pmc}}$ and $\mathrm{WE}_{\mathrm{wiki}}$ are locally trained using different training data and techniques; the third one is a pre-trained $\mathrm{WE}_{\text{trec-skip}}$ model[18] (Zuccon et al., 2015).

Model $\mathbf{WE}_{\mathbf{pmc}}$. The training data for model $\mathrm{WE}_{\mathrm{pmc}}$ is from medical domain and obtained from PubMed Central (PMC)[19]. The PMC Open Access Subset[20] is a part of the total collection of articles in PMC which contain free full-text archive of biomedical and life sciences journal literature at the U.S. National Institutes of Health's National Library of Medicine; we used the subset's non-commercial collection snapshot on date 16th Feb, 2017 for the vectors training. As a result, a file containing 25,140,380 vectors (number of distinct terms), with size 200 was obtained.

Model $\mathbf{WE}_{\mathbf{wiki}}$. Model $\mathrm{WE}_{\mathrm{wiki}}$ was trained using data from the non-medical specific area and obtained from Wikipedia English articles using a snapshot on 16th Nov, 2016 and a file containing 8,689,917 vectors (number of distinct terms), with size 200 was obtained.

Concerning the training tool, we choose the widely used Word2vec to train these two word embedding models (Le and Mikolov, 2014). Details about the parameter setting for each model is presented in Table 5.9.

[18] See table 3.8 for more information.

[19] https://www.ncbi.nlm.nih.gov/pmc/

[20] https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/

Table 5.9: Locally trained word embedding models.

	WE$_{\text{pmc}}$	**WE$_{\text{wiki}}$**	**WE$_{\text{trec-skip}}$**
Training tool	Word2vec	Word2vec	Word2vec
Architecture	CBOW	CBOW	Skipgram
Vector dimension	200	200	1000
Training data	PMC Open Access Subset	Wikipedia English articles	TREC Medical Records Track 2011 and 2012
Snapshot date	16th Feb, 2017	16th Nov, 2016	-
Word vectors (distinct terms)	25,140,380	8,689,917	25,469

5.3.2 FIRE'2016 Data Experiments

This section presents the assessments of the word embeddings models on FIRE data. Corresponding experiments are designed; the evaluations are performed and the relevant results are presented; the conclusion are then made based on the observations from the results.

Experiments

The first experiment set was performed over FIRE'2016 CHIS data collection and six experiments were designed in total, as shown in Table 5.10. These experiments are tested with and without the use of the pseudo relevance feedback technique and three different query expansion resources: UMLS Metathesaurus, two word embedding models WE$_{\text{wiki}}$ and WE$_{\text{pmc}}$.

Table 5.10: WE in QE: rankers built on FIRE'2016 data.

Ranker	**Methods Description**
TFIDF_UMLS_PRF$_{\text{no}}$	UMLS as QE resource, no PRF technique
TFIDF_UMLS_PRF	UMLS as QE resource, with PRF technique
TFIDF_WE$_{\text{wiki}}$_PRF$_{\text{no}}$	WE$_{\text{wiki}}$ as QE resource, no PRF technique
TFIDF_WE$_{\text{wiki}}$_PRF	WE$_{\text{wiki}}$ as QE resource, with PRF technique
TFIDF_WE$_{\text{pmc}}$_PRF$_{\text{no}}$	WE$_{\text{pmc}}$ as QE resource, no PRF technique
TFIDF_WE$_{\text{pmc}}$_PRF	WE$_{\text{pmc}}$ as QE resource, with PRF technique

Assessment

Baselines. We build two baselines, one is using PRF and the other is without PRF. We use TFIDF model as the retrieval model and parameters are set to default in Terrier. Both the queries and the documents of the dataset are pre-processed to remove stop-words and stem the words; we use Terrier's standard stop-words list and the Porter stemmer respectively.

Evaluation Metrics. Accuracy is chosen by organizers as the official evaluation measure in the campaign (Sinha et al., 2016). To compare to the baselines, as well as comparing to other team scores, we also adopt accuracy as one evaluation measure in our experiments. Besides, to further study how different these query expansion techniques performed, Precision, Recall and F1 score measures were also used to evaluate the results. For every evaluation metric, performance of each query expansion method was evaluated on each query and the average was then calculated.

Results. In this section, the comparison between different query expansion techniques using different evaluation metrics is done. Then the best results attained from the experiments are compared to the best team score in CHIS FIRE'2016 task.

Table 5.11 presents the obtained precision values. From it we can see that expansion using model WE_{pmc} greatly improve the baseline; in average the increase is about 17%; for expansion based on model WE_{wiki}, no improvement over baseline is observed in average. We note that using this model (WE_{wiki}) achieves almost the same performance as using the UMLS Metathesaurus. Besides, we also note that when evaluated on precision, any expansion technique combined with pseudo relevance feedback performs worse than using the technique alone.

When evaluated on recall (Table 5.12), using word embedding, either trained on Wikipedia or on PubMed, improves the baseline; the average improvement is 17% and 20% respectively. We can also see that expansion based on word embedding performs better than using the UMLS Metathesaurus and observe that any expansion technique combined with pseudo relevance feedback performs better than using the technique alone.

Table 5.13 presents the F1 results. From the table we can see that query expansion with word embedding trained on PubMed achieves the best results and shows a great improvement compared with other methods. The average increase over baseline is about 24%. When comparing word embedding to the UMLS Metathesaurus query expansion, a much higher performance is

Table 5.11: WE in QE: rankers' precision on FIRE'2016 data.

Algorithm	**Ranker**	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Avg.**
Baselines	baseline_PRF$_{\text{no}}$	0.587	0.900	0.893	0.836	0.770	0.797
	baseline_PRF	0.584	0.760	0.862	0.826	0.769	0.760
UMLS	TFIDF_UMLS_PRF$_{\text{no}}$	0.585	0.892*	0.882	0.852	0.769	0.796
	TFIDF_UMLS_PRF	0.593	0.778	0.849	0.846	0.759	0.765
Word embedding	TFIDF_WE$_{\text{wiki}}$_PRF$_{\text{no}}$	0.592	0.872	0.889	0.838	0.764	0.791
	TFIDF_WE$_{\text{wiki}}$_PRF	0.588	0.777	0.848	0.825	0.758	0.759
	TFIDF_WE$_{\text{pmc}}$_PRF$_{\text{no}}$	1*	0.833	0.985*	1*	1*	0.964*
	TFIDF_WE$_{\text{pmc}}$_PRF	1*	0.645	0.985*	1*	0.971	0.920

* : the best score achieved by developed rankers

Table 5.12: WE in QE: rankers' recall on FIRE'2016 data.

Algorithm	**Ranker**	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**	**Avg.**
Baselines	baseline_PRF$_{\text{no}}$	0.851	0.061	0.923	0.933	0.966	0.747
	baseline_PRF	0.881	0.259	0.962	0.962	0.970	0.807
UMLS	TFIDF_UMLS_PRF$_{\text{no}}$	0.856	0.113	0.938	0.827	0.962	0.739
	TFIDF_UMLS_PRF	0.954	0.311	0.976	0.846	0.971	0.812
Word embedding	TFIDF_WE$_{\text{wiki}}$_PRF$_{\text{no}}$	0.959	0.140	0.928	0.942	0.966	0.787
	TFIDF_WE$_{\text{wiki}}$_PRF	0.979*	0.642	0.995	0.976	0.976	0.914
	TFIDF_WE$_{\text{pmc}}$_PRF$_{\text{no}}$	0.961	0.4	1*	0.945	0.971	0.855
	TFIDF_WE$_{\text{pmc}}$_PRF	0.961	0.8*	1*	1*	0.985*	0.949*

* : the best score achieved by developed rankers

achieved with the former technique. Also, when evaluated on F1 score, an expansion technique combined with pseudo relevance feedback performs better than using the technique alone.

Comparison to the best team score. The two best performing techniques in our experiments are compared to the best results reported for the task (Sinha et al., 2016) in Table 5.14. We can see that expansion based on word embedding trained with PubMed outperforms the best team score about 15% in average accuracy[21].

When comparing the performance for each query using different query expansion techniques, it was observed that for Query 2, all expansion techniques decreased the baseline when evaluated on precision; inversely, when evaluated on recall, a surprisingly improvement was observed when using pseudo

[21] The calculation follows the same measure adopted in CHIS FIRE'2016 track.

Table 5.13: WE in QE: rankers' F1 score on FIRE'2016 data.

Algorithm	Ranker	Q1	Q2	Q3	Q4	Q5	Avg.
Baselines	baseline_PRF$_{\text{no}}$	0.695	0.115	0.908	0.882	0.857	0.691
	baseline_PRF	0.702	0.387	0.909	0.889	0.860	0.749
UMLS	TFIDF_UMLS_PRF$_{\text{no}}$	0.695	0.290	0.909	0.839	0.854	0.717
	TFIDF_UMLS_PRF	0.731	0.444	0.908	0.846	0.852	0.756
Word embedding	TFIDF_WE$_{\text{wiki}}$_PRF$_{\text{no}}$	0.732	0.241	0.908	0.887	0.854	0.724
	TFIDF_WE$_{\text{wiki}}$_PRF	0.735	0.703	0.916	0.894	0.853	0.820
	TFIDF_WE$_{\text{pmc}}$_PRF$_{\text{no}}$	0.980*	0.540	0.992*	0.972	0.985*	0.893
	TFIDF_WE$_{\text{pmc}}$_PRF	0.980*	0.714*	0.992*	1*	0.978	0.933*

* : the best score achieved by developed rankers

Table 5.14: WE in QE: rankers' accuracy on FIRE2016 data.

Ranker	Q1	Q2	Q3	Q4	Q5	Avg.
The best team score	0.796	0.810*	0.875	0.641	0.784	0.781
TFIDF_WE$_{\text{pmc}}$_PRF$_{\text{no}}$	0.966*	0.734	0.966	0.948	0.973*	0.917
TFIDF_WE$_{\text{pmc}}$_PRF	0.966	0.750	0.986*	1*	0.959	0.932*

* : the best score achieved by developed rankers

relevance feedback. With this in mind, we can state that query expansion techniques perform abnormally for Query 2.

Conclusions

In this experiment set, two word embedding models were trained using the Word2vec algorithm on two large text corpora (Wikipedia and PubMed) and were applied for query expansion. Query expansion techniques based on word embedding and the UMLS Metathesaurus were compared in a thoroughly evaluation. Firstly, we find that query expansion using word embedding is useful; we observe that using word embedding for query expansion achieves higher performance compared to the state-of-the-art the UMLS Metathesaurus technique. Secondly, our results also show that word embedding trained on a medical corpus (PubMed) obtains much better results than models trained on general Wikipedia data.

When compared with the CHIS FIRE'2016 best team score, we observe that expansion using word embedding trained with medical PubMed exceeds that score with a large margin. From our observation, we conclude that using word embedding can be an effective way to perform query expansion in consumer health information search.

5.3.3 CLEF'2016-2017 Data Experiments

The second experiment set is performed over CLEF'2016-2017 data collection. This section discuss the designed experiments, the evaluation on both topical relevance and understandability, and then the overall results.

Experiments

Six rankers are designed in total, as shown in Table 5.15.

Table 5.15: WE in QE: rankers built on CLEF'2016-2017 data.

Ranker	**Methods Description**
BM25_UMLS_PRF$_{\text{no}}$	UMLS as QE; no PRF technique
BM25_UMLS_PRF	UMLS as QE; with PRF technique
BM25_MCM$_{\text{UMLS}}$_PRF$_{\text{no}}$	MCM Model: UMLS as QE; no PRF technique
BM25_MCM$_{\text{UMLS}}$_PRF	MCM Model: UMLS as QE; with PRF technique
BM25_WE$_{\text{pmc}}$_PRF$_{\text{no}}$	WE$_{\text{pmc}}$ as QE; no PRF technique
BM25_WE$_{\text{pmc}}$_PRF	WE$_{\text{pmc}}$ as QE; with PRF technique

These rankers were all built using BM25 retrieval model. They employed either the UMLS Metathesaurus or the trained word embedding models as the resource for query expansion, with or without using the PRF techniques.

Rankers *BM25_ UMLS_ PRF$_{no}$* and *BM25_ UMLS_ PRF* are built using the the UMLS Metathesaurus expansion. First, we take use of the cTAKES tool to identify the medical terms inside a query; next, similar or related words to these medical terms are searched out from the UMLS Metathesaurus; finally, these new words are added to the original query.

Rankers *BM25_ MCM$_{UMLS}$_ PRF$_{no}$* and *BM25_ MCM$_{UMLS}$_ PRF* are designed to take use of the proposed Medical Concept Model: first, medical terms are identified in a query with cTAKES; then, we apply the MCM model to process the original query; next, we use the the UMLS Metathesaurus expansion approaches; and finally, the processed medical concepts and their weighted synonyms obtained from the the UMLS Metathesaurus are added to the query. Concerning the tuning-able parameters used in MCM model, we use the ones which were tested to be most effective in the previous experiments: for words expanded from a phrase concept, an extra weight of 2 is set and for words expanded from a term concept, an extra weight of 1.5 is set.

Rankers *BM25_ WE$_{pmc}$_ PRF$_{no}$* and *BM25_ WE$_{pmc}$_ PRF* are based on word embedding model WE$_{\text{pmc}}$, which was locally trained using PubMed dataset.

First, referring to each query term, 10 expanding words are selected from model WE_{pmc} using cosine similarity scores; then, these selected words are expanded to the query.

Topical Relevance Assessment

Results. Table 5.16 presents the topical relevance assessments for the experiments in assessing the word embeddings in QE and carried out on the CLEF'2016-2017 eHealth IR data collection. All rankers are evaluated in three evaluation metrics: P@10, NDCG@10 and MAP[22].

Table 5.16: WE in QE: rankers' topical relevance on CLEF'2016-2017 data.

Algorithm	**Ranker**	**P@10**	**NDCG@10**	**MAP**
Baselines	baselineDirichletLM_PRF$_{\text{no}}$	0.2507	0.2011	0.0733
	baselineDirichletLM_PRF	0.2297	0.1797	0.0632
	baselineBM25_PRF$_{\text{no}}$	0.2903	0.2346	0.0914
	baselineBM25_PRF	$\underline{\underline{0.3167}}$	$\underline{\underline{0.2512}}$	$\underline{\underline{0.1149}}$
	baselineTFIDF_PRF$_{\text{no}}$	0.2897	0.2335	0.0909
	baselineTFIDF_PRF	0.3147	0.2509	0.1147
UMLS	BM25_UMLS_PRF$_{\text{no}}$	0.2753	0.2225	0.0839
	BM25_UMLS_PRF	0.3020*	0.2413*	0.1025
	BM25_MCM$_{\text{UMLS}}$_PRF$_{\text{no}}$	0.2757	0.2246	0.0890
	BM25_MCM$_{\text{UMLS}}$_PRF	0.3017	0.2413*	0.1079*
Word embedding	BM25_WE$_{\text{pmc}}$_PRF$_{\text{no}}$	0.2043	0.1707	0.0392
	BM25_WE$_{\text{pmc}}$_PRF	0.2203	0.1824	0.0446

$\underline{\underline{\quad}}$: strongest baseline
* : the best score achieved by developed rankers

As it can be observed, no rank built in our experiments is able to exceed the strongest baseline. Also, we can see that using the PRF technique achieves better results than not using it.

In what concerns performance using UMLS, all four rankers built are able to outperform the two ones using word embedding model in all three evaluation metrics. When considering the performance using the UMLS Metathesaurus expansion: for P@10 measure, the best one is ranker *BM25_ UMLS_-PRF*; the best results in NDCG@10 and MAP were obtained with *BM25_-*

[22] The baselines for topical relevance assessment are the same as we built in assessing the MCM model.

MCM_{UMLS}_PRF which used MCM model, the UMLS Metathesaurus expansion and PRF technique. Also, rankers built using the MCM model did improve the performance compared to the ones using the UMLS Metathesaurus expansion only. Finally, when comparing the two rankers built with word embedding model, we can see that using PRF achieves better result.

Summing up, on CLEF'2016-2017 eHealth IR data collection, the rankers built using the UMLS Metathesaurus expansion were able to exceed the ones using pre-trained word embedding model WE_{pmc}, able to outperforming most baselines and close to the strongest baseline.

Understandability Assessment

Results. Table 5.17 presents the understandability assessment for the experiments on CLEF'2016-2017 data collection, using understandability measures RBP, uRBP and uRBPgr[23].

As it can be observed, using the PRF technique achieved better results than without using it in all rankers. Comparing the performance between using the UMLS Metathesaurus and the word embedding model, rankers built with the first techniques were able to outperform the ones built with word embedding model in all cases with a large margin. Looking at the the UMLS Metathesaurus expansion, the best result in all three measures was obtained with $BM25\_MCM_{umls}\_PRF$, which was able to surpass the strongest baseline *baselineBM25_PRF* in uRBP metric and close to the strongest baseline in other two metrics; the other three rankers were able to surpass most baselines but not the strongest baseline. Looking at the word embedding model, both rankers underperformed the strongest baseline in all understandability measures.

As a sum, when tested on CLEF'2016-2017 eHealth IR data collection, the rankers built using the UMLS Metathesaurus expansion are able to: surpass most baselines and some ranker outperforms the strongest baseline in uRBP metric; exceed the ones built with pre-trained word embedding model WE_{pmc} in all cases.

Overall Results

In this section, we carried out six experiments on CLEF'2016-2017 eHealth IR Data Collection. Six rankers were built using two different query expansion resources. The results were evaluated both in topical relevance and

[23] The baselines for understandability assessment are the same as we built in assessing the MCM model.

Table 5.17: WE in QE: rankers' understandability on CLEF'2016-2017 data.

Algorithm	**Ranker**	**RBP**	**uRBP**	**uRBPgr**
Baselines	baselineTFIDF_PRF$_{\text{no}}$	0.2961	0.1110	0.1175
	baselineTFIDF_PRF	$\underline{\underline{0.3199}}$	$\underline{\underline{0.1222}}$	$\underline{\underline{0.1273}}$
	baselineBM25_PRF$_{\text{no}}$	0.2972	0.1126	0.1189
	baselineBM25_PRF	0.3193	0.1220	0.1272
	baselineDirichletLM_PRF$_{\text{no}}$	0.2616	0.1106	0.1082
	baselineDirichletLM_PRF	0.2394	0.1082	0.1040
	baselineBM25CLI	0.0897	0.0170	0.0256
	baselineBM25GFI	0.0979	0.0213	0.0315
	baselineBM25spam50	0.2874	0.1022	0.1120
	baselineBM25spam60	0.2887	0.1054	0.1152
	baselineBM25spam70	0.2792	0.1007	0.1110
	baselineBM25spam80	0.2629	0.0890	0.1005
	baselineBM25spam90	0.2028	0.0660	0.0753
UMLS expansion	BM25_UMLS_PRF$_{\text{no}}$	0.2834	0.1106	0.1122
	BM25_UMLS_PRF	0.3006	0.1219	0.1193
	BM25_MCM$_{\text{umls}}$_PRF$_{\text{no}}$	0.2835	0.1126	0.1143
	BM25_MCM$_{\text{umls}}$_PRF	0.3107*	$\underline{\underline{0.1262{+}*}}$	0.1262*
Word embedding model	BM25_WE$_{\text{pmc}}$_PRF$_{\text{no}}$	0.2112	0.0963	0.1012
	BM25_WE$_{\text{pmc}}$_PRF	0.2304	0.1027	0.1069

$\underline{\underline{}}$: strongest baseline
* : the best score achieved by developed rankers
$\underline{\underline{+}}$: better than the strongest baseline

understandability assessments.

We can have these conclusions from the results observed on the experiments carried out on CLEF'2016-2017 data collection. For topical relevance: (i) the classic UMLS Metathesaurus expansion presented better performance than the pre-trained word embedding model WE$_{\text{pmc}}$; (ii) the rankers built using either two of the expansion resources were not able to surpass the strongest baseline; (iii) the rankers built using the UMLS Metathesaurus expansion were close to the strongest baseline; And for understandability: (i) using UMLS expansion also showed better performance than using the word embedding model; (ii) one ranker using MCM model and UMLS expansion was able to surpass the strongest baseline in uRBP and quite close to it in RBP and uRBPgr metrics.

5.3.4 CLEF2018 Data Experiments

The third experiment set is performed over CLEF2018 data collection.

Experiments

Five experiments are designed in total, as shown in Table 5.18.

Table 5.18: WE in QE: rankers built on CLEF'2018 data.

Ranker	**Methods Description**
BM25_MCM$_{\text{UMLS}}$_PRF$_{\text{no}}$	BM25, MCM: UMLS as QE, no PRF technique
BM25_MCM$_{\text{UMLS}}$_PRF	BM25, MCM: UMLS as QE, with PRF technique
InL2_MCM$_{\text{UMLS}}$_PRF$_{\text{no}}$	InL2, MCM: UMLS as QE, no PRF technique
InL2_MCM$_{\text{UMLS}}$_PRF	InL2, MCM: UMLS as QE, with PRF technique
BM25_WE$_{\text{trec-skip}}$_PRF$_{\text{no}}$	BM25, WE$_{\text{trec-skip}}$ as QE, no PRF
BM25_WE$_{\text{trec-skip}}$_PRF	BM25, WE$_{\text{trec-skip}}$ as QE, with PRF

On CLEF'2016-2017 data collection, using MCM model and UMLS expansion achieved much better results than using UMLS expansion only. On CLEF'2018 data, we selectively tested with the first method. Similar to experiment set 1, the rankers built and tested on CLEF'2018 data employed either the UMLS Metathesaurus or word embedding model query expansion, with or without using pseudo relevance feedback techniques.

The first two rankers were built using BM25 retrieval model and followed the same procedure as we took on the CLEF'2016-2017 data: the query terms identified by cTAKES were processed by MCM model and expanded using the UMLS Metathesaurus; the techniques and parameters used were set as the optimal ones where an extra weight of 2 was assigned to words expanded from a phrase concept and weight 1.5 to words expanded from a term concept. Those processed medical concepts as well as their weighted synonyms were added to the query.

Rankers *InL2_MCM$_{UMLS}$_PRF$_{no}$* and *InL2_MCM$_{UMLS}$_PRF* followed the same procedures as the first two rankers and are based on InL2 retrieval model.

The other two rankers use the pre-trained word embedding model WE$_{\text{trec-skip}}$ as the expanding resource. Simple techniques in finding related words were used: first, we use the word embedding model to find the top 10 related words returned by the model were added to the query.

Topical Relevance Assessment

Results. Table 5.19 presents the topical relevance assessments for the experiments over CLEF'2018 eHealth IR data collection. All the rankers are evaluated with three evaluation metrics: P@10, NDCG@10 and MAP[24].

Table 5.19: WE in QE: rankers' topical relevance on CLEF'2018 data.

Algorithm	**Ranker**	**P@10**	**NDCG@10**	**MAP**
Baselines	baselineDirichletLM_PRF$_{\text{no}}$	0.7120	0.6054	$\underline{\underline{0.2752}}$
	baselineDirichletLM_PRF	0.6520	0.5521	0.1455
	baselineTFIDF_PRF$_{\text{no}}$	$\underline{\underline{0.7360}}$	$\underline{\underline{0.6292}}$	0.2586
	baselineTFIDF_PRF	0.7200	0.6080	0.2526
	baselineBM25_PRF$_{\text{no}}$	0.7100	0.5919	0.2575
	baselineBM25_PRF	0.6900	0.5698	0.2471
	baselineBing	0.4940	0.4856	0.0185
UMLS	BM25_MCM$_{\text{UMLS}}$_PRF$_{\text{no}}$	0.6940	0.5949	0.2496
	BM25_MCM$_{\text{UMLS}}$_PRF	0.6820	0.5709	0.2377
	InL2_MCM$_{\text{UMLS}}$_PRF$_{\text{no}}$	$\underline{\underline{0.7480+}}$*	$\underline{\underline{0.6333+}}$*	0.2604*
	InL2_MCM$_{\text{UMLS}}$_PRF	0.7080	0.6008	0.2445
Word embedding	BM25_WE$_{\text{trec-skip}}$_PRF$_{\text{no}}$	0.5400	0.4375	0.1774
	BM25_WE$_{\text{trec-skip}}$_PRF	0.5600	0.4483	0.1742

$\underline{\underline{\ \ }}$: strongest baseline
* : the best score achieved by developed rankers
$\underline{\underline{+}}$: better than the strongest baseline

As it can be observed, ranker $InL2\_MCM_{UMLS}\_PRF_{no}$ was able to exceed the strongest baseline with evaluation metrics P@10 and NDCG@10, but underperform it with MAP; The other developed rankers were not able to outperform the strongest baseline. Comparing the effectiveness between using the UMLS Metathesaurus and the word embedding model, rankers built using the first approach were able to outperform the rankers built with word embedding model in all cases. Considering the performance with the UMLS Metathesaurus expansion, the rankers using InL2 retrieval model obtained better results than the ones using BM25 model. Finally, when we look at the two rankers using word embedding model, we can see that the one using the PRF techniques achieved better results that the one without using it in RBP and uRBP.

Concluding, and for CLEF'2018 eHealth IR data collection, all four rankers

[24] The baselines for topical relevance are the same as we built

built using the UMLS Metathesaurus expansion (using MCM model) were able to exceed the ones using pre-trained word embedding model $\text{WE}_{\text{trec-skip}}$ when evaluated with topical relevance. More specifically, one ranker using the UMLS Metathesaurus expansion was able to surpass the strongest baseline in P@10 and NDCG@10 measures.

Understandability Assessments

Results. Table 5.20 presents the understandability assessment results using understandability measures RBP, uRBP and uRBPgr[25].

Table 5.20: WE in QE: rankers' understandability on CLEF'2018 data.

Algorithm	**Ranker**	**RBP**	**uRBP**	**uRBPgr**
Baselines	baselineTFIDF_PRF$_{\text{no}}$	0.7297	$\underline{\underline{0.7370}}$	$\underline{\underline{0.3170}}$
	baselineTFIDF_PRF	0.7199	0.7348	0.3030
	baselineBM25_PRF$_{\text{no}}$	0.6987	0.7076	0.3030
	baselineBM25_PRF	0.6873	0.7006	0.2820
	baselineDirichletLM_PRF$_{\text{no}}$	$\underline{\underline{0.7735}}$	0.7241	0.3010
	baselineDirichletLM_PRF	0.6654	0.6818	0.2750
	baselineBM25CLI	0.6005	0.6126	0.2280
	baselineBM25GFI	0.5981	0.6030	0.2340
UMLS	BM25_MCM$_{\text{UMLS}}$_PRF$_{\text{no}}$	0.6907	0.7173	0.2950
	BM25_MCM$_{\text{UMLS}}$_PRF	0.6732	0.6977	0.3060
	InL2_MCM$_{\text{UMLS}}$_PRF$_{\text{no}}$	$\underline{\underline{0.7396+}}$*	$\underline{\underline{0.7506+}}$*	$\underline{\underline{0.3260+}}$*
	InL2_MCM_UMLS_PRF	0.7168	0.7297	0.3000
Word embedding	BM25_WE$_{\text{trec-skip}}$_PRF$_{\text{no}}$	0.5520	0.5579	0.2370
	BM25_WE$_{\text{trec-skip}}$_PRF	0.5543	0.5672	0.2320

$\underline{\underline{\ }}$: strongest baseline
* : the best score achieved by developed rankers
$\underline{\underline{+}}$: better than the strongest baseline

Comparing rankers to the baseline, one ranker built using the UMLS Metathesaurus expansion was able to exceed the strongest baseline in all cases while the other developed rankers underperform the strongest baseline. Looking at the effectiveness between using the UMLS Metathesaurus and the word embedding model, rankers built using the first technique were able to outperform the rankers built with word embedding model in all cases with a large

[25] The baselines for understandability assessment are the same as we built

margin. Considering the performance with the UMLS Metathesaurus expansion, the best results in all three measures were obtained with ranker $InL2\_MCM_{UMLS\_}PRF_{no}$, surpassing the strongest baseline with a large margin. Ranker built using InL2 retrieval model achieved better results than the ones using BM25 model in most cases. Finally, and looking at the use of word embedding model, both rankers underperformed the strongest baseline in all understandability measures.

Concluding, the rankers built using the UMLS Metathesaurus expansion were able to: (i) exceeded the ones built with pre-trained word embedding model $\text{WE}_{\text{trec-skip}}$ in all cases; (ii) the ranker using InL2 retrieval model improved the strongest baseline in all evaluation metrics.

Overall Results

In this section, we carried out experiments on CLEF'2018 eHealth IR Data Collection. Six rankers were built with two different query expansion resources and the results were evaluated both in terms of topical relevance and understandability.

From the results, one can conclude that: (i) the classic UMLS Metathesaurus expansion achieve better performance than our pre-trained word embedding model $\text{WE}_{\text{trec-skip}}$; (ii) in both topical relevance and understandability assessment, the ranker built using UMLS expansion resources and InL2 retrieval model obtained obvious improvement over the strongest baseline in almost all evaluation metrics.

5.3.5 Conclusions

During Research Phase 1 Task B, we proposed the idea of using different query expansion resources to to bridge the language gap between non-expert consumers and medical experts. In this section, we compared the performance between classic UMLS Metathesaurus and pre-trained word embedding models on three sets of data collections. First, we carried out the experiments on FIRE'2016 CHIS data collection and compared expansion performance between the UMLS Metathesaurus and two word embedding models WE_{wiki} and WE_{pmc}. Then, on CLEF2016-2017 eHealth IR data collections, we compared UMLS to the WE_{pmc} model. And the third experiment set was performed on CLEF2018 eHealth IR Data Collection, comparing UMLS to the $\text{We}_{\text{trec-skip}}$ model.

First, we make conclusions based on our observations of the results obtained from the experiments carried out on the FIRE CHIS data collection: both word embedding models $\mathrm{WE}_{\mathrm{wiki}}$ and $\mathrm{WE}_{\mathrm{pmc}}$ achieved much better results than using the UMLS Metathesaurus. Moreover, rankers built using the Vector Model PubMed expansion scored the best results in our experiments and greatly outperformed other teams scores. We can conclude that on FIRE CHIS data, using vectors model as expansion was more effective than using UMLS and can greatly improve the state-of-the-art techniques in CHIR.

Next, we turn to the experiments performed on CLEF eHealth data collection. We observed totally different results from the ones we obtained from the FIRE data: (i) classic UMLS Metathesaurus expansion achieved better performance than the word embedding models on both two data collections in all designed experiments; (ii) when evaluated with topical relevance assessment, the ranker built combining UMLS Metathesaurus and the MCM model was able to perform the best of all and able to surpass the strongest baselines in most cases; (iii) when evaluated with understandability assessment, we can see that the ranker built using UMLS Metathesaurus expansion and MCM model was able to outperform the strongest baseline in all cases. We can conclude that: from our testing on CLEF eHealth data, combining classic UMLS Metathesaurus expansion with the proposed Medical Concept Model was an effective approach in improving state-of-the-art techniques and especially in improving understandability in the area of CHIR.

Chapter 6

Learning Understandability from Experience

LETOR approach has been successfully used to improve information retrieval performance as well as been studied in the context of consumer health information retrieval.

In the two-stage LETOR module, we propose using LETOR approach to promote understandability in our research: first exploring field-based features to training single field-based LETOR models; then combining these LETOR models using different rank aggregation methods.

In this chapter, we first preliminarily test the validity of our proposed approach on CLEF'2016-2017 eHealth IR data collection.

Then we design the following experiments and carry them out on the three sets of CLEF eHealth IR data collections: first, we plan to train a number of field-based LETOR models using CLEF'2016-2017 eHealth IR data collections; next, we apply and test these LETOR models on CLEF'2018 eHealth IR data collection during retrieval process, where a set of ranking lists are to be produced based on each LETOR model respectively; then, new and combined ranking lists are generated by applying different aggregation methods on this set of ranking lists; and finally, all produced ranking lists or results are to be assessed using the assessments files with the introduced evaluation metrics. At the end of this section, we draw a conclusion over our experiments and the observations from the results.

6.1 Preliminary Experiments

To test the effectiveness of field-based features in training a LETOR model, we first carry out a group of experiments on CLEF'2016-2017 data collection. These experiments can be generally explained as: first, two groups of features are defined, one composed of field-based features and other of non-field based features; then, two LETOR rankers are trained using each of the groups of features; finally, a group of rankers are built using simple linear combination methods on the two LETOR rankers. All rankers are evaluated with NDCG@10 metric and compared to a group of state-of-the-art baselines.

6.1.1 Features

We start with feature exploring. When applying LETOR techniques, the very first and important step is to define the features. Potential useful features are gathered in a feature list and used to train a model afterwards.

To compare the performance between field-based and non-field features, two groups of features, $\mathrm{F}_{\text{field}}$ and $\mathrm{F}_{\text{non-field}}$ are defined, as shown in Table 6.1.

Table 6.1: Two-stage LETOR model: features defined on CLEF'2016-2017 data.

Group	**Nr.**	**Features**	**Description**
$\mathrm{F}_{\text{field}}$	1	Dl,Title	score of text length on Title
	2	Dl,H1	score of text length on H1
	3	Dl,Else	score of text length on Else
	4	BM25,Title	score of BM25 on Title
	5	BM25,H1	score of BM25 on H1
	6	BM25,Else	score of BM25 on Else
$\mathrm{F}_{\text{non-field}}$	7	Dl	score of text length on whole document
	8	BM25	score of BM25 on whole document
	9	TFIDF	average score of tf*idf of query terms on whole document
	10	DirichletLM	score of LM with Dirichlet smoothing on whole document
	11	HiemstraLM	score of Hiemstra's language model on whole document
	12	LemurTFIDF	score of LemurTFIDF on whole document

Feature group $\mathrm{F}_{\text{field}}$ includes 6 field-based features extracted from three fields: Title, H1 and Else; IR weighting model BM25 and Dl (Document length) are used during the retrieval process to obtain these features. $\mathrm{F}_{\text{non-field}}$ group includes 6 non-field features extracted using seven state-of-the-art IR weighting models on the whole document. All 12 features in both groups are query and document dependent.

6.1.2 Experiments

To apply LETOR techniques, first, CLEF'2016-2017 query set is divided into into three parts: train, validation and test. The query set is carefully and equally split so that queries coming from the same post can have the same split. Then, five fold cross validation is used when to train a LETOR model and tuned with the validation set. Concerning LETOR algorithms available in IR, LambdaMART is chosen since this algorithm is shown to achieve better performances when compared to other algorithms in previous work (Soldaini and Goharian, 2017).

Totally 6 rankers are designed, as shown in Table 6.2. Rankers $R_{\textit{field}}$ and $R_{\textit{non-field}}$ are two LETOR models trained using features from $\mathrm{F}_{\text{field}}$ and $\mathrm{F}_{\text{non-field}}$ groups respectively. Rankers $R_{\textit{sum}}$, $R_{\textit{sum-w}}$, $R_{\textit{sum-28}}$ and $R_{\textit{sum-82}}$ are built using simple linear combination methods on the two basic LETOR rankers $R_{\textit{field}}$ and $R_{\textit{non-field}}$.

Table 6.2: Two-stage LETOR model: rankers trained on CLEF'2016-2017 data.

Ranker	**Method Description**
$\mathrm{R}_{\text{field}}$	LETOR model trained using $\mathrm{F}_{\text{field}}$ features
$\mathrm{R}_{\text{non-field}}$	LETOR model trained using $\mathrm{F}_{\text{non-field}}$ features
R_{sum}	sum the results from $\mathrm{R}_{\text{field}}$ and $\mathrm{R}_{\text{non-field}}$
$\mathrm{R}_{\text{sum-w}}$	weighted sum the results from $\mathrm{R}_{\text{field}}$ and $\mathrm{R}_{\text{non-field}}$
$\mathrm{R}_{\text{sum-28}}$	$0.2 \times \text{Score } (\mathrm{R}_{\text{field}}) + 0.8 \times \text{Score } (\mathrm{R}_{\text{non-field}})$
$\mathrm{R}_{\text{sum-82}}$	$0.8 \times \text{Score } (\mathrm{R}_{\text{field}}) + 0.2 \times \text{Score}$

When building $R_{\textit{sum}}$, for every query-document pair, we sum the pair's score obtained from $R_{\textit{field}}$ and $R_{\textit{non-field}}$; then the summed score is used as the new score and re-ranking is performed according to this newly summed score. For ranker $R_{\textit{sum-w}}$, we calculate the weights like this: first, we use number 100 divide the best score obtained from each group accordingly, the quotient is used as the weight for each group; then, score obtained from each group is multiplied by this calculated weight; finally, the addition of the weighted two scores from each group are used as the final results. Rankers $R_{\textit{sum-28}}$

and $R_{sum\text{-}82}$ follow the same techniques as $R_{sum\text{-}w}$, except that we set the weights to fixed numbers. For $R_{sum\text{-}28}$, we assign 0.2 as the weight to the best results obtained in group 1, and 0.8 to group 2. And for $R_{sum\text{-}82}$, we assign 0.8 to the best results obtained in group 1 and 0.2 to group 2 [1].

6.1.3 Results

Five indexes using different indexing algorithms were available from the CLEF'2017 eHealth IR task. In our work, we note them as SBN, SBY, PTN, PTY and Field index accordingly, with the name implying different stemming techniques, stop words removal, and field information. We build one baseline on each index on Terrier platform: the retrieval is performed on the original queries without pre-processing; IR weighting model BM25 was used during the retrieval process and all parameters were set to default. Although the baselines used very simple techniques, all participants teams were not able to outperform the best baseline available from the organizers (Soldaini and Goharian, 2017). In our experiments, we use these group of baselines as comparison to our developed models. As shown in Table 6.3, the five baselines were evaluated with NDCG@10 metric[2] and *baselinePTY* with a score of 0.2412 was the strongest one.

The results for the six designed experiments evaluated with NDCG@10 were presented in Table 6.3 as well[3].

As the two LETOR rankers were built using different indexes: R_{field} on field index and $R_{non\text{-}field}$ on PTY index, the results are specially analyzed and compared to *baselineField* and *baselinePTY*.

From the results, we can see that ranker R_{field} was able to surpass the strongest baseline *baselinePTY* and improved *baselineField* with a large margin. Ranker $R_{non\text{-}field}$ was not able to surpass the strongest baseline. All rankers built using linear combinations over R_{field} and $R_{non\text{-}field}$ were able to surpass the strongest baselines. More specifically, the best result was obtained with R_{sum} which simply sums up the scores of R_{field} and $R_{non\text{-}field}$. Furthermore, comparing R_{sum} to R_{field}, we see that the combination reinforced the improvement of $\mathrm{R_{field}}$: an improvement of 1.3% was obtained compared R_{sum} to R_{field} (from 12.79% to 14%). And for $R_{non\text{-}field}$, which has lower performance than the strongest baseline, improved this baseline when

[1] We choose 0.2 and 0.8 as the weighted score since this two valued presented the best performance with our tested experimental iteration.

[2] The results were evaluated with the help of task organizers and only NDCG@10 was available.

[3] This set of experiments were evaluated with the help of task organizers and only NDCG@10 evaluation results were available.

Table 6.3: Two-stage LETOR model: rankers tested on CLEF'2016-2017 data.

Algorithm	**Ranker**	**NDCG@10**	**Vs. baselineField**	**Vs. baselinePTY**
Baselines	baselineSBN	0.2286	-	-
	baselineSBY	0.2405	-	-
	baselinePTN	0.2295	-	-
	baselinePTY	0.2412	-	-
	baselineField	0.2246	-	-
LETOR	R_{field}	0.2533+	12.79%	5.02%
	$R_{\text{non-field}}$	0.2341	4.23%	-0.71%
Combination	R_{sum}	0.2557+*	14.00%	6.02%
	$R_{\text{sum-w}}$	0.2520+	12.20%	4.48%
	$R_{\text{sum-28}}$	0.2506+	11.58%	3.90%
	$R_{\text{sum-82}}$	0.2481+	10.46%	2.86%

= : strongest baseline
+ : better than the strongest baseline
* : the best score achieved by developed rankers

combined with R_{field}.

6.1.4 Conclusions

This section preliminarily tests our proposed two-stage LETOR model on CLEF'2016-2017 eHealth IR data collections. Two feature groups were defined including 6 features and 7 non-field features. Six rankers were built: two were LETOR models, each was trained using one group of features, respectively; the other four were built using simple linear combination methods on the two original LETOR models. All rankers were evaluated using NDCG@10 and compared to state-of-the-art baselines.

Conclusions can be made based on the observations obtained: (i) field-based features were shown to be more effective than non-field features; (ii) the ranker built using field-based features was able to improve the strongest baseline with a large margin; (iii) rankers built using combined methods surpassed the strongest baselines in all cases; (iv) the combined ranker using simple sum combination improved of the one using field-based features only.

We can clearly conclude that using field-based features to train a model is an effective solution in applying LETOR techniques in CHIR. And the combination of different LETOR models can enhance the improvement, which presents rank combination can be a very useful approach in improving state-of-the-art techniques in CHIR.

6.2 Rank Aggregation on Field-based LETOR Models

Based on the observations obtained from the preliminary experiments, the following experiments are designed: a set of LETOR models using single-field features are trained and then these LETOR models are combined by applying different rank aggregation methods.

These experiments are carried out on the three sets of CLEF eHealth IR data collections. First, we train a number of field-based LETOR models using CLEF'2016-2017 eHealth IR data collections; next, we apply and test these LETOR models on CLEF'2018 eHealth IR data collection during retrieval process, where a set of ranking lists are to be produced based on each LETOR model respectively; then, new and combined ranking lists are generated by applying different aggregation methods on this set of ranking lists; and finally, all produced ranking lists or results are to be assessed using the assessments files with the introduced evaluation metrics.

6.2.1 Features

Following a similar feature exploration as in LETOR Benchmark dataset by Liu et al. (2007), here we extract features on four fields: Title, H1, Else[4] and full text of a document[5]. We also consider the features that perform well in consumer health search applications mentioned in two related works (Soldaini and Goharian, 2017; Palotti et al., 2016). A total set of 36 features are used in our experiments and they are presented in Table 6.4. These features are mostly based on 8 classic IR weighting models except feature 1, 10, 19 and 28 which are query independent and extracted from text length information. Here, features are clustered into four groups based on the four different fields of a document that the features are extracted from.

[4] Else field means all the other parts of a document besides Title and H1.

[5] We regard the full text as a field information as well.

Table 6.4: Two-stage LETOR model: features explored on CLEF'2016-2018 data.

Group	Nr.	Features	Description
F_{title}:	1	Text Length,Title	Length of title.
title features	2	TF,Title	Average term frequency of query terms on title.
	3	IDF	Average inverse document frequency of query terms on title.
	4	TFIDF,Title	Average score of tf*idf of query terms on title.
	5	BM25,Title	Score of BM25 on title.
	6	HiemstraLM,Title	Score of Hiemstra's language model on title
	7	DirichletLM,Title	Score of LM with Dirichlet smoothing on title
	8	PL2,Title	Score of PL2 on title
	9	BB2,Title	Score of PL2 on title
F_{H1}:	10	Text Length,H1	Length of H1.
H1 features	11	TF,H1	Average term frequency of query terms on H1.
	12	IDF,H1	Average inverse document frequency of query terms on H1.
	13	TFIDF,H1	Average score of tf*idf of query terms on H1.
	14	BM25,H1	Score of BM25 on H1.
	15	HiemstraLM,H1	Score of Hiemstra's language model on H1
	16	DirichletLM,H1	Score of LM with Dirichlet smoothing on H1.
	17	PL2,H1	Score of PL2 on H1.
	18	BB2,H1	Score of BB2 on H1.
F_{Else}:	19	Text Length,Else	Length of Else.
Else features	20	TF,Else	Average term frequency of query terms on Else.
	21	IDF,Else	Average inverse document frequency of query terms on Else.
	22	TFIDF,Else	Average score of tf*idf of query terms on Else.
	23	BM25,Else	Score of BM25 on Else.
	24	HiemstraLM,Else	Score of Hiemstra's language model on Else
	25	DirichletLM,Else	Score of LM with Dirichlet smoothing on Else.
	26	PL2,Else	Score of PL2 on Else.
	27	BB2,Else	Score of BB2 on Else.
F_{full}:	28	Text Length	Length of the whole document.
whole document	29	TF	Average term frequency of query terms on the whole document.
features	30	IDF	Average inverse document frequency of query terms on the whole document.
	31	TFIDF	Average score of tf*idf of query terms on the whole document.
	32	BM25	Score of BM25 on the whole document.
	33	HiemstraLM	Score of Hiemstra's language model on Else.
	34	DirichletLM	Score of LM with Dirichlet smoothing on the whole document
	35	PL2	Score of PL2 on the whole document.
	36	BB2	Score of BB2 on the whole document.

6.2.2 Experiments

Hypothesizing that different field information contributes differently to a document, a crude combination of all features extracted from different fields together will make the features blur; also, an aggregated models which is aggregated on a set of models trained using single field features each is supposed to be more effective than a model trained including in all fields features together.

To prove these hypotheses, we design a two-stage LETOR framework and different aggregation methods are applied on a set of LETOR models which are presented in Table 6.5.:

- During the first stage, five LETOR rankers were built in total. R_T, R_H, R_E and R_F are built using the four feature groups respectively; R_A is built using all blindly joint 36 features.
- During the second stage, the results from these four field-based LETOR rankers are aggregated. For example, R_{TH} means combining the results from R_T and R_H.

Table 6.5: LETOR model: rankers trained on CLEF'2016-2017 data.

Ranker	**Method Description**
$\mathrm{R_T}$	LETOR model trained with $\mathrm{F_{title}}$ features
$\mathrm{R_H}$	LETOR model trained with $\mathrm{F_{H1}}$ features
$\mathrm{R_E}$	LETOR model trained with $\mathrm{F_{Else}}$ features
$\mathrm{R_F}$	LETOR model trained with $\mathrm{F_{full}}$ features
$\mathrm{R_A}$	LETOR model trained with all 36 features
$\mathrm{R_{TH}}$	aggregation over $\mathrm{R_T}$ and $\mathrm{R_H}$
$\mathrm{R_{THE}}$	aggregation over $\mathrm{R_T}$, $\mathrm{R_H}$ and $\mathrm{R_E}$
$\mathrm{R_{THEF}}$	aggregation over $\mathrm{R_T}$, $\mathrm{R_H}$, $\mathrm{R_E}$ and $\mathrm{R_F}$

We use the learning to rank framework provided in Terrier 4.27 with Jforest and LambdaMART (Macdonald et al., 2013). To train a LETOR model, Okapi BM25 weighting model is used with all parameters set to default values; up to 1,000 documents per query are retrieved during retrieval process; all models are trained and tuned with a separate validation set from CLEF'2016-2017 data collection and tested on CLEF'2018 data.

6.2.3 Topical Relevance Results

In this section, we first compare the performance among all built rankers as well as to the baselines in topical relevance assessment metrics. Next, we analyze the effect of how different aggregation methods vary the aggregation performance. Then, we look at how different fields and their combination affect the retrieval performance.

Aggregated Rankers Evaluation

We use the same CLEF'2018 data baselines as we built in previous tasks, including seven topical relevance baselines. Table 6.6 presents the topical relevance assessments for the experiments over CLEF 2016-2018 eHealth IR data collections. All rankers are evaluated in three evaluation metrics: P@10, NDCG@10 and MAP.

Considering aggregated rankers R_{THEF}, as it can be observed, $R_{THEF\_}$-*med* and $R_{THEF\_}\,sum$ outperformed the strongest baseline *baselineTFIDF_*-PRF_{no} with all three evaluation metrics[6]. And other aggregated R_{THEF} were able to surpass baselineBM25_PRF which was built using the same experimental settings as our developed rankers in most cases.

Looking at rankers R_{THE}. Specifically, all R_{THE} were able to surpass baselineBM25_PRF with NDCG@10 evaluation metric. $R_{THE\_}\,med$ and $R_{THE\_}$-*sum* were able to exceed baselineBM25_PRF in P@10 and MAP metrics.

Then, we look at rankers R_{TH}. $R_{TH\_}\,max$ was able to surpass baselineBM25_-PRF in P@10; four $\mathrm{R_{TH}}$ (aggregated using CombMAX, CombMED, CombMNZ and CombSUM) were able to exceed this baseline in NDCG@10; and three R_{TH} (aggregated using CombMAX, CombMED and CombSUM) outperformed this baseline in MAP.

Turning our attention to R_A, which was trained with all 36 features blindly joined. It failed to exceed all baseline except *baselineBing* at P@10; with NDCG@10 and MAP metric, this ranker was only able to exceed the lowest baselines $DirichletLM_{PRF}$ and *baselineBing*, underperforming all the other baselines.

We now look at the LETOR models trained with single field features. We observe that R_T trained with title field outperformed baselineBM25_PRF with NDCG@10 metrics. R_H, R_E and R_F performed almost similarly, failing to exceed most baselines. Nevertheless, when comparing these field-based rankers to R_A, some of them (R_T and R_F) still outperformed R_A.

[6] CombSUM and CombMED presented the same performance in our experiments.

Table 6.6: Two-stage LETOR model: rankers' topical relevance on CLEF'2018 data.

Algorithm	**Ranker**	**P@10**	**NDCG @10**	**MAP**
Baselines	baselineTFIDF_PRF$_{\text{no}}$	$\underline{\underline{0.7360}}$	$\underline{\underline{0.6292}}$	$\underline{\underline{0.2586}}$
	baselineTFIDF_PRF	0.7200	0.6080	0.2526
	baselineBM25_PRF$_{\text{no}}$	0.7100	0.5919	0.2575
	baselineBM25_PRF	$\underline{0.6900}$	$\underline{0.5698}$	$\underline{0.2471}$
	baselineDirichletLM_PRF$_{\text{no}}$	0.7120	0.6054	0.2752
	baselineDirichletLM_PRF	0.6520	0.5521	0.1455
	baselineBing	0.4940	0.4856	0.0185
LETOR (field-based features)	R$_{\text{T}}$	0.6820	0.6131+	0.2428
	R$_{\text{H}}$	0.6340	0.5683	0.2279
	R$_{\text{E}}$	0.5700	0.4753	0.2115
	R$_{\text{F}}$	0.6620	0.5395	0.2404
LETOR(all features)	R$_{\text{A}}$	0.6420	0.5687	0.2177
Aggregation	R$_{\text{TH}}$_anz	0.6300	0.5751	0.1831
	R$_{\text{TH}}$_max	0.6900+	0.6128+	0.2479+
	R$_{\text{TH}}$_med	0.6740	0.6101+	0.2492+
	R$_{\text{TH}}$_min	0.6540	0.5847	0.2341
	R$_{\text{TH}}$_mnz	0.6580	0.6017+	0.1845
	R$_{\text{TH}}$_sum	0.6740	0.6101+	0.2492+
	R$_{\text{THE}}$_anz	0.6820	0.6176+	0.1808
	R$_{\text{THE}}$_max	0.6800	0.5919+	0.2413
	R$_{\text{THE}}$_med	0.7040+	0.6246+	0.2500+
	R$_{\text{THE}}$_min	0.6660	0.5886+	0.2310
	R$_{\text{THE}}$_mnz	0.6820	0.6118+	0.1822
	R$_{\text{THE}}$_sum	0.7040+	0.6246+	0.2500+
	R$_{\text{THEF}}$_anz	0.7060+	0.6214+	0.1889
	R$_{\text{THEF}}$_max	0.7060+	0.5937+	0.2542+
	R$_{\text{THEF}}$_med	$\underline{\underline{0.7440+}}$*	$\underline{\underline{0.6630+}}$*	$\underline{\underline{0.2660+}}$*
	R$_{\text{THEF}}$_min	0.6840	0.6061+	0.2414
	R$_{\text{THEF}}$_mnz	0.7200+	0.6516+	0.1903
	R$_{\text{THEF}}$_sum	$\underline{\underline{0.7440+}}$*	$\underline{\underline{0.6630+}}$*	$\underline{\underline{0.2660+}}$*

= : strongest baseline
_ : baseline using the same experimental settings as the built rankers
$\underline{+}$: better than the baseline using the same experimental settings as the built rankers
$\underline{\underline{+}}$: better than the strongest baseline
* : the best score achieved by developed ranker

Summing up, with all three evaluation metrics, the results clearly show that the aggregated rankers produced very competitive results as the best solutions. These aggregated rankers exceeded the ranker trained using blindly joint features R_A with a large space, and showed to be superior in all cases.

Aggregation Methods Analyses

Now we look at the performance of the different aggregation methods on LETOR models in terms of topical relevance assessments. We conducted an empirical evaluation to better understand the effectiveness of how different aggregation methods performed, as shown in Figure 6.1. The effectiveness of aggregation methods were evaluated by analyzing with three assessment metrics (P@10, NDCG@10 and MAP), varying the aggregation method on R_{TH}, R_{THE} and R_{THEF} respectively. The results were compared with the strongest baseline $baselineTFIDF\_PRF_{no}$ and the baseline using same experimental settings ($baselineBM25\_PRF$).

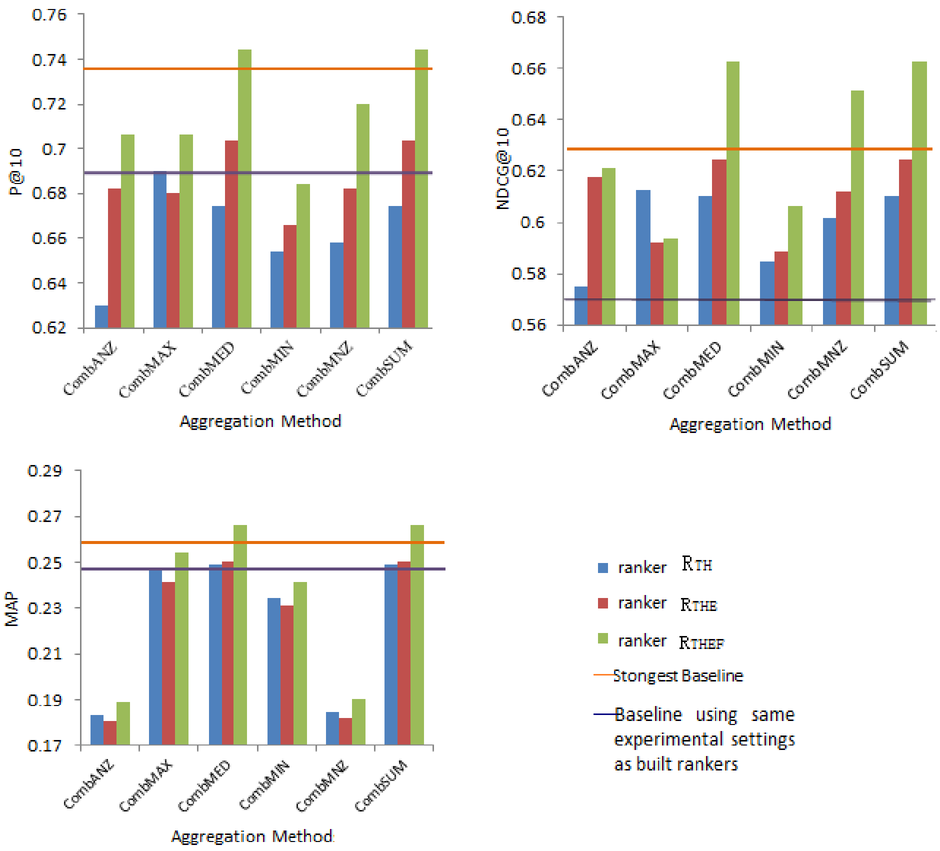


Figure 6.1: Rank aggregation performance on topicality assessment.

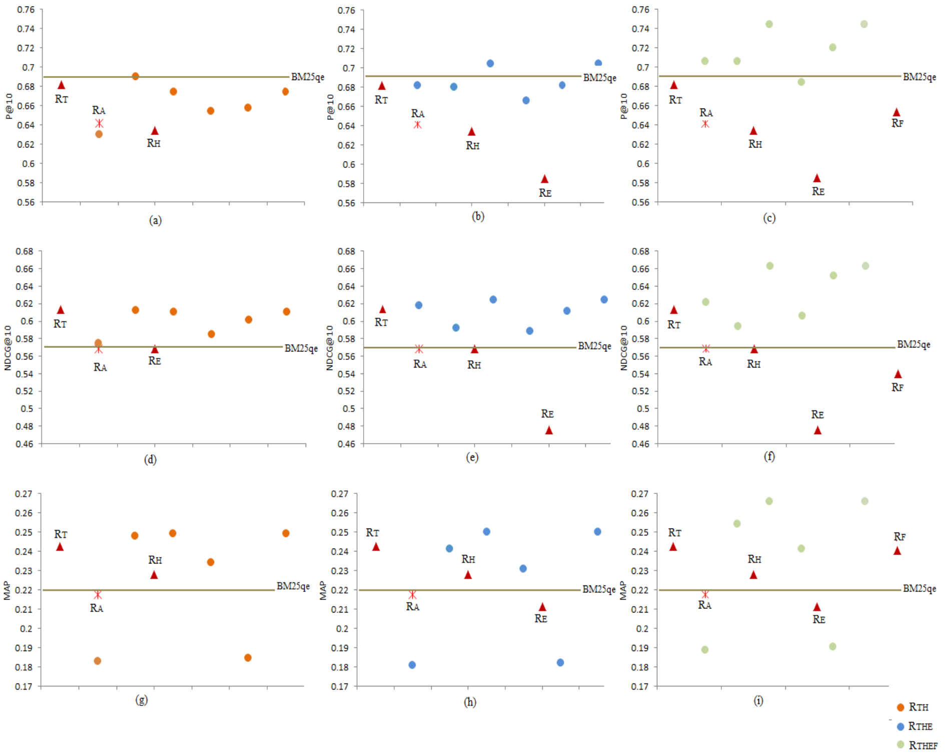


Figure 6.2: Performance of field features on topicality assessment.

As it can be observed and specially, with all three assessment metrics, CombMED and CombSUM performed the same and achieved the best scores compared to other four aggregation methods. Considering P@10 only, we can observe that CombMIN method performed the worst, underperforming baselineBM25_PRF in all three rankers. For NDCG@10, we can see that all rankers using these aggregation methods were able to outperform baselineBM25_PRF. Finally, with MAP, CombANZ and CombMNZ performed almost the same and far from the baseline; CombMAX achieved almost similar results as CombMED and CombSUM.

Comparison between Field Features

Now we look at how different fields and their combination affect the retrieval performance in topical relevance assessment. As shown in Figure 6.2, picture (a), (b) and (c) present the evaluation in P@10 metric; picture (d), (e) and (f) are in NDCG@10 metric; picture (g), (h) and (i) are in MAP metric.

We first look at the results evaluated in P@10,. As it can be observed in picture (c), *title* features (R_T) are shown to be the most effective compared to *full document* features (R_F), *H1* features (R_H) and *else* features (R_E);

blindly combined features (R_A) are able to surpass *H1* and *else* features, underperforming *title* and *full document* features; no rankers built using these features separately are able to surpass the baseline. Moreover, as seen from picture (a) to (c), aggregating more features can greatly improve the retrieval performance; most of the rankers which were built over all aggregated features (*title*, *H1*, *else* and *full document* features) are able to surpass the baseline. Next, we consider the performance in NDCG@10. As shown in picture (f), *title* features (R_T) present to be most effective, followed by the blindly combined features (R_A), then *H1* and *full document* features, *else* features are the most ineffective. Similarly, as seen from picture (d) to (f): all aggregated rankers are able to surpass the baselines; aggregating more features can achieve better retrieval performance. Finally, with MAP, as seen in picture (i), *title* features perform the best, followed by *full document* features, then *H1* and blindly combined features, *else* features present to be the most ineffective ones. Comparing picture (g) to (h), the aggregated rankers achieve similar results, no obvious improvements are observed when aggregating *else* features. As shown in picture (i), when aggregating full document features, we can obtain better results.

6.2.4 Understandability Results

In this section, we first compare the performance among all built rankers as well as to the baselines in understandability assessment metrics. Next, we analyze the effect of how different aggregation methods vary the aggregation performance. Then, we look at how different fields and their combination affect the retrieval performance.

Aggregated Rankers Evaluation

We use the same CLEF'2018 data baselines as we built in previous tasks, including eight understandability baselines. Table 6.7 presents the understandability results for the experiments over CLEF 2016-2018 data. All rankers are evaluated on three evaluation metrics: RBP, uRBP and uRBPgr.

As it can be observed, for aggregated rankers R_{THEF}, $R_{THEF\_}$*med*, $R_{THEF\_}$*mnz* and $R_{THEF\_}$*sum* outperformed the strongest baseline *baselineTFIDF_PRF*$_{no}$ on all three evaluation metrics. The other aggregated R$_{\text{THEF}}$ were able to surpass *baselineBM25_PRF* which was built using the same experimental settings as the developed rankers in all cases.

Next, for rankers R_{THE}, $R_{THE\_}$*med*, $R_{THE\_}$*mnz* and $R_{THE\_}$*sum* were able to exceed *baselineBM25_PRF* in all three metrics; $R_{THE\_}$*anz* and $R_{THE\_}$*max* surpassed this baseline in RBP and uRBPgr metrics; on the contrary,

Table 6.7: Two-stage LETOR model: ranker understandability on CLEF2018 data.

Algorithm	**Ranker**	**RBP**	**uRBP**	**uRBPgr**
Baselines	baselineTFIDF_PRF$_{\mathrm{no}}$	$\underline{\underline{0.7297}}$	$\underline{\underline{0.7370}}$	$\underline{\underline{0.3170}}$
	baselineTFIDF_PRF	0.7199	0.7348	0.3030
	baselineBM25_PRF$_{\mathrm{no}}$	0.6987	0.7076	0.3030
	baselineBM25_PRF	$\underline{0.6873}$	$\underline{0.7006}$	$\underline{0.2820}$
	baselineDirichletLM_PRF$_{\mathrm{no}}$	0.7735	0.7241	0.3010
	baselineDirichletLM_PRF	0.6654	0.6818	0.2750
	baselineBM25CLI	0.6005	0.6126	0.2280
	baselineBM25GFI	0.5981	0.6030	0.2340
LETOR	$\mathrm{R_T}$	0.7034+	0.7131+	0.3020+
(field-based	$\mathrm{R_H}$	0.6395	0.6493	0.2630
features)	$\mathrm{R_E}$	0.5705	0.5849	0.2380
	$\mathrm{R_F}$	0.6463	0.6539	0.2750
LETOR(all features)	$\mathrm{R_A}$	0.6332	0.5821	0.2550
Aggregation	$\mathrm{R_{TH}}$_anz	0.6739	0.6650	0.2880+
	$\mathrm{R_{TH}}$_max	0.6960+	0.6952	0.2950+
	$\mathrm{R_{TH}}$_med	0.7078+	0.7055+	0.3000+
	$\mathrm{R_{TH}}$_min	0.6767	0.6747	0.2840+
	$\mathrm{R_{TH}}$_mnz	0.7005+	0.6921	0.2980+
	$\mathrm{R_{TH}}$_sum	0.7078+	0.7055+	0.3000+
	$\mathrm{R_{THE}}$_anz	0.6941+	0.6990	0.2890+
	$\mathrm{R_{THE}}$_max	0.6953+	0.6948	0.2930+
	$\mathrm{R_{THE}}$_med	0.7211+	0.7220+	0.3100+
	$\mathrm{R_{THE}}$_min	0.6746	0.6766	0.2770
	$\mathrm{R_{THE}}$_mnz	0.6989+	0.7039+	0.2960+
	$\mathrm{R_{THE}}$_sum	0.7211+	0.7220+	0.3100+
l	$\mathrm{R_{THEF}}$_anz	0.7201+	0.7214+	0.3050+
	$\mathrm{R_{THEF}}$_max	0.7114+	0.7134+	0.2900+
	$\mathrm{R_{THEF}}$_med	$\underline{\underline{0.7664+*}}$	$\underline{\underline{0.7658+*}}$	$\underline{\underline{0.3280+*}}$
	$\mathrm{R_{THEF}}$_min	0.7027+	0.7002+	0.2930+
	$\mathrm{R_{THEF}}$_mnz	$\underline{\underline{0.7468+}}$	$\underline{\underline{0.7479+}}$	$\underline{\underline{0.3170+}}$
	$\mathrm{R_{THEF}}$_sum	$\underline{\underline{0.7664+*}}$	$\underline{\underline{0.7658+*}}$	$\underline{\underline{0.3280+*}}$

= : strongest baseline
_ : baseline using the same experimental settings as the built rankers
$\underline{+}$: better than the baseline using the same experimental settings as the built rankers
$\underline{\underline{+}}$: better than the strongest baseline
* : the best score achieved by developed ranker

$R_{THE\_}min$ was not able to exceed this baseline.

Then, for rankers R_{TH}, all R_{TH} were able to surpass *baselineBM25_PRF* with uRBPgr evaluation metric. Four R_{TH} (aggregated using CombMAX, CombMED, CombMNZ and CombSUM) were able to exceed *baselineBM25_-PRF* in RBP; two rankers $R_{TH\_}med$ and $R_{TH\_}sum$ were able to surpass this baseline in uRBP.

Turning our attention to R_A, which was trained with all 36 features blindly joined. It underperformed most baselines in RBP and uRBPgr metrics; underperformed all baseline on uRBP metric.

Considering LETOR models trained with single field features, we observe that R_T trained with title field outperformed baseline BM25_PRF with all three metrics. R_H, R_E and R_F performed almost similarly, failing to exceed this baselines. When comparing these field-based rankers to R_A, some of them (R_T, R_E and R_F) still outperformed R_A obviously.

Summing up, with all three evaluation metrics, the results clearly show that the aggregated rankers produced very competitive results as the best solutions. These aggregated rankers exceeded the ranker trained using blindly joint features R_A with a large space and showed to be superior in all cases.

Aggregation Methods Analyses

We now consider how different aggregation methods perform with understandability assessments.

Figure 6.3 depicts the effectiveness of aggregation methods which were evaluated by analyzing three assessment metrics (RBP, uRBP and uRBPgr), with varying aggregation method on R_{TH}, R_{THE} and R_{THEF} respectively. The results were compared with the strongest baseline *baselineTFIDF_PRF*$_{no}$ and the baseline *baselineBM25_PRF* using same experimental settings .

As it can be observed, for all three metrics, the best results were obtained by CombMED and CombSUM which achieved all the best scores in all cases; next was CombMNZ and then CombANZ and CombMAX which achieved similar results. Finally, CombMIN presented the worst performance compared with other aggregation methods.

Looking at RBP only, we can see that three rankers built using CombMED, CombMNZ or CombSUM methods were able to surpass the strongest baseline. A few rankers built using CombANZ and CombMIN were not able to exceed the baseline using same experimental settings.

Considering uRBPgr, we can observe that only rankers built using Comb-

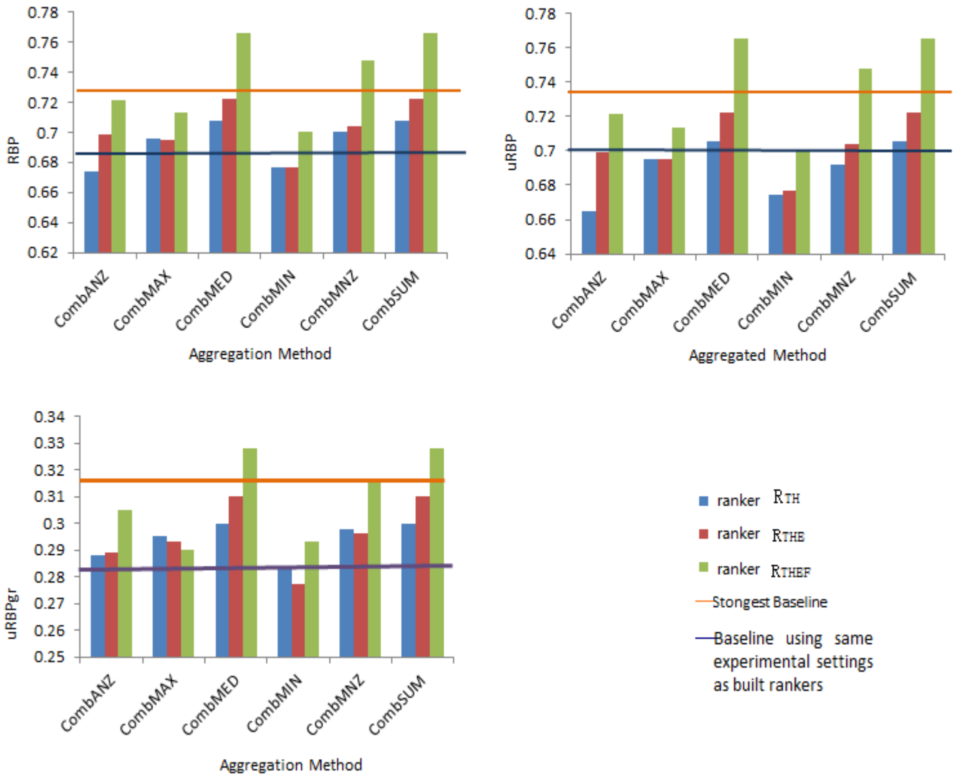


Figure 6.3: Rank aggregation performance on understandability assessment.

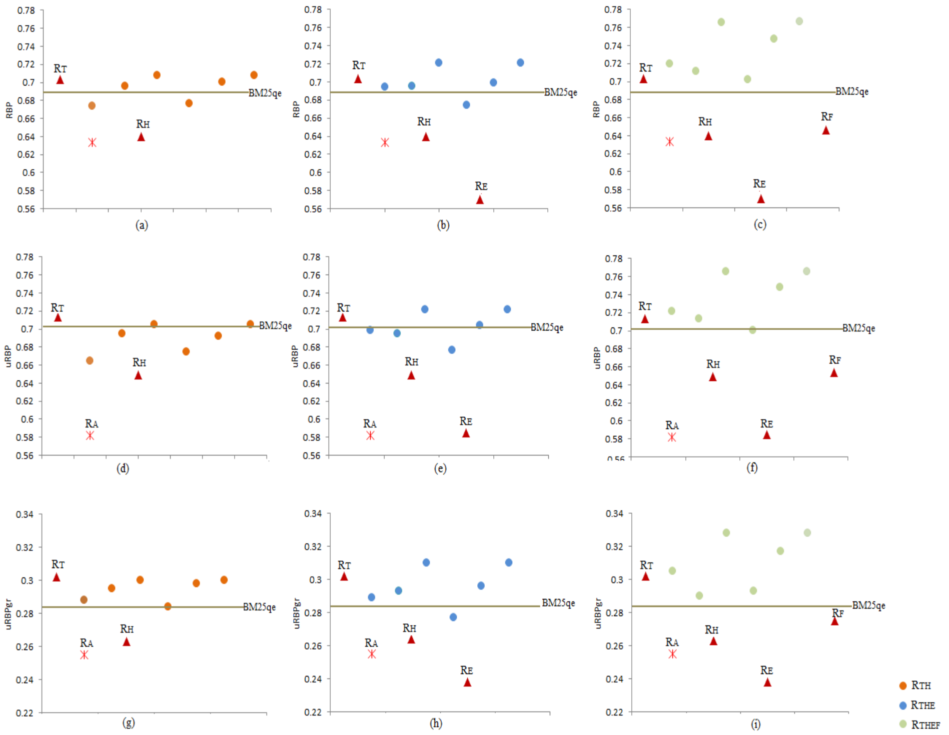


Figure 6.4: Performance of field features on understandability assessment.

MIN method were not able to surpass the baseline using same experimental settings.

Finally, for uRBP, we observe similar results as we got from RBP: three rankers built using CombMED, CombMNZ or CombSUM methods were able to surpass the strongest baseline; however, some aggregated rankers were not able to surpass the baseline using same experimental settings.

Comparison between Field Features

Now we look at how different fields and their combination affect the retrieval performance. First, we evaluate and compare the performance in understandability assessment. As shown in Figure 6.4, picture (a), (b) and (c) present the evaluation in uRBP metric; picture (d), (e) and (f) are in uRBPgr metric.

As it can be observed, in both two evaluation metrics, *title* features (R_T) are shown to be the most effective and able to surpass the baseline[7], followed

[7] In this figure, we only compare to baseline BM25_{PRF} which used the same experimental setup as the developed rankers.

by *full document* features (R_F), *H1* features (R_H) and *else* features (R_E); blindly combined features (R_A) are shown to be most ineffective. First, in picture (a), we can see that the aggregation over *title* and *else* features neutralize the results[8], underperforming R_T and outperforming R_E; most aggregated rankers were not able to surpass the baseline. Next, in picture (b), *else* features were added and we can see that these aggregated ranker achieved much better performance than the ones using *title* and *H1* features. Finally, in picture (c), when we add *full document* features information in, great improvement are achieved. When evaluated in uRBPgr, we can obtain similar observation from picture (d) to (f).

6.2.5 Conclusions

In the two-stage LETOR module, we propose using aggregation over field-based LETOR models. This section presents the results obtained over CLEF 2016-2018 eHealth data collections. We tested the models effectiveness on CLEF 2016-2018. We trained a set of LETOR models on CLEF'2016-2017 data and tested them on CLEF2018 collection. Different rank aggregation methods were applied to generate new ranking lists.

In what concerns to topical relevance, the following conclusions can be drawn from our observations:

- All aggregated rankers were able to surpass the baseline (using the same experimental settings as our developed rankers) in almost all cases on all three evaluation measures;
- two of the aggregated rankers outperformed the strongest baseline with obvious improvements;
- the rankers built using single field features did not perform as well as the aggregated rankers, none of them was able to surpass the strongest baseline and most failed to surpass the corresponding baseline built using same experimental settings;
- ranker built using all blindly joint features underperformed most baselines;
- comparing field-based LETOR models to the one using all features, some of the former ones were still be able to present better performance than the latter one.

[8] In this section, the aggregation over features means taking use of the two-stage L2R method proposed in our work.

For understandability analyses, we get much better results than we observed with topical relevance assessments:

- Almost all aggregated rankers were able to surpass the baseline using the same experimental settings as our developed rankers with all three evaluation measures and three of these rankers outperform the strongest baseline with obvious improvements;
- The ranker built using single field features did not perform as well as the aggregated rankers, but one of them were able to surpass the baseline built using same experimental settings;
- Most field-based LETOR models exceeded the ranker built using all blindly joint features together.

In summary, from the results obtained, we can conclude that rank aggregation is an effective solution for improving topical relevance and presents even better performance for improving understandability in the area of consumer health search.

Moreover, based on our comparison among different rank aggregation methods, we can conclude that: all aggregation methods studied are useful for improving state-of-the-art techniques, with CombMED and CombSUM being the best ways for combining the ranking lists compared with other four aggregation methods.

Concluding, the hypotheses presented in Section 4.3.2 are shown to be valid: different field information like title and H1 contribute differently to ranking and crude combination of all features from different fields together makes the feature blur. Moreover, fusing models learned with single field-based features individually has been shown to be more effective than a model learned including with all fields features together.

Chapter 7

Conclusions and Future Work

In this dissertation, research on promoting understandability in Consumer Health Information Retrieval where both the topical relevance and the understandability are considered, is presented. Many existing IR systems merely consider the topical relevance of the retrieved documents without taking into account the dimension of understandability. A topically relevant but not understandable document is of no value to a consumer.

In health domain search, this is even more important since non understandable information may cause other issues. For example, a non understandable health document may cause exceeded concerns about common symptoms.

To solve the problem, two sub-problems were defined, and related to each one, two research questions were raised. The first sub-problem aims at bridging the language gap between non-expert consumers and medical professionals, while the second proposes to learn understandability from experience.

Next sections present the conclusions reached for each sub-problem.

7.1 Bridge the Language Gap

According to the established theories and research in related areas, the understandability issue in the consumer health domain often arises from the language gap between a non-expert consumer and a medical expert. The consumers have usually limited medical knowledge while health documents are usually written by medical experts.

Our research paid special attention to the characteristics of the medical language and the hypotheses were:

- Each query term does not contribute equally to a query when searching relevant documents.
- Phrases are more effective than single, separate terms when finding relevant and understandable documents.

With these hypothesis in mind, the proposal was to combine NLP techniques with state-of-the-art query expansion ones. First, the Medical Concept Model (MCM) was proposed: the identified medical terms were classified as *term concepts* or *phrase concepts* and specific processing was done for each kind; *loose phases* were introduced and constructed and the techniques for word selecting were improved.

Then, the usefulness of the MCM model was tested on CLEF'2016- 2017 data and re-assessed on the CLEF'2018 data. The results were evaluated with carefully selected metrics and the following key observations were made:

- Regarding topical relevance, the proposed MCM model demonstrated its effectiveness compared to state-of-the art techniques, being able to surpass most baselines and be very close to the strongest one.
- Regarding understandability relevance, the MCM model demonstrated even better performance than the one achieved for topical relevance assessment, being able to surpass the strongest baseline in most cases. On CLEF'2016-2017 data, the improvements over the strongest baseline were 9.7% in uRBP and 3.3% in uRBPgr[1]; on CLEF'2018, the improvements were 2.0% in uRBP[2] and 1.3% in uRBPgr[3].
- Also on understandability, jointly using the MCM model with the UMLS Metathesaurus expansion achieved much better performance then using UMLS Metathesaurus alone. The improvements of were 4.5% in uRBP and 2.5% in uRBPgr[4].

These observations proved our hypotheses which enables to conclude that **classifying medical terms as groups and applying specific processing is an effective approach** to bridge the existing language gap. In turn, this provides a feasible solution for retrieving not only topically relevant, but also more understandable documents to the consumers.

[1] Ranker *BM25_MCM_PRF* which uses BM25 as the retrieval model, the MCM model independently and the PRF technique.

[2] Ranker *TFIDF_MCM_*PRF_{no} which uses TFIDF as the retrieval model, the MCM model independently and without using the PRF technique.

[3] Ranker *InL2_MCM_*PRF_{no} which uses InL2 as the retrieval model, the MCM model independently and without using the PRF technique.

[4] The comparison was done on CLEF'2016-2017 data.

In addition to the MCM model, another proposal tested to bridge the language gap was the use of word embeddings as the expansion resource in CHIR. Three models locally trained and one pre-trained model were used and their usefulness was examined through a comparison to the UMLS Metathesaurus query expansion on three sets of data collections. Different results were achieved and the following key observations could be made:

- In FIRE'2016, query expansion using word embedding models shown to be much more effective than using UMLS Metathesaurus, with an increase of 23.4% in F1 score, of 21.1% in precision and 16.9% in recall;
- Conversely, on CLEF data collections, the UMLS Metathesaurus expansion surpassed the word embedding models.

As a conclusion, it's possible to say that applying trained **word embedding models as query expansion resource is not always effective**, but still shows its usefulness in some data.

7.2 Learn Understandability from Experience

According to the established theories and research in related areas, using machine learning techniques for solving ranking problems has been shown to be an effective approach; this is known as Learning to Rank.

Instead of researching on potential features, more attention was paid on how to better take use of the existing features, and the hypotheses were:

- Blind combination of features extracted from different fields into a single feature list makes the features blur.
- An aggregated model over a set of pre-trained field-based models is more effective than a model trained with all features from different fields.

A two-stage LETOR model was proposed that first builds a set of single field-based learning to rank models and then applies rank aggregation to create aggregated rankers.

Experiments were done with this improved LETOR model on the CLEF eHealth data collections. First, the model was tested conducting a preliminary experiment on CLEF'2016-2017 data: a number of field features and simple combination methods were used; rankers built using the field-based features showed to be more effective than the ones using non field-based features.

Based on the results observed, the model was assessed with a more intensive experiment. A set of models were trained on CLEF'2016-2017 data and tested on CLEF'2018 data: 36 features were extracted from four different fields of the documents, four single field-based LETOR models and one with all features were trained and six different rank aggregation methods were tested producing, at the end, 18 different rankers.

Concerning the single field-based LETOR rankers built, the following key observations were made:

- Regarding topical relevance, all single field-based LETOR models (except R_E) R_T, R_H and were able to surpass the one trained with blindly mixed features (R_A) in most cases.
- Regarding understandability relevance, all 4 single field-based LETOR models presented even better results in surpassing the one trained with blindly mixed features.

These observations prove the rightness of our hypothesis: blind combination of features extracted from different fields into one single list makes features blur and learning more difficult (the performance is worse).

Concerning the application of rank aggregation methods over the LETOR models, the following key observations can be made:

- Regarding topical relevance, in all cases, all aggregated models (except one) surpassed the one with all features. The highest improvement[5] reached was 15.9% in P@10, 16.6% in NDCG@10 and 22.2% in MAP. And the improvement over the baseline was 7.8%, 16.4% and 7.6% in P@10, NDCG@10 and MAP, respectively.
- Regarding understandability relevance, also in all cases, all 18 aggregated models surpassed the one trained with all features. The highest improvement[6] reached was 17.4% in RBP, 24.0% in uRBP and 28.6% in uRBPgr. And the improvement over the baseline was 11.5%, 9.3% and 16.3% in RBP, uRBP and uRBPgr, respectively.

Moreover, field features demonstrate different effects in learning models. Comparing to *H1*, *else* and *full document* features, *Title* features are shown

[5] Rankers R_{THEF_sum} and R_{THEF_med} which use CombSUM and CombMED aggregation methods over four single field based LETOR models $\mathrm{R_T}$, $\mathrm{R_H}$, $\mathrm{R_E}$ and $\mathrm{R_F}$.

[6] Rankers R_{THEF_sum} and R_{THEF_med}.

to be the most effective group in learning both understandability and topical relevance from the past data; furthermore, the effectiveness was strengthened when aggregated with *H1*, *else* and *full document* features using the proposed two-stage L2R framework.

These results prove our second hypothesis that an aggregated model over a set of pre-trained field-based models is obviously more effective than a model trained by blindly joining together all features from different fields.

Based on the observations from the two sets of experiments, one could concluded that **understandability can be learned with the proposed two-stage LETOR model**.

7.3 Main Contributions

To summarize, the research presented in this dissertation has the following significant contributions:

- A thorough survey about state-of-the-art techniques in HIR and CHIR is made. Three classically and widely used thesauri are discussed and compared (MeSH, OAC-CHV and UMLS), the valuable domain data source PubMed and its use in HIR and CHIR is presented, medical concepts identification techniques and the useful open-source tools are introduced, significant campaigns in the related area are detailed and finally, related work by previous researchers are thoroughly reviewed.
- The language gap between consumers and medical experts is shortened. By improving the expressions of original queries, state-of-the-art query expansion techniques are improved:
  - A Medical Concept Model is proposed. Rather than using general purpose query expansion techniques, the characteristics of the medical language are fully considered and original queries issued by non-expert consumers are processed using the proposed model;
  - Loose phrases are introduced, aiming to build more flexible query expressions. Loose phrases are constructed using the terms from the original queries or words expanded from the QE resources.

  These proposed techniques for medical query processing proved to be useful in bridging the language gap between non-expert consumer and medical professional. It is useful to health queries processing and may also be applied to other related tasks. They can, as well, be general enough to be applied to other IR research work with similar language gap issues as the one presented in CHIR.

- A two-stage LETOR model is proposed where rank aggregations are combined with the LETOR approach. This model is not only useful in CHIR research work, but can be also easily generalized to other IR research work:
    - A set of different field based information models are built. Rather than building a single model with all the potential features, a set of field-based models is built with features extracted from each corresponding information field. Besides the field-based grouping, other grouping ways can be easily used.
    - A score-based rank aggregation model is suggested based on the field-based LETOR models. Instead of score-based rank aggregation methods other methods can also be applied based on the needs of the tasks.
- We participated in three different CHIR related campaigns:
    - In the FIRE'2016 CHIS task (team UÉvora) participation, different from all other 8 teams, we proposed to use IR techniques (mainly state-of-the-art QE techniques in CHIR) to solve the task; this method secured *the second rank* (Yang and Gonçalves, 2016). As a continued work from this task, the locally trained word embedding models $\mathrm{WE}_{\mathrm{wiki}}$ and $\mathrm{WE}_{\mathrm{pubmed}}$ were used as query expansion resources showing to have high performances; specially, the $\mathrm{WE}_{\mathrm{pubmed}}$ model surpassed the best team score with a large margin (19.3% in averaged accuracy) (Yang and Gonçalves, 2018b).
    - In the CLEF'2017 eHealth IRTask 1 participation (team UÉvora), the MCM model was used to built the rankers. This method ranked *the best place* and the work was published in *the best of the labs track* at CLEF 2018 (Yang and Gonçalves, 2017, 2018a).
    - In the CLEF'2018 eHealth IR task1 (team UÉvora) participation, the proposed two-stage LETOR model was used to build the rankers. This method was ranked the second place in P@10 and the third place in BPREF[7] evaluation metrics. More specifically, this method was obviously superior in some of the queries compared with other teams[8].

[7] BPREF, Binary Preference Based Measure, is an IR evaluation metric.

[8] The evaluation results are available at https://github.com/CLEFeHealth/CLEFeHealth2018IRtask/blob/master/presentations%40CLEF2018/from_organisers/CHS%20Session%20-%202018.pdf.

7.4 Future Work

The research on promoting understandability in consumer health information retrieval presented in this dissertation can be improved and extended in several ways in the future.

In this work we mainly tested the proposals on the CLEF eHealth data collections. Their queries were generated by experts and non-experts consumers; the consumers' understandability was roughly mapped into two levels: good and poor medical knowledge. In reality, even as non-experts consumers, their medical knowledge can vary a lot; an ideal corpus would have queries issued by non-expert consumers who have different levels of medical knowledge. As future research, it would be interesting to evaluate and refine these methods aiming at this kind of corpora.

Using the trained word embedding models as query expansion resource in comparison to the UMLS Metathesaurus in CHIR, presented contradictory results in different data collections. Simple methods for word similarity selection (using the cosine similarity) were used in this research; as a future work, it would be interesting to improve these methods and test their effectiveness in two ways: (i) jointly using the MCM and the word embedding model; (ii) improving the techniques used for word similarity selection.

One valuable finding of this research is that field-based LETOR models shown to be more effective than one built with all mixed features. Thirty six features, mainly query-document dependent, were experimented in total, extracted using IR weighting models. It would be interesting to include other features and test the proposed approaches with those features in the future. For example, document dependent features, such as linguistic ones, readability scores or medical terms statistics, could be explored. Moreover, query dependent features, like the consumers' readability or the assessment of consumers' understandability in health information, could also be explored. It would also be interesting to group features using other categories besides the fields of the document.

Besides being able to achieve better results in averaged scores, our improved LETOR method presented its superiority in some queries. It would be very interesting to analyze the queries based on their performance and find out solutions to improve the results. Also, since *Title* features shown to be more effective than other field features, it would be interesting to investigate the approach with other data collections to test its applicability.

Appendix A

Publications

Conference papers (peer review)

Hua Yang and Teresa Gonçalves. "How does Word Embeddings-based Query Expansion Perform in Consumer Health Information Search?". In: FIRE'18 Proceedings of the 10th annual meeting of the Forum for Information Retrieval Evaluation. Prasenjit Majumder et al. (Eds.). ACM, NY, USA,2018. P:35-40.

Hua Yang and Teresa Gonçalves. "A Compound Model for Consumer Health Search". In: Lecture Notes in Computer Science. CLEF 2018. Experimental IR Meets Multilinguality, Multimodality, and Interaction. Volume: 11018. Patrice Bellot et al. (Eds.). Springer, Cham, 2018. P: 231-236.

Hua Yang and Teresa Gonçalves. "Promoting Understandability in Consumer Health Information Search". In : Advances in Information Retrieval. ECIR 2017. Lecture Notes in Computer Science. Volume: 10193. Jose J. et al. (Eds.). Springer, Cham, 2017. P: 727-734.

Campaign Working Notes

Hua Yang and Teresa Gonçalves. "Improving Personalized Consumer Health Search: Notebook for eHealth at CLEF 2018". In: Working Notes of Conference and Labs of the Evaluation Forum (CLEF18). Vol-2125. Linda Cappellato et al. (Eds.). CEUR Workshop Proceedings. 2018.

Hua Yang and Teresa Gonçalves. "UEvora at CLEF eHealth 2017 Task". In: Working Notes of Conference and Labs of the Evaluation Forum (CLEF17).

Vol-1866. Linda Cappellato et al. (Eds.). CEUR Workshop Proceedings. 2017.

Hua Yang and Teresa Gonçalves. "Improving Understandability in Consumer Health Information Search: UEVORA@2016 FIRE CHIS". In: Working Notes of 8th Forum for Information Retrieval Evaluation (FIRE16). Vol-1737. Prasenjit Majumder et al. (Eds.). CEUR Workshop Proceedings. 2016. P: 228-232.

Workshops

Hua Yang and Teresa Gonçalves. "Query Expansion Techniques in Consumer Health Information Search". In Jornadas de Informática da Universidade de Évora, Escola de Ciências e Tecnologia, Universidade de Évora (JIUE17), 2017.

Hua Yang and Teresa Gonçalves. "Survey report on text classification". In Jornadas de Informática da Universidade de Évora, Escola de Ciências e Tecnologia, Universidade de Évora (JIUE16), 2016.

Appendix B

Campaign Participation

FIRE 2016 CHIS track[1]

In FIRE2016, we participated in the CHIS (Consumer Health Information Search) track. The goal was to research and develop techniques to support users in complex multi-perspective health information queries. Task A aimed at classifying sentences in the document as relevant to the query or not; Task B was to further classify the relevant sentences as supporting or opposing the claim made in the query.

In this task, different from all other 8 teams who all used machine learning techniques and trained classifiers, we proposed to use IR techniques to solve Task 1, more specifically, state-of-the-art query expansion techniques were used. This novel method secured *the second rank* (Yang and Gonçalves, 2016).

As a continuous work from this task, locally trained word embedding models were used as query expansion resources. This method surpassed the best team score with a large margin. This extended work was submitted and published in FIRE2018 (Yang and Gonçalves, 2018b).

CLEF 2017 eHealth IR Task[2]

In CLEF'2017, we participated in eHealth IRTask 1. This task was a standard ad-hoc search task, aiming at retrieving information relevant to people (non-expert users) seeking health advice on the web.

[1] http://fire.irsi.res.in/fire/2016/home

[2] https://sites.google.com/site/clefehealth2017/task-3

In this task, we proposed a medical concept model (MCM) to improve state-of-the-art query expansion techniques applied in health IR. Our method ranked the best place and our work was published in *the best of the labs track* at CLEF 2018 (Yang and Gonçalves, 2017, 2018a).

CLEF 2018 eHealth IR Task[3]

In CLEF2018, we participated in IRTask 1 and IRTask 2. IRTask 1 was the same as previous year, IRTask 2 was developed on top of the IRTask1, aiming to personalize the retrieved list of search results to match user expertise, measured by how likely the person understands the content of a document (with respect to the health information).

In IRTask 1, we re-tested the MCM model on CLEF 2018 data collection and experimented on using trained word embeddings as the query expansion resource as well. In IRTask 2, I proposed a two-stage learning to rank model in which rank aggregation was applied over a set of single field-based learning to rank models. The models were trained using data from CLEF16 and CLEF17 and tested on CLEF18 data collection. Our submissions were ranked at the top place (Yang and Gonçalves, 2018c).

[3] https://sites.google.com/view/clef-ehealth-2018/task-3-consumer-health-search

Bibliography

Abacha, A. B. (2016). Nlm nih at trec 2016 clinical decision support track. In *TREC*.

Abacha, A. B. and Zweigenbaum, P. (2011). Medical entity recognition: A comparison of semantic and statistical methods. In *Proceedings of BioNLP 2011 Workshop*, pages 56–64. Association for Computational Linguistics.

Ai, Q., Bi, K., Luo, C., Guo, J., and Croft, W. B. (2018). Unbiased learning to rank with unbiased propensity estimation. *arXiv preprint arXiv:1804.05938*.

ALMasri, M., Berrut, C., and Chevallet, J.-P. (2016). A comparison of deep learning based query expansion with pseudo-relevance feedback and mutual information. In *European Conference on Information Retrieval*, pages 709–715. Springer.

Amati, G. (2003). *Probability models for information retrieval based on divergence from randomness*. PhD thesis, University of Glasgow.

Aronson, A. R. (2001). Effective mapping of biomedical text to the umls metathesaurus: the metamap program. In *Proceedings of the AMIA Symposium*, page 17. American Medical Informatics Association.

Aronson, A. R. (2006). Metamap: Mapping text to the umls metathesaurus. *Bethesda, MD: NLM, NIH, DHHS*, pages 1–26.

Aronson, A. R. and Lang, F.-M. (2010). An overview of metamap: historical perspective and recent advances. *Journal of the American Medical Informatics Association*, 17(3):229–236.

Aronson, A. R. and Rindflesch, T. C. (1997). Query expansion using the umls metathesaurus. In *Proceedings of the AMIA Annual Fall Symposium*, page 485. American Medical Informatics Association.

Aslam, J. A. and Montague, M. (2001). Models for metasearch. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 276–284. ACM.

Badarudeen, S. and Sabharwal, S. (2010). Assessing readability of patient education materials: current role in orthopaedics. *Clinical Orthopaedics and Related Research®*, 468(10):2572–2580.

Balaneshin-Kordan, S., Kotov, A., and Xisto, R. (2015). Wsu-ir at trec 2015 clinical decision support track: Joint weighting of explicit and latent medical query concepts from diverse sources. Technical report, Wayne State University Detroit United States.

Białecki, A., Muir, R., Ingersoll, G., and Imagination, L. (2012). Apache lucene 4. In *SIGIR 2012 workshop on open source information retrieval*, page 17.

Bodenreider, O. (2004). The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_-1):D267–D270.

Budaher, J., Almasri, M., and Goeuriot, L. (2016). Comparison of several word embedding sources for medical information retrieval. In *CLEF (Working Notes)*, pages 43–46.

Burges, C. J. (2010). From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11(23-581):81.

Cao, Z., Qin, T., Liu, T.-Y., Tsai, M.-F., and Li, H. (2007). Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136. ACM.

Carpineto, C. and Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Computing Surveys (CSUR)*, 44(1):1.

Christopher, D. M., Prabhakar, R., and Hinrich, S. (2008). Introduction to information retrieval. *An Introduction To Information Retrieval*, 151(177):5.

Cohen, D., Mitra, B., Hofmann, K., and Croft, W. B. (2018). Cross domain regularization for neural ranking models using adversarial learning. *arXiv preprint arXiv:1805.03403*.

Collins-Thompson, K. (2014). Computational assessment of text readability: A survey of current and future research. *ITL-International Journal of Applied Linguistics*, 165(2):97–135.

Collins-Thompson, K., Bennett, P. N., White, R. W., De La Chica, S., and Sontag, D. (2011). Personalizing web search results by reading level. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 403–412. ACM.

Craswell, N. (2009). Precision at n. *Encyclopedia of Database Systems.*

Cuadra, C. A. and Katter, R. V. (1967). Opening the black box of relevance. *journal of Documentation*, 23(4):291–303.

D., H. (2009). Language models. *Encyclopedia of Database Systems.*

Darmoni, S. J., Griffon, N., and Névéol, A. (2012). Improving information retrieval using medical subject headings concepts: a test case on rare and chronic diseases. *Journal of the Medical Library Association*, 100(3):176.

De Vine, L., Zuccon, G., Koopman, B., Sitbon, L., and Bruza, P. (2014). Medical semantic similarity with a neural language model. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 1819–1822. ACM.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.

Deng, K., Han, S., Li, K. J., and Liu, J. S. (2014). Bayesian aggregation of order-based rank data. *Journal of the American Statistical Association*, 109(507):1023–1039.

Diaz, F., Mitra, B., and Craswell, N. (2016). Query expansion with locally-trained word embeddings. *arXiv preprint arXiv:1605.07891.*

Dwork, C., Kumar, R., Naor, M., and Sivakumar, D. (2001). Rank aggregation methods for the web. In *Proceedings of the 10th international conference on World Wide Web*, pages 613–622. ACM.

Efthimiadis, E. N. (1996). Query expansion. *Annual review of information science and technology (ARIST)*, 31:121–87.

Eltorai, A. E., Ghanian, S., Adams Jr, C. A., Born, C. T., and Daniels, A. H. (2014). Readability of patient education materials on the american association for surgery of trauma website. *Archives of trauma research*, 3(2).

Fitzsimmons, P., Michael, B., Hulley, J., and Scott, G. (2010). A readability assessment of online parkinson's disease information. *The journal of the Royal College of Physicians of Edinburgh*, 40(4):292–296.

Fox, E. A. and Shaw, J. A. (1994). Combination of multiple searches. *NIST special publication SP*, 243.

Fox, S. and Duggan, M. (2013). Health online 2013. *Washington, DC: Pew Internet & American Life Project.*

Goeuriot, L., Jones, G. J., Kelly, L., Müller, H., and Zobel, J. (2016). Medical information retrieval: introduction to the special issue. *Information Retrieval Journal*, 1(19):1–5.

Goodwin, T. and Harabagiu, S. M. (2014). Utd at trec 2014: Query expansion for clinical decision support. Technical report, DTIC Document.

Harter, S. P. (1992). Psychological relevance and information science. *Journal of the American Society for information Science*, 43(9):602–615.

Hedman, A. S. (2008). Using the smog formula to revise a health-related document. *American Journal of Health Education*, 39(1):61–64.

Hiemstra, D. (Sep 2009). *UMLS Reference Manual [Internet]*. Bethesda (MD): National Library of Medicine (US). Available from https://www.ncbi.nlm.nih.gov/books/NBK9684/.

Hollada, J. L., Zide, M., Speier, W., and Roter, D. L. (2017). Readability assessment of patient-centered outcomes research institute public abstracts in relation to accessibility. *Epidemiology*, 28(4):e37–e38.

Humphreys, B. L., Lindberg, D. A., Schoolman, H. M., and Barnett, G. O. (1998). The unified medical language system. *Journal of the American Medical Informatics Association*, 5(1):1–11.

Jimmy, Zuccon, G., and Palotti, J. (2018). Overview of the clef 2018 consumer health search task. In *Working Notes of Conference and Labs of the Evaluation (CLEF) Forum. CEUR Workshop Proceedings.*

Jones, K. S. (1971). Automatic keyword classification for information retrieval. *The Library Quarterly.*

Keselman, A., Smith, C. A., Divita, G., Kim, H., Browne, A. C., Leroy, G., and Zeng-Treitler, Q. (2008). Consumer health concepts that do not map to the umls: where do they fit? *Journal of the American Medical Informatics Association*, 15(4):496–505.

Kher, A., Johnson, S., and Griffith, R. (2017). Readability assessment of online patient education material on congestive heart failure. *Advances in preventive medicine*, 2017.

Kutner, M., Greenburg, E., Jin, Y., and Paulsen, C. (2006). The health literacy of america's adults: Results from the 2003 national assessment of adult literacy. nces 2006-483. *National Center for Education Statistics.*

Kuzi, S., Shtok, A., and Kurland, O. (2016). Query expansion using word embeddings. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 1929–1932. ACM.

Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196.

Lee, J. H. (1997). Analyses of multiple evidence combination. In *ACM SIGIR Forum*, volume 31-SI, pages 267–276. ACM.

Lioma, C., Macdonald, C., Plachouras, V., Peng, J., He, B., and Ounis, I. (2006). University of glasgow at trec 2006: Experiments in terabyte and enterprise tracks with terrier. In *TREC*.

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

Liu, T.-Y., Xu, J., Qin, T., Xiong, W., and Li, H. (2007). Letor: Benchmark dataset for research on learning to rank for information retrieval. In *Proceedings of SIGIR 2007 workshop on learning to rank for information retrieval*, volume 310. ACM Amsterdam, The Netherlands.

Lopes, C. T. and Ribeiro, C. (2016). Effects of language and terminology on the usage of health query suggestions. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 83–95. Springer.

Lu, Z., Kim, W., and Wilbur, W. J. (2009). Evaluation of query expansion using mesh in pubmed. *Information retrieval*, 12(1):69–80.

Luo, G. and Tang, C. (2008). On iterative intelligent medical search. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 3–10. ACM.

Macdonald, C., Santos, R. L., Ounis, I., and He, B. (2013). About learning models with multiple query-dependent features. *ACM Transactions on Information Systems (TOIS)*, 31(3):11.

Manmatha, R., Rath, T., and Feng, F. (2001). Modeling score distributions for combining the outputs of search engines. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 267–275. ACM.

Manning, C. D., Manning, C. D., and Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT press.

Manning, C. D., Raghavan, P., and Schütze, H. (2009). Probabilistic information retrieval. *Introduction to Information Retrieval*, pages 220–235.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Mizzaro, S. (1998). How many relevances in information retrieval? *Interacting with computers*, 10(3):303–320.

Montague, M. and Aslam, J. A. (2001). Relevance score normalization for metasearch. In *Proceedings of the tenth international conference on Information and knowledge management*, pages 427–433. ACM.

Narwani, V., Nalamada, K., Lee, M., Kothari, P., and Lakhani, R. (2016). Readability and quality assessment of internet-based patient education materials related to laryngeal cancer. *Head & neck*, 38(4):601–605.

Oh, H.-S. and Jung, Y. (2016). Kisti at clef ehealth 2016 task 3: Ranking medical documents using word vectors. In *CLEF (Working Notes)*, pages 103–108.

Oliffe, M., Johnston, J., Freeman, D., Bagga, H., and Wong, P. (2017). Assessing the readability and patient comprehension of medicine information sheets provided to patients by australian rheumatologists.

Ounis, I., Amati, G., Plachouras, V., He, B., Macdonald, C., and Lioma, C. (2006). Terrier: A high performance and scalable information retrieval platform. In *Proceedings of the OSIR Workshop*, pages 18–25.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Palotti, J., Goeuriot, L., Zuccon, G., and Hanbury, A. (2016). Ranking health web pages with relevance and understandability. In *Proceedings of the 39th international ACM SIGIR conference on Research and development in information retrieval*, pages 965–968. ACM.

Palotti, J. and Rekabsaz, N. (2018). Exploring understandability features to personalize consumer health search. In *CEUR-WS, Working Notes of CLEF 2017 - Conference and Labs of the Evaluation Forum.*

Palotti, J., Zuccon, G., Goeuriot, L., Kelly, L., Hanbury, A., Jones, G., Lupu, M., and Pecina, P. (2015). Clef ehealth evaluation lab 2015, task 2: Retrieving information about medical symptoms. In *Proc. of CLEF.*

Pavlopoulos, I., Kosmopoulos, A., and Androutsopoulos, I. (2014). Continuous space word vectors obtained by applying word2vec to abstracts of biomedical articles.

Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Ponte, J. M. and Croft, W. B. (1998). *A language modeling approach to information retrieval*. PhD thesis, University of Massachusetts at Amherst.

Qin, T. and Liu, T.-Y. (2013). Introducing letor 4.0 datasets. *arXiv preprint arXiv:1306.2597*.

Qin, T., Liu, T.-Y., Xu, J., and Li, H. (2010). Letor: A benchmark collection for research on learning to rank for information retrieval. *Information Retrieval*, 13(4):346–374.

Roberts, K., Simpson, M., Demner-Fushman, D., Voorhees, E., and Hersh, W. (2016). State-of-the-art in biomedical literature retrieval for clinical cases: a survey of the trec 2014 cds track. *Information Retrieval Journal*, 19(1-2):113–148.

Roberts, K., Simpson, M. S., Voorhees, E. M., and Hersh, W. R. (2015). Overview of the trec 2015 clinical decision support track. In *TREC*.

Robertson, S., Zaragoza, H., et al. (2009). The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Robertson, S. E., Walker, S., Jones, S., Hancock-Beaulieu, M. M., Gatford, M., et al. (1995). Okapi at trec-3. *Nist Special Publication Sp*, 109:109.

Rocchio, J. J. (1971). Relevance feedback in information retrieval. *The SMART retrieval system: experiments in automatic document processing*, pages 313–323.

Roy, D., Paul, D., Mitra, M., and Garain, U. (2016). Using word embeddings for automatic query expansion. *arXiv preprint arXiv:1606.07608*.

Saracevic, T. (1996). Relevance reconsidered. In *Proceedings of the second conference on conceptions of library and information science (CoLIS 2)*, pages 201–218. ACM New York.

Saracevic, T. (2016). The notion of relevance in information science: Everybody knows what relevance is. but, what is it really? *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 8(3):i–109.

Savova, G. K., Masanz, J. J., Ogren, P. V., Zheng, J., Sohn, S., Kipper-Schuler, K. C., and Chute, C. G. (2010). 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.

Scells, H., Zuccon, G., Deacon, A., and Koopman, B. (2017). Qut ielab at clef ehealth 2017 technology assisted reviews track: Initial experiments with learning to rank. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, volume 1866, pages Paper–98. CEUR Workshop Proceedings.

Seung-Hyeon Jo, K.-S. L. (2016). Cbnu at trec 2016 clinical decision support track. In *Text REtrieval Conference (TREC 2016).*

Severyn, A. and Moschitti, A. (2015). Learning to rank short text pairs with convolutional deep neural networks. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 373–382. ACM.

Shen, W. and Nie, J.-Y. (2015). Is concept mapping useful for biomedical information retrieval? In *International conference of the cross-language evaluation forum for European languages*, pages 281–286. Springer.

Sinha, M., Mannarswamy, S., and Roy, S. (2016). Chis@ fire: Overview of the shared task on consumer health information search. In *FIRE (Working Notes)*, pages 193–196.

Soldaini, L. and Goharian, N. (2017). Learning to rank for consumer health search: a semantic approach. In *European Conference on Information Retrieval*, pages 640–646. Springer.

Soldaini, L., Yates, A., Yom-Tov, E., Frieder, O., and Goharian, N. (2016). Enhancing web search in the medical domain via query clarification. *Information Retrieval Journal*, 19(1-2):149–173.

Song, F. and Croft, W. B. (1999). A general language model for information retrieval. In *Proceedings of the eighth international conference on Information and knowledge management*, pages 316–321. ACM.

Song, Y., He, Y., Hu, Q., He, L., and Haacke, E. M. (2015). Ecnu at 2015 ehealth task 2: User-centred health information retrieval. In *CLEF (Working Notes).*

Srinivasan, P. (1996). Query expansion and medline. *Information Processing and Management*, 32(4):431–443.

Strohman, T., Metzler, D., Turtle, H., and Croft, W. B. (2005). Indri: A language model-based search engine for complex queries. In *Proceedings of the International Conference on Intelligent Analysis*, volume 2-6, pages 2–6. Citeseer.

Thuma, E., Anderson, G., and Mosweunyane, G. (2015). Ubml participation to clef ehealth ir challenge 2015: Task 2. In *CLEF (Working Notes)*.

Vogt, C. C. and Cottrell, G. W. (1998). Predicting the performance of linearly combined ir systems. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 190–196. ACM.

Vogt, C. C. and Cottrell, G. W. (1999). Fusion via a linear combination of scores. *Information retrieval*, 1(3):151–173.

Voorhees, E. M. (1994). Query expansion using lexical-semantic relations. In *SIGIR94*, pages 61–69. Springer.

Voorhees, E. M. (2001). The philosophy of information retrieval evaluation. In *Workshop of the cross-language evaluation forum for european languages*, pages 355–370. Springer.

Voorhees, E. M., Harman, D. K., et al. (2005). *TREC: Experiment and evaluation in information retrieval*, volume 1. MIT press Cambridge.

Voorhees, E. M. and Hersh, W. R. (2012). Overview of the trec 2012 medical records track. In *TREC*.

Wang, C., Cao, L., and Zhou, B. (2015). Medical synonym extraction with concept space models. *arXiv preprint arXiv:1506.00528*.

Wang, H., Langley, R., Kim, S., McCord-Snook, E., and Wang, H. (2018). Efficient exploration of gradient space for online learning to rank. *arXiv preprint arXiv:1805.07317*.

Wang, R., Lu, W., and Ren, K. (2016a). Whuirgroup at the clef 2016 ehealth lab task 3. In *CLEF (Working Notes)*, pages 193–197.

Wang, X., Bendersky, M., Metzler, D., and Najork, M. (2016b). Learning to rank with selection bias in personal search. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 115–124. ACM.

Wang, Y., Wang, L., Li, Y., He, D., and Liu, T.-Y. (2013). A theoretical analysis of ndcg type ranking measures. In *Conference on Learning Theory*, pages 25–54.

White, R. W. and Horvitz, E. (2009). Cyberchondria: studies of the escalation of medical concerns in web search. *ACM Transactions on Information Systems (TOIS)*, 27(4):23.

Xia, X., Lo, D., Wang, X., Zhang, C., and Wang, X. (2014). Cross-language bug localization. In *Proceedings of the 22nd International Conference on Program Comprehension*, pages 275–278. ACM.

Xu, J. and Li, H. (2007). Adarank: a boosting algorithm for information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 391–398. ACM.

Xu, Y. and Chen, Z. (2006). Relevance judgment: What do information users consider beyond topicality? *Journal of the American Society for Information Science and Technology*, 57(7):961–973.

Yang, H. and Gonçalves, T. (2016). Improving understandability in consumer health information search: Uevora @ 2016 fire chis. In Majumder, P., Mitra, M., Mehta, P., Sankhavara, J., and Ghosh, K., editors, *Working notes of FIRE 2016 – Forum for Information Retrieval Evaluation*, volume 1737, pages 228–232, Kolkata, IN. CEUR.

Yang, H. and Gonçalves, T. (2017). Promoting understandability in consumer health information search. In *European Conference on Information Retrieval*, pages 727–734. Springer.

Yang, H. and Gonçalves, T. (2017). Uevora at clef ehealth 2017 task 3. In *Conference and Labs of the Evaluation Forum (CLEF) 2017*.

Yang, H. and Gonçalves, T. (2018a). A compound model for consumer health search. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 231–236. Springer.

Yang, H. and Gonçalves, T. (2018b). How does word embeddings-based query expansion perform in consumer health information search?: A novel solution for chis fire'2016 track. In *Proceedings of the 10th annual meeting of the Forum for Information Retrieval Evaluation*, pages 35–40. ACM.

Yang, H. and Gonçalves, T. (2018c). Improving personalized consumer health search: Notebook for ehealth at clef 2018. In *Conference and Labs of the Evaluation Forum (CLEF) 2018*. CEUR.

Yilmaz, E., Verma, M., Craswell, N., Radlinski, F., and Bailey, P. (2014). Relevance and effort: an analysis of document utility. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 91–100. ACM.

Zeng, Q. T., Crowell, J., Plovnick, R. M., Kim, E., Ngo, L., and Dibble, E. (2006). Assisting consumer health information retrieval with query recommendations. *Journal of the American Medical Informatics Association*, 13(1):80–90.

Zeng, Q. T. and Tse, T. (2006). Exploring and developing consumer health vocabularies. *Journal of the American Medical Informatics Association*, 13(1):24–29.

Zhai, C. and Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS)*, 22(2):179–214.

Zhai, C. and Lafferty, J. (2017). A study of smoothing methods for language models applied to ad hoc information retrieval. In *ACM SIGIR Forum*, volume 51-2, pages 268–276. ACM.

Zhu, D. and Carterette, B. (2012a). Exploring evidence aggregation methods and external expansion sources for medical record search. Technical report, DELAWARE UNIV NEWARK DEPT OF COMPUTER AND INFORMATION SCIENCES.

Zhu, D. and Carterette, B. (2012b). Improving health records search using multiple query expansion collections. In *Bioinformatics and Biomedicine (BIBM), 2012 IEEE International Conference On*, pages 1–7. IEEE.

Zielstorff, R. D. (2003). Controlled vocabularies for consumer health. *Journal of biomedical informatics*, 36(4-5):326–333.

Zuccon, G. (2016). Understandability biased evaluation for information retrieval. In *European Conference on Information Retrieval*, pages 280–292. Springer.

Zuccon, G., Koopman, B., Bruza, P., and Azzopardi, L. (2015). Integrating and evaluating neural word embeddings in information retrieval. In *Proceedings of the 20th Australasian document computing symposium*, page 12. ACM.

Zuccon, G., Palotti, J., Goeuriot, L., Kelly, L., Lupu, M., Pecina, P., Mueller, H., Budaher, J., and Deacon, A. (2016). The ir task at the clef ehealth evaluation lab 2016: user-centred health information retrieval. In *CLEF 2016-Conference and Labs of the Evaluation Forum*, volume 1609, pages 15–27.

UNIVERSIDADE DE ÉVORA
INSTITUTO DE INVESTIGAÇÃO
E FORMAÇÃO AVANÇADA

Contactos:
Universidade de Évora
Instituto de Investigação e Formação Avançada — IIFA
Palácio do Vimioso | Largo Marquês de Marialva, Apart. 94
7002 - 554 Évora | Portugal
Tel: (+351) 266 706 581
Fax: (+351) 266 744 677
email: iifa@uevora.pt