The Knowledge and Language Gap in Medical Information Seeking

A Dissertation
submitted to the Faculty of the
Graduate School of Arts and Sciences
of Georgetown University
in partial fulfillment of the requirements for the
degree of
Doctor of Philosophy
in Computer Science

By

Luca Soldaini, M.S.

Washington, DC
March 26, 2018

The Knowledge and Language Gap in Medical Information Seeking

Luca Soldaini, M.S.

Dissertation Advisor: Nazli Goharian, Ph.D.

Abstract

Interest in medical information retrieval has risen significantly in the last few years. The Internet has become a primary source for consumers looking for health information and advice; however, their lack of expertise causes a language and knowledge gap that affects their ability to properly formulate their information needs. Health experts also struggle to efficiently search the large amount of medical literature available to them, which impacts their ability of integrating the latest research findings in clinical practice. In this dissertation, I propose several methods to overcome these challenges, thus improving search outcomes.

For queries issued by lay users, I introduce *query clarification*, a technique to identify the most appropriate expert expression that describes their information need; such expression is then used to expand the query. I experiment with three existing synonym mappings, and show that the best one leads to a 7.3% improvement over non-clarified queries. When a classifier that predicts the most appropriate mapping for each query is used, an additional 5.2% improvement over non-clarified queries is achieved. Furthermore, I introduce a set of features to capture semantic similarity between consumer queries and retrieved documents, which are then exploited by a learning to rank framework. This approach yields a 26.6% improvement over the best known results on a dataset designed to evaluate medical information retrieval for lay users.

To improve literature search for medical professionals, I propose and evaluate two query reformulation techniques that expand complex medical queries with relevant latent and explicit medical concepts. The first is an unsupervised system that combines a statistical

query expansion with a medical terms filter, while the second is a supervised neural convolutional model that predicts which terms to add to medical queries. Both approaches are competitive with the state of the art, achieving up to 8% improvement in inferred nDCG. Finally, I conclude my dissertation by showing how the convolutional model can be adapted to reduce clinical notes that contain significant noise, such as medical abbreviations, incomplete sentences, and redundant information. This approach outperforms the best query reformulation system for this task by 27% in inferred nDCG.

INDEX WORDS: Query reformulation, Health informatics, Information retrieval, Learning to rank, Convolutional neural networks, Web search

Acknowledgments

This dissertation would have not been possible without the input and support of many colleagues, friends, and loved ones.

I would like to thank my advisor Nazli Goharian for being a superb mentor for the past five years. Beside invaluable research input and feedback about my work, she relentlessly motivated me to pursue my research interests and focus on the necessary milestones to complete my degree.

One of the highlights of my journey at Georgetown has been to be a member of the Information Retrieval Lab. I learned invaluable research lessons from the director of the lab Ophir Frieder, and I must also thank him for the many opportunities he gave me to perfect my coffee brewing and wine tasting skills. It was also a joy to collaborate and spend time with many former and current students in the lab, including Sean MacAvaney, Jon Parker, Jason Soo, Eugene Yang, and Andrew Yates. A special thank you goes to my brother-in-research Arman Cohan, with whom I shared many of the successes and hurdles along this journey, and to his wife Maryam.

My Ph.D. would have not been the same without many members of the computer science department. Among them, thank you Jakob Prange, Katina Russell, Tavish Vaidya, and Mohammad Zaheri for being wonderful friends.

In the second half of my Ph.D., I had the chance to witness the start of a fierce effort in establishing a union of graduate workers at Georgetown. Today, the Georgetown Alliance of Graduate Workers (GAGE) is a strong network all of us can rely on for support and assistance. I am particularly thankful to have met all the wonderful workers who are part of GAGE's organizing committee: you all were so incredibly kind to me, and showed me that change is possible through true collective organizing.

I would like to thank my mom Anna, my dad Andrea, and my brother Riccardo for being my biggest cheerleaders from thousands of kilometers away. A call once a week is really not enough to fill the distance from home, and I wish we spent more time together.

Finally, the last thank you goes to my husband James. It is hard to imagine getting this far without you on my side. You had no manual on how to deal with a busy spouse working on his doctorate, and yet you always had the right words to support and encourage me. I love you.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Among domain specific applications, medical information retrieval has gained significant prominence in the last few years.

On one hand, the Internet has become a primary source for consumer health information seeking. A recent survey [41] showed that 72% of all adults in the U.S. seek information about health issues online (mostly focusing on diseases and treatments), while 8% have asked questions or shared their experiences [40]. As a result, many approaches designed to improve consumer health search have been proposed (e.g., [173, 84, 109, 101, 177, 130]). Furthermore, researchers have studied how to characterize the behavior of lay people looking for health information online (e.g., [103, 152, 153, 154, 165, 167, 169, 171]), analyze aggregate health trends (e.g., [23, 42, 166, 162, 105, 161, 79]), and identify search engine users affected by specific diseases (e.g., [104, 106, 168, 129, 10]). However, efficiently finding medical information remains a challenge for lay users as many online resources, even those addressed to consumers, employ abundant medical terminology that consumers might not know or be familiar with [174].

On the other hand, the amount of medical information available to health experts has increased dramatically in the last few years. For example, the number of articles in PubMed[1], one of the largest repository of biomedical literature, increases by approximately 1 million documents each year[2]; today, it is over 28 million; this growth has been directly attributed to rapid advances in clinical research [36]. At the same time, the adoption rate of electronic health records (EHR) soared in the last few years, going from 41% of hospitals in 2012

[1] `https://www.ncbi.nlm.nih.gov/pubmed/`

[2] `https://www.nlm.nih.gov/bsd/licensee/baselinestats.html`

to 59% in 2015 in the United States alone [3]. Large repositories of research and clinical information are both a blessing and a curse for the medical community: while they enable clinical practices such as evidence-based [49] and precision medicine [29], they also represent a new set of challenges for health professionals, who often struggle to keep up-to-date with current literature [15, 142, 97] or adapt to new technologies [67, 34, 56]. Understandably, this has led to the introduction of shared tasks designed to advance the state of the art in medical search systems: OHSUMED [61] focused on retrieving biomedical literature for short, keyword-heavy clinical queries; TREC Genomics [60] tackled search in support of genomics research (i.e., retrieval of literature about protein interaction, gene mutations, etc.); ImageCLEFmed [72] studied multimodal retrieval for clinical practice; MedTrack [150, 151] was concerted with improving retrieval of clinical notes; the Clinical Decision Support (CDS) track at TREC was created to evaluate search systems designed to retrieve relevant literature for a patient's clinical note [116, 117, 118]. The latter task is, in many ways, a good model for complex medical search tasks in support of clinical practice:

- The ability to retrieve relevant literature in support of the medical decision process is a pressing need, as indicated by the popularity of evidence-based medicine.
- Health professionals struggle to keep up with advances in clinical research.
- PubMed and other search systems currently used by clinicians to retrieve literature are primarily designed to handle short, keyword-heavy queries[3].

Over the years, several approaches have been proposed to improve information systems designed to support clinical practice; some leverage medical ontologies, such as Medical Subject Headings[4] (MeSH) or the Unified Medical Language System[5] (UMLS) to perform query reformulation (e.g., [134, 35, 84, 86, 83, 94, 95]); others explored the use of pseudo

[3]For example, PubMed search tool can only perform Boolean search; therefore, adapting clinical note to an appropriate Boolean query is a time-consuming task that requires clinical expertise, as well as technical knowledge of PubMed itself.

[4]`https://www.nlm.nih.gov/mesh/`

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

relevance feedback (PRF) (e.g., [2, 25, 65, 178]). Systems combining the two have also been proposed (e.g., [138, 7, 143, 65, 131, 128]). Furthermore, some have exploited the use of multiple auxiliary collections for both clinical notes and medical literature retrieval (e.g., [179, 99]). Finally, graph-based approaches have been proposed as a mean to perform query expansion (e.g., [35, 53, 59, 125]) and query to document inference (e.g., [76, 80, 48]).

In this dissertation, I will argue that challenges in health search are caused by a knowledge and language gap both laypeople and health experts suffer from; I will propose several methods to reduce it, thus improving the search outcomes. While previous works acknowledged the effect of this gap on medical search, this dissertation explicitly frames query reformulation with respect to this gap for both consumer and health professionals. This gap is directly tied to several unique aspects of the medical domain:

- **Domain breath:** from healthcare management to genomics, the medical domain encompasses a vast set of disciplines and fields. Its broadness poses challenges in defining what a domain expert is, as an in-depth knowledge in a field does not necessarily translate to familiarity with others.

- **Specialized vocabulary:** the medical domain is characterized by a rich and wide vocabulary, routinely used by health professional to describe diseases, symptoms, and treatments. This represents a challenge not only for medical entity recognition systems [124], but also for health professionals, as it can lead to clinical errors [75]. Such specialized language is also widely used on health websites that supposedly target consumers, which leads to a higher-than-recommended readability level [52, 63, 149].

- **Synonymy and polysemy:** in the medical domain, the same concept is often represented by multiple expressions (synonymy; for example, *alopecia* and *hair loss* refer to the same condition) or the same expression can stand for two or more concepts (polysemy; for example, *progesterone* can either refer to the hormone or the pharmacological substance.)

These aspects hinder the search experience of different cohorts of users differently, mostly depending on their level of expertise: for lay people, the complexity of the medical vocabulary represents a significant barrier for retrieving relevant and reliable information. For experts, the gap could be due to searching in a field different from their area of expertise or due to time constrains (physicians in demanding clinical setting might not have sufficient time to craft an appropriate Boolean query to submit to PubMed.)

1.1 Hypotheses

The challenges described in the section above can be reformulated in two sets of hypotheses, which will be discussed in the reminder of this dissertation.

Hypothesis 1: *Lay health searchers suffer from language gap that can be bridged using semantic analysis to reformulate queries or rerank search results.*

The first part of my dissertation is concerned with examining, quantifying, and ultimately reducing the language gap lay people suffer from in consumer-oriented health searches. My efforts in this domain are detailed in Chapter 2.

> **Hypothesis 1.1**: *The language gap between laypeople and online health resources negatively affects their ability to retrieve answers to health questions.*
>
> I show and quantify the language and knowledge gap laypeople suffer from through a task-based experiment, which is described in Section 2.2.2. Users were asked to find the answer to specific questions using search results from a commercial search engine. Some of the users were health experts, while the other were laypeople. Results demonstrating the negative effect of the language gap are presented in Section 2.2.3.1.

Hypothesis 1.2: *Query reformulation reduces language gap of laypeople searching health-related information online.*

I validate this hypothesis by showing that reformulated queries are preferred by lay people, lead to better understanding of health topics, and retrieve more relevant documents. Three existing laypeople-to-expert synonym mappings (Section 2.2.1.1) were used to reformulate queries and reduce the language gap through a process called "query clarification". The clarification process is detailed in Section 2.2.1.2. Query clarification is evaluated in Section 2.2.3.1. Furthermore, a classifier designed to select the most appropriate synonym mapping is also proposed and evaluated favorably, as shown in Section 2.2.4.

Hypothesis 1.3: *Re-ranking search results based on semantic health features improves search quality for consumer health queries.*

Medical web pages — even those addressed to consumers — are likely to use proper medical terminology laypeople might not be familiar with. This mismatch is another form of language gap between lay people and health resources. I show how this gap can be closed by introducing a supervised learning to rank approach for this problem. In particular, I propose a novel set of features that capture semantic similarity between queries and web pages; this approach is detailed and validated in Section 2.3.

Hypothesis 2: *Reformulating complex medical queries by taking into account explicit and latent medical concepts improves retrieval of medical literature.*

As previously mentioned, medical experts also struggle to formulate appropriate medical questions, either due to lack of knowledge in a specific field of medicine, or because they cannot invest time in formulating complex Boolean queries. In Chapter 3, I investigate the

use of supervised and unsupervised techniques to reformulate patient clinical notes, such that they can be employed as queries for medical literature retrieval. In particular, I focus on proving the following hypotheses:

Hypothesis 2.1: *Well-formed clinical notes can be expanded using unsupervised or supervised techniques to increase the precision of retrieval systems.*

In Section 3.2, I discuss two techniques — one supervised, the other unsupervised — that generate queries suitable for medical literature retrieval. Both take advantage of pseudo relevance feedback to obtain a list of candidate terms for query expansion; the unsupervised method uses term distribution heuristics to filter candidate terms, while the supervised method learns to predict the importance of candidate terms using a convolutional neural model. Performance of both models are thoroughly analyzed in Section 3.4; both are competitive with the state of the art, achieving up to 8% improvement in nDCG.

Hypothesis 2.2: *Clinical notes that contain significant noise (i.e., medical and clinical abbreviations, incomplete sentences, redundant or unnecessary information) can be reduced to be used as queries for medical literature retrieval.*

As the implementation of clinical notes varies from institution to institution, there is no consistent format in which clinical notes are written [66]. Notes in publicly available collections make heavy use of abbreviations, are heavily comprised of not-fully specified sentences, and include unnecessary information about patients' treatment and hospitalization history [71, 121, 145, 146, 147]. In Section 3.5, I detail a system designed to reformulate noisy clinical notes; the proposed method achieves an improvement of 67% over the unmodified clinical note, and a 27% improvement over state of the art query reduction methods.

Through proving the hypotheses listed above, I will demonstrate, in Chapters 2 and 3, that challenges connected with the effect of the language and knowledge gap can be effectively mitigated to improve search outcomes in the medical domain. Then, in Chapter 4, I will examine the impact of this dissertation and discuss potential future work that could extend this research effort.

Parts of Chapters 2 and 3 are reproductions of my jointly authored publications [27, 125, 126, 127, 128, 130, 131, 132].

Chapter 2

Laypeople as Health Searchers

As pointed out in Chapter 1, the Internet has become a primary source of health information for the majority of adults living in the United States [41, 40]. Lay people have come to rely on the Internet as a tool to seek information about specific diseases or medical problems. However, this process is often challenging due to the gap between language used by consumers to describe their conditions and proper medical vocabulary. Trustworthy health care resources, even those addressed to consumers, employ appropriate medical terminology; yet laypeople do not have the necessary knowledge to express their information need using such vocabulary, thus struggling to satisfy their information needs [174].

This gap is difficult to overcome either by searchers, who need to learn a specialized vocabulary to describe their information need, or by experts who are trying to assist them, as they have to speculate on the ways in which laypeople will phrase their intent. This language gap was noted as one of the primary reasons for failures of retrieval engines [21].

In this chapter, I focus on understanding and reducing such language gap. In particular, Section 2.2 will discuss how to quantify this language gap; furthermore, a method to reduce the language gap with query reformulation will be introduced. I will be referring to this process as "query clarification", as the lay concept in each query is "clarified" using the equivalent expert concept. This approach takes advantage of three existing laypeople-to-expert synonym mappings; each map associates one or more layperson expressions to one or more expressions used by medical professionals. Then, in Section 2.3, I will propose and evaluate a learning to rank (LtR) approach that leverages statistical and semantic features

to address the language gap. LtR algorithms have been successfully employed to promote understandability in medical health queries [101] and retrieve medical literature [81].

The reminder of this chapter is organized as follows: in Section 2.1, I will present an overview of research efforts related to improving consumer health search. Then, in Section 2.2, the process of query clarification will be formally introduced and its impact evaluated, addressing hypotheses *1.1* and *1.2*. Section 2.3 presents the methodology and the results of the proposed LtR system, addressing *1.3*. Finally, in Section 2.4, I will provide a brief summary of the innovations described in this chapter.

2.1 Related Works

Interest in medical search is steadily increasing, and many approaches to improve its accuracy have been proposed. For laypeople, researchers have focused on building systems to retrieve relevant and trustworthy health information on the web.

2.1.1 Laypeople as Health Information Seekers

Interaction between consumer seeking health information and web search engines has been extensively studied in recent years. Early on, Eysenbach and Köhler [39] noticed that consumers' query formulation is often suboptimal. Moreover, they observed that laypeople struggle with identifying trustworthy websites. Spink et al. [133] examined a large query log from Excite[1] and AlltheWeb[2]. Their findings suggest that most consumers fail to understand the limitations of web search when searching medical advices; furthermore, they rarely reformulate queries to include synonyms or alternate health expressions that could increase the quality of retrieved results. Toms and Latter [144] also noticed that consumers are often unable to properly formulate queries when looking for health resources.

[1] `http://www.excite.com/`
[2] `http://www.alltheweb.com/`

More recently, Cartright, White, and Horvitz [22] studied the behavior of consumers when searching for health information. Their findings suggest that users perform evidence-directed and hypothesis-directed exploratory health searches. Powell et al. [110] conducted a comparative study between popular search engines (Google, Bing, Yahoo! and Ask.com) in retrieving health information about breast cancer. They noticed that, while all the search engines were able to provide somewhat satisfactory results, the rankings of retrieved web page was often suboptimal, therefore leaving room for improvement to help users get more relevant information.

Finally, Zuccon, Koopman, and Palotti [180] analyzed the results retrieved by two commercial web search engine (Google and Bing) on a set of queries formulated by laypeople describing medical symptoms. For both engines, only three of the top ten retrieved results were both relevant and from trustworthy websites. Their analysis suggests that current search engines are not sufficiently equipped to satisfy the information need associated with the laymen queries in their dataset.

2.1.2 Influence of Domain Expertise in Health Search Behaviors

Researches have also studied the differences between experts and laypeople when performing health-related searches. White, Dumais, and Teevan [153] analyzed interaction logs from Google, Yahoo!, and Microsoft Live Search. Based on their analysis, the authors concluded that health experts—compared to laypeople—are more likely to visit authoritative medical websites, issue long queries, use domain appropriate terms, spend more time searching, and reformulate queries often. Palotti, Hanbury, and Müller [100] proposed a set of features that could help discern queries issued by health professionals from queries issued by laypeople.

While our experiments confirm some of the aforementioned findings, our work focuses on how to bridge the gap between laypeople and medical experts rather than analyzing the differences between the two groups.

2.1.3 Efforts in Improving Consumer Health Search

The interest in helping laypeople access reliable medical resources has increased in the last few years. Zeng et al. [173] started the Consumer Health Vocabulary (CHV) initiative, a resource designed to link medical terms and expression used by consumers to terms health care professionals use. CHV is included in UMLS since version 2011AA. Similarly, [160] constructed MedSyn, a database that includes the mapping of layperson vocabulary to 12 expert terms extracted from both UMLS and social media posts; Yates, Goharian, and Frieder [163] proposed a system to programmatically extract synonyms from a corpus of medical forum posts; they utilized their approach later to extract the mentions of adverse drug reactions (ADR) from social media [161]. Can and Baykal [17] created MedicoPort, a retrieval engine that enhances health queries using UMLS. Luo et al. [84] built MedSearch, a search engine designed to process long, discursive queries and retrieve trustworthy results from a set of hand picked sources. The proposed system increased search results diversity, as well as suggesting new queries.

Other efforts include Stanton, Ieong, and Mishra [137], who studied the use of circumlocution in diagnostic medical queries (i.e., situations in which a non-expert uses many words to describe a symptom in place of the appropriate medical term). The authors proposed a supervised approach to link circumlocutory queries to medical concepts. Shen et al. [123] considered a concept-based similarity model; MetaMap [6] was used to extract medical concepts from the queries and documents; furthermore, the authors experimented with using concept-based pseudo relevance feedback. Their best approach also resulted in a 11% improvement over the baseline. Oh and Jung [98] used a combination of rule-based expansion of medical abbreviations, expansion through terms in the clinical notes, and pseudo relevance feedback. Their system achieved a 8% improvement over the baseline. Palotti et al. [101] suggested to use statistical and readability features to promote web pages that are relevant and are at an appropriate reading level for lay people.

In recent years, a shared task aimed at improving consumer health search was introduced at the CLEF eHealth Evaluation Lab [45]; the task ran in subsequent years [46, 182, 181]. In the first three editions, partecipating systems were asked to retrieve relevant documents from the Khreshmoi collection [51], a set of certified websites by the Health-On-Net foundation[3] and other hand-picked trusted resources; more recently, a general-domain dataset (ClueWeb 12 Category B13[4]) was used as test collection, making this shared task more realistic. Alongside some collaborators, this dataset was used to explore the use of semantic relationships between terms for query expansion [125]. In a closely related work, Goeuriot, Kelly, and Leveling [44] provided a more detailed analysis of the impact of query complexity on the performance of the participating systems; their findings suggest that the increase in query complexity affected the retrieval performances.

2.2 Closing the Language Gap through Query Clarification

Our goal was to evaluate whether the language gap affects negatively the search result accuracy; further, we wanted to evaluate whether three synonym mappings could be used to reduce the language gap and improve the quality user search experience and the relevancy retrieved results. As such, we performed a task-bases user study as shown in [130].

2.2.1 Methodology

We bridge the gap between laypeople and experts in the health search domain to improve users' ability to answer medical questions. As such, we investigated using three different synonym mappings to perform query clarification.

For each query, we generated three clarified queries using the synonym mappings described in Section 2.2.1.1. Each mapping associates an expression from layperson's vocabulary (i.e., a word or phrase a non-expert would use to describe a health concept) to

[3] *Health On the Net (HON) Foundation* (`http://www.healthonnet.org`) is an organization that certifies those health-related websites that meet specific reliability standards ("HONcode" of conduct)

[4] `http://www.lemurproject.org/clueweb12.php`

one or more expressions used by medical professionals, which we refer to as "clarification candidates". Section 2.2.1.2 describes the algorithm used to select the most appropriate expression among clarification candidates.

For each of the four query versions (the original and the three derived from clarification), we used Bing to retrieve relevant search results. In Section 2.2.1.3, we discuss the overlap between each synonym mapping, as well as the overlap between the retrieved results.

2.2.1.1 Medical Synonym Mappings

The following are the three medical synonym mappings that are utilized here to clarify layperson user queries; some filtering is done to adjust to our approach.

Behavioral

Based on Yom-Tov and Gabrilovich [166], this mapping links expressions commonly used by laypeople to describe their medical condition to 195 symptoms listed in the International Statistical Classification of Diseases and Related Health Problems, 10th Revision (ICD-10)[5]. The synonyms were generated in two ways. First, the most frequent search terms that led users to click on Wikipedia pages describing symptoms were selected. Second, frequently occurring lexical affinities [20] were added to the list. Lexical affinities are word pairs appearing in close proximity in the 50 highest ranked search results retrieved when symptoms were used as queries. The list was validated by medical professionals, and 88% of terms were found to be appropriate expansion terms for the symptoms. The list was generated using search information from the Yahoo! search engine collected in 2010. A detailed description of this mapping can be found in [166].

MedSyn

Based on Yates and Goharian [159], this synonym mapping focuses on diseases and symptoms. It was generated from a subset of UMLS filtered to remove irrelevant terms types.

[5] `http://www.who.int/whosis/icd10/`

SIDER 2 [77] was used to keep only terms with UMLS semantic types that were assigned to side effects listed on drug labels. Synonyms of these terms were identified using UMLS' semantic network and added to the map. Finally, relevant common terms from a drug review data set [159] were added to the map as synonyms of the appropriate terms. To ensure that only expert terms were added to queries, we kept only terms designated as *preferred terms*[6] in UMLS as candidate expressions (i.e., expressions used to clarify a query).

DBpedia

This mapping takes advantage of Wikipedia redirect pages as a mean to map laypeople expressions to expert terminology. Redirect pages are meant to route users to the most appropriate expression for a concept. For example, the Wikipedia page for "acid reflux"[7] redirects to "gastroesophageal reflux disease"[8]. Wikipedia redirect pages have been successfully employed in building general ontologies [140], creating domain specific thesauri [93], and improving query reformulation [92, 158]. We took advantage of DBpedia[9], a project aimed at extracting structured information from Wikipedia, to parse redirect pages. Through this knowledge base, we label two expressions X and Y as synonyms if there exits a redirect from page X to page Y. To prevent query drift, we only kept those redirect terms which led to a Wikipedia page describing a medical symptom, drug, or disease. This ensures that those terms in the query that are not health-related are not attempted to be clarified.

2.2.1.2 Candidate Selection

In some instances, a synonym mapping associates an expression (which could be either a word or a phrase) in a query with more than one clarification candidate $\{c_1, \ldots, c_m\}$. However, not all clarification candidates are equally suitable for expansion: some are more apt at representing the medical concept in the query and are therefore preferred in medical pages

[6] In UMLS, an expression is labeled as *preferred term* if it is found to be the most appropriate to represent a concept.

[7] `http://en.wikipedia.org/wiki/Acid_reflux/`

[8] `http://en.wikipedia.org/wiki/Gastroesophageal_reflux_disease/`

[9] `http://dbpedia.org/`, accessed July 2013

containing reliable information. Therefore, our goal is to select the clarification candidate $\mathbf{c_k}$ that better represents the medical concept expressed by consumers in the query. The following heuristic was considered to achieve this goal: when multiple clarification candidates are identified by a mapping, we choose the candidate c_k whose probability of appearing in health-related Wikipedia pages is maximized. Wikipedia was deemed appropriate to determine the best clarification candidate because of its strict manual of style[10] and the expertise of the editors curating the Medicine Portal[11] (more than half of the editors are medical practitioners, 85.5% holds a university degree [55]).

Let $\mathcal{W} = \{P_i\}_{i=1}^{i=|\mathcal{W}|}$ be the set of all pages in English Wikipedia (special pages, such as category or disambiguation pages, are not included), $\mathcal{W}_H$ the set of all health-related pages. Then, for each candidate term $t_j \in \mathcal{T}$, we estimate its odds ratio of being health related as follows:

$$\mathrm{OR}(t_j) = \frac{\Pr\{t_j \in P_i \wedge P_i \in \mathcal{W}_H\}}{\Pr\{t_j \in P_i \wedge P_i \in \mathcal{W}\}} \tag{2.1}$$

The two probabilities are estimated using Maximum Likelihood Estimation (MLE); that is, they are calculated by dividing the number of documents with term t_j by the total number of documents.

In accordance with the previously stated heuristic, the candidate maximizing the following equation is selected for clarification:

$$\underset{c_k \in \{c_1,\ldots,c_m\}}{\arg\max} \left(\mathrm{OR}(c_k)\right) \tag{2.2}$$

Intuitively, the more a clarification candidate appears in health-related Wikipedia pages, the more likely it is that the candidate is the most appropriate expression to describe the concept in the query. Therefore, we clarify a query with the expression c_k that maximizes Equation 2.1.

[10] `http://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style/Medicine-related_articles`

[11] `http://en.wikipedia.org/wiki/Portal:Medicine`

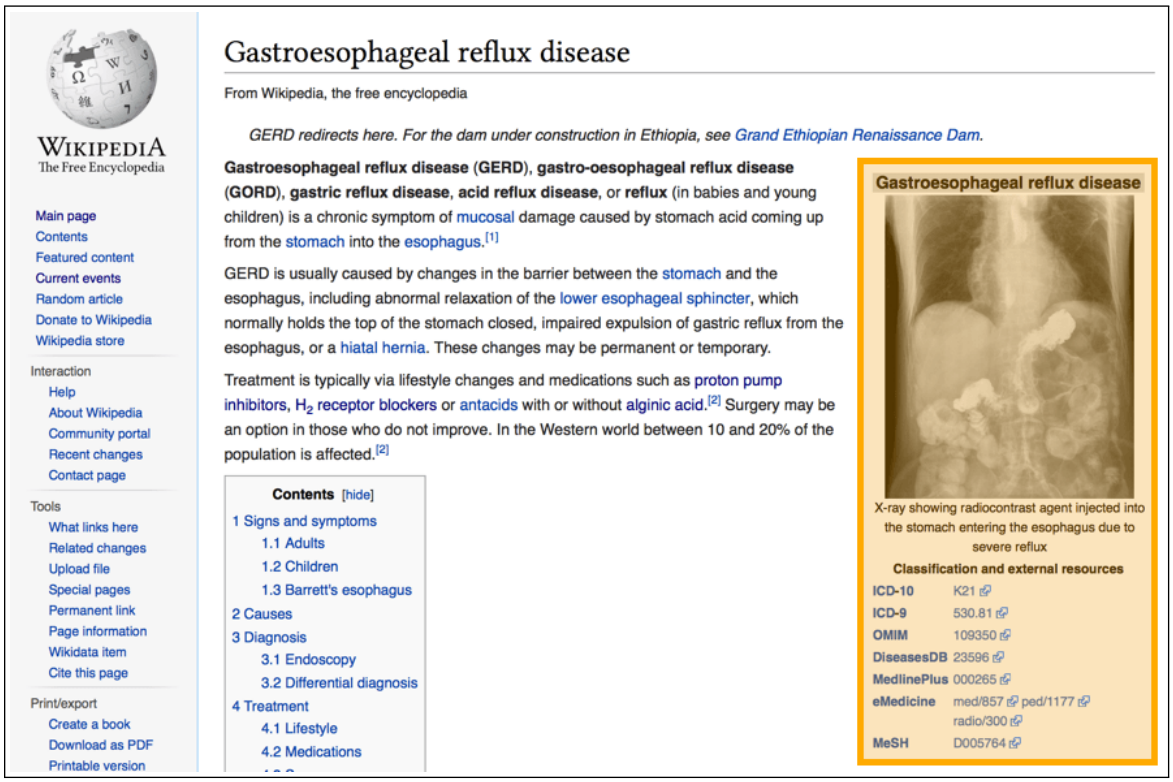

Figure 2.1: A screen shot of the Wikipedia entry for "Gastroesophageal reflux disease". The information box is displayed on the right side of the page, highlighted in orange. Because it contains several medically-related identification codes, this page was identified as health-related.

The set $\mathbb{W}$ was defined over a snapshot of Wikipedia obtained on November 4, 2013. We took advantage of the content of the information box (e.g., Figure 2.1) of each Wikipedia entry to define the set $H(\mathbb{W})$ (i.e., to determine which pages should be considered health-related). In detail, any page whose information box contained one of the following medically-related identification codes was designated as health-related: *MedlinePlus*, *DiseasesDB*, *eMedicine*, *MeSH*, or *OMIM*. Of $2,794,145$ unique pages indexed, about 0.88% ($24,654$ pages) were identified as health-related.

Table 2.1: Size of the synonym mappings.

	Unique expressions	Synonym pairs
Behavioral	593	611
MedSyn	6,760	43,703
DBpedia	64,652	177,116

We avoided augmenting a query with more than one clarification candidate to minimize the likelihood of query drift. Because all queries in the dataset contain no more than a medical concepts, if multiple expressions in a query can be mapped to an expert term using a synonym mapping, we consider the longest, as it fully captures the information need of the user. If multiple expressions of the same length can be clarified, we choose the one with the highest conditional probability.

2.2.1.3 Overlap Between Mappings

We compare and contrast the synonym mappings introduced in Section 2.2.1.1 as a means of providing a greater understanding of their differences and similarities. In detail, we examine the size of the mappings, as well as the overlap between each pair. Finally, we analyze the overlap of set of results retrieved for each query in our dataset before and after being clarified by each synonym mapping.

Table 2.1 shows the size of each synonym mapping in terms of unique expressions and in terms of synonym pairs (i.e., pairs of non-expert expression X and expert expression Y). An expression may either be a single word ("GERD") or a multi word phrase ("gastroesophageal reflux disease"). *Behavioral* has the fewest number of expressions, whereas *DBpedia* has the most. In fact, *Behavioral* is much closer to a one-to-one mapping than *MedSyn* and *DBpedia*, as both include relationships between many more pairs of synonyms. Note, however, that *Behavioral* only includes medical symptoms, which may explain its size in comparison to the other synonym mappings. The size difference shown in Table 2.1 unsurprisingly affects the

Table 2.2: Percentage overlap between the lists of synonyms.

	Behavioral	***MedSyn***	***DBpedia***
Behavioral	-	21.3% (126 expressions)	98.5% (584 expressions)
MedSyn	1.9% (126 expressions)	-	8.0% (540 expressions)
DBpedia	0.9% (584 expressions)	0.8% (540 expressions)	-

Each cell (i, j) in the table represents the overlap of synonym mapping i with synonym mapping j as a percentage of the size of mapping i. To better understand the relative size of each overlap, the number of overlapping expressions is also reported.

number of clarification candidates of each mapping. *Behavioral* selected, on average, M=1.02 (SD=0.24) candidates per query, while *MedSyn* selected M=1.16 (SD=1.07) candidates. The difference between the two is not statistically significant (Mann-Whitney U test, $p = 0.243$). *DBpdia*, the largest mapping, consistently selected the largest number of candidates per query: M=2.46 (SD=4.42) (difference is statistically significant over *Behavioral* and *MedSyn*, $p < 0.05$).

The overlap between each list of synonyms is shown in Table 2.2. For each cell (i, j) in the table, we report the overlap of synonym mapping i with synonym mapping j as a percentage of the size of mapping i. *Behavioral*, the mapping with the smallest synonym list (as shown in Table 2.1), is almost completely contained (98.5%) within *DBpedia*, the largest mapping. *Behavioral* and *MedSyn* have far fewer expressions in common, as about one fifth (21.3%) the expressions in *Behavioral* are also present in *MedSyn*.

Table 2.3 shows the overlap between the unclarified queries and the queries clarified by each mapping (as described in Sections 2.2.1.1 and 2.2.1.2). In cases where a synonym mapping had no clarification expression to add, we say that the *null* term was added; this allowed us to compute overlap between the unclarified query (which we refer to as "no clar.") and each synonym mapping. By definition, "no clar." adds the *null* term to each query. *MedSyn* added the *null* term (i.e., did not add any clarification expression to the query) 30%

Table 2.3: Query overlap between the unclarified query ("no clar.") and the queries clarified by each mapping.

	no clar.	***Behavioral***	***MedSyn***	***DBpedia***
no clar.	-	2%	30%	0%
Behavioral	2%	-	28%	74%
MedSyn	30%	28%	-	36%
DBpedia	0%	74%	36%	-

MedSyn is the most similar to the baseline, while *Behavioral* and *DBpedia* are the most similar synonym mappings. Unlike Table 2.2, this table is symmetrical, as all queries in the dataset were clarified using all synonym mappings.

of the time, while both *Behavioral* and *DBpedia* added a expression to the vast majority of queries. *Behavioral* and *DBpedia* often lead to similar clarification (74% overlap), which is to be expected given the high overlap between the two synonym lists. Finally we note that, despite the fact that only 8% of the synonyms found in *MedSyn* occurred in *DBpedia* (Table 2.2), the overlap in terms of expressions added to the queries by the two mapping was considerably higher (36%). This outcome is likely due to the fact that the queries in our dataset, which are among the 500 most common health queries on Bing (Section 2.2.2.1), contain health expressions that are very likely to be included in both synonym mappings.

The overlap between the URLs of the retrieved results is shown in Table 2.4. Results confirm that *Behavioral* and *DBpedia* are the most similar mappings. Both have little overlap with the URLs of results retrieved with the unclarified query (13% and 14%, respectively); a slight increase can be observed for both mappings when overlap is measured with respect to the snippets of retrieved results. Queries clarified with *MedSyn* retrieved, on average, 38% of the results retrieved by the unclarified query.

Summarizing our comparison, queries clarified using *Behavioral* and *DBpedia* retrieve the most similar set of results, even though the former mapping comprises of only a small subset of the latter. Of all synonym mappings, *MedSyn* yields the most similar results to the

Table 2.4: Overlap of the URLs of results retrieved by the unclarified query ("no clar.") and by the queries clarified by each mapping.

	no clar.	***Behavioral***	***MedSyn***	***DBpedia***
no clar.	-	14%	38%	13%
Behavioral	14%	-	36%	74%
MedSyn	38%	36%	-	42%
DBpedia	13%	74%	42%	-

MedSyn is most similar to the unexpanded baseline, but still adds a significant number of URLs.

baseline; yet, it still adds a significant number of clarification expressions and URLs over the unclarified query.

2.2.2 Experimental Setup for Task-based User Study

To evaluate the effectiveness of our clarification strategy, we used the three synonym lists introduced in Section 2.2.1.1 to clarify 50 queries from a Bing query log. Details regarding the set of queries are provided in Section 2.2.2.1. Laypeople and medical experts were enrolled to assess the impact of the proposed methodology. For each query, we created a multiple-choice question; participants were required to answer it to demonstrate their understanding of the retrieved results. We overview the query creation process in Section 2.2.2.2. Query clarification was evaluated using an online platform we introduce in Section 2.2.2.3.

All the resources detailed in this section (queries, questions, and anonymized user interaction reports) are publicly available at the authors' GitHub page[12].

2.2.2.1 Queries Dataset

As previously mentioned, we studied the impact of query clarification on a sample of common health-related queries from a Bing query log. To do this, we extracted the set of all English-

[12] `https://github.com/Georgetown-IR-Lab/query-clarification-data`

Table 2.5: An example of query in our dataset.

Mapping	Query	Question
no clar.	excessive burping	"*Which of the following solution does NOT help with excessive ructus?*" (avoiding drinking through a straw, taking an antacid, eating slowly, **swallowing air**)
Behavioral	excessive burping belching	
MedSyn	excessive burping eructation	
DBpedia	excessive burping belching	

The first mapping, "no clar.", represents the original unclarified query as extracted from the Bing query log. The last column contains the question formulated by the authors. In parentheses we report the four corresponding answers (the correct one is in **bold**).

language queries submitted to Bing by users in the United States during November 2013. This set was filtered to extract those queries which contained a symptom, drug name, or disease name, or one of their synonyms, as listed in Wikipedia. We randomly sampled 50 out of the 500 most common queries in the resulting list. Sampling was done to reduce the dimensionality of the dataset, thus making the experimentation more tractable.

The 50 queries in the dataset contain 93 unique terms and have an average length of 2.6 terms (median length is equal to 2). This is not statistically significantly different (rank-sum test) from the queries in the larger set of 500 queries, which have an average length of 2.5 (median is 2) and contains 463 unique terms.

2.2.2.2 Evaluation Questions

The process laypeople follow while looking for medical information on the Internet is akin to a task-based retrieval scenario: consumers have a specific information need that they try to satisfy through web search engine. Thus, for our task-based experiment, we created, for each query, a question that would estimate the quality of the retrieved results in providing helpful information to a user. Users in our scenario are given a similar task to [58], where medical students were asked to use a search system to gather information to answer a question.

Such approach is also common in focus groups examining the behavior of laypeople seeking health information on the web [39, 144]. Since a users' ability to correctly answer questions is uncorrelated with the number of relevant documents read [58] or precision and recall [57], we consider the users' question answering accuracy when we analyze our results.

Our design goal was to formulate questions that (a) were highly relevant to the query, (b) required reading at least one, if not many, of the links shown and (c) were not easily intelligible by reading the snippets provided with each search result. Each question was created using the following procedure: first, the authors read the query and content of the search results; then, they formulated a question based on the content of the retrieved web pages; finally, they generated four possible answers—one correct, three wrong. The volume of data needed by our study ruled out the option of proposing open questions.

2.2.2.3 Online Evaluation Platform

We developed a website (Figure 2.2) to determine the effectiveness of the proposed clarification methodology. Through this website, laypeople and medical experts answered a set of health-related, multiple-choice questions using a set of search results retrieved using Bing. For each query in the dataset, we showed participants the query itself and the question simulating the information need associated with the query. Users were asked to find the answer to the question presented to them by using the displayed search results. We required the participants to open (click) at least one link before choosing the correct answer among four possible choices to prevent bias in results selection. To minimize the number of factors involved in the study, users were not allowed to modify the displayed query. For each respondent and each query, an interaction report consisting of the links clicked and the answer given was created.

We interleaved search results to quantify the impact of each synonym mapping we used for query clarification. Interleaving, introduced by [68], is a technique designed to receive implicit user feedback about two retrieval methods without introducing bias due to the

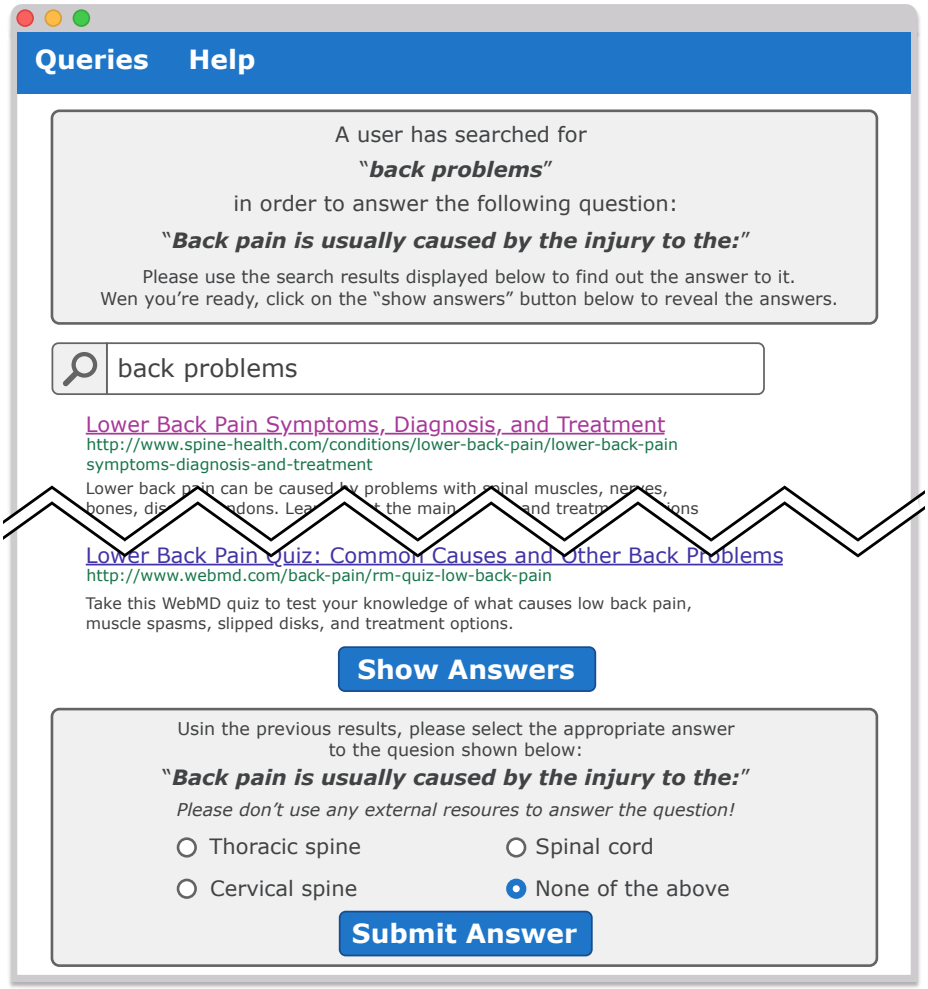

Figure 2.2: The main interface of the website. The top third of the screen shows the question for the user, while the middle part displays the original query and ten interleaved results. The bottom section shows the question which the user is asked to answer. Even when results obtained via a clarified query are presented, the original query is shown; users are not allowed to reformulate the query at any point. The multiple choice options to the question are initially hidden and can be revealed by the user after opening (clicking) at least one result.

presentation of the results. Team draft interleaving [112] was chosen for the evaluation platform; as its name might suggest, it mimics how players are usually divided in teams at the beginning of friendly matches. Given two ranked lists A and B of retrieved results,

$A = \{a_1, \ldots, a_n\}, B = \{b_1, \ldots, b_n\}$, we operate as follows: for each pair of results a_i and b_i of rank i, an unbiased coin is flipped; if heads, a_i is ranked before b_i in the interleaved set of result; if tails, b_i is ranked first. As detailed in [111], team draft interleaving shows comparable levels of expert agreement to other interleaving methods, and it is less prone to introducing bias.

We tested query clarification among laypeople recruited using Amazon Mechanical Turk[13]. Each participant was asked to answer 20 medical questions. Workers were paid between \$2.00 and \$4.50 (M=\$3.53, SD=\$0.99), depending on when they accepted the task. We enrolled as many workers as needed to obtain at least 5 interaction reports per query per pair of methods. In total, 80 workers registered for the task.

We also enrolled 12 freelance medical experts using Elance[14]. These workers were paid \$20.00 for their efforts. We provided interleaved results retrieved using original queries and queries clarified by *MedSyn* to this group of participants. *MedSyn* was chosen because its promising results on preliminary tests. The size of this group was also determined by the need of at least 5 interaction reports for each query.

2.2.3 Results

We analyzed the results collected in the two task-based retrieval experiments. In particular, we were interested in finding out (a) whether laypeople were impacted by a language gap, (b) whether lay users would prefer clarified queries, and (c) whether clarification benefits searchers with no medical experience. Furthermore, we wanted to quantify potential differences between laypeople and experts in our experiments. We discuss the impact of query clarification in Section 2.2.3.1; the differences between lay and expert users are highlighted in Section 2.2.3.2.

[13] `http://mturk.amazon.com/`
[14] `http://www.elance.com/`

2.2.3.1 Impact of Query Clarification

Quantifying the Language Gap

The first step in studying the effect of query clarification was to compare the success rate of lay and experts users when no clarification is used. We found that laypeople answered correctly to 63.2% of the questions, while experts were able to determine the correct answer in 73.3% of the cases (difference is statistically significant, Welch's t-test, 2-tailed, $p < 0.05$). This observation confirms a prominent gap between experts and lay users exists in our experimental setup.

Do Lay Users Prefer Query Clarification?

To measure whether lay users preferred results retrieved through query clarification, we used the implicit feedback given by users through team draft interleaving. This implicit feedback allows us to use a voting scheme to determine which mapping is the preferred one, as the feedback is akin to a vote on a ballot.

Team draft interleaving assumes that, for each query, the method preferred by a user is the one that retrieved the majority of web pages they visited. Thus, we assigned a preference to synonym mapping i when compared with mapping j if a user clicked more results retrieved by a query clarified with mapping i than results retrieved by a query clarified with mapping j.

The Kemeny-Young method [172] was used to determine the users' preferred ranking among the three synonyms lists and original query ("no clar."), which we will refer to as "candidates" throughout the rest of this section. The Kemeny-Young method was originally designed to combine prioritized votes; in information retrieval, it has been used to perform rank aggregation on search result sets [38, 31], on candidates in question answering tasks [4], and on short texts in social media [139].

The score for each ranking (which, in this context, is a permutation of the list {no clar., *Behavioral*, *MedSyn*, *DBpedia*}) is computed by summing the number of votes for each pair of candidates in the ranking. The ranking with the highest score is the Kemeny ranking.

Formally, given a ranking $r = \{c_1, \ldots, c_m\}$, its score $S(r)$ is calculated as:

$$S(r) = \sum_{\substack{i,j \in \{1,\ldots,m\} \\ i<j}} \sum_{u \in U} \begin{cases} 1 & \text{rank}_u(c_i) > \text{rank}_u(c_j) \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

Where $u \in U$ is a user in our experiment, and $\text{rank}_u(c_m)$ is the rank assigned by user u to candidate c_m. In other words, equation 2.3 computes the sum of the number of users who ranked candidate c_i over candidate c_k for all possible candidate pairs c_i, c_j.

By definition, the Kemeny ranking maximizes the number of pairwise agreement between users, where two users agree if they have expressed preference of a candidate over another candidate. In other words, a ranking $r = \{c_1, \ldots, c_m\}$, will score high if, for all $i, j \in \{1, \ldots, m\}, i < j$ many users prefer candidate c_i over candidate c_j.

Table 2.6 shows the Kemeny rankings for the Mechanical Turk users with respect of the set of all questions (left), the set of questions which were answered correctly (center), and the set of questions which were answered incorrectly (right). When the set of all questions is considered, results retrieved by queries clarified via *MedSyn* are preferred by Mechanical Turk users, followed by web pages retrieved by unclarified queries. If only the set of correctly answered questions is considered, two rankings achieve the same Kemeny score; in both cases, results retrieved by clarified queries are preferred (*Behavioral* and *MedSyn*). When only the set of incorrectly answered questions is considered, an identical ranking to the set of all queries is observed. This symmetry, while perhaps counterintuitive, is due to the fact that the results retrieved by the unclarified queries ("no clar.") are preferred more highly in those cases when a question is incorrectly answered; this preference skews the results when all questions are considered, thus causing the symmetric behavior observable in Table 2.6.

Table 2.6: The best synonym mappings as determined by the Kemeny-Young method.

All questions	**Correctly answered** (tie between two rankings)		Incorrectly answered
1st: *MedSyn*	**1st: *Behavioral***	**1st: *MedSyn***	1st: *MedSyn*
2nd: no clar.	**2nd: *MedSyn***	**2nd: *Behavioral***	2nd: no clar.
3rd: *DBpedia*	**3rd: no clar.**	**3rd: no clar.**	3rd: *DBpedia*
4th: *Behavioral*	**4th: *DBpedia***	**4th: *DBpedia***	4th: *Behavioral*

"no clar." represents the set of retrieved results by the original (unclarified) query. The leftmost column indicates that results retrieved by queries clarified with *MedSyn* were the preferred over all queries. However, when only considering those instances where questions were correctly answered, *Behavioral* was the preferred mapping, shortly followed by *MedSyn* (central columns). When only preferences associated with incorrectly answered queries (rightmost column), *MedSyn* is, once again, the preferred mapping.

Results retrieved by queries clarified through *MedSyn* are preferred more highly across all questions, regardless of whether questions were answered correctly or not. *Behavioral*, while being the preferred clarification mapping for correctly answered questions, ranks last when the set of all questions is considered. We hypothesize that such behavior is due to the skewness induced by the aforementioned preference expressed for unclarified queries. We observe that *Behavioral* does not exhibit such skewness with respect of the set of correctly answered questions; this could be caused by the fact that users seem to equally prefer queries clarified by *Behavioral* and *MedSyn*.

Is Query Clarification Beneficial?

While the Kemeny-Young method provides great insights about the preference expressed by participants towards results retrieved using clarified queries, its findings are insufficient to properly determine whether query clarification increased the understanding of health topics and which synonym mapping is the most appropriate for query clarification. In particular,

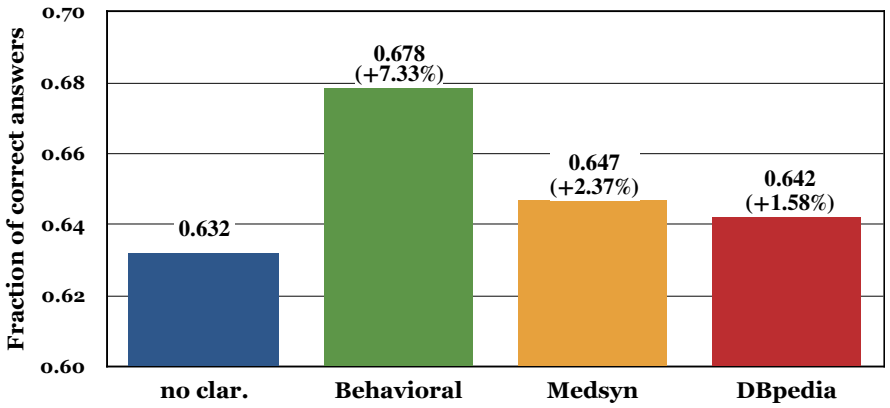

Figure 2.3: Average fraction of correct answers for each clarification candidate. For each candidate, the fraction is calculated over all the query/user combinations where the candidate is preferred. *Behavioral*, the method with the highest fraction of correct answers, improves over the baseline (no clarification, leftmost bar in blue) by 7.33% (statistically significant, Welch's t-test, $p < 0.05$).

the Kemeny ranking does not measure the difference between *MedSyn* and *Behavioral*, the two most preferred mappings for the set of correctly answered questions (Table 2.6, center). To quantify such difference, we calculate the average fraction of correct answers for each clarification candidate when the query clarified by such candidate is preferred (Figure 2.3).

Of the three synonym mappings presented in this dissertation, *Behavioral* resulted in the highest fraction of correct answers (0.678). In other words, when users express a preference for results retrieved by a query clarified with *Behavioral*, they were able to correctly answer the question associated with the query 68% of the time. This results represent an improvement of 4.63% over *MedSyn*, an improvement of 5.38% over *DBpedia*, and an improvement of 7.33% over no query clarification (statistically significant, Welch's t-test, $p < 0.05$). This suggests that *Behavioral* is to be considered the best-performing synonym mapping, since it both achieves the highest Kemeny ranking for correctly answered questions and yields the highest fraction of correct to incorrect question answers.

Table 2.7: Correct/incorrect number of answers when users clicked HON-certified websites.

	Certified by HON	Not Certified
Questions answered correctly	426	566
Questions answered incorrectly	158	270

These resources led to an 7.7% statistically significant increase (Fisher's exact test, $p < 0.05$) in correct answers.

The findings detailed in this subsubsection corroborate our observations regarding the Kemeny ranking: *MedSyn*, while being the most preferred synonyms mapping across all questions, is associated with a lower rate of correct answers, due to the strong preference expressed for it for the set of incorrectly answered questions. On the other hand, *Behavioral* achieves the highest fraction of correct answers; to the fact that it is one of the most preferred clarification mappings in the set of correctly answered questions, and the least preferred for the set of incorrectly answered questions.

Reliability of Results

The *Health On the Net Foundation* (HON)[15] is an organization that publishes a code of good conduct ("HONcode") for health-related online resources, issuing a certification for those websites that conform to it. The HONcode ensures that a website is reliable and useful in the medical information it provides. On average, M=3.43 interleaved results were certified by the HON foundation (SD=2.02, Mdn=3), while M=4.78 were not certified (SD=2.45, Mdn=4).

We studied the impact of HON-certified results on the fraction of correct answers given by Mechanical Turk workers. Table 2.7 shows the number of health-related questions answered

[15] `http://www.healthonnet.org/`

correctly and incorrectly when Mechanical Turk users clicked on and did not click on websites certified by HON. Users were 7.7% statistically more likely (significant at $p < 0.05$, Fisher's exact test) to answer the question correctly after visiting a website with HONcode certification. Such increase remains statistically significant ($p < 0.05$) when the performance of each user are normalized by the number of results visited. Therefore, we conclude that HON certified websites help laypeople answer medical questions, lending credence to the importance of such certification.

The majority (88%) of the clicks were on HON-certified websites returned by a clarified query, which again confirms the effectiveness of our system in promoting pages whose content was verified as reliable. Furthermore, the ratio of HON-certified vs. not certified websites remains constant at any rank position (Spearman's rank correlation coefficient $r_s = 0.921$, significant at $p < 0.01$), although the number of clicks exponentially decreased for lower ranked results. This bias toward higher ranked results is to be expected, as shown by previous research [70].

2.2.3.2 Users Analysis

As previously mentioned, the synonym mappings were tested on two groups of users: laypeople, recruited via Amazon Mechanical Turk, and freelance medical professionals, enrolled on Elance. Given the differences between the members of the two sets, we compare the two groups. Descriptive statistics are reported in Table 2.8, while the distributions of users are represented in Figure 2.4. All users answered questions better than would be expected by chance (i.e., 25% of the time). Furthermore, the vast majority ($> 95\%$) of users answered questions correctly over 50% of the time.

As shown in Table 2.8, the expert group correctly answered a higher number of questions (statistically significant, Welch's t-test, $p < 0.05$). Moreover, Experts were found to visit more web pages before answering to each question, which is consistent with the findings reported in previous studies [153]. Users in both groups were found to click on more results

Table 2.8: Overview of the differences between laypeople and experts.

	Laypeople	**Experts**
Number of survey participants	80	12
Fraction of correct answers Sig. difference between groups, $p < 0.05$	M=0.655, SD=0.135	M=0.723, SD=0.116
Average clicks per correct answer Sig. difference between groups, $p < 0.05$	M=1.94, SD=0.84	M=3.19, SD=1.42
Average clicks per wrong answer Sig. difference between groups, $p < 0.01$	M=1.60, SD=0.93	M=2.86, SD=1.23
Intra-agreement within groups (Fleiss' kappa)	0.4477	0.6528

The significance of differences between the two groups were measured using Welch's t-test (2-tailed).

before correctly answering a question, although the difference was not found to be significant (Welch's t-test, $p = 0.687$ for laypeople, $p = 0.556$ for experts).

We quantified the inter-agreement between the two sets of participants using Fleiss' kappa (Table 2.8). Experts were found to have a substantially higher agreement than laypeople. This observation, alongside the higher success rate, confirms the intuition that experts are more likely to correctly answer the proposed questions. This could be due to the fact that health professionals, thanks to their background, are able to successfully infer the necessary information from the retrieved results to satisfy their information need. We hypothesize that laypeople are instead more likely to randomly guess when they are presented with a difficult question, thus exhibiting both lower agreement and lower success rate.

For the laypeople group, we observed a moderate positive correlation between the average number of web pages visited and the fraction of correct answers (Spearman's correlation, $r_s = 0.228$, $p < 0.05$). In other words, those users who visited more web pages were more likely to correctly select the correct answer. For the expert group, we noticed a strong but

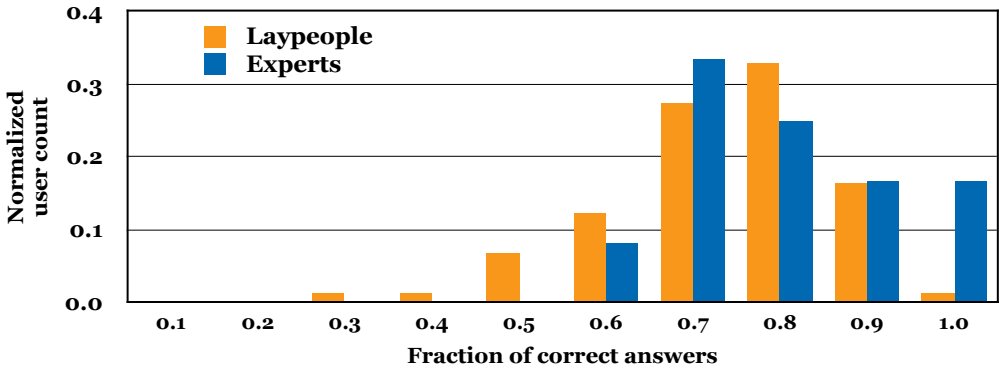

Figure 2.4: Distributions of the fraction of correct answers by laypeople (orange, N=80 M=0.655, SD=0.135) and experts (blue, N=12, M=0.723, SD=0.116).

not significant negative correlation between the average number of web pages visited and the fraction of correct answers (Spearman's correlation, $r_s = -0.558$, $p = 0.083$). This finding, while not conclusive, may suggest that more skilled experts—who have a higher success rate—may need to visit less web pages to correctly answer a question.

For both groups, a very strong correlation was found between the number of results clicked by a user before correctly answering a question and the number of results clicked before incorrectly answering a question (Spearman's correlation, $r_s = 0.780$ for experts, $r_s = 0.882$ for experts, $p < 0.01$ for both groups). This suggest that the number of visited results is unique to each user, and it is not influenced by the perceived difficulty of each question.

A fixed compensation was given to experts throughout the experiment; on the other end, the reward per task for laypeople increased over time to speed up data collection. To verify that higher compensation rates did not skew the performances of workers, we tested whether any relationship existed between retribution and fraction of questions correctly answered. However, no correlation was found between the two variables (Spearman's correlation, $r_s = 0.110$, $p = 0.405$).

Table 2.9: Percentage of queries where the baseline ("no clar.") is outperformed by each synonym mapping.

Synonym mapping	Percentage of queries in which baseline ("no clar.") is outperformed
Behavioral	66% (33 queries)
MedSyn	62% (31 queries)
DBpedia	50% (25 queries)
any synonym mapping	86% (43 queries)

Queries clarified using *Behavioral*—the best mapping—outperformed the unclarified query in 66% of the cases. The last row of the table contains the percentage of queries where any of the synonym mappings outperforms the baseline.

Finally, we note that unlike laypeople, experts seem to prefer the unclarified queries over the clarified ones. Nevertheless, the difference in success rate between the two is not significant (Welch's t-test, $p = 0.409$). We hypothesize that such findings could be explained by the fact that experts are more likely to effectively determine those documents that could satisfy their information need from the text snippet, thus not benefiting from query clarification. Such hypothesis would be consistent with previous studies investigating the relationship between domain knowledge and search results click-through events [28].

2.2.4 Learning to Select the Optimal Synonym Mapping

As shown in Section 2.2.3.1, query clarification increases the fraction of correctly answered questions. However, while all the mappings showed an overall improvement over the baseline, no single clarification technique consistently outperformed all others; moreover, for some queries, the unclarified query led to a higher success rate than any of the clarified queries. These observations are supported by the findings reported in Table 2.9. *Behavioral*, the best performing synonym mapping, improves over the baseline in 66% of the cases, while *MedSyn* and *DBpedia* outperform the baseline only in 62% and 50% of the cases, respectively.

Table 2.10: Features used as predictor variables for each logistic regression model M_k.

Features over query q_i and clarification candidate $\mathcal{C}_k$
Probability of bigrams and trigrams in q_i of appearing in Wikipedia
Probability of unigrams (stopwords excluded) in q_i of appearing in Wikipedia
Probability of bigrams and trigrams in q_i of appearing in health-related Wikipedia pages
Probability of unigrams (stopwords excluded) in q_i of appearing in health-related Wikipedia pages
Normalized longest common subsequence between clarified query $\mathcal{C}_k(q_i)$ and q_i
Presence of clarified query $\mathcal{C}_k(q_i)$ in any other clarification candidate $\mathcal{C}_h, h \neq k$ for query q_i

Features over query each web page p retrieved by clarified query $\mathcal{C}_k(q_i)$
Domain name of p (e.g., `nlm.nih.gov`)
Normalized longest common subsequence between page title of p and $\mathcal{C}_k(q_i)$
Normalized longest common subsequence between search result snippet of p and $\mathcal{C}_k(q_i)$
p is certified by HON

Finally, when considering any synonym mapping, we notice that, for 86% of the queries in the dataset, the baseline is outperformed; this implies that, for the remaining 14% of queries in our dataset, results retrieved by the unclarified query yield the highest rate of correctly answered questions. Motivated by these findings, we investigated whether the most appropriate mapping can be predicted to further increase the benefits of query clarification.

Previous work on query performance prediction [170, 19] has demonstrated that selective query expansion through a predictor achieves significant performance gains compared to either always expanding or always not expanding queries. In this section, we introduce a classifier that, given a query, either predicts which synonym mapping among *Behavioral*, *MedSyn*, and *DBpedia* should be used to clarify the query, or predicts to perform no clarification. For the reminder of this section, we will refer to the four possible outcome of the classifier as "clarification candidates".

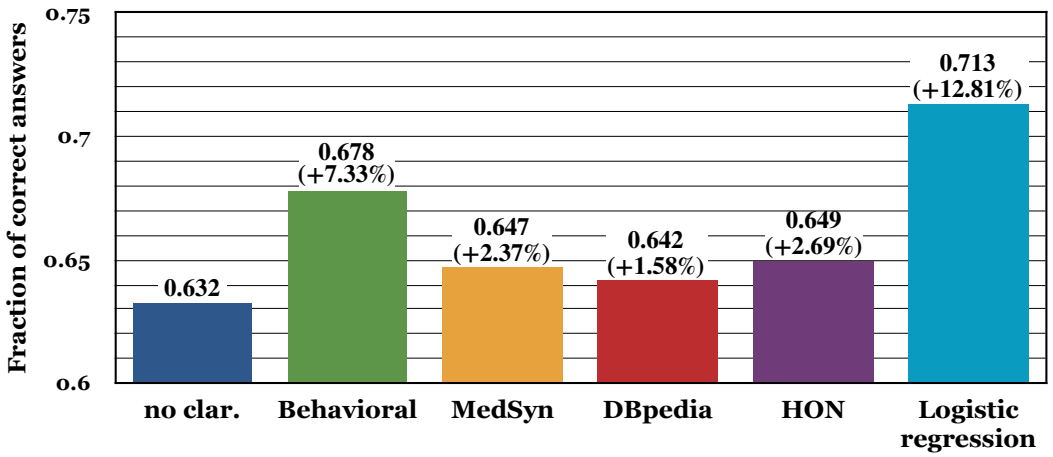

Figure 2.5: Average fraction of correct answers by laypeople. Six approaches are compared: unclarified query (no clar.), three synonym mappings (*Behavioral*, *MedSyn*, *DBpedia*), a baseline classifier trained on the number of HON-certified pages retrieved (HON) and the proposed classifier (Logistic regression). Logistic regression outperforms the baseline by 12.81% (statistically significant, Welch's t-test, $p < 0.05$).

The classifier was implemented as ensemble of four classifiers, one for each clarification candidate. In detail, four binary logistic regression models $\mathbb{M} = \{M_1, \ldots, M_4\}$ were trained as one-vs-the-rest classifiers: given a query q_i and its best clarification candidate $\mathcal{C}_k$, we trained model M_k with class label 1, and models $M_h \in \mathbb{M}, h \neq k$ with class label 0.

Two sets of features were used to train each model. The first one was defined over each query and each clarification candidate; it includes estimations of the likelihood of unigrams, bigrams, and trigrams in the query of appearing in any Wikipedia page, as well as their likelihood of appearing in health-related Wikipedia pages (as defined in Section 2.2.1.2). The longest common subsequence (LCS) between the clarified and unclarified (normalized by the length of the unclarified query) was also considered, as well as an indicator of the presence of the clarified query in any other clarification candidate. The second set of features was defined over each web page retrieved by a query q_i processed by a clarification candidate $\mathcal{C}_k$; in particular, we considered the domain name, LCS between the clarified query and the

page title, LCS between the clarified query and the search snipped of the page, and the presence of the page in the Health on Net database as predictor variables. The detailed list of features is presented in Table 2.10.

To determine the optimal clarification mapping for a query q_i, we used each model M_k to calculate an estimation $p_{i,k}$ of the likelihood of clarification candidate $\mathcal{C}_k$ of being the optimal mapping for q_i. For each q_i, the system chose as clarification mapping the one with the highest likelihood, i.e., $\text{argmax}_k(p_{i,k})$.

The system was implemented using the *Scikit-learn* Python package [107] and tested under ten-fold cross validation. The results are presented in Figure 2.5. We compared the performance of the logistic regression classifier with the results obtained by each individual synonym mapping. We also considered a simple multinomial logistic regression classifier trained on the fraction of retrieved results that are certified by HON as an additional baseline.

The logistic regression classifier performs well, improving over every individual synonym mapping. In detail, it achieves a 12.81% increase over the unclarified query, an 11.06% increase over *DBpedia* (Welch's t-test, $p < 0.05$), a 10.20% increase over *MedSyn* ($p < 0.05$) and a 5.16% increase over *Behavioral* ($p < 0.1$). Furthermore, it also outperforms (9.86% improvement, $p < 0.05$) the simple classifier trained on the number of HON-certified pages retrieved.

The positive results presented in this section confirm that query clarification can be further improved by selecting the most appropriate clarification candidate for each query.

2.3 Search Results Semantic Reranking

In this section, I propose and validate a novel set of syntactic and semantic features to be used in a learning to rank (LtR) framework that is aimed at capturing the semantic similarity between the information need expressed by a query and the content of relevant documents (Section 2.2.1). Then, in Section 2.3.2, the dataset used to evaluate the system is introduced. Finally, results are presented in Section 2.3.3.

Table 2.11: Features for each document.

Features group	Description	Note
STAT	Term frequency (*tf*) of query terms in document	†
	Inverse document frequency (*idf*) of query terms in document	†
	tf-idf score of query terms in the document	†
	Length of the body of the document	
	Length of document URL	
	Number of slashes in document URL	
	tf of terms in the document	†
	idf of terms in the document	†
	tf-idf of terms in the document	†
	idf of domain name	†
	Vector space model, BM25, language model with Dirichlet smoothing, language model with Jelinek-Mrcer smoothing scores	
	Spam scores	
ST-HEALTH	Document has *Health-on-Net* certification	
	tf of terms in the document that appear in health pages on Wikipedia	†
	idf of terms in the document that appear in health pages on Wikipedia	†
UMLS	Number of *n-grams* in the document matched to one or more UMLS concepts	*
	Number of UMLS concepts matched to one or more *n-grams* in the document	*
	Number of UMLS concepts matched to the query and the document	*
	Fraction of UMLS concepts matched for each semantic type	
	idf of UMLS concepts in health pages on Wikipedia	†
LSA	Euclidean distance between LSA representations of the query and the document	
	Euclidean distance between LSA representations of the query and the document, weighted by the odds of each term of appearing in health Wikipedia	
W2V	Euclidean distance between words embedding of the query and the document.	‡
	Euclidean distance between word embeddings of the query and the document, weighted by the odds of each term of appearing in health Wikipedia.	‡

For features marked with *, we use both the raw sum and the sum normalized by the document length as features. A † indicates a feature group where the sum, mean, variance, and median of the values were used as features. For features marked with ‡, we used word embeddings trained on PubMed [24] and Google News [91] to derive document and query representations.

2.3.1 Methodology

2.3.1.1 Features

We proposed in [128] a combination of statistical and semantic features to train a LtR model.

The feature set can be partitioned in five groups: An overview of all the features introduced

in this section is presented in Table 2.11. We remand the reader to Section 2.3.3.2 for an analysis of the impact of each set of features.

Statistical Features (STAT)

We considered a subset of features from the LETOR benchmark dataset, which have shown to be useful in many LtR systems [81]. These features encode statistical information about the terms in the query and documents (e.g., term frequency (*tf*), inverse document frequency (*idf*)). We excluded some features because they are not available for our dataset (e.g., HITS scores). We also excluded all features that relied on the titles of webpages, as they showed poor correlation with relevance judgments in our tests. In total, 36 features were extracted.

Statistical Health Features (ST-HEALTH)

We expanded the set of statistical features by including health-specific features. We consider whether a document is certified by the *Health on Net Foundation*[16], an organization that publishes a code of good conduct for health websites. Such signal has been shown to be a good indicator of informative web sites [130]. We also extracted *tf* and *idf* of all terms in the document that can be found in the subset of health-related pages in Wikipedia, which were extracted following as in [130]. The average, variance, mode, and sum of *tf* and *idf* were used as features. In total, 9 features belong to this group.

Unified Medical Language System Features (UMLS)

The Unified Medical Language System[17] (UMLS) is a medical ontology maintained by the U.S. National Library of Medicine. Terms in this ontology are organized by concepts, each of which is associated with one or more semantic type. Palotti, et al. [102] observed that more than 77% medical queries issued by laypeople contain medical concepts from UMLS. Therefore, we explored the use of medical concepts as semantic features to identify relevant

[16] `https://www.healthonnet.org/`
[17] `https://www.nlm.nih.gov/research/umls/`

search results. UMLS concepts are often present in queries issued by laypeople; thus, we explored their used as to identify relevant search results. To obtain the set of UMLS concepts in each document and in the query we used QuickUMLS a medical concept extraction system that we proposed in [126]. We match UMLS expressions belonging to 16 semantic types that are associated with symptoms, diagnostic tests, diagnoses, or treatments, as we previously suggested [126]. 26 UMLS features were extracted from each document and query.

Latent Semantic Analysis Features (LSA)

To extract semantic relationships between terms, we built a 100-dimension Latent Semantic Analysis (LSA) model using a collection of 9,379 entries from the A.D.A.M. Medical Encyclopedia[18] (a consumer-oriented medical encyclopedia) and the MedScape[19] reference guide. The model was used to obtain vector representations of terms in the query and documents, which were summed using two strategies: simple sum and sum weighted by the probability of each term appearing in the health section of Wikipedia. This composition technique, while simple, has been shown to be very effective [14]. To extract LSA features, we computed the euclidean distance between the vector representing the query and the vector for the document. We used the similarity scores from the weighted and unweighted models as features. We obtained two features using this approach.

Word Embeddings Features (W2V)

Similar to [26], we used word embeddings trained on PubMed[20] and Google News[21] to obtain dense vector representations for terms in the document and in the query. Word embeddings from the medical domain provide a strong representation for medical terms, while general domain word embeddings should capture the terms lay people are be more familiar with. Unlike the LSA model, which was trained on documents describing diseases, treatments, and tests, the PubMed model has a broader scope; thus, we used both to generate

[18] https://medlineplus.gov/encyclopedia.html
[19] http://reference.medscape.com/
[20] https://github.com/cambridgeltl/BioNLP-2016/
[21] https://code.google.com/archive/p/word2vec/

features. As in LSA, we used a sum and a weighted sum to compose the term vectors into the vector representation of the document or query. In total, 4 features were extracted: weighted and unweighted similarities between document and query using PubMed and Google News models.

2.3.1.2 Ranking Algorithms

LtR algorithms are typically partitioned in three groups: point-wise, pair-wise, and list-wise learners. point-wise algorithms are trained to predict the relevance of each example in the collection; pair-wise learners are trained to predict, for any two documents, which one should be ranked before the other. Finally, list-wise algorithms attempt at finding a permutation of the retrieved results such that the value of a loss function on the list of results is minimized. Because the features described in Section 2.3.1.1 do not preclude algorithms from any of the three groups to be used, we experimented with a set of algorithms that is representative of all three.

We considered the following LtR algorithms: logistic regression, random forests, LambdaMART [155], AdaRank [156], and ListNet [18]. Logistic regression and random forests are point-wise algorithms; we trained them to predict, for each document, its likelihood of being relevant. LambdaMART, a pair-wise learner, is an ensemble method that aims at minimizing the number of inversions in ranking. ListNet and AdaRank are list-wise learners that are designed to find a permutation of the retrieved results such that the value of a loss function on the list of results is minimized. Point-wise learners were implemented using the scikit-learn library[22] v.0.18; We used the implementation of LambdaMART, AdaRank, and ListNet available in RankLib[23] v.2.7 for the experiments described below.

Table 2.12: Six queries from the 2016 CLEF eHealth IR Task dataset from two distict topics.

Query ID	Topic ID	Query Text
...		
101004	1	inguinal hernia surgery or surgical complications
101005	1	inguinal hernia laparoscopic with mesh surgery reviews
101006	1	inguinal hernia surgery story, is it safe?
...		
103004	3	headaches caused by too much blood or high blood pressure
103005	3	headache that only goes away with blood loss
103006	3	strong headaches at base of skull, stops with blood donation
...		

Compared to the queries described in Section 2.2.2.1, these queries encode a much narrower information need.

2.3.2 Experimental Setup

2.3.2.1 Dataset

The proposed LtR approach to laypeople medical search was evaluated on the 2016 CLEF eHealth IR Task dataset [182]. The dataset consists of 300 queries modeled after 50 distinct topics. A sample of queries is shown in Table 2.12. The topics were created by health professional from forum posts from the *AskDocs* section of Reddit; well written posts containing demographic information and medical history of the authors and expressing a single information need were selected to generate plausible queries; 6 queries were created for each forum post. Results for the queries were retrieved from the ClueWeb12 category B dataset, a collection of 53 million web pages. In total, 25,000 documents were evaluated; to each one, a score between 0 and 2 was assigned. Because all queries created from the same forum post share the same information need, relevance judgments of queries on the same topic are

[22] `http://scikit-learn.org/stable/`
[23] `https://sourceforge.net/p/lemur/wiki/RankLib/`

identical. On average, 74.1 documents were deemed relevant for each query (min: 1; max: 335; median: 45; std.dev.: 74.7).

2.3.2.2 Experiments

Documents were indexed using the Terrier search engine, v. 4.0[24]. As a baseline, we consider the BM25 scoring function defined by the CLEF eHealth organizers in [182]. While simple, this baseline outperformed all 10 teams (29 runs) who participated in shared task[25]. We use this baseline to retrieve up to 1,000 documents per query to train the LtR methods. All learners were trained under five fold cross validation and manually tuned using a separate validation set. Pair-wise and list-wise learners were configured to optimize NDCG@10 on the validation set. To avoid overfitting, we carefully generated the training, validation, and test set so that all queries from the same group are part of the same split. Finally, P@10 and NDCG@10 were used to evaluate all the approaches, as users of online search engines are more likely to pay attention to the first page of retrieved results than the subsequent ones.

2.3.3 Results

In this section, we analyze the impact of different classification algorithms and features set on the outcome of the LtR task. Specifically, we compare the performance of different point-wise, pair-wise, and list-wise algorithms in Section 2.3.3.1; then, we study the impact of each class of features in Section 2.3.3.2; finally, we present a per-query analysis of the performance of the best algorithm in Section 2.3.3.3.

2.3.3.1 LtR Algorithms

We compare the LtR approaches from Section 2.3.1.2 with the baseline used in [182]. For all experiments, learners are trained on all the features described in Section 2.3.1.1.

[24] `http://terrier.org/`
[25] `https://goo.gl/6kpCFJ`

Table 2.13: Performance of LtR algorithms on the dataset.

Method	Type of approach	NDCG@10		P@10	
BM25 baseline [182]	*n/a*	0.241		0.291	
Random Forests	point-wise	0.249	(+3.3%)	0.293	(+0.6%)
Logistic Regression	point-wise	0.262*	(+8.7%)	0.317	(+8.9%)
LambdaMART [155]	pair-wise	**0.305***	**(+26.6%)**	**0.361***	**(+24.1%)**
AdaRank [156]	list-wise	0.239	(-0.8%)	0.292	(- 0.7%)
ListNet [18]	list-wise	0.267*	(+10.8%)	0.333*	(+ 14.4%)

Runs marked with * are significantly different from the baseline (Paired Student's t-test, Bonferroni-adjusted, $p < 0.01$).

Of all learners reported in Table 2.13, LambdaMART achieves the best performance (+26.6% NDCG@10, +24.1% P@10 over the baseline). This demonstrates that LtR can be successfully exploited to improve the access to relevant medical resources that satisfy the need of online health seekers. As expected, LambdaMART outperforms point-wise LtR approaches, as it is often the case [81]. LambdaMART also achieves better performance than the two list-wise methods, AdaRank and ListNet (difference is statistically significant for both). This is to be expected, as previous work found LambdaMART to be very competitive in LtR tasks on web results when optimizing for NDCG@10 [141].

2.3.3.2 Feature Analysis

The performance of the model trained on each set of features is presented in Table 2.14. We observe that the model trained only on the statistical features (STAT) obtains better performances than models trained on other sets of features. This is to be expected, as statistical features were modeled after the LETOR feature set, which has been shown to be very effective for LtR tasks [81]. The model trained solely on statistical health features (ST-HEALTH) ranks second, suggesting that the presence and frequency of health terms plays an important role in identifying relevant results. This intuition is reinforced by the findings

Table 2.14: Performance of LambdaMART trained on each set of features.

Features group	NDCG@10	P@10
BM25 baseline	0.241	0.291
STAT	0.274*	0.322*
ST-HEALTH	0.260	0.311
UMLS	0.253	0.307
W2V	0.160*	0.210*
LSA	0.121*	0.188*
All features	**0.305**	**0.361**

Runs marked with * are significantly different from the baseline (Paired Student's t-test, Bonferroni-adjusted, $p < 0.0083$).

shown in Table 2.15, where ST-HEALTH features are among the highest ranked in terms of importance.

The UMLS features set shows limited improvements over the BM25 baseline. However, based on their ranking in Table 2.15, we argue that they have an important role in model built using all features, as they capture information about symptoms and diseases mentioned in the queries.

Lastly, we note that neither word embedding similarity features (W2V) nor latent semantic analysis similarity features (LSA) features are enough to train an effective LtR model by themselves. This outcome could be due to the fact that these features sets, which contain just 4 and 2 features, do not encode enough information to train a comprehensive model. However, while W2V features improve the effectiveness of the model when combined with other features (Table 2.15), LSA features have less of an impact on the model built by LambdaMART. This might be due to the fact that the LSA model was trained using a set of 9,379 pages, which could be too small to properly capture the semantic similarity between the query and the retrieved documents. This might be due to the fact that the set of 9,379 pages the LSA model was trained on is too small to capture the semantic similarity

Table 2.15: Top 10 features ranked by weight.

Feature	Group	Weight
Avg. *idf* in health Wikipedia	ST-HEALTH	0.0995
# of matching UMLS concepts in document	UMLS	0.0776
Avg. *tf* in health Wikipedia	ST-HEALTH	0.0616
BM25 similarity score	STAT	0.0605
# concepts in "*Sign or Symptom*" UMLS semantic type	UMLS	0.0579
Similarity weighted word embeddings PubMed	W2V	0.0521
# concepts in "*Injury or Poisoning*" UMLS semantic type	UMLS	0.0418
LM similarity score	STAT	0.0408
Similarity weighted word embeddings Google News	W2V	0.0393
Spam scores	STAT	0.0335

The weight of each feature was computed by averaging their information gain and ℓ_2-normalized.

between queries and the retrieved documents. Conversely, similarity features derived by dense word representations are effective for this task as long as the model used to derive them is accurate.

2.3.3.3 Query Performance

In this section, we compare the per-query performance of the baseline with the best ranker from Table 2.13. Results are shown in Figure 2.6. Rather than reporting the individual NDCG@10 for each query, we average the results of all queries that belong to the same query group. This approach is motivated by the fact that all queries in the same group share the same information need (and document relevance judgments). Therefore, by averaging the performance of all queries in the same group, we can study whether the performance of the best ranker relative to the baseline is due to the information need associated with each query. To convince the reader that this representation is justified, the variance for each

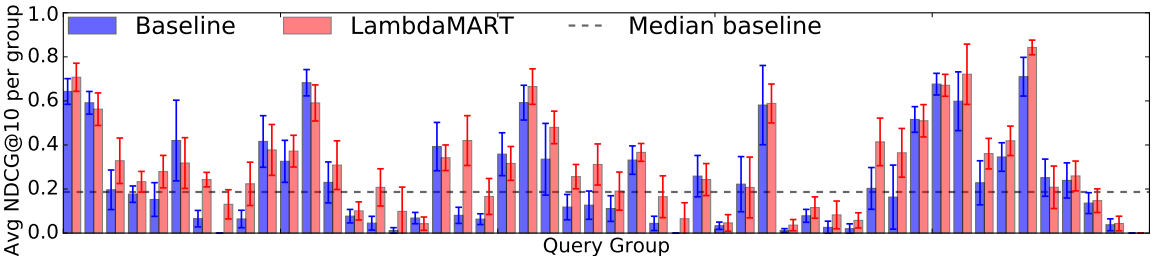

Figure 2.6: NDCG@10 of the baseline and LambdaMART. To increase the clarity of the figure, we averaged the value of NDCG@10 of all queries from the same query group (i.e., all queries sharing the same information need.)

query group is shown in Figure 2.6. As the variance for each topic is moderate, we conclude that our approach is appropriate.

The proposed ranker outperforms the baseline on 36 out of 50 topics. Interestingly, LambdaMART outperforms the baseline in all but one query whose NDCG@10 is below median. In other words, there exists a statistically significant correlation between the performance of the baseline on each query and the difference between the NDCG@10 of the baseline and LambdaMART (Spearman's rank correlation, $r_s = -0.38$, $p < 0.05$). This suggests that LtR is a viable strategy for addressing difficult queries; however, its performance are still bounded by the quality of results retrieved by the baseline.

2.4 Conclusions

Seeking information on medical topics is a common task for search engine users. Arguably, this information need also has one of the most important and immediate effects on the well-being of users. However, the technical nature of this information makes it inaccessible to many users, partly because of the jargon used by medical professionals. A significant effort has been made by providers of information in the medical domain to make their content

accessible to laypeople; yet, many users still struggle to retrieve reliable literature to their needs, due to a language gap.

This gap was quantified by measuring the ability of lay and experts users to complete a task-based retrieval task in which they were asked to answer health-related questions using search engine results. The evidence presented in this chapter suggests that the difference between the two group is significant. Yet, experiments showed that by clarifying queries submitted by non-experts to a major Internet search engine the likelihood that a user will answer health-related questions correctly increases significantly. Thus, our approach bridges the language gap between medical professionals and laypeople.

Three existing synonym mappings were used for query clarification; results show that all three are effective resources for such task. Of the three, queries clarified using *Behavioral* achieve the highest fraction of correctly answered questions (67.8% correctly answered questions, +7.3% over baseline); *MedSyn*, while being preferred by laypeople over no clarification, does not yield a significant improvement over the baseline (+2.3% over the baseline), partly due to the fact that it is strongly preferred in the case of incorrectly answered questions. Furthermore, I presented a supervised classifier that is able to select the most appropriate synonym mapping for query clarification. This classifier outperformed every individual synonym mapping, further validating the effectiveness of query clarification (71.3% correctly answered questions, +12.8% over baseline).

Finally, a learning to rank approach was proposed as another mean to close the language gap consumers suffer from. This approach uses a novel set of syntactic and semantic features to match the information need expressed by lay queries with the content of each document. The proposed approach led to a 26.2% increase in NDCG@10 over existing methods. The impact of several Learning to Rank algorithms was studied; furthermore, we discussed the effectiveness of our proposed features. This work demonstrates that semantic features can be effectively exploited by a LtR framework to improve laypeople health search.

Chapter 3

Medical Literature Retrieval for Health Experts

In Chapter 1, I briefly discussed how health professional struggle to search for medical literature while practicing. As many have argued, computational solutions are needed to handle the increase in volume of medical literature and adoption of electronic health records in the last few years. In particular, there is a growing interest in clinical decision support (CDS) systems that "provides timely information, usually at the point of care, to help inform decisions about a patient's care." [54]

In this chapter, I will explore a specific type of CDS system designed to retrieve medical literature in support of clinical practice. In detail, I will be focusing on the problem of using medical case reports, such as the one shown in Figure 3.1, to obtain high-quality publications to be used by health experts to diagnose or treat a patient; I will be referring to this task as *"CDS search"*.

As highlighted in Chapter 1, CDS search is an example of a complex search task in support of clinical practice: (*i*) there is a growing need for systems that could facilitate evidence-based medicine [29, 49]; (*ii*) healthcare professionals — especially those who practice — struggle to keep up-to-date with current advances in clinical research [15, 142], which, in turn, makes clinical errors more likely [32, 47]; (*iii*) search systems commonly used to retrieve medical literature are designed to handle short, keyword-heavy queries, thus achieving poor performance when clinical notes are used as queries [118].

Compared to other search settings, CDS search also presents unique challenges: (*i*) clinical case reports are substantially longer than queries in traditional search domains; (*ii*) unlike other search applications characterized by long queries (e.g., systematic review or

patent search), case reports often consist of multiple sentences and contain narrative elements, such as temporal dependencies between sentences and cause and effect relations. (*iii*) most importantly, CDS search highly favors precision over recall, since many healthcare professionals stated that they can only afford to spend limited time reading medical literature.

Biomedical literature retrieval has previously been studied in the TREC genomics track [60]. CDS search, while sharing some aspects with it — descriptive queries, domain specific lexicon — is not limited to the genomics domain, but spans across multiple fields in medicine. Consequently, CDS search systems must process a variety of literature styles written with a wide domain specific vocabulary. Therefore, it is necessary to re-evaluate the effect of known IR techniques for this domain.

In [128, 131, 132], I studied the impact of query expansion and reduction methods that take advantage of medical domain knowledge, as well as general purpose IR techniques. In particular, two novel methods for CDS search are introduced in Section 3.2. Both systems reformulate long, discursive queries by adding relevant terms to the information need expressed in the query. The first method expands the query by using pseudo relevance feedback; then it prunes the list of expansion candidates by removing those that are not medically related. The second method is a supervised approach to query expansion; it uses a multilayer neural network to predict, given a list of possible candidate terms, which terms to add to the original query to improve document retrieval. The two methods were tested on two datasets designed to evaluate CDS search (USMLE dataset [132], TREC CDS dataset [116, 118]), where they achieved comparable or better performance than state of the art systems, especially in precision-oriented metrics. This confirms that expanding medical queries with latent health concepts represents a viable strategy to improve medical literature retrieval systems.

Finally, I conclude this chapter by presenting a technique to reduce noise in clinical queries. Because the implementation of electronic health records varies from institution to

A 46-year-old woman presents with a 9 month history of weight loss (20 lb), sweating, insomnia and diarrhea. She reports to have been eating more than normal and that her heart sometimes races for no reason. On physical examination her hands are warm and sweaty, her pulse is irregular at 110bpm and there is hyperreflexia and mild exophthalmia.

Figure 3.1: An example of a query in the TREC dataset. (Query #6, TREC 2015)

institution, there is no consisted format in which clinical notes are written [66]. This results in clinical notes of varying quality, with little to no accompanying structured information [120]. As an example, many collections that have been released by institutions for research purposes make heavy use of abbreviation, are heavily comprised of partial sentences (e.g., missing subject or verb), and include unnecessary information about patients' medical history and stay [71, 121, 145, 146, 147]. This poses a challenge for CDS search system, as this noise hinders performance of systems that are designed and tuned for properly written clinical notes (such as the ones trained on TREC CDS 2014 and 2015 datasets.) Therefore, in Section 3.5, I introduce a system designed to reformulate noisy clinical notes; when evaluated on the 2016 TREC dataset [117], the proposed method achieves an improvement of 67% over the unmodified clinical note, and a 14% improvement over state of the art query reformulation methods.

3.1 Related Works

Search in the health domain has been a topic of interest for more than two decades. Over the years, many systems that rely on query reformulation have been proposed to improve retrieval in this domain. In this section, we present an overview of query reformulation techniques to expand and reduce queries using statistical techniques or domain-specific resources.

3.1.1 Domain-specific Query Expansion

Domain specific resources have long been used to perform query reformulation; in particular, they are commonly used to expand the query, in an effort to mitigate issues of polysemy and synonymity of medical expressions.

Early on, Srinivasan [135] introduced SMART, a retrieval system that uses the MeSH ontology[1]—a controlled vocabulary used by the US National Library of Medicine to tag and index articles in PubMed—to expand queries in the OHSUMED collection. Similarly, Hersh, Price, and Donohoe [59] expanded queries with terms manually selected from the UMLS, although experimental results showed that thesaurus based query expansion did not always improve search efficiency. More recently, Liu and Chu [82] also used UMLS to perform query expansion; their system automatically expands the query using scenario-specific terms (where a scenario could be "make a diagnosis" or "finding a treatment") and chooses the most appropriate UMLS terms for any given scenario. Dong, Srimani, and Wang [35] adapted PageRank to perform query expansion using the UMLS ontology. Specifically, terms in UMLS are used as nodes for the PageRank; relationships between concepts are used to determine popularity.

Query expansion through domain-specific resources has been found to be particularly effective for long queries. In the context of the TREC Genomics track, Hersh and Voorhees [60] noted that, the groups who used domain-specific query expansion (e.g., synonym based expansion) achieved the best performance (e.g., [16]).

For CDS search, Mourao, Martins, and Magalhaes [94] used MeSH terms to expand the query; the modified query was then used to retrieve and rank documents using multiple scoring functions (BM25L, BM25+ [85], tf-idf, language model with Dirichlet smoothing [175]). Finally, the rank of retrieved documents was determined by combining the ranks given by each scoring function using the Reciprocal Rank Fusion algorithm [30].

[1] `https://www.ncbi.nlm.nih.gov/mesh`

3.1.2 Statistical Query Expansion

Several statistical approaches to query reformulation for health search have been studied; overall, such approaches seems to be more effective in case of shorter queries, such as those in the OHSUMED collection. For example, Abdou and Savoy [2] introduced a variant of the Rocchio query expansion formula [119] for search in MEDLINE; their system improved up to 13.5% over SMART.

Many explored the use of statistical query expansion through pseudo relevance feedback for CDS. Choi and Choi [25] used titles, abstracts, and MeSH terms from the MEDLINE collection to obtain expansion terms for each query. Documents retrieved by the expanded query were then re-ranked using three classifiers trained to identify papers that matched the scenario. Xu, McNamee, and Oard [157] and McNamee [87] combined HAIRCUT [88], a character n-grams search engine, with pseudo relevance feedback. Their system achieved an increase in inferred Normalized Discounted Cumulative Gain (infNDCG) [164] when PRF is used over their non-expanded baseline.

Interestingly, some researchers have experimented with combining statistical evidence from multiple collections to select terms to expand a query. For example, Zhu et al. [179] explored the use of four auxiliary collections of clinical records, medical literature, and general domain web pages to build a mixture of relevance model for query expansion for improving clinical cases retrieval. Similarly, Oh and Jung [99] proposed a method that employs external collections to generate candidate terms to add to the queries. Documents retrieved from external collections are clustered; terms from each cluster are then employed to expand the query. The proposed method was tested on three collections: TREC CDS, OHSUMED, and CLEF eHealth; however, it achieved statistically significant improvement over a language model baseline in the first two cases (+10.32% and +12.33% respectively).

3.1.3 Hybrid Approaches

Approaches that combine statistical techniques with domain specific resources have also been proposed. Jalali and Borujerdi [65] proposed a method that incorporates medical concepts in the PRF process. In detail, MeSH terms are used in conjunction with query terms to rank MEDLINE documents. In the context of TREC Genomics, Stokes et al. [138] noted that "query expansion has a positive effect on genomic retrieval performance ... [but] expansion terms should be gleaned for manually-derived domain specific resources." Similarly, Lu, Kim, and Wilbur [83] and Matos et al. [86] proposed concepts-based query expansions systems. Limsopatham, Macdonald, and Ounis [80] uses a combination of medical concepts extracted from the top retrieved documents and concept relationship obtained from ontologies and external collections to expand the query. Balaneshin Kordan, Kotov, and Xisto [7] (and subsequent work [8]) used Markov Random Field Parameterized Query Expansion, a mixture model that weights terms based on whether they appeared in the query, in top retrieved documents, or in the UMLS ontology. Goodwin and Harabagiu [48] reformulated the problem of retrieving medical literature as a question answering problem; their system extracts questions from clinical notes, and retrieves answers using a probabilistic knowledge graph generated from a collection of electronic medical records.

3.1.4 Query Reduction

Query reduction algorithms have been extensively studied as a way to remove noisy terms from the original query. Their impact has mostly been tested in the web search domain.

For example, Kumaran and Carvalho [78] used SVM^{rank} Joachims [69] to find the best sub-query using a series of clarity predictors and similarity measures as features. Balasubramanian, Kumaran, and Carvalho [9] also studied how to improve performance by reducing queries using quality predictors; however, their system only removes up to one term from the query. This approach is not viable when dealing with long, descriptive case reports. Bendersky and Croft [11] used a supervised method for identifying key concepts in long

queries; in a subsequent work, they assigned different weights to concepts extracted from the query [13]. The framework introduced in the latter work inspired the system introduced by Balaneshin Kordan, Kotov, and Xisto [7] for CDS search. Luo et al. [84] has also adapted query reduction techniques in their MedSearch engine. The engine performs query reduction by filtering non-important terms based on their tf-idf score. Their system is designed for lay people performing health search on the Web and does not focus on medical literature retrieval. However, as shown by Balaneshin-kordan and Kotov [8] and Soldaini, Yates, and Goharian [127], existing query reduction algorithms are typically outperformed by ad-hoc methods for CDS search.

3.2 Methodology

As documented in the previous section, researchers have shown that query reformulation techniques are very effective at improving retrieval performance of CDS search systems.

Informed by such findings, we propose a three-stage approach to reformulate long, discursive queries. The first stage takes advantage of the PRF method introduced in [132] to generate term candidates (Section 3.2.1.) In the second stage, a subset of candidate terms are selected for query expansion. Two candidate selection methods are compared: the first is an improved version of health terms pseudo relevance feedback (*HTPRF*) [132]; the second is a supervised approach (Sections 3.2.2 and 3.2.3.) Finally, in the third stage, the query is expanded using the terms selected in the previous step; furthermore, we also experimented with statistical and syntactical query reduction methods to remove terms from the query that could cause query drift (Section 3.2.4.)

3.2.1 Candidates Generation

Candidate terms for query expansion are generated using the pseudo relevance feedback method introduced in [132]. For each query, the algorithm assigns a score s_j to each term t_j appearing in the k highest ranked documents.

In detail, the method works as follows: given a query Q and a document collection $\mathcal{D}$, it firstly retrieves and tokenizes k documents $\{d_1, \ldots, d_k\}$ from document collection $\mathcal{D}$; then, it builds the root set of query Q; that is, it generates the set $\mathcal{P}_Q$ of all terms appearing in any of the documents $\{d_i, \ldots, d_k\}$. Each term $t_j \in \mathcal{P}_Q$ is associated with a score s_j defined as follows:

$$\begin{aligned} s_j &= \log_{10}(10 + w_j) \\ w_j &= \alpha \cdot tf(t_j, Q) + \frac{\beta}{k} \sum_{i=1}^{k} tf(t_j, D_i) \cdot idf(t_j, \mathcal{D}) \end{aligned} \tag{3.1}$$

where $tf(t_j, Q)$ is the term frequency of term t_j in Q, $tf(t_j, D_i)$ is the term frequency of term t_j in document d_i, and $idf(t_j, \mathcal{D})$ is the inverse document frequency of the j-th term in the collection $\mathcal{D}$, as defined in [50, ch. 2]. α and β are smoothing factors; the value of w_j is increased by ten before calculating s_j to ensure that all scores are greater or equal to one.

In our implementation, the top 500 candidate terms ranked by s_j are considered for query expansion. This choice is due to efficiency reason and does not impact the performance of the system, as the final number of expansion terms is, in all experiments, an order of magnitude smaller.

In our experiments, we found the scoring method shown in Equation 3.1 is quite stable with respect to the choice of parameters α and β: increasing or decreasing either of the two parameters by up to an order of magnitude causes little variation in the performance of the system. Therefore we set $\alpha = 2.0$ and $\beta = 0.75$ as suggested in [27, 132]. On the other hand, the number of top documents k does affect the retrieval performance of the algorithm; therefore, we will discuss the tuning of this parameter in Section 3.4.5.

3.2.2 *HTPRF* Candidate Selection

Our unsupervised method uses the same candidate selection technique described in Section 2.2.1.2: given a list of candidate terms $\{t_1, \ldots, t_j\}$, a candidate term t_j is kept if and only if $\mathrm{OR}(t_j) \geqslant \delta$, where δ is a tuning parameter of our system. Of the remaining candidate

terms, the top m ranked by s_j are considered for query expansion. This approach has two advantages: (i) it removes terms that are either not-medically related, or very common in both domains, and (ii) it prevents clinical jargon, which is not typically used to describe medical conditions and treatment, from being included in the expansion terms.

As with k, the value of δ and m influence the performance of the retrieval algorithm; we analyze the effect of different values for δ and m in Section 3.4.5.

3.2.3 Deep Neural Network (*DNN*) Supervised Candidate Selection

We also approached query expansion as a supervised learning task where the goal is to predict which candidate terms should be used to expand the query.

It is generally challenging to optimize systems that are evaluated with respect to ranking metrics (such as mean average precision or discounted cumulative gain), as systems are designed to predict a score for documents while loss functions are defined in terms of ranking of documents [148]. For this reason, we decided to train our *DNN* to optimize a surrogate metric, which we call weighted relevance ratio (WRR). WRR is designed to capture the importance of a candidate terms extracted in the first stage for each query.

We used three groups of features to train our supervised model: word embedding representations of the query and terms, statistical features over multiple auxiliary collections, and other syntactical and semantic features. Word embeddings, a means of representing terms from a vocabulary into a dense, low-dimensionality space, were obtained using the `word2vec` model [90]. We detail the statistical features over external collections, as well as syntactical and semantic features in Section 3.2.3.1.; we will refer to them as "candidate features" (opposed to "candidate word embedding") throughout the rest of the manuscript.

The WRR of a candidate is defined as the ratio of its probability of appearing in a relevant document over its probability of appearing in the entire collection, weighted by its own frequency in the relevant documents. Similarly to [89], we found odds ratio to be a reliable indicator of importance in the relevant category. We scale the odds ratio of each

term to prevent extremely rare terms from having a very high WRR score, as we empirically noticed that such terms are often spelling errors or non-relevant terms. Formally, given a term t, a collection $\mathcal{D} = \{P_i\}_{i=1}^{i=|\mathcal{D}|}$ of documents, and the set $\mathcal{R}_Q$ of relevant documents for query Q, $\mathcal{R}_Q \subset \mathcal{D}$, we defined WRR as follows:

$$\text{WRR}(t) = \log_{10}\left(cf(t, \mathcal{R}_Q) + 1\right) \cdot \frac{\Pr\{t \in P_i \wedge P_i \in \mathcal{R}_Q\}}{\Pr\{t \in P_i \wedge P_i \in \mathcal{D}\}} \tag{3.2}$$

We note that we scale the collection frequency $cf(t, \mathcal{R}_Q)$ of term t in set of relevant documents $\mathcal{R}_Q$ by taking its log to prevent very frequent terms from having a high WRR. The two probabilities are estimated using MLE, i.e. by dividing the number of documents with term t by the total number of documents. We predict WRR using a regression with mean squared error (MSE) as the loss function.

Our neural network consists of two components: a component that learns query and term representations in order to compute the similarity between them, and a component that predicts the candidate term's WRR based on the term's similarity with the query and the candidate term's features.

This design is modeled after the neural network proposed in [122], which learns query and document representations in order to rerank pairs of short documents (i.e., pairs of sentences and pairs of tweets). Our model primarily differs in that we use a single dense layer to learn term representations, whereas Severyn and Moschitti [122] use a convolutional network to learn representations of the sequences of terms in two short documents. This change is due to that fact that, unlike their work, our system predicts the score of a single candidate rather than a passage.

The purpose of the second component of our neural network model is to combine the query-term similarity with additional features in order to make a WRR prediction. It consists of two layers: (*i*) a dense (i.e., fully connected) layer that takes the query-term similarity and candidate features as input (shown as *query-term similarity* and *features* in Figure 3.2) and filters them with a ReLU activation function [96], and (*ii*) a dense layer that takes the

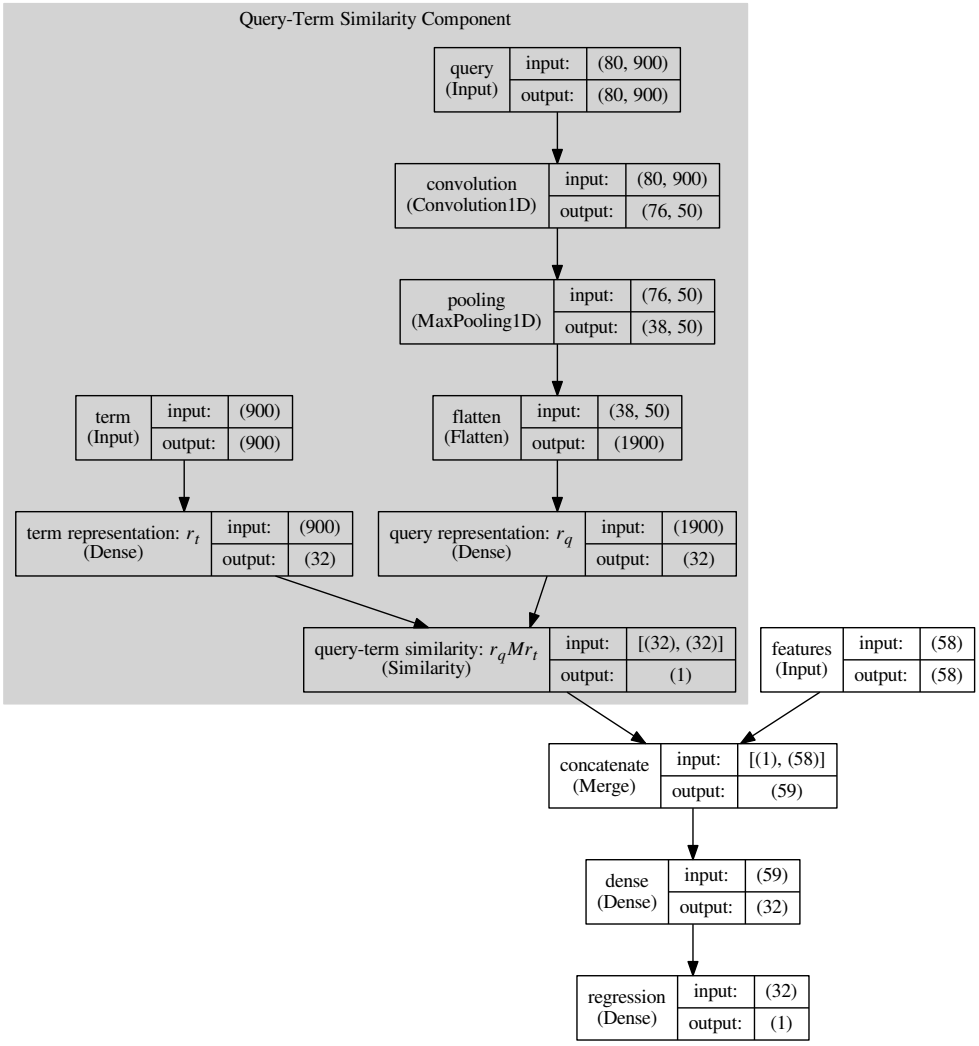

Figure 3.2: Overview of the Deep Neural Network (*DNN*) model. This supervised candidate selection model consists of a query-term similarity component and a features component. Each square represents a layer. Arrows indicate each layer's input. Layer types are shown in parentheses; *Flatten* and *Merge* layers modify their inputs' shape without modifying the input itself. Query-term similarity is computed by the *query-term similarity* layer (shaded in gray in the figure) as described in the query-term similarity section, combined with other features in the *concatenate* layer, and input to two dense layers (i.e., *dense* and *regression*) to perform the regression based on the query-term similarity and the term's features.

previous dense layer's output as input and predicts the term's WRR (i.e., *concatenate* and its inputs in Figure 3.2). A detailed overview of the feature set is provided in Section 3.2.3.2. Given a candidate term, query, and features, the neural network outputs the predicted term's predicted WRR.

We describe the training setup, as well as hyperparameters of the network in Section 3.2.3.3.

3.2.3.1 Query-Term Similarity

Our query-term similarity component learns compact query and term representations and computes their similarity with the help of a learned similarity matrix M. That is,

$$sim(r_q, r_t) = r_q M r_t \tag{3.3}$$

where r_q and r_t are compact query and term representations, respectively.

We learn the query representation by using a 1-dimensional convolution over a `word2vec` representation of the query, and applying $n_{filters}$ filters to the convolution followed by max pooling and a dense layer with a ReLU activation function and $n_{representation}$ neurons (i.e., *query representation* in Figure 3.2). That is, the convolution layer combines each w-term sliding window with $n_{filters}$ filters to produce $n_{filters}$ features for each sliding window, before using max pooling to take the top 50% of query term sliding windows and creating a compact representation of the query.

The convolutional layer's purpose is to apply position-independent filters to w-term windows of query terms. Without the convolution, the query representation would be dependent on the exact position in the query each term appears in. The dense layer's purpose is to learn to reduce the dimensionality of the query representation; the representation vector must be small both to generalize from training to testing and to match the dimensionality of the term representation vector.

Similarly, we learn the term representation by feeding a `word2vec` representation of the term to a single dense layer with a ReLU activation function and $n_{representation}$ neurons (i.e.,

term representation in Figure 3.2). As with the query representation dense layer, the term representation dense layer's purpose is to learn to reduce the dimensionality of the term representation.

The output of these steps is a query representation vector r_q and a term representation vector r_t with $n_{representation}$ dimensions. Finally, we compute the similarity between r_q and r_t as described above (i.e., using *query-term similarity* in Figure 3.2) and pass $sim(r_q, r_t)$ to the neural network's second component (i.e., *concatenate* in Figure 3.2).

3.2.3.2 Features

Recently, Oh and Jung [99] showed that taking advantage of multiple document collections leads to significant improvements in medical literature retrieval. Similarly, we consider several collections of health documents to capture medical soundness of candidate terms, as well as relationships between expansion candidates and query terms. The following collections were used to obtain features for candidate terms:

- **Khreshmoi project**[2] [51]: a collection of approximately 1.1 million web pages in the health domain. Pages in the collection were sampled from websites that have been certified by the HON foundation. Other known trustworthy websites were also included.
- **Health Wikipedia**: 22,943 Wikipedia pages from its Portal of Medicine[3]. This set of pages was extracted using the previously described information box heuristic.
- **Wikipedia**: a set of 5.9 million English Wikipedia pages collected on May 5, 2016. While pages in this collection are not necessarily from the medical domain, it should help discerning medical terminology from general domain terms.

[2] `http://www.khresmoi.eu/`
[3] `https://en.wikipedia.org/wiki/Portal:Medicine`

- **PubMed Central**: the open access subset of PubMed Central[4]. The snapshot we use — obtained on January 21, 2014 — is the same test collection used in the CDS track at TREC.

- **A.D.A.M. Medical Encyclopedia**: a consumer-oriented medical encyclopedia. We use the subset available through Medline Plus[5], which consists of 1,789 pages. This dataset was retrieved in May 2016.

- **MedScape**: a collection of 7,590 pages containing educational material (e.g., summaries of diseases, descriptions of symptoms, lists of drugs interactions, differential diagnosis sheets, etc.) for medical specialists, primary care physicians, and other health professionals. The collection was retrieved in June 2016.

For each collection $\mathcal{C}$ and each candidate term t, we consider the inverse document frequency (idf) of the term in the collection as a feature. Specifically, the following formulation of idf is used:

$$idf(t, \mathcal{C}) = \log_{10}\left(\frac{|\mathcal{C}| + 1}{df(t, \mathcal{C}) + 1}\right) \tag{3.4}$$

where $df(t, \mathcal{C})$ is the document frequency of term t in collection C, i.e., the number of documents in $\mathcal{C}$ that contain t.

To capture the semantic relationship between query terms and candidate terms, we extract, for each candidate term t, query term q, and collection $\mathcal{C}$, the number $N_{t,q,\mathcal{C}}$ of documents in which t and q co-occur; Then, for each t, we consider as feature the minimum, maximum, average, and standard deviation of $N_{t,q,\mathcal{C}}$ for all terms in the query.

Finally, similarly to [128], we also consider the following features for each candidate term:

- The PRF score of the term, as defined in Equation 3.1.

- The odds ratio of the term, as defined in Equation 2.1.

[4] `https://www.ncbi.nlm.nih.gov/pmc/`
[5] `https://medlineplus.gov/encyclopedia.html`

Table 3.1: Top 16 features ranked by the absolute value of their Spearman's rank correlation coefficient (ρ_s) with WRR.

Rank	Feature	ρ_s
1	*HTPRF* score	0.426
2	odds of being in health Wikipedia	0.134
3	term is a noun	0.095
4	term is a verb	-0.094
5	term is a UMLS concept	0.093
6	MedScape *co-occurrence st.dev.*	0.089
7	English Wikipedia *idf*	-0.083
8	MedScape *co-occurrence max.*	0.082
9	term is part of UMLS concept	-0.068
10	MedScape *co-occurrence avg.*	0.067
11	Khreshmoi *co-occurrence min.*	0.066
12	Khreshmoi *co-occurrence st.dev.*	0.064
13	Khreshmoi *co-occurrence max.*	0.061
14	Health Wikipedia *co-occurrence min.*	0.060
15	A.D.A.M. *co-occurrence st.dev.*	0.059
16	length of term	0.033

- The number of concepts in the UMLS metathesaurus that can be matched to the candidate term; QuickUMLS [126] was used to identify concepts.
- The number of concepts in UMLS that contain the candidate term; note that this differs from the previous features, as a term that is not a UMLS concept (e.g., "swine") can still appear as part of one ("african swine fever").
- The length in characters of the candidate terms.
- The Part of Speech (PoS) of the candidate term (e.g., the candidate term is a noun, verb, adjective, etc.).

In Table 3.1, we report the top 16 features, as determined by the absolute value of their Spearman's rank correlation coefficient (ρ_s) with WRR. We choose Spearman's rank correlation because the target value WRR—as well as many of the features—is not normally

distribuited (Shapiro-Wilk test, two-tailed, $p < 0.05$). All correlations reported in the table are statistically significant (two-tailed, $p < 0.05$).

We note the two top ranked features are the *HTPRF* score and the odds ratio of a term appearing in health Wikipedia, two features that are used by *HTPRF* to select terms for expansion. This implies that the improved *HTPRF* is a strong baseline for the supervised method. Interestingly, the rank correlation suggests that candidate terms that are nouns are more likely to appear in relevant search results ($\rho_s = 0.095$), while verbs are more likely to appear in non-relevant search results ($\rho_s = -0.094$). As expected, collections whose content is mainly health-related (MedScape, Khreshmoi, health Wikipedia, A.D.A.M.) all have positive correlation with WRR, while English Wikipedia — which includes pages over many domains — correlates negatively with WRR.

3.2.3.3 Implementation Details

For *DNN*, we train the neural network using the Adam algorithm [74] for up to 30 epochs. Training is stopped early if loss fails to decrease on the validation set; in practice this happens after approximately 15 epochs. Term `word2vec` representations [90] are obtained by concatenating 300-dimensional `word2vec` representations trained for 25 epochs on the PMC and Kreshmoi datasets described in the previous section. In the neural network's second component, we use a dense layer with 32 neurons. Our implementation of *DNN* method leverages Gensim[6] [113] and Theano[7] [114]. Furthermore, spaCy[8] [62] was used for PoS extraction.

3.2.4 Query Reformulation

Both *HTPRF* and *DNN* can be used to expand the preprocessed query. For the former, terms are ranked by their score; then, the top m candidate terms are used for expansion. As

[6] `https://radimrehurek.com/gensim/`
[7] `http://deeplearning.net/software/theano/`
[8] `https://spacy.io/`

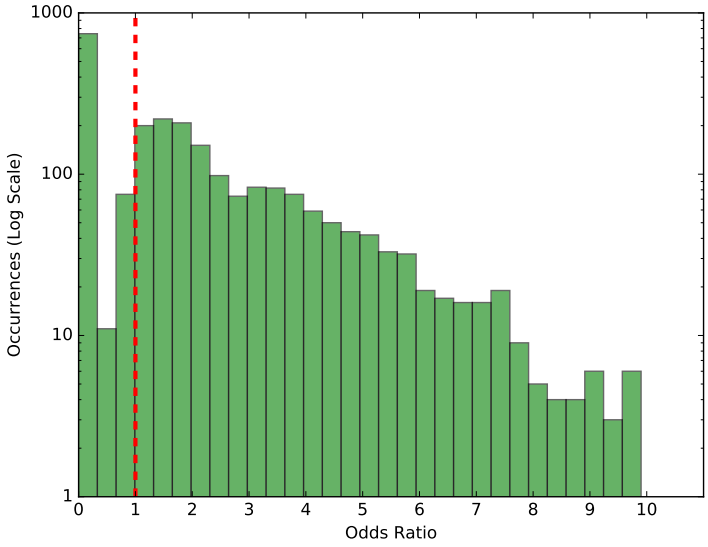

Figure 3.3: Distribution of the odds ratio of being relevant among terms in the query. Terms whose odds ratio is less than 1 (left of red dashed line) are more likely to appear in non-relevant documents than relevant documents. In our dataset, 832 query terms (34.6% of terms) have odds ratio less than 1.

expected, the value of m affects the performance of the algorithm, as we will later discuss. For *DNN*, the top 30 terms by predicted WRR are added to the query.

As previously mentioned, queries for this task are long and discursive. Through statistical analysis, we determined that some terms in the queries are less likely to appear in relevant documents than others; thus, we experimented with query reduction algorithms to improve retrieval performance. In detail, the distribution of the odds ratio of terms is shown in Figure 3.3. Of 2,403 terms across 60 queries, 35% of them have an odds ratio less than 1, meaning that they are more likely to appear in non-relevant documents than in relevant documents. It follows that an effective query reformulation strategy that removes most of such terms would improve retrieval performances. However, Soldaini et al. [132] have shown that query reduction techniques that rely on extraction of UMLS concepts do not improve the

performance of a CDS search system. Therefore, in this, work, we investigated whether part-of-speech (PoS) tags or syntactic dependencies could be used instead. We proceed as follows: first, we extract Part-of-Speech (PoS) tags and syntactic dependencies associated with the query. The two are coupled to identify all Noun Phrases (NP) in the query. The union of all noun phrases are considered as reformulated query. Furthermore, in [27], we suggested that Verb Phrases (VP) could have a significant impact in conveying the information need of each query. In this dissertation, we set to study this by considering a query reduction algorithm that keeps both VPs and NPs.

To summarize, the following types of queries are expanded using the candidate terms as determined by the *HTPRF* and *DNN*:

- Preprocessed query (stopwords, numbers, and units of measurement removed). We will refer to this method as "*stopwords removal*".
- Reduced preprocessed with terms t whose odds ratio of appearing in health Wikipedia is greater than or equal to δ (i.e., $\text{OR}(t) \geqslant \delta$). We will refer to this method as "*odds ratio reduction*".
- Reduced preprocessed query with only noun phrases. We will refer to this method as "*NP reduction*".
- Reduced preprocessed query with only noun phrases and verb phrases. We will refer to this method as "*NP+VP reduction*".

3.3 Experimental Setup

At the time we started investigating CDS search systems, no benchmark dataset containing case reports or medical publications could be used to evaluate our system. As previously noted, the information retrieval task introduced in CLEF eHealth Evaluation Lab [45, 46, 181, 182] is designed to evaluate search systems for consumers, not experts. OHSUMED [61] provides relevance annotations on medical literature, but its queries are considerably shorter

than a case report (6 vs 67.6 terms on average) and are keyword based. ImageCLEFmed [72] studied multimodal literature retrieval for clinical practice; that is, it promoted systems designed for combining information from images and textual descriptions to retrieved relevant clinical literature. Therefore, the system that we introduced in [132] was developed using an alternative experimental framework based on practice material for the United States Medical Licensing Examination (USMLE). Construction and characteristics of this dataset are detailed in Section 3.3.1. The dataset was also made publicly available[9] for other researchers to study.

In 2014, the Clinical Decision Support shared task was introduced at the Text REtrieval Conference (TREC) [118]. The shared task captured the interest of many research teams, and ran again the following years [116, 117]. Therefore, the two approaches introduced in Section 3.2 were also evaluated TREC CDS 2014 and 2015 dataset, which I discuss in Section 3.3.2.

Finally, I briefly discuss unsupervised and supervised baselines in Section 3.3.3. These methods are compared to the proposed systems in the results section 3.4.

3.3.1 Synthetic USMLE Dataset

At the time we started working on CDS search, the lack of datasets suitable to evaluated CDS search system required us to create our own. To create a benchmark for evaluation, we developed [132] an approach to automatically identify relevant documents to case reports by making use of external information about each case report (the correct diagnosis, treatment or test associated with each one as well as explanations about the correctness of such relations). Our dataset contains two components: medical papers and medical case reports. The medical literature was obtained from Open Access Subset of PubMed central[10], a free full-text archive of health journals (728,455 documents retrieved January 1, 2014).

[9] `https://github.com/Georgetown-IR-Lab/CDS-search-dataset`

[10] `http://www.ncbi.nlm.nih.gov/pmc/tools/openftlist`

88. A previously healthy 40-year-old man is brought to the emergency department because of constant substernal chest pain for 12 hours that is exacerbated by coughing and inspiration. The pain is relieved with sitting up and leaning forward. There is no family history of heart disease. His temperature is 38°C (100.4°F), pulse is 120/min, and blood pressure is 110/60 mm Hg. The lungs are clear to auscultation. Cardiac examination shows distant heart sounds. An ECG shows diffuse ST-segment elevation in all leads. An x-ray of the chest shows normal findings. The most likely cause of his condition is injury to which of the following tissues?

(A) Aortic intima
(B) Esophageal sphincter
(C) Myocardium
(D) Pericardium
(E) Pleura

Figure 3.4: Sample of case report for a USMLE Step 1 prep book exam.

495 medical case reports were obtained from three USMLE preparation books[11]. An example is shown in Figure 3.4. Each case report contains a description of a patient followed by a question asking for the correct diagnosis, treatment, or test that should be executed. Case reports from USMLE are modeled after real clinical situations with goal of assessing the ability of future physicians in applying clinical knowledge, concepts and principles for effective patient care[12].

Given a case report, our goal was to retrieve documents (medical publications) that can help a physician diagnose the patient, treat the patient's condition, or request a test relevant to the case; the content of three USMLE prep books were used to determine which documents in our collection were relevant. In detail, we took advantage of the multiple answer choices associated with the case reports as well as the explanation of why an answer is correct. To determine relevant documents for each case report, we separately issued as queries the explanation paragraph (q_{E}) and each answer choice individually ($q_{a_0}, \ldots, q_{a_3}$). Documents retrieved by the correct answer $q_{a_{\mathrm{corr}}}$ and q_{E} received a relevance score of two, while documents retrieved by q_{E} and any incorrect answer choice were given a score of one. By using this approach, we were able to take into account that not only the correct documents retrieved by

[11] `https://github.com/Georgetown-IR-Lab/CDS-search-dataset`
[12] Bulletin of Information, `http://www.usmle.org/pdfs/bulletin/2012bulletin.pdf`

querying the correct answer contribute to determine the right treatment/test/diagnosis, but also those related to the incorrect options. Any answer choice query ($q_{a_i}, \in \{0, \ldots, 3\}$) that contained more than 200 documents was discarded under the assumption that the query was too broad. A case report was discarded if its correct answer choice query was discarded. This process left us with 195 valid queries (i.e., case reports).

Three human assessors were then instructed to read each of these case reports and determine their validity. Specifically, they were asked to categorize each one as invalid or as asking for a diagnosis, treatment, or test. Invalid queries were those that were primarily quantitative (i.e., contained only numeric values about some tests or vital signs e.g. blood pressure, heart rate, body temperature, etc). The three assessors' inter-rater agreement was 0.56 as measured by Fleiss' kappa[13]. Any query deemed invalid by at least two assessors was discarded. This left us with 85 case reports; of those, 17 were reserved for parameters tuning, while the remaining 68 were used for testing.

We used Elasticsearch[14], a search server built on top of Lucene[15], to index the medical documents in our dataset and to retrieve results. The default tokenizer and the divergence from randomness retrieval model [5] were used.

3.3.2 TREC CDS Dataset

The effectiveness of the proposed methods was also studied on the datasets introduced in the CDS track at TREC 2014 [118] and TREC 2015 [116]. The two dataset share the same documents collection, but have different sets of test queries. A summary of the characteristics of the two datasets is provided in Table 3.2.

The document collection in the datasets is of a snapshot of the open access subset of PubMed Central (PMC), a database of biomedical literature available online free of charge.

[13]The moderate level of agreement between assessors is attributable to the hardness of the task. The evaluators reported that many reports laid in the spectrum between fully quantitative and fully qualitative, thus representing a noteworthy challenge.

[14]`https://www.elastic.co/products/elasticsearch`

[15]`https://lucene.apache.org`

Table 3.2: Statistics of datasets used in the 2014 and 2015 CDS track at TREC.

Dataset year	Documents				Queries		Qrels	
	number	*has title*	*has abstract*	*has body*	*number*	*average length*	*relevant*	*non relevant*
2014	733,138	100%	86%	88%	30	78.6	3,356	34,594
2015						83.3	4,990	32,818

The same documents collection was used both years. "Qrels" is set of documents whose relevancy has been assessed by TREC organizers.

The snapshot was defined by the organizers of the CDS track as the subset of all documents in PMC published before January 21, 2014. It contains 733,138 documents, totaling approximately 9.5 GB in size. Each article is in NXML format[16]. From each article, we extract the title, the abstract, and all sections in the body of the paper. Although all articles in PMC have a title, not all of them include a body or an abstract section. In the snapshot provided by the organizers of the CDS track, 14% of the articles have no abstract and 12% have no body. However, all articles have at least one of the two sections.

Gobeill et al. [43] computed the distribution of article types on the collection and on the set of relevant documents for the 2014 queries. Their work showed that 74.3% of articles in the collection are research articles (case reports: 4%; review articles: 6.9%; other: 15.8%). Similarly, 52.2% of relevant articles are research articles, 20.4% are case reports, and 17.9% are review articles. The remaining 9.5% belong to other categories.

Compared to our previous system [27, 132], citation markers were removed using regular expressions; furthermore, we also removed table and figure captions. This more thorough preprocessing step is partially to credit for the improvements over our previous results.

The queries in the CDS dataset were created by clinical informatics experts (all of whom were physicians) at the US National Library of Medicine. In the reminder of this section,

[16]NXML is a XML-compliant format whose tags are specified in the US National Library of Medicine's Journal Archiving and Interchange Tag Library. A full specification is available at the following location: `http://jats.nlm.nih.gov/archiving/versions.html`.

we provide a brief description of them; we remand the reader to [118] for more details. Each query is comprised of three sections: a title, a summary, and a description. The description field was created to resemble a typical sign-out note (that is, a clinical note containing a brief history of a patient) in use at many hospitals when a patient is transfered across departments. In the words of the CDS track organizers, this process was done to "replicate the types of information contained in EHR notes, thus providing as near as possible a realistic evaluation of how such a retrieval system would perform in a clinical environment."

The information need of each query falls into one of these three categories: make a diagnosis, determine a test to confirm a diagnosis, establish the most appropriate treatment after diagnosis. We will refer to this categories as "diagnoses", "treatments", and "tests" throughout the rest of the manuscript. The three categories were chosen because previous research has shown that questions regarding diagnoses, treatments, and tests account for a 58% of the clinical questions posed by primary care physicians [33]. For each query, a summary and title were also provided. However, none of the proposed methods consider them for retrieval, as the description field is a more accurate representation of the search task studied in this dissertation.

3.3.3 Baselines

3.3.3.1 Unsupervised Techniques

UMLS Concepts Selection (*MMselect*)

We extract concepts from queries based on concepts defined in the Unified Medical Language System[17] (UMLS) to perform query reduction. For this extraction we utilize MetaMap[18], a tool designed for UMLS concept extraction. We reformulated the query by removing all the terms that did not have a mapping to any UMLS concepts.

[17] `http://www.nlm.nih.gov/research/umls/`
[18] `http://metamap.nlm.nih.gov/`

UMLS Concepts Extraction (*MMexpand*)

Similar to *MMselect* method, this method identifies UMLS Metathesaurus concepts that exist in the query using MetaMap. However, rather than filtering out terms, this method expands the query using new terms associated with the concepts identified. After detecting the concepts in the query, expansion terms were chosen by querying UMLS for new terms that were synonyms of the concepts in the query and were marked as preferred terms by UMLS; the query was expanded with all these terms. Given the extensive coverage of UMLS, we limited concept expansion to concepts containing drugs, diseases, and findings to prevent query drift.

Health-related Terms Selection (*HT*)

In an effort to estimate the impact of each component of the *HTPRF* method, we tested the health term filter described in Section 2.2.1.2 (equation 2.1) as a query reduction technique. Parameter δ for this method were tuned independently from *HTPRF*.

Pseudo Relevance Feedback (*PRF*, SNUMedinfo, NovaSearch)

Similarly to *HT*, we also evaluated the impact of the *PRF* component of *HTPRF* and *DNN* separately. This method uses the candidates' scores described in Section 3.2.1; in particular, the score s_j computed for each term t_j using equation 3.1 is considered when selecting the top k terms for expansion. Like in the case of *HT*, parameters k and m for this method are tuned independently from *HTPRF*.

Beside our original formulation of *PRF*, we also compared our approach with the query expansion methods introduced by Choi and Choi [25] and Mourao, Martins, and Magalhaes [94]. These two approaches, previously described in the related works section (Section 3.1), are two top-performing query reformulation methods on the TREC CDS 2014 dataset [118].

3.3.3.2 Supervised Techniques

Query Quality Predictors for Query Reduction (QQP)

We implemented the learning to rank framework described by Kumaran and Carvalho [78]. This method uses quality predictors as features to rank sub-queries of the original query using SVM$^{\text{rank}}$ [69]. The following predictors are considered as features:

Mutual information Each sub-query is represented as a fully connected weighted graph, where each vertex represents a term in the sub-query. Edges are weighted by mutual information as follows:

$$\text{MI}(t_i, t_j) = \log \frac{\frac{\text{co}(t_i,t_j)}{T}}{\frac{\text{n}(t_i)}{T} \cdot \frac{\text{n}(t_j)}{T}} \tag{3.5}$$

Where $\text{co}(t_i, t_j)$ is the number of times terms t_i and t_j appear within 100 tokens in any document in the collection, $\text{n}(t)$ is the number of times terms t appears in the collection, and T is the size of the collection. For each graph, the heaviest spanning tree is extracted; the average weight of the edge is used as query predictor.

Query clarity Estimation of the divergence of the query model from the collection model using the top 500 documents retrieved per sub-query.

$$\text{QC} = \sum_{t \in Q} \Pr(t|Q) \cdot \log_2 \frac{\Pr(t|Q)}{\Pr_{\text{coll}}(t)} \tag{3.6}$$

Where $\Pr t|Q$ is the probability of token t of occurring in the query model and $\Pr_{\text{coll}}(t)$ is the probability of t of appearing in the collection.

Simplified clarity score Simplified version of clarity score that estimates the probability of a term in the language model by considering the likelihood of it appearing in the query.

Query scope Measure of the size of the retrieved set of documents relative to the size of the collection.

$$\mathrm{QS} = -\log(\frac{n_q}{T}) \tag{3.7}$$

Where n_q is number of documents in the collection containing at least one query term. Sub-queries showing high query scope are expected to perform poorly since they contain terms that are too broad.

Similarity to original query *Tf-idf* similarity is considered as one of the quality predictors under the hypothesis that the closer a sub-query is to the original query, the less likely it is to cause intent drift.

In addition to the previously listed features, *QQP* considers, for each sub-query, statistical measures[19] over the term frequency, document frequency and collection frequency of the terms in the sub-query as features for $\mathrm{SVM}^{\mathrm{rank}}$. The length of each sub-query is also considered as a feature. Interested readers should consult the original paper [78] for more details.

Since most of the query predictors are query dependent, they cannot be computed ahead of time, thus slowing the sub-query selection process. Therefore, as suggested by the authors, we implemented a set of heuristics to reduce the number of candidate sub-queries, which, prior to pruning, is exponential to the size of the original query: (*i*) select queries with length between three and six terms; (*ii*) select only the top twenty five sub-queries ranked by MI; (*iii*) select only the sub-queries containing name entities. The parameters for $\mathrm{SVM}^{\mathrm{rank}}$ were set as suggested in [78].

Fast Query Quality Predictors (*Fast QQP*)

Since *QQP* was not designed specifically for CDS search, its performance is negatively affected by the greatly reduced length of the generated sub-queries and by the lack of domain-specific features. Because of the unique formulation of case reports, we implemented

[19] Maximum and minimum value; arithmetic, harmonic, and geometric mean; standard deviation and coefficient of variation.

a set of sub-query candidates pruning heuristics that resulted in statistically significant improvements over the original formulation while reducing the processing time.

First, we increased the maximum length M_{subq} of a sub-query candidate from 6 to 16 terms (empirically determined). This is motivated by the fact that case reports are, on average, much longer than the queries in [78] (16.2 vs. 67.6 terms). The minimum length of a sub-query was not altered (i.e., $m_{\text{sub-q}} = 3$).

As the size of the candidates set grows exponentially when the maximum number of tokens increases linearly, *Fast QQP* prunes the list of candidates after each increase in length of candidate sub-queries. In other words, for each $i \in \{m_{\text{subq}}, \ldots, M_{\text{subq}}\}$, the set of candidates C_i is ranked by MI; the top-k sub-queries are then extracted (set $C_{i,k}$) and used to build the set C_{i+1} accordingly with the following formula:

$$C_{i+1} = \{s_l \cup \{q_h\} \mid s_l \in C_{i+1} \wedge q_h \in Q\} \cup C_{i,k} \tag{3.8}$$

where Q is the original query. After empirical evaluation, we set $k = 50$.

We further improved *Fast QQP* by including some domain-specific features:

- number of UMLS concepts in the candidate sub-query;
- semantic type of the UMLS concepts in the candidate sub-query;
- statistical features over the likelihood of each term in the candidate sub-query of being health related, as estimated by equation (2.1);
- number of MeSH terms in the candidate sub-query.

WSU-IR

We also compare our work with the model introduced by Balaneshin Kordan, Kotov, and Xisto [7]. This system leverages a Markov Random Field model to parameterize query expansion. Briefly, a mixture model is used to estimate importance weights of expansion terms with respect to the primary metric of the task (in this case, inferred nDCG). Features that

Table 3.3: Performance of baselines and proposed methods on the USMLE dataset.

	nDCG		Recall		P@5	
baseline	0.2855	–	0.2741	–	0.1824	–
MMselect$^{\circ}$	$0.1622^{\triangledown}$	(−43.2%)	$0.1486^{\triangledown}$	(−45.8%)	$0.1059^{\triangledown}$	(−41.9%)
MMexpand$^{\bullet}$	0.3020	(+5.8%)	0.2958	(+7.9%)	0.1676	(−8.1%)
QQP$^{\circ}$	$0.2557^{\triangledown}$	(−10.4%)	0.2494	(−9.0%)	$0.1118^{\triangledown}$	(−38.7%)
Fast QQP$^{\circ}$	$0.3177^{\triangle}$	(+11.3%)	$0.3129^{\triangle}$	(+14.2%)	$0.1471^{\triangledown}$	(−19.4%)
HT$^{\circ}$	$0.3328^{\triangle}$	(+16.5%)	$0.3262^{\triangle}$	(+19.0%)	0.1882	(+3.2%)
PRF$^{\bullet}$	$0.3390^{\triangle}$	(+16.5%)	$0.3263^{\triangle}$	(+19.0%)	0.1765	(−3.4%)
HTPRF$^{\bullet}$	**0.3768** $\triangle$	(**+32.0**%)	**0.3520** $\triangle$	(**+28.9**%)	**0.2382** $\triangle$	(**+30.5**%)

The symbol $\circ$ indicates query reduction methods, while $\bullet$ indicates query expansion methods. A $\triangle/\triangledown$ indicate a significant improvement/worsening (Student t-test, Bonferroni-adjusted, $p < 0.0071$) over the baseline.

are used to weight terms are based on whether they appeared in the query, in top retrieved documents, or in the UMLS ontology. This method achieved state of the art performance at TREC 2015 [116].

3.4 Results

In detail we first compare the unsupervised method with several baseline on the work introduced in [132]. Then, we compare both methods with the state of the art; we follow by studying the effect of query reduction techniques when combined with *HTPRF* and *DNN* query expansion methods; furthermore, we analyze the impact of individual features on the performance of the *DNN* method; finally, we detail the process of tuning parameters for *HTPRF* and *DNN* on the TREC datasets.

3.4.1 Comparison of Reformulation Methods on USMLE Dataset

As previously mentioned, CDS search is a precision oriented task; it is meant to support healthcare professionals who are looking for findings that could help them determine the next action in the care of a patient. For this reason, performance at the first ten points

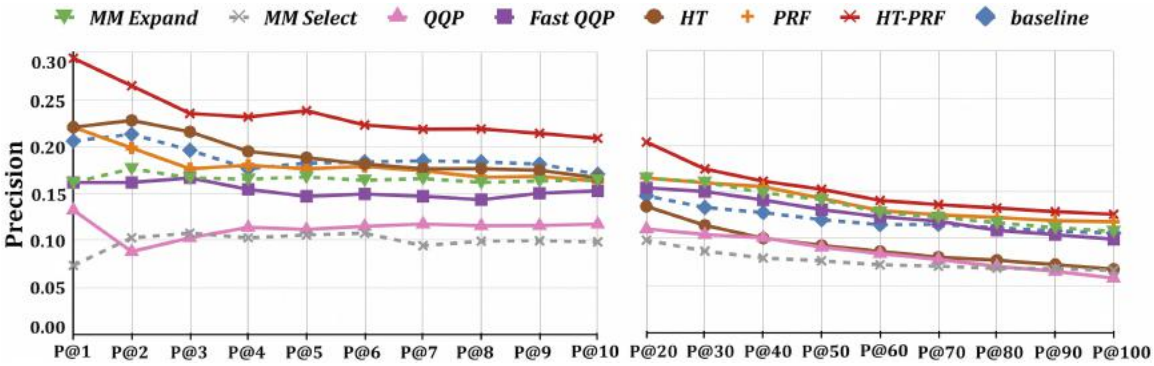

Figure 3.5: Points of precision for each method. The best performing method, *HTPRF*, achieves a 43% increase over the baseline for P@1.

of precision (Fig. 3.5) is key to assert the quality of a reformulation method. We focus our analysis on precision at five documents retrieved (P@5), as the performance of each method is consistent throughout the first ten points (Fig. 3.5, left) of precision and show no significant difference up to P@100 (Fig. 3.5, right). Recall and nDCG are also reported (Table 3.3); these metrics, albeit less key to the task, are still useful indicators to assert the overall quality of each method. We use a paired Student's t-test to measure whether the difference between any two methods is statistically significant ($p < 0.01$).

MMselect performed significantly worse than the baseline. We attribute such difference to the fact that, while it successfully identifies most medical concepts in the query, it often discards terms that have a key role connecting domain specific expression. *MMexpand* showed a minor but significant gain in terms of nDCG and recall over the baseline, but it performed worse (although not significantly) than the baseline in terms of P@5. We attribute the modest difference to the limited coverage of the portion of the synonym map in UMLS *MMexpand* uses with respect to the size of our dataset. This trade-off was necessary to prevent query drift.

QQP performed very poorly. Its limited performance is due to its aggressive reduction algorithm, which reduces the original query to at most six terms. As result, the reduced query loses most of the information content of the case report.

Fast QQP showed substantially better nDCG and recall results, but fell short in terms of P@5. We attribute the improvement to the fact that the inclusion of domain specific features and a more conservative approach lead to a more effective reduction. On the other hand, the worsening in terms of P@5 is likely due to the insufficient coverage of medical terms in the query: in medical literature, the same concept is often expressed using different terms and expression; thus a method that only performs reduction is likely to miss documents that are relevant to the case report, but differ from it in terms of vocabulary.

Both *HT* and *PRF* methods showed a statistically significant improvement over the baseline in terms of nDCG and recall; *HT* removes common non-health-related terms, whereas *PRF* reweights the entire query, increasing the importance of health-related terms, which naturally have a high IDFQE coefficient given the domain of the dataset. In *HT* some improvement is expected, as it keeps more generalized medical concepts in comparison with the UMLS concept selection method. Neither *HT* nor *PRF* showed significant improvement in terms of P@5. *HT* is likely to suffer from the same limitation in terms of vocabulary coverage *Fast QQP* has, while *PRF* is partially affected by query drift.

We achieved the most noteworthy results by using the *HTPRF*. The nDCG and recall values shown in Table 1 are statistically significant not only with respect to the baseline but also over simple *PRF* and *HT*. Moreover, *HTPRF* consistently improves over the baseline for each precision level shown in Fig. 2 ($p < 0.01$). The substantial increase in performances of *HTPRF* is due to the fact that it combines two very effective techniques: by expanding the query using the most relevant document, it is able to broad its vocabulary; on the other side, filtering the list of candidate terms for expansion prevents query drifting.

Table 3.4: Comparison of the proposed systems (last two rows) with a baseline method and the state of the art.

System	2014 dataset		2015 dataset	
	infNDCG	**P@10**	**infNDCG**	**P@10**
Baseline case report used as query	0.1546 -84.7%	0.2500 -56.0%	0.1729 -70.0%	0.3133 -54.2%
HTPRF baseline (tuned on USMLE dataset) [131]	0.2272 -24.7%	0.3200 -21.9%	0.2296 -28.0%	0.3367 -43.5%
SNUMedinfo [25]	0.2674 -5.9%	0.3633 -7.3%	n/a	n/a
NovaSearch [94]	0.2631 -7.7%	**0.3900**	0.2242 -31.1%	0.3567 -35.5%
WSU-IR [7]	n/a	n/a	**0.2939**	0.4667 -3.6%
HTPRF	*0.2567* *-10.3%*	*0.3733* *-4.5%*	*0.2653* *-10.8%*	***0.4833***
DNN	***0.2833***	*0.3600* *-8.3%*	*0.2744* *-7.7%*	*0.4300* *-12.4%*

For each column, the best result is in **bold**.

3.4.2 Comparison with State of the Art Systems

In Table 3.4, we report the performance of our methods on the 2014 and 2015 TREC CDS datasets. As previously mentioned, we compare the proposed approaches with the best approaches for the task, as well as with our previously proposed method. We also include a baseline system that uses the case report as query (no expansion; stopwords, numbers, and units of measurement removed). This baseline represents an important point of comparison with the two methods introduced in this dissertation, since it is used as a first step to retrieve the top documents used to generate candidate terms for query expansion. We note that some results are missing due to the fact that some of the teams have not participated in both years.

Table 3.5: Example of terms added to the query shown in Figure 3.1 by the *HTPRF* (left) and *DNN* (right) methods.

HTPRF	***DNN***
anorexia **autonomic case diagnosis disorder distress episodes excessive** fatigue **gastrointestinal medication** nervosa **nocturnal onset patient psychiatric report restless severe signs sleep** symptoms syndrome tachycardia thyroid thyrotoxic thyrotoxicosis **treatment tremor**	anorexia **antithyroid emptying** fatigue **gastric graves hyperthyroidism hypoglycemia insomniacs meal methimazole milnacipran** nervosa **prandial propranolol remission remittent reuptake sertraline** symptoms syndrome tachycardia thyroid **thyroiditis** thyrotoxic thyrotoxicosis **triazolam**

Terms in bold are exclusive to a method. For this query, *HTPRF* achieves higher P@10 (0.6 vs 0.3), but *DNN* achieves better infNDCG (0.419 vs 0.2506).

Both systems proposed in this dissertation fare well against the state of the art. On the 2014 dataset, the *DNN* expansion approach outperforms any other method in terms of inferred NDCG, while NovaSearch achieves a better precision at 10. This behavior is expected, as NovaSearch uses a formulation of PRF in which expansion terms are chosen among high tf-idf terms in few top-ranked documents; this implicitly optimizes for precision at top ranked results. On the other hand, our *DNN* method is trained to choose terms based on WRR, which does not take into account their tf-idf score. On the 2015 dataset, the *DNN* method underperforms the state of the art, as well as the other method proposed in this dissertation, when measured by precision at 10.

The improved *HTPRF* method is also very competitive with respect to state of the art methods. The run reported in Table 3.4 uses odds ratio on Wikipedia to reduce the query before expanding it; a more detail analysis of query reduction is provided in a later section. Overall, we notice that, unlike the *DNN* expansion technique, *HTPRF* favors precision at 10 over inferred NDCG. This could be a desirable characteristic of this method in those situations were obtaining a small set of highly relevant literature is preferred. On the 2014 dataset, *HTPRF* achieves a precision at 10 comparable to NovaSearch; on the 2015 dataset, it

outperforms the state of the art, although WSU-IR achieves better infNDCG. We explain the substantial improvement in performance of *HTPRF* by observing that the baseline method — which is used to obtain the top k documents from which expansion terms are extracted — is also much more effective on the 2015 dataset, especially in terms of precision at top ranked results. This causes *HTPRF* to select more relevant terms from the top documents, which explains the increase in performance.

When comparing *HTPRF* with the *DNN* method, a few interesting observations can be made. First, we note that, for both methods, precision at 10 results and inferred NDCG strongly correlate (Pearson'r, $\rho = 0.7612$ for *HTPRF*, $\rho = 0.7885$ for supervised query expansion, $p < 0.05$ for both).

However, as shown in Figure 3.6, the relative performance of two methods vary depending on the query. In 25 out of 60 queries *HTPRF* outperforms the *DNN* method; the opposite occurs in the remaining 35 queries. On average, the *DNN* method outperforms *HTPRF* on diagnosis and tests, while the opposite happens for treatments. However, the difference is not statistically significant (Student t-test , two-tailed, $p = 0.83$, $p = 0.87$, and $p = 0.77$ respectively). Thus, we cannot conclude that the difference in infNDCG between the two methods is due to type of information need associated with the query.

Finally, we point out that the *DNN* method is more likely to choose UMLS concepts as expansion terms; on average, 82.3% of expansion terms selected by the *DNN* method are UMLS concepts, while only 72.5% of terms chosen by *HTPRF* are present in the metathesaurus (difference is statistically significant, Student t-test, two-tailed, $p < 0.05$). Using the semantic type associated with each concept and the taxonomy introduced in [80], we were able to determine the aspects of the medical decision that the concepts chosen by the two methods belong to. For *HTPRF*, 18.5% of the terms are a diagnostic procedure or test (*DNN*: 19.3%), 17.1% are diseases (*DNN*: 19.7%), 32.5% are symptoms (*DNN*: 26.4%), and 20.3% are treatments (*DNN*: 23.4%). The remaining (11.6% and 11.2%) refer to other semantic types.

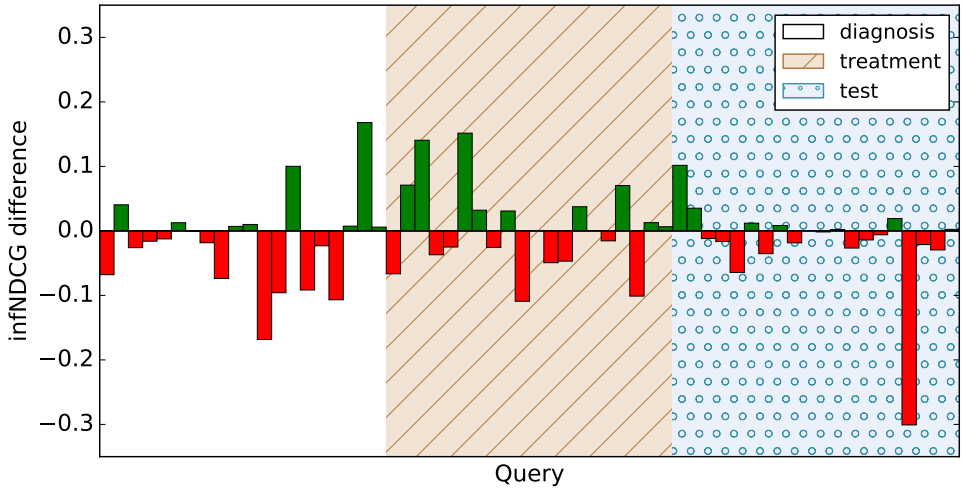

Figure 3.6: Difference in infNDCG between *HTPRF* and the *DNN* method for each query. Negative values are associated with queries in which the *DNN* method outperforms *HTPRF*.

3.4.3 Impact of Query Reduction

Previous work [27, 132] has suggested that query reduction could improve retrieval performance; therefore, we studied the impact of several query reduction techniques on the performance of both methods introduced in this dissertation. As previously mentioned, we set out to evaluate three query reduction techniques, and compared them with the original query (with stopwords, numbers, and units of measurement removed). Results for both methods are shown in Tables 3.6 and 3.7.

As shown in Table 3.6, removing terms that are less likely to appear in medical pages on Wikipedia is an effective strategy when combined with *HTPRF*, However, this technique is equally effective when combined with the *DNN* expansion method. We hypothesize that this is due to the fact that Equation 3.1 is likely to assign higher scores to terms that are semantically close to those in the query; thus, by removing less medically sound terms from the original query, we achieve an improvement in inferred NDCG. Conversely, the

Table 3.6: Comparison of several query reduction techniques on the improved *HTPRF* method.

System	2014 dataset		2015 dataset	
	infNDCG	**P@10**	**infNDCG**	**P@10**
improved *HTPRF* *stopword removal*	0.2541	0.3567	**0.2703**	0.4800
improved *HTPRF* *odds ratio reduction*	**0.2567**	**0.3733**	0.2653	**0.4833**
improved *HTPRF* *NP reduction*	0.2523	0.3633	0.2634	0.4367
improved *HTPRF* *NP+VP reduction*	0.2512	0.3533†	0.2621	0.4433*

Query reduction using odds ratio achieves the best results except for a modest decrease in infNDCG on the 2015 dataset. However, the difference between runs is not statistically significant (Student t-test, two tailed, $p \geqslant 0.05$).

DNN expansion method selects more diverse terms, thus increasing the need of keeping less medically sound terms in the query. This is evidenced by the fact that average distance between UMLS concepts in the query and UMLS concepts in the candidate terms selected by *HTPRF* is 3.25 nodes in the UMLS graph, while the average distance for terms selected by the *DNN* method is 5.68 (difference is statistically significant, Student t-test, two-tailed, $p < 0.05$.)

Furthermore, we notice that, for the *DNN* expansion method, there exists a trade-off between infNDCG and P@10 when more aggressive query reduction algorithms are used (Table 3.6 and 3.7). "*NP reduction*" and "*NP+VP reduction*", which shorten the query substantially, cause an increase in inferred NDCG, but negatively affect precision at 10 retrieved results.

Overall, we note that query reduction techniques show limited improvement over the original method. As evidenced in Table 3.6 and 3.7, none of the reduction methods show statistically significant improvements in terms of infNDCG over simple stopwords removal

Table 3.7: Comparison of several query reduction techniques on the *DNN* expansion method.

System	2014 dataset		2015 dataset	
	infNDCG	**P@10**	**infNDCG**	**P@10**
DNN expansion *stopwords removal*	0.2833	0.3600	0.2729	**0.4300**
DNN expansion *odds ratio reduction*	0.2842	**0.3700**	0.2698	0.4167
DNN expansion *NP reduction*	0.2865	0.3500	**0.2744**	0.4133
DNN expansion *NP+VP reduction*	**0.2919**	0.3400	0.2695	0.4267

NP reduction achieves the best infNDCG on the 2014 dataset, *NP+VP reduction* on the 2015 dataset, but both perform poorly in terms of P@10. Overall, the difference between runs is not statistically significant (Student t-test, two tailed, $p \geqslant 0.05$).

(Student t-test, two-tailed, $p \geqslant 0.05$). The biggest improvements are with respect to P@10; when used with *HTPRF*, *odds ratio reduction*; however, it is less effective when paired with the supervised expansion method (*DNN*). When compared with the baseline, *odds ratio reduction* shows the best improvements on the 2014 dataset, while it performs similarly or worse on the 2015 dataset.

3.4.4 Impact of *DNN* Method Features

In this section we study the impact of different feature types on our *DNN* expansion method. To do so we individually evaluate (*i*) the query-term similarity component and feature component of our model, (*ii*) classes of features, and (*iii*) features derived from specific collections. The results are shown in Table 3.8. None of the changes in infNDCG and P@10 are statistically different from the model's performance with *DNN expansion: query-term similarity & features* (Student t-test, two-tailed, $p \geqslant 0.05$). This can be attributed to the low number of queries in our datasets. While excluding features can cause the average

Table 3.8: Impact of model components, feature groups, and document collections on the *DNN* model's performance.

System	2014 dataset		2015 dataset	
	infNDCG	**P@10**	**infNDCG**	**P@10**
DNN expansion *both (query-term sim. & features)*	0.2833	0.3600	0.2744	0.4300
DNN expansion *query-term similarity only*	0.2501	0.3100	0.2785	0.4200
DNN expansion *features only*	0.2726	0.3467	0.2714	0.4167
DNN expansion *both excluding IDF features*	0.2766	0.3033	0.2808	0.4400
DNN expansion *both excluding co-occurr. features*	0.2640	0.3600	0.2727	0.4233
DNN expansion *both excluding UMLS features*	0.2709	0.3633	0.2665	0.4167
DNN expansion *both excluding PRF features*	0.2545	0.3567	0.2785	0.4233
DNN expansion *both excluding odds ratio feature*	0.2762	0.3500	0.2761	0.4300
DNN expansion *both using only Wikipedia features*	0.2631	0.3567	0.2748	0.4100
DNN expansion *both using only A.D.A.M. features*	0.2606	0.3433	0.2767	0.4233
DNN expansion *both using only PubMed features*	0.2517	0.3567	0.2854	0.4233
DNN expansion *both using only MedScape features*	0.2627	0.3433	0.2691	0.4167

infNDCG and P@10 to change substantially, this change in the average metric is caused by substantial changes to a small number of queries. Over all the runs shown in Table 3.8, no more than 9 queries per run ever experience a change in infNDCG or P@10 greater than 0.1. The average number of queries experiencing such a change is much smaller; 3 queries for infNDCG and 7 queries for P@10 on the 2014 dataset, and 1 query for infNDCG and 4 queries for P@10 on the 2015 dataset. These values are much smaller than the number of queries for which P@10 changes in Tables 3.6 and 3.7: all runs that show a statistically significant difference experience a change in at least 13 out of 30 queries. We attribute this difference to the fact that query reduction methods potentially modify the entire expanded

query, while the process of tuning the feature set for the supervised method only affects which new query expansion terms are added to the initial query.

The model's performance using both components, only the query-term similarity component, and only the feature component are shown in the first three rows, respectively. While the 2015 infNDCGs are similar regardless of which components are used, using only the query-term similarity component substantially harms infNDCG and P@10 on the 2014 data set. The model performs better on the 2014 data when using only the feature component, but both components are necessary to achieve the best results.

The model's performance when different classes of features are excluded is shown in the next five rows of Table 3.8. The biggest change in performance as measured by infNDCG occurs when the UMLS features are excluded, causing the 2014 infNDCG to decrease from 0.2833 to 0.2709 and the 2015 infNDCG to decrease from 0.2744 to 0.2665. Excluding co-occurrence features and excluding PRF features both cause substantial decreases in performance on the 2014 data, but do not substantially affect the results on the 2015 dataset. Similarly, excluding the IDF features and excluding the odds ratio feature cause smaller decreases on the 2014 infNDCG, but slightly increase the 2015 infNDCG. We conclude that the UMLS features have the most impact on our model's performance, followed by the co-occurrence and PRF features.

Table 3.8's final four rows show the impact on performance when only features from specific collections are used (i.e., the co-occurrence and IDF features derived from a given collection). PubMed features perform the worst in terms of 2014 infNDCG, but perform the best in terms of 2015 infNDCG. The other three collections perform similarly, with MedScape performing slightly worse on 2015 infNDCG but not on 2014 infNDCG. This suggests that, when they are used independently, these three collections are somewhat interchangeable for the purpose of deriving co-occurrence and IDF features. The results improve when all the collections are used, however, suggesting that they are also complementary and it is beneficial to use multiple collections.

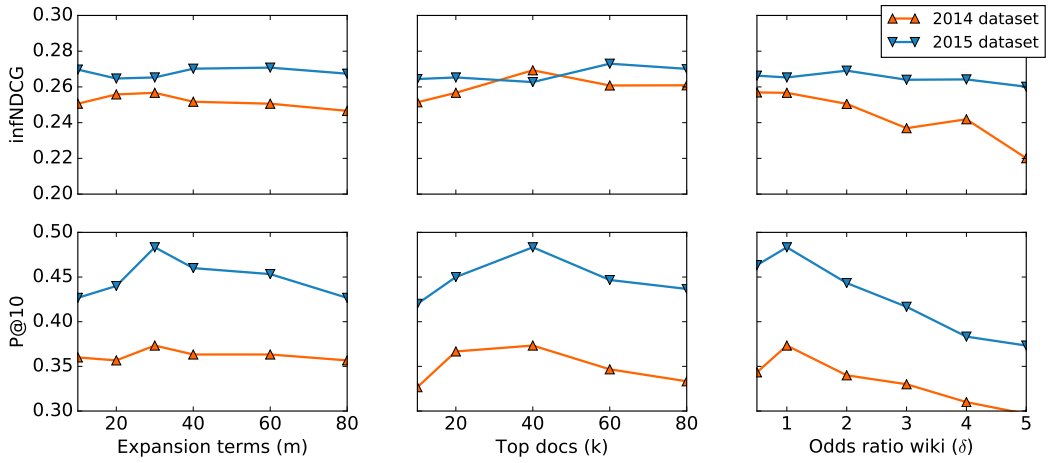

Figure 3.7: Effects of number of expansion terms (m, left), top documents (k, center), and minimum odds ratio (δ, right) on the performance of *HTPRF*, as measured by infNCCG (top) and P@10 (bottom.) We chose $m = 30$, $k = 40$, $\delta = 1$.

3.4.5 Parameter Tuning

In this section we describe the tuning process we followed for our methods. In the case of *HTPRF* we chose the number of expansion terms m, the number of top documents k, and the minimum odds ratio δ for *HTPRF*. Our goal was to choose parameters that would maximize infNDCG across both datasets. The results of our optimization phase are shown in Figure 3.7. Alongside the effect of each parameter on infNDCG, we also present their effect on P@10.

We observed that *HTPRF* is moderately stable with respect to the choice of its parameters: even when varying m or k by two orders of magnitude, infNDCG was affected by at most 15%. However, *HTPRF* behaved differently between the two datasets. On the 2014 dataset, a smaller number of expansion terms and top documents achieves the best performances, while larger values of m and k were necessary to achieve better infNDCG on the 2015 dataset. Since queries in the two datasets are of similar length and structure, we

suspect that the differences in size of the pool of relevant documents (as shown in Table 3.2) might explain the different behavior: fewer relevant documents exists for the queries in the 2014 dataset. Thus, large values of k and m may cause query drift. Conversely, larger values of k and m are appropriate for the 2015 dataset, as more presumably relevant documents are considered to choose expansion candidates. Ultimately, because infNDCG is less sensitive to changes in k and m than P@10, we choose the values of k and m that maximize P@10; that is, we set $m = 30$ and $k = 40$. We stress that we did not intentionally choose the same parameters for the 2014 and 2015 datasets; rather, because of the heuristic described above, the two parameters set happen to be the same.

Contrary to k and m, the behavior of δ was consistent across the two dataset. Large values of δ caused too few terms to be selected for expansion, thus reducing the performance of *HTPRF*. Unlike infNDCG, P@10 behaves similarly across the two datasets when tuning parameters are varied.

To achieve a good balance between the two datasets, we chose the tuning parameters for our dataset by performing ten fold cross validation on the 2014 and 2015 datasets combined. In seven out of ten folds, parameters $m = 30$, $k = 40$, $\delta = 1$ maximized infNDCG; therefore, we chose such combination for all experiments reported in this section.

The *DNN*'s parameters include the number of expansion terms m, the convolution size, the number of filters $n_{filters}$, and the term and query representation size $n_{representation}$. We found $m = 30$ terms to perform best on the 2014 dataset in terms of infNDCG. On the 2015 dataset varying the number of terms between 5, 10, 20, and 30 changed the average infNDCG by less than 1%. We thus used $m = 30$ terms in all experiments.

We empirically chose a convolution size of 5 (i.e., we consider 5 query terms at a time) with $n_{filters} = 50$ and $n_{representation} = 32$. Substantially increasing the number of filters (i.e., by more than 15%), the size of $n_{representation}$, or the dense layer harms performance by causing the neural network to overfit quickly, whereas substantially decreasing them reduces the network's ability to fit the training data and also harms performance. While there are

Original Note	Clean note
Mr. [[PATIENT]] is an 80yo M with dementia, CAD status post CABG in [[DATE]] (LIMA-LAD, SVG to OM2, SVG to RPDA), then status post CABG redo in [[DATE]], then status post 2 caths this year with patent LIMA, totally occluded SVG to RPDA, SVG to OM2, status post BMS to LCX on [[DATE]] who presented to [[HOSPITAL]] Hospital with increasing chest pain and nausea over the past few days. Per report, patient has presented several times since last cathed for recurrent angina. Admitted to [[HOSPITAL]] on [[DATE]] with recurrent chest pain. Ruled out for MI. Last episode of chest pressure was the morning of transfer, associated with dry heaves and belching relieved with morphine. Patient was continued on ASA, Plavix, Statin, BBker, Imdur and placed on Heparin gtt. Cath last [[DATE]] here at [[HOSPITAL]] showed a patent BMS in LCX and no new lesions. According to the family he usually has angina once every day or two, but for the past 2 weeks he has been having angina with any minimal exertion (eg putting on his shirt), and waking him several times per night.	A 80 yo male with demantia and past medical history of coronary artery bypass graft surgery presented with increasing chest pain and nausea over the past few days. The patient has history of repeated episodes of recurrent chest pain with relief with morphine. Patient is on Aspirin, Statins, Imdur, and Heparin. According to the family, the patient has increasing episodes of chest pain with minimal exertion in the last two weeks.

Figure 3.8: An example of noisy clinical note from the 2016 TREC CDS dataset (left, red), and a "clean" version of the same note created by NLM residents at the U.S. National Institute of Health (right, blue.) Text in `monospace font` represent information that has been anonymized by TREC organizers.

many candidate terms to use as training data, the number of training queries is a limiting factor; additional training queries would likely allow these parameters to be increased.

3.5 Clinical Decision Support with Noisy Queries

While CDS TREC 2014 and 2015 relied on fictional clinical descriptions created by health experts, the TREC 2016 dataset [117] provided real clinical notes as search topics alongside "clean" clinical descriptions. Compared with fictitious clinical descriptions, raw clinical notes present additional challenges for existing CDS systems, due to "terse language and heavy use of abbreviations and clinical jargon" [117]. An example of such raw clinical notes, as well as its clean counterpart, is shown in Figure 3.8.

In this section, we argue that query reduction techniques that address such challenges ought to be studied, as they improve CDS search by enabling the use of real clinical notes as queries. In particular, we propose a convolutional neural model that is able to predict, for each term in the clinical note, its importance in relevant documents. To do so, it employs several convolutional filters to learn local interactions between terms appearing in clinical notes. Predicted importance is then used to weight terms at retrieval time. Our approach is explained in detail in Section 3.5.1; then, in Section 3.5.2, we highlight the main differences between the experimental setup for noisy clinical notes and the one described in Section 3.3; finally, we evaluate our system on the TREC CDS dataset in Section 3.5.3.

3.5.1 Methodology

Similar to the work of Kumaran and Carvalho [78] and Bendersky, Metzler, and Croft [12], the approach proposed in this dissertation is designed to predict, for each term in a clinical note, a coefficient that encodes its importance. However, unlike these approaches, we do not use heuristics to select informative query terms, nor we rely on feature engineering to train our supervised method; rather, we use a convolutional neural network (CNN) to directly estimate the importance of each query term by learning from terms in its proximity. Our approach is described in Section 3.5.1.1.

We experimented with two different training strategies for our model. The first one mimics the training strategy used for *DNN* (Section 3.2.3) in that it minimizes, for each term in the training queries, the error between the importance weight predicted by the CNN and its WRR value. The second one is most similar to supervised pairwise learning to rank algorithms: given a query, a relevant document for the query, and a non-relevant document for the query, we first use the CNN to determine the weights of terms in the query; then, using these weights, we derive the scores of the two documents; finally, we backpropagate a positive loss if the non-relevant document is scored higher than the relevant document. A more detailed description of the learning strategy is provided in Section 3.5.1.2.

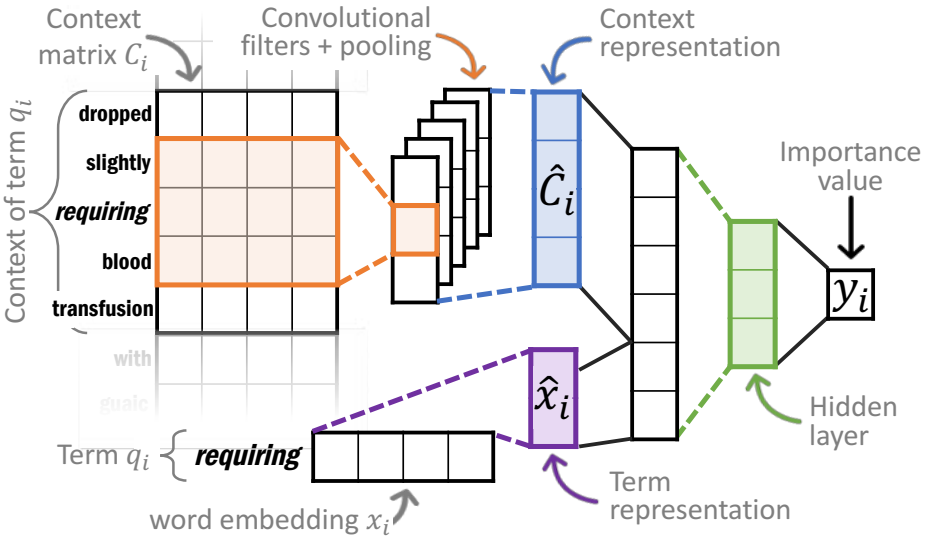

Figure 3.9: Diagram of the proposed convolutional neural model (CNN). The term being evaluated is *"requiring"*, while the context is *"dropped slightly requiring blood transfusion"*. Dotted lines represent transformations for which batch normalization [64] and dropout [136] are used.

3.5.1.1 Neural model topology

As previously mentioned, we used a CNN to capture local interactions between terms in clinical notes. On a high level, our system includes several convolution filters of different sizes to exploit interactions between terms in the proximity of each query term; the output of the filters is then reduced to a dense vector, which we refer to as *context representation* $\hat{C}_i$. The context representation of each term is then concatenated with a *term representation vector* $\hat{x}_i$ and used to derive the importance value y_i for each term in the query. A visual overview of the system is presented in Figure 3.9.

Term representation $\hat{x}_i$

For each query $\tilde{\mathbf{q}} = \{q_1, .., q_n\}$, we first obtain its dense representation $\mathbf{x} = \{x_1, .., x_n\}$. Two source of evidence were used to obtain, for each term q_i, its word embedding x_i: GloVe

vectors [108] pre-trained on the common crawl corpus[20] and SkipGram vectors pre-trained on PubMed[21]. We found that concatenating domain-specific with domain-agnostic embeddings yielded the best results; this is consistent with findings in other neural clinical applications [115]. We preserved the case of terms when obtaining word embeddings: this ensures that medical abbreviations, which are often capitalized, are properly captured. In order to reduce the dimensionality, the system learns a task dependent representation of the term feature x_i through a dense layer with ReLU activation function, which we denote as $\hat{x}_i$.

Context representation $\hat{C}_i$

For each term q_i in query $\vec{q}$, we define the context of q_i as the c terms preceding q_i and the c terms following it. In other words, the context of q_i consists of the terms appearing in a window of size $2c+1$ centered in q_i. For each query term q_i, we stack the word embeddings (obtained as described above under "term representation") of the terms in its context to obtain the context matrix $C_i = \{x_{i-c}, .., x_i, .., x_{i+c}\}$. If less than c terms precede q_i or less than c terms follow it, we pad C_i with zeros in order to keep its size consistent with other context matrices.

We chose to define context as the terms appearing in window around each query term, rather than the entire clinical note, as we argue that terms in close proximity to each other contain strong signals that can be used to estimate term relevance, while considering a larger window would add unnecessary noise. Results supporting this observation are presented in Section 3.5.4. Overall, the approach used to obtain a representation of the context of a term was modeled after the architecture proposed by Severyn and Moschitti [122] to predict similarity between short documents.

To obtain the context representation $\hat{C}_i$, we use convolutional filters of size $k = 2, 3, 4,$ and 5, as proposed in [73]. This approach allows to capture local features with different granularities. The convolution layer produces $(c - 2\lfloor k/2 \rfloor)$ features per filter per size (stride

[20] `http://commoncrawl.org`
[21] `https://github.com/cambridgeltl/BioNLP-2016/`

size was kept at 1). We indicate the number of filters used for each size as h; we use the same number of filters for each filter size. To reduce dimensionality, we transform each filter using a max pooling layer of size k and stride $\lfloor k/2 \rfloor$ (i.e., from size 2 and stride 1 for $k = 2$ to size 5 and stride 2 for $k = 5$). Finally, after flattening and merging all filters, compact context representation $\hat{C}_{i,c}$ is obtained through a dense layer with ReLU activation function.

We combine term representation $\hat{x}_i$ and context representation $\hat{C}_i$ by concatenation (Figure 3.9). The resulting layer is first encoded using an intermediate hidden layer with ReLU activation function; then, the predicted importance value y_i for term q_i is obtained by linearly combining the output of the hidden layer, as typically done for regression networks. For simplicity, we will use the notation $y_\theta(\vec{q}) = \{y_1, \ldots, y_n\}^\top$ to indicate the vector of predicted importance values for terms in $\vec{q}$ by the model with weights θ.

3.5.1.2 Learning strategies

Predicting WRR

Similarly to our *DNN* approach on clean clinical notes reformulation, we experimented with training our CNN model to predict WRR of terms in the query. Note that, unlike the model described in Section 3.2.3, we do not train on candidate terms; rather, we are interested in predicting WRR for terms in the query. Furthermore, based on our experiments, we determined that — unlike *DNN*— the CNN designed for this reduction task does not benefit from the additional features described in Section 3.2.3.2.

Optimizing Document Ranking

In order to learn to predict the importance y_i of each query term q_i, we also experimented with training our model in a pairwise learning to rank framework. In particular, given triples $\langle \vec{q}, \vec{d_+}, \vec{d_-} \rangle$, where $\vec{d_+}$ is a relevant document for the query, and $\vec{d_-}$ is a non-relevant document for the query, we proceeded as follows: let $Sim(\vec{d}, \vec{q})$ be a function that estimates the similarity of document $\vec{d}$ with query $\vec{q}$. Many similarity functions used in information

retrieval (including BM25, which we used in our experiments), are linear with respect to query term coefficients, i.e., they can be written as:

$$Sim(\vec{d}, \vec{q}) = w(\vec{d}, \vec{q}) \cdot \mathbf{1}_n \tag{3.9}$$

where n is the length of query $\vec{q}$, $w(\vec{d}, \vec{q})$ is a vector of size $1 \times n$ whose elements are the weight of each query term with respect to document $\vec{d}$, and $\mathbf{1}_n$ is a all-ones vector of size $n \times 1$.

In the method we propose, the predicted importance values for terms in $\vec{q}$ are integrated in the similarity function as follows:

$$Sim(\vec{d}, \vec{q}) = w(\vec{d}, \vec{q}) \cdot y_\theta(\vec{q}) \tag{3.10}$$

Leveraging this notation, we can finally define a pairwise max margin loss function with respect to the training triple $\langle \vec{q}, \vec{d_+}, \vec{d_-} \rangle$ and model weights θ:

$$\mathcal{L}_\theta(\vec{q}, \vec{d_+}, \vec{d_-}) = \max\left(0, 1 - w(\vec{d_-}, \vec{q}) y_\theta(\vec{q}) + w(\vec{d_+}, \vec{q}) y_\theta(\vec{q})\right) \tag{3.11}$$

We combine the loss function defined in Equation 3.11 with a regularizing function designed to prevent the model from assigning negative importance to query terms:

$$\mathcal{O}(\vec{q}, \vec{d_+}, \vec{d_-}; \theta) = \mathcal{L}_\theta(\vec{q}, \vec{d_+}, \vec{d_-}) + \textstyle\sum_{y_i \in y_\theta(\vec{q})} \min(0, y_i)^2 \tag{3.12}$$

We train the proposed model by minimizing this objective function.

3.5.2 Experimental Setup

3.5.2.1 Dataset

We studied the effectiveness of the proposed method on the 2016 TREC CDS dataset [117]. It is comprised of 30 topics (each containing a clinical note), 1.25 million articles from the open access subset of PubMed Central, and 28,349 documents whose relevancy to topics

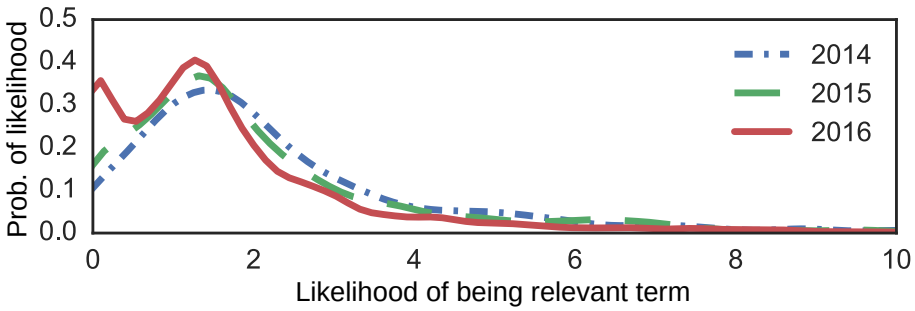

Figure 3.10: Probability density function (PDF) of word relevance ratio (WRR) of terms on the 2014 (blue dashes & dots), 2015 (green dashes), and 2016 (solid red) datasets. Unedited clinical notes (2016 dataset) contain more non-relevant terms (i.e., WRR < 1) than artificial reports (2014/2015 datasets). Since the distributions are comparable, we augment the training set using the 2014 and 2015 datasets.

have been assessed. On average, clinical notes in this dataset have a length of 184 terms and a median of 188; for each note, an average of 182 documents were found to be relevant (median: 119). Note that this dataset is similar, but not identical to the ones used in previous editions of the TREC CDS track we described in Section 3.3.2.

Because of the limited amount of training data proved by the 2016 TREC CDS dataset, we expanded the training set using fictitious clinical descriptions from previous years' collections. Since these collections do not contain clinical notes, we use the fictitious clinical descriptions instead. While descriptions are substantially shorter than actual clinical notes (average length: 81 terms), the distribution of query terms that are likely to appear in relevant documents is sufficiently similar to the one of query terms in the clinical notes dataset (Figure 3.10); the likelihood of a query term being relevant was defined as the probability of appearing in relevant documents for a query over the probability of appearing in non-relevant documents for the query.

3.5.2.2 Model Training

We partition the 2016 dataset in training, development, and test sets. The system was evaluated under three-fold cross validation by rotating the subsets. For all three runs, the training set was always expanded using the 2014 and 2015 TREC CDS datasets.

Optimal model topology was determined through empirical evaluation on the development set. The size of the context and term representation layers was set to 128, while the size of the hidden layer was set to 64. To prevent over-fitting, outputs of all layers (except the last one) were regularized using batch normalization; batch size was set to 32. A 30% dropout was also applied at training time to the input of all layers denoted by a dashed line in Figure 3.9. As illustrated in Section 3.5.4, we experimented with several filter sizes k; the number of filters per size was set to $h = 256$.

Both models were trained using the Adagrad optimizer [37]. For the model trained on WRR, the number of epochs was fixed to 100. For the model trained on rank prediction, each fold was trained until no improvement in infNDCG was achieved on the development set for 30 epochs (at the end of training, the model was rolled back to the last iteration with improvement). Using this heuristic, each fold model was trained, on average, for 105 epochs. Both models were implemented using Tensorflow[22] [1].

3.5.3 Results

In this section, we investigate the impact of the proposed method on the TREC CDS 2016 ad-hoc retrieval task. Performance was measured using the two main metrics of 2016 TREC CDS track: inferred nDCG [164] (primary metric) and precision at 10 retrieved results (P@10). The proposed method was compared with several well-known query reformulation techniques, as well as the best approach from TREC CDS 2016. In detail, we compared the proposed method with the following approaches (reported in Table 3.9; *i* to *iv* are baselines, while *v* to *ix* are state-of-the-art techniques):

[22] `https://www.tensorflow.org/`

Table 3.9: Performance of the proposed approach (*x* and *xi*), several baselines (*i* to *iv*), and state of the art methods (*v* to *xi*) on the TREC CDS 2016 dataset.

		Query reduction approach	TREC CDS 2016	
			infNDCG	**P@10**
baselines	*i*	No query reduction	0.1138*	0.1967*
	ii	*idf* filter	0.1242*	0.2067*
	iii	*MMselect*	0.1580*	0.2400*
	iv	*HT*	0.1670*	0.2300*
st. of the art	*v*	*QQP* [78]	0.1312*	0.2133
	vi	*Fast QQP* [132]	0.1520*	0.2433*
	vii	*HTPRF* [128]	0.1926*	0.2800†
	viii	PCW [12]	0.1833*	0.2900†
	ix	NKU [176] *(best at CDS TREC 2016)*	0.1978*	0.2900†
	x	CNN *(trained on WRR prediction)*	0.1896*	0.2700†
	xi	**CNN *(trained on document rank optimization)***	**0.2518**	**0.3167**

Compared to the proposed method (*ix*), results marked with * are significantly different (paired Student *t*-test, Bonferroni-adjusted, $p < 0.005$). Resutls marked with † show large, yet not statistically significant, differences with the best method ($p < 0.05$).

(i) **No query reduction**: we left the clinical note as-is, except removing numbers, stop words, and units of measurement.

(ii) ***idf* filter**: we removed terms whose *idf* is less than 1 (term appears in more than 10% of the documents) and more than 5.5 (term appears in less than 3 documents in the collection); values were determined through manual tuning on the development set. Just like the previous method, we also removed numbers, stop words, and units of measurement.

(iii) ***MMselect***: similarly to the 2014 and 2015 TREC dataset, we tested the UMLS method described in Section 3.3.3.1 on the 2016 dataset. Expressions in the clinical notes were mapped to concepts in UMLS using QuickUMLS [126]; terms that are not in UMLS were removed.

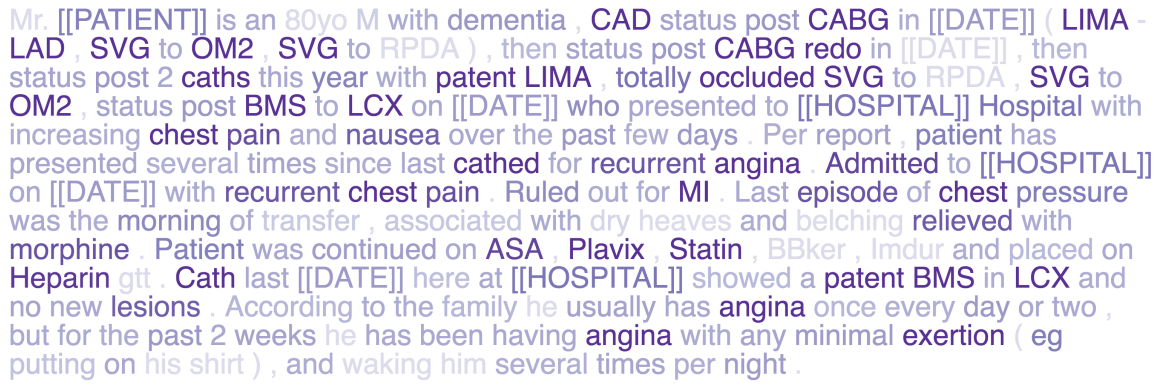

Figure 3.11: Weights assigned by the CNN model trained on document rank optimization to terms in the query shown in Figure 3.8. For the condition the patient is suffering from ("recurrent angina"), the model is able to accurately predict important (e.g., "CABG", an acronym for "coronary artery bypass graph") and irrelevant (e.g., "dementia") medical terms for CDS search.

(iv) ***HT***: we also consider the health terms filter introduced in Section 2.2.1.2 as baseline.

Furthermore, we compared the proposed method with several state of the art methods:

(v) ***QQP*** and ***Fast QQP***: we evaluated the impact of the state of the art methods described in Section 3.3.3.2; while these methods showed limited to no improvement in on clean clinical notes, we include them in the evaluation due to the significant differences in the CDS TREC 2016 dataset. This method uses quality predictors as features to learn to rank sub-queries of clinical notes.

(vi) **Parameterized concept weighting (PCW)**: we implemented the supervised model introduced by Bendersky, Metzler, and Croft [12] to learn weights of concepts in the query. This model uses statistical features (e.g., term and document frequency in target collection) to learn the importance weight of three concept types: unigrams, bigrams phrases, and proximity bigrams. We expanded the set of concept types with medical

concepts extracted with QuickUMLS [126], and the set of features with term and document frequencies of candidate concepts in several medical collections.

(vii) **NKU team**: we compared our system with the work of Zhang and Liu [176], which obtained the best performance on clinical notes at TREC 2016. This method combines concept extraction, query expansion using the MeSH[23], and pseudo relevance feedback.

As shown in Table 3.9, the CNN approach trained on document rank optimization (Table 3.9, line *xi*) outperformed all baselines and state-of-the-art methods. Compared to the CNN trained on predicting WRR, it achieved an improvement of 27% in infNDCG. This justifies the new training strategy over the one proposed for the 2014 and 2015 datasets.

The difference between the proposed CNN and the other methods' performance is more prominent in terms of inferred NDCG, as we observed an improvement of 121% over the unmodified clinical note (line *i*), 37% over the best general domain query reduction (PCW, line *viii*), and 27% over the best system proposed for this task (NKU, line *ix*).

The proposed CNN showed a less pronounced improvement over state of the art methods in terms of P@10; nevertheless, it outperforms all state of the art methods by at least 9% (line *ix*) and up to 30% (line *v*). We attribute this outcome to the fact that the proposed method was trained to maximize the difference in scores between relevant and non-relevant documents; thus, it suffers in precision-oriented metrics with early cutoff, such as P@10.

Finally, we observed that approaches that explicitly take advantage of domain specific resources, such as medical concept extraction using UMLS (*iii*), *HT* (*iv*), and *Fast QQP* (*vi*) outperform methods that do not leverage such resources (*ii* and *v*). This confirms the finding of Balaneshin-kordan and Kotov [8] and Soldaini et al. [132].

3.5.4 Parameter Tuning

We studied the impact of the hyperparameters detailed in section 3.5.1.1 on performance of the proposed method. In detail, we conducted two experiments: we evaluated the impact of

[23] `https://www.nlm.nih.gov/mesh/`

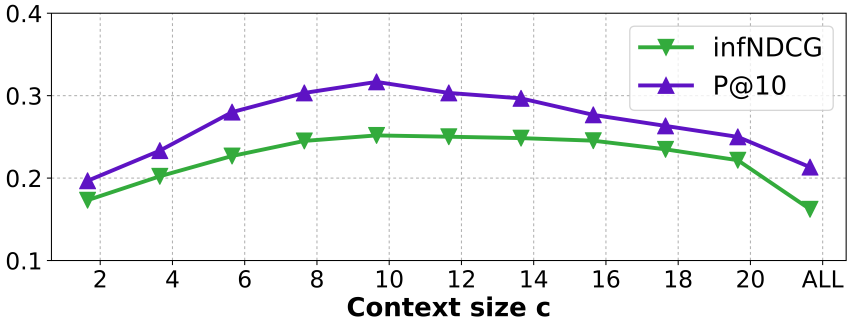

Figure 3.12: Impact of context size on the best method's performance.

context size c on infNDCG and P@10 (Figure 3.12), and we performed an ablation study to quantify the impact of convolutional filter sizes (Table 3.10).

We experimented with context sizes ranging from $c = 2$ (that is, considering two terms before and two terms after each query term) to using the entire clinical note as context ($c = \text{ALL}$). As shown in Figure 3.12, the best performance is obtained when $c = 10$. While the performance of the system is not affected by small deviations from the optimal value, choosing a context that is too small ($c \leqslant 4$) or too large ($c \geqslant 15$) notably reduced its effectiveness. In particular, we note that the model that uses the entire clinical note as context performed worse than any other context size c in terms of infNDCG, supporting our decision to limit the context size.

Finally, we evaluated the impact of the convolutional filters size k using an ablation study. The results presented in Table 3.10 suggest that using multiple values for k has positive impact on capturing local features, as each filter size learn specific aspects of term interaction in the context. However, we note that the improvement in performance got smaller as larger filters were introduced in the model.

Table 3.10: Ablation study on the size of convolutional filters.

Size(s) of convolutional filters used	infNDCG	P@10
$k = 2$	0.2342	0.2867
$k = 2, 3$	0.2435	0.3033
$k = 2, 3, 4$	0.2498	0.3100
$\mathbf{k = 2, 3, 4, 5}$	**0.2518**	**0.3167**

3.6 Conclusions

In this chapter, I studied whether expanding complex medical queries by latent medical concepts improves the retrieval quality of medical literature. The problem was studied in the context of clinical decision support (CDS) search, which is a search task intended to help medical practitioners retrieve relevant publications to clinical case reports.

Two query reformulation approaches were introduced for this task. The first (*HTPRF*) combines pseudo relevance feedback with a health term filter designed to remove non-health related terms from the expansion candidates. The second method (*DNN*) is a supervised approach to query expansion that leverages a deep neural network to predict each candidate term's weighted relevance ratio, a measure of importance of each term in relevant documents. To train the model, we use a combination of word embeddings, syntactical and semantic features over the candidate terms, and statistical features derived from the distribution of candidates and query terms in several auxiliary collections.

The proposed approaches was validated using two benchmarks: an artificial dataset derived from USMLE prep book questions, and a manually annotated dataset that was introduced in TREC 2014 and 2015. When compared to state of the art, the two systems fair well, outperforming them up to 8% in infNDCG. Overall, *DNN* achieved better infNDCG, while *HTPRF* obtained better P@10 performance.

Finally, a convolutional neural model to reduce noise in clinical notes was presented. This method was designed to handle clinical notes that contain an abundance of medical and clinical abbreviations, incomplete sentences, redundant or unnecessary information. For each

term in a clinical note, the proposed approach takes advantage of the context surrounding the term to predict its importance. The proposed approach was evaluated on the TREC CDS 2016 dataset, and compared several query reduction baselines, as well as state of the art methods, outperforming them all.

Chapter 4

Conclusions

In this dissertation, I argued that both laypeople and experts suffer from language and knowledge gaps when seeking health information. For laypeople, this is mostly due to limited knowledge of the medical domain; for experts, this is due to lack of expertise in a specific domain or limited time to invest in interacting with retrieval systems.

As evidenced by my scholarly work [27, 125, 126, 127, 128, 130, 131, 132], I have studied how to quantify such gap in searches issued by laypeople (*hypothesis 1.1*), close the language gap in health searches with query reformulation (*hypothesis 1.2*), utilize search results reranking to promote documents that are semantically close to health queries (*hypothesis 1.3*), reformulate complex health queries to improve medical literature retrieval (*hypothesis 2.1*), and de-noise clinical notes, making them suitable for document retrieval (*hypothesis 2.2*).

The impact of the work put forward in this dissertation is two-fold: first, it introduces a framework to quantify and close the knowledge gap laypeople experience when looking for medical information online. As more and more individuals rely on the Internet to get informed about their health, it is crucial to provide access to reliable websites that can adequately satisfy their information needs. Given a query submitted by a lay user, the work presented in Chapter 2 focuses on closing this gap either (*i*) by expanding the query through *query clarification* or (*ii*) by re-ranking search results based on their semantic similarity to the query. The former was proved effective at improving the success rate of users in understanding health topics. In particular, results show that, when presented with search results retrieved using *clarified* queries, users are up to 12.8% more likely to correctly

answer a medical question related to the queries. The best result, achieved by a linear classifier that automatically choose the best synonym mapping to *clarify* the query, confirms that explicitly bridging the vocabulary gap by "translating" form lay vocabulary to expert vocabulary is a simple, yet effective way to improve search outcomes. In Chapter 2, I also showed that learning to rerank queries based on their semantic similarity is also an effective approach to ameliorate the knowledge gap between lay searcher and medical content available online. Experiments showed that a random forest regressor trained to predict query-result similarity within a learning to rank framework achieved a 26.2% improvement over the baseline, validating the proposed approach. Further feature analysis showed that features designed to capture (i) the distribution of query and document terms in several health collections, or (ii) semantic similarity between query and document are particularly effective for this task. This suggest relevant pages for a query not only contain semantically related terms, but also that the type of medical content in a page is an important indicator of relevance.

This dissertation also presented several clinical decision support systems aimed at integrating medical literature in clinical practice. The problem of bringing the latest findings in clinical research to those who practice medicine has been long studied; research shows that healthcare professionals struggle to keep up-to-date with current advances in clinical research, and that this might lead to dangerous clinical errors due to misinformation. The two solutions proposed in this manuscript — one supervised, the other unsupervised — effectively address this problem by automatically reformulating clinical notes as queries, which can then by used to retrieve relevant medical literature. The unsupervised method ($HTPRF$) pairs pseudo relevance feedback with a statistical health term filter; the former is designed to identify candidate for query expansion, while the latter removes terms that are not medical concepts or medically-related terms. The supervised method (DNN) leverages a convolutional neural network to predict, for each expansion candidate, its importance in relevant documents for the query; the model takes advantage of word embedding representations of

the query and candidate terms, as well as statistical features designed to capture the use of candidate terms across several medical and non-medical collections. When compared to state of the art, the two systems fair well, outperforming them up to 8% in infNDCG. Overall, *DNN* achieved better infNDCG, while *HTPRF* obtained better P@10 performance. The latter is a particularly desirable characteristic, as it suggests that *HTPRF* could employed to retrieve few, highly relevant papers that physicians could quickly consult during practice, thus addressing limitation of current literature search systems.

Finally, I presented an improved version of *DNN* specifically designed to remove noisy terms (i.e., medical and clinical abbreviations, incomplete sentences, redundant or unnecessary information) that are often present in clinical notes. Such noise is common in clinical notes, and, as shown in Section 3.5, it negatively impacts performance of CDS search systems. The proposed approach is based on a convolutional neural network; it estimates, for each term in a clinical note, its likelihood of being a noisy term. It is designed to optimize the rank of relevant documents in the retrieved set for a given note. When compared with state of the art query reformulation techniques for noisy clinical notes, it achieves an improvement of 27% in inferred nDCG (+10% in P@10); it also outperforms the best domain-agnostic query reduction techniques by 37% (+10% in P@10). This demonstrates that effective CDS systems can be designed even for non-ideal clinical search environments.

4.1 Future Work

While the systems and methods put forward in this dissertation represent effective solutions for closing the language and knowledge users suffer in medical information retrieval, they could be further enhance through user modeling or semantic analysis.

For lay users, this work has proposed solutions to reduce the language gap by exploiting the information need expressed in a query; however, further improvements could be achieved by considering a model of the user who submitted it. For example, session information could be leveraged to refine search results to only include documents that are consistent

with previous medical searches; similarly, health-related information shared by users (for example, through social media) could also represent an important signal to improve modeling of their needs. Beside a user profile, which might contain sensitive personal information, population-level statistics could be used to resolve ambiguous information needs (for example, by providing information about diseases that are most likely in area where a user is roughly located.) Approaches in this family would cause only a minimal compromise of users' privacy.

Clinical decision support search systems could be improved by considering methods outside core information retrieval research. First, natural language processing techniques could be applied to improve reduction of noisy clinical notes, either through semantic analysis or summarization. However, in order to succeed these approaches will have to be coupled with an increase in data annotated for the task. Alternatively, inference between symptoms, diseases, and treatments could be applied to increase precision of CDS systems: semantic relationships between medical concepts in clinical notes and retrieved literature could be used to promote papers about affine conditions, and demote literature about unrelated topics. In this case, sparseness of existing ontologies, particularly in terms of symptoms-disease and disease-treatment relationships, ought to be addressed to create an effective system.

Bibliography

[1] Martin Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: `https://www.tensorflow.org/`.

[2] Samir Abdou and Jacques Savoy. "Searching in Medline: Query expansion and manual indexing evaluation". In: *Information Processing & Management* (2008).

[3] Julia Adler-Milstein et al. "More than half of US hospitals have at least a basic EHR, but stage 2 criteria remain challenging for most". In: *Health Affairs* (2014), pp. 10–1377.

[4] Arvind Agarwal et al. "Learning to Rank for Robust Question Answering". In: *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM. 2012, pp. 833–842.

[5] Gianni Amati and Cornelis Joost Van Rijsbergen. "Probabilistic models of information retrieval based on measuring the divergence from randomness". In: *TOIS* 4 (2002).

[6] Alan R Aronson and François-Michel Lang. "An overview of MetaMap: historical perspective and recent advances". In: *Journal of the American Medical Informatics Association* 17.3 (2010), pp. 229–236.

[7] Saeid Balaneshin Kordan, Alexander Kotov, and Railan Xisto. "WSU-IR at TREC 2015 Clinical Decision Support Track: Joint Weighting of Explicit and Latent Medical Query Concepts from Diverse Sources". In: *Proceedings of the 2015 Text Retrieval Conference*. 2015.

[8] Saeid Balaneshin-kordan and Alexander Kotov. "Optimization Method for Weighting Explicit and Latent Concepts in Clinical Decision Support Queries". In: *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*. ICTIR '16. New York, NY, USA: ACM, 2016, pp. 241–250.

[9] Niranjan Balasubramanian, Giridhar Kumaran, and Vitor R Carvalho. "Exploring reductions for long web queries". In: *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*. ACM. 2010.

[10] Ayelet Ben-Sasson and Elad Yom-Tov. "Online Concerns of Parents Suspecting Autism Spectrum Disorder in Their Child: Content Analysis of Signs and Automated Prediction of Risk". In: *Journal of Medical Internet Research* 18.11 (2016).

[11] Michael Bendersky and W. Bruce Croft. "Discovering Key Concepts in Verbose Queries". In: *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '08. Singapore, Singapore: ACM, 2008, pp. 491–498. ISBN: 978-1-60558-164-4. DOI: 10.1145/1390334.1390419. URL: http://doi.acm.org/10.1145/1390334.1390419.

[12] Michael Bendersky, Donald Metzler, and W Bruce Croft. "Parameterized concept weighting in verbose queries". In: *SIGIR*. ACM. 2011.

[13] Michael Bendersky, Donald Metzler, and W. Bruce Croft. "Learning Concept Importance Using a Weighted Dependence Model". In: *WSDM*. 2010.

[14] William Blacoe and Mirella Lapata. "A comparison of vector-based representations for semantic composition". In: ACL. Association for Computational Linguistics. 2012.

[15] David T Burke et al. “Reading habits of physical medicine and rehabilitation resident physicians”. In: *American journal of physical medicine & rehabilitation* (2004).

[16] Stefan Büttcher, Charles LA Clarke, and Gordon V Cormack. “Domain-Specific Synonym Expansion and Validation for Biomedical Information Retrieval (MultiText Experiments for TREC 2004).” In: *TREC*. 2004.

[17] Aysu Betin Can and Nazife Baykal. “MedicoPort: A medical search engine for all”. In: *Computer methods and programs in biomedicine* 86.1 (2007), pp. 73–86.

[18] Zhe Cao et al. “Learning to rank: from pairwise approach to listwise approach”. In: ICML. 2007.

[19] David Carmel and Elad Yom-Tov. “Estimating the query difficulty for information retrieval”. In: *Synthesis Lectures on Information Concepts, Retrieval, and Services* 2.1 (2010), pp. 1–89.

[20] David Carmel et al. “Automatic query wefinement using lexical affinities with maximal information gain”. In: *Proceedings of SIGIR ’02*. ACM. 2002, pp. 283–290.

[21] David Carmel et al. “What makes a query difficult?” In: *Proceedings of SIGIR ’06*. ACM. 2006, pp. 390–397.

[22] Marc-Allen Cartright, Ryen W White, and Eric Horvitz. “Intentions and attention in exploratory health search”. In: *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM. 2011, pp. 65–74.

[23] Emily H Chan et al. “Using web search query data to monitor dengue epidemics: a new model for neglected tropical disease surveillance”. In: *PLoS Negl Trop Dis* 5.5 (2011), e1206.

[24] Billy Chiu et al. “How to train good word embeddings for biomedical NLP”. In: ACL (2016).

[25] Sungbin Choi and Jinwook Choi. *SNUMedinfo at TREC CDS track 2014: Medical case-based retrieval task*. Tech. rep. DTIC Document, 2014.

[26] Arman Cohan et al. "A Neural Attention Model for Categorizing Patient Safety Events". In: ECIR. 2017.

[27] Arman Cohan et al. "On clinical decision support". In: *Proceedings of the 5th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics*. ACM. 2014, pp. 651–652.

[28] Michael J Cole et al. "Knowledge effects on document selection in search results pages". In: *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM. 2011, pp. 1219–1220.

[29] Francis S Collins and Harold Varmus. "A new initiative on precision medicine". In: *New England Journal of Medicine* 372.9 (2015), pp. 793–795.

[30] Gordon V Cormack, Charles LA Clarke, and Stefan Buettcher. "Reciprocal rank fusion outperforms condorcet and individual rank learning methods". In: *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*. ACM. 2009, pp. 758–759.

[31] Ali Dasdan, Chris Drome, and Santanu Kolay. "Thumbs-up: A Game for Playing to Rank Search Results". In: *Proceedings of the 18th International Conference on World Wide Web*. WWW '09. Madrid, Spain: ACM, 2009, pp. 1071–1072.

[32] Guilherme Del Fiol, T Elizabeth Workman, and Paul N Gorman. "Clinical questions raised by clinicians at the point of care: a systematic review". In: *JAMA Intern. Med.* 174.5 (2014), pp. 710–718.

[33] Guilherme Del Fiol, T Elizabeth Workman, and Paul N Gorman. "Clinical questions raised by clinicians at the point of care: a systematic review". In: *JAMA internal medicine* 174.5 (2014), pp. 710–718.

[34] Catherine M DesRoches et al. “Electronic health records in ambulatory care—a national survey of physicians”. In: *New England Journal of Medicine* 359.1 (2008), pp. 50–60.

[35] Liang Dong, Pradip K Srimani, and James Z Wang. “Ontology graph based query expansion for biomedical information retrieval”. In: *Bioinformatics and Biomedicine (BIBM), 2011 IEEE International Conference on*. IEEE. 2011, pp. 488–493.

[36] Benjamin G Druss and Steven C Marcus. “Growth and decentralization of the medical literature: implications for evidence-based medicine”. In: *Journal of the Medical Library Association* 93.4 (2005), p. 499.

[37] John Duchi, Elad Hazan, and Yoram Singer. “Adaptive Subgradient Methods for Online Learning and Stochastic Optimization”. In: *J. Mach. Learn. Res.* 12 (July 2011), pp. 2121–2159.

[38] Cynthia Dwork et al. “Rank Aggregation Methods for the Web”. In: *Proceedings of the 10th International Conference on World Wide Web*. WWW ’01. Hong Kong, Hong Kong: ACM, 2001, pp. 613–622.

[39] Gunther Eysenbach and Christian Köhler. “How do consumers search for and appraise health information on the world wide web? Qualitative study using focus groups, usability tests, and in-depth interviews”. In: *Bmj* 324.7337 (2002), pp. 573–577.

[40] Susannah Fox. *The social life of health information*. Web: `http://www.pewresearch.org/fact-tank/2014/01/15/the-social-life-of-health-information/`. 2014.

[41] Susannah Fox and Maeve Duggan. *Health Online 2013*. Web: `http://www.pewinternet.org/Reports/2013/Health-online.aspx`. 2013.

[42] Jeremy Ginsberg et al. “Detecting influenza epidemics using search engine query data”. In: *Nature* 457.7232 (2009), pp. 1012–1014.

[43] Julien Gobeill et al. *Full-texts Representations with Medical Subject Headings, and Co-citations Network Reranking Strategies for TREC 2014 Clinical Decision Support Track*. Tech. rep. DTIC Document, 2014.

[44] Lorraine Goeuriot, Liadh Kelly, and Johannes Leveling. "An analysis of query difficulty for information retrieval in the medical domain". In: *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. ACM. 2014, pp. 1007–1010.

[45] Lorraine Goeuriot et al. "ShARE/CLEF eHealth Evaluation Lab 2013, Task 3: Information retrieval to address patients' questions when reading clinical reports". In: (2013).

[46] Lorraine Goeuriot et al. "ShArE/CLEF eHealth evaluation lab 2014, task 3: User-centred health information retrieval". In: *Proceedings of CLEF*. Vol. 2014. 2014.

[47] Ana I González-González et al. "Information needs and information-seeking behavior of primary care physicians". In: *The Annals of Family Medicine* 5.4 (2007), pp. 345–352.

[48] Travis R Goodwin and Sanda M Harabagiu. "Medical Question Answering for Clinical Decision Support". In: *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. CIKM '16. New York, NY, USA: ACM, 2016, pp. 297–306.

[49] Trisha Greenhalgh, Jeremy Howick, and Neal Maskrey. "Evidence based medicine: a movement in crisis?" In: *Bmj* 348 (2014), g3725.

[50] David A Grossman and Ophir Frieder. *Information Retrieval: Algorithms and Heuristics*. Springer, 2012.

[51] Allan Hanbury et al. "KHRESMOI: towards a multi-lingual search and access system for biomedical information". In: *Med-e-Tel, Luxembourg* 2011 (2011), pp. 412–416.

[52] David R Hansberry et al. “A critical review of the readability of online patient education resources from RadiologyInfo. Org”. In: *American Journal of Roentgenology* 202.3 (2014), pp. 566–575.

[53] Taher H Haveliwala. “Topic-sensitive pagerank”. In: *Proceedings of the 11th international conference on World Wide Web*. ACM. 2002, pp. 517–526.

[54] Agency for Healthcare Research and Quality — U.S. Department of Health & Human Services. *Programs: Prevention & Chronic Care: Clinical Decision Support*. Apr. 2014. URL: `https://www.ahrq.gov/professionals/prevention-chronic-care/decision/clinical/index.html`.

[55] James M Heilman and Andrew G West. “Wikipedia and Medicine: Quantifying Readership, Editors, and the Significance of Natural Language”. In: *Journal of medical Internet research* 17.3 (2015), e62.

[56] Dawn Heisey-Grove et al. “A national study of challenges to electronic health record adoption and meaningful use”. In: *Medical care* 52.2 (2014), pp. 144–148.

[57] William R Hersh et al. “Factors associated with success in searching MEDLINE and applying evidence to answer clinical questions.” In: *J Am Med Inform Assoc* 9.3 (2002), pp. 283–293.

[58] William Hersh, Jeffrey Pentecost, and David Hickam. “A Task-oriented Approach to Information Retrieval Evaluation”. In: *J. Am. Soc. Inf. Sci.* 47.1 (Jan. 1996), pp. 50–56.

[59] William Hersh, Susan Price, and Larry Donohoe. “Assessing thesaurus-based query expansion using the UMLS Metathesaurus.” In: *Proceedings of the AMIA Symposium*. American Medical Informatics Association. 2000, p. 344.

[60] William Hersh and Ellen Voorhees. “TREC genomics special issue overview”. In: *Information Retrieval* 12.1 (2009), pp. 1–15.

[61] William Hersh et al. “OHSUMED: An interactive retrieval evaluation and new large test collection for research”. In: *SIGIR’94*. Springer. 1994.

[62] Matthew Honnibal and Mark Johnson. “An Improved Non-monotonic Transition System for Dependency Parsing”. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, Sept. 2015, pp. 1373–1378. URL: https://aclweb.org/anthology/D/D15/D15-1162.

[63] Nora Hutchinson, Grayson L Baird, and Megha Garg. “Examining the reading level of internet medical information for common internal medicine diagnoses”. In: *The American journal of medicine* 129.6 (2016), pp. 637–639.

[64] Sergey Ioffe and Christian Szegedy. “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. In: *arXiv:1502.03167* (2015).

[65] Vahid Jalali and Mohammad Reza Matash Borujerdi. “Information retrieval with concept-based pseudo-relevance feedback in MEDLINE”. In: *Knowledge and information systems* 29.1 (2011), pp. 237–248.

[66] Peter B Jensen, Lars J Jensen, and Søren Brunak. “Mining electronic health records: towards better research applications and clinical care”. In: *Nat. Rev. Genet.* 13 (May 2012), p. 395.

[67] Ashish K Jha et al. “Use of electronic health records in US hospitals”. In: *New England Journal of Medicine* 360.16 (2009), pp. 1628–1638.

[68] Thorsten Joachims. “Optimizing search engines using clickthrough data”. In: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2002, pp. 133–142.

[69] Thorsten Joachims. “Training linear SVMs in linear time”. In: *Proceedings of the 12^{th} SIGKDD international conference on Knowledge Discovery and Data Mining*. ACM. 2006.

[70] Thorsten Joachims et al. “Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search”. In: *ACM Transactions on Information Systems (TOIS)* 25.2 (2007), p. 7.

[71] Alistair E W Johnson et al. “MIMIC-III, a freely accessible critical care database”. en. In: *Sci Data* 3 (May 2016), p. 160035.

[72] Jayashree Kalpathy-Cramer et al. “Evaluating performance of biomedical image retrieval systems—An overview of the medical image retrieval task at ImageCLEF 2004–2013”. In: *Computerized Medical Imaging and Graphics* 39 (2015), pp. 55–61.

[73] Yoon Kim. “Convolutional neural networks for sentence classification”. In: *arXiv:1408.5882* (2014).

[74] Diederik Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv:1412.6980* (2014).

[75] Linda T Kohn, Janet M Corrigan, Molla S Donaldson, et al. “Errors in health care: a leading cause of death and injury”. In: (2000).

[76] Bevan Koopman et al. “Information retrieval as semantic inference: graph inference model applied to medical search”. In: *IR Journal* (2016).

[77] Michael Kuhn et al. “A side effect resource to capture phenotypic effects of drugs.” In: *Molecular systems biology* 6 (Jan. 2010), p. 343.

[78] Giridhar Kumaran and Vitor R Carvalho. “Reducing long queries using query quality predictors”. In: *SIGIR*. 2009.

[79] Vasileios Lampos et al. “Advances in nowcasting influenza-like illness rates using search query logs”. In: *Scientific reports* 5 (2015), p. 12760.

[80] Nut Limsopatham, Craig Macdonald, and Iadh Ounis. “Inferring Conceptual Relationships to Improve Medical Records Search”. In: *Proceedings of the 10th Conference on Open Research Areas in Information Retrieval*. OAIR ’13. Lisbon, Portugal: LE CENTRE DE HAUTES ETUDES INTERNATIONALES

D'INFORMATIQUE DOCUMENTAIRE, 2013, pp. 1–8. ISBN: 978-2-905450-09-8. URL: http://dl.acm.org/citation.cfm?id=2491748.2491750.

[81] Tie-Yan Liu et al. "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*. 2007, pp. 3–10.

[82] Zhenyu Liu and Wesley W Chu. "Knowledge-based query expansion to support scenario-specific retrieval of medical free text". In: *Information Retrieval* 10.2 (2007), pp. 173–202.

[83] Zhiyong Lu, Won Kim, and W John Wilbur. "Evaluation of query expansion using MeSH in PubMed". In: *Information retrieval* 12.1 (2009), pp. 69–80.

[84] Gang Luo et al. "MedSearch: a specialized search engine for medical information retrieval". In: *Proceedings of the 17th ACM conference on Information and knowledge management*. ACM. 2008.

[85] Yuanhua Lv and ChengXiang Zhai. "When documents are very long, BM25 fails!" In: *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. ACM. 2011, pp. 1103–1104.

[86] Sérgio Matos et al. "Concept-based query expansion for retrieving gene related publications from MEDLINE". In: *BMC bioinformatics* 11.1 (2010), p. 1.

[87] Paul McNamee. "A Domain Independent Approach to Clinical Decision Support". In: (2015).

[88] Paul McNamee and James Mayfield. "Character n-gram tokenization for European language text retrieval". In: *Information retrieval* 7.1-2 (2004), pp. 73–97.

[89] Saket SR Mengle and Nazli Goharian. "Ambiguity measure feature-selection algorithm". In: *Journal of the American Society for Information Science and Technology* 60.5 (2009), pp. 1037–1050.

[90] Tomas Mikolov et al. “Distributed representations of words and phrases and their compositionality”. In: *NIPS*. 2013.

[91] Tomas Mikolov et al. “Efficient estimation of word representations in vector space”. In: *arXiv* (2013).

[92] David N Milne, Ian H Witten, and David M Nichols. “A knowledge-based search engine powered by Wikipedia”. In: *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. ACM. 2007, pp. 445–454.

[93] David Milne, Olena Medelyan, and Ian H Witten. “Mining domain-specific thesauri from Wikipedia: A case study”. In: *Proceedings of the 2006 IEEE/WIC/ACM international conference on web intelligence*. IEEE Computer Society. 2006, pp. 442–448.

[94] André Mourao, Flávio Martins, and Joao Magalhaes. *NovaSearch at TREC 2014 clinical decision support track*. Tech. rep. DTIC Document, 2014.

[95] Xiangming Mu and Sukjin You. “TREC 2015 paper submission UWM-UO at 2015 Clinical Decision Support Track: QE by Weighted Keywords using PRF.” In: *TREC*. 2015.

[96] Vinod Nair and Geoffrey E. Hinton. “Rectified Linear Units Improve Restricted Boltzmann Machines”. In: *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*. Ed. by Johannes Fürnkranz and Thorsten Joachims. Omnipress, 2010, pp. 807–814.

[97] Eitan Naveh, Tal Katz-Navon, and Zvi Stern. “Resident physicians’ clinical training and error rate: the roles of autonomy, consultation, and familiarity with the literature”. In: *Advances in Health Sciences Education* 20.1 (2015), pp. 59–71.

[98] Heung-Seon Oh and Yuchul Jung. “A Multiple-stage Approach to Re-ranking Clinical Documents.” In: *CLEF (Working Notes)*. 2014, pp. 210–219.

[99] Heung-Seon Oh and Yuchul Jung. “Cluster-based query expansion using external collections in medical information retrieval”. In: *Journal of biomedical informatics* 58 (2015), pp. 70–79.

[100] João Palotti, Allan Hanbury, and Henning Müller. *Exploiting health related features to infer user expertise in the medical domain*. 2014.

[101] Joao Palotti et al. “Ranking health web pages with relevance and understandability”. In: *Proceedings of the 39th international ACM SIGIR conference on Research and development in information retrieval*. ACM. 2016, pp. 965–968.

[102] João Palotti et al. “How users search and what they search for in the medical domain”. In: *IR Journal* (2016).

[103] João Palotti et al. “How users search and what they search for in the medical domain”. In: *Information Retrieval Journal* 19.1-2 (2016), pp. 189–224.

[104] John Paparrizos, Ryen W White, and Eric Horvitz. “Detecting devastating diseases in search logs”. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2016, pp. 559–568.

[105] Jon Parker et al. “Health-related hypothesis generation using social media data”. In: *Social Network Analysis and Mining* 5.1 (2015), pp. 1–15.

[106] Michael J Paul, Ryen W White, and Eric Horvitz. “Search and breast cancer: On episodic shifts of attention over life histories of an illness”. In: *ACM Transactions on the Web (TWEB)* 10.2 (2016), p. 13.

[107] Fabian Pedregosa et al. “Scikit-learn: Machine learning in Python”. In: *The Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[108] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. “GloVe: Global Vectors for Word Representation”. In: *EMNLP*. 2014.

[109] Veronica Pinchin. *I'm Feeling Yucky :(Searching for symptoms on Google*. Web: https://blog.google/products/search/im-feeling-yucky-searching-for-symptoms/. 2016.

[110] John Powell et al. "The characteristics and motivations of online health information seekers: cross-sectional survey and qualitative interview study". In: *Journal of Medical Internet Research* 13.1 (2011).

[111] Filip Radlinski and Nick Craswell. "Optimized interleaving for online retrieval evaluation". In: *Proceedings of the sixth ACM international conference on Web search and data mining*. ACM. 2013, pp. 245–254.

[112] Filip Radlinski, Madhu Kurup, and Thorsten Joachims. "How does clickthrough data reflect retrieval quality?" In: *Proceedings of the 17th ACM conference on Information and knowledge management*. ACM. 2008, pp. 43–52.

[113] Radim Řehůřek and Petr Sojka. "Software Framework for Topic Modelling with Large Corpora". English. In: *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. http://is.muni.cz/publication/884893/en. Valletta, Malta: ELRA, May 2010, pp. 45–50.

[114] Rami Al-Rfou et al. "Theano: A Python framework for fast computation of mathematical expressions". In: *arXiv e-prints* abs/1605.02688 (May 2016). URL: http://arxiv.org/abs/1605.02688.

[115] Kirk Roberts. "Assessing the Corpus Size vs Similarity Trade-off for Word Embeddings in Clinical NLP". In: *ClinicalNLP workshop at COLING 2016*. 2016.

[116] Kirk Roberts et al. "Overview of the TREC 2015 Clinical Decision Support Track". In: (2016).

[117] Kirk Roberts et al. "Overview of the TREC 2016 Clinical Decision Support Track." In: *TREC*. 2017.

[118] Kirk Roberts et al. “State-of-the-art in biomedical literature retrieval for clinical cases: a survey of the TREC 2014 CDS track”. In: *Information Retrieval Journal* 19.1-2 (2016), pp. 113–148.

[119] Joseph John Rocchio. “Relevance feedback in information retrieval”. In: (1971).

[120] S Trent Rosenbloom et al. “Data from clinical notes: a perspective on the tension between structure and flexible documentation”. en. In: *J. Am. Med. Inform. Assoc.* 18.2 (Mar. 2011), pp. 181–186.

[121] Mohammed Saeed et al. “Multiparameter Intelligent Monitoring in Intensive Care II: a public-access intensive care unit database”. en. In: *Crit. Care Med.* 39.5 (May 2011), pp. 952–960.

[122] Aliaksei Severyn and Alessandro Moschitti. “Learning to Rank Short Text Pairs with Convolutional Deep Neural Networks”. In: *SIGIR*. 2015.

[123] Wei Shen et al. “An investigation of the effectiveness of concept-based approach in medical information retrieval GRIUM at CLEF2014eHealthTask 3”. In: *Proceedings of the ShARe/CLEF eHealth Evaluation Lab* (2014).

[124] Matthew S Simpson and Dina Demner-Fushman. “Biomedical text mining: A survey of recent progress”. In: *Mining text data*. Springer, 2012, pp. 465–517.

[125] Luca Soldaini, Will Edman, and Nazli Goharian. “Team GU-IRLAB at CLEF eHealth 2016: Task 3”. In: CLEF. 2016.

[126] Luca Soldaini and Nazli Goharian. “QuickUMLS: a fast, unsupervised approach for medical concept extraction”. In: *Proceedings of the 2nd Medical Information Workshop (MedIR) at the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. 2016.

[127] Luca Soldaini, Andrew Yates, and Nazli Goharian. “Denoising Clinical Notes for Medical Literature Retrieval with Convolutional Neural Model”. In: *26th ACM*

International Conference on Information and Knowledge Management (CIKM 2017). 2017.

[128] Luca Soldaini, Andrew Yates, and Nazli Goharian. "Learning to reformulate long queries for clinical decision support". In: *Journal of the Association for Information Science and Technology* 68.11 (2017), pp. 2602–2619.

[129] Luca Soldaini and Elad Yom-Tov. "Inferring individual attributes from search engine queries and auxiliary information". In: *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee. 2017.

[130] Luca Soldaini et al. "Enhancing web search in the medical domain via query clarification". In: *IR Journal* (2016).

[131] Luca Soldaini et al. "Query Reformulation for Clinical Decision Support Search". In: *The Twenty-Third Text REtrieval Conference Proceedings (TREC 2014)*. 2015.

[132] Luca Soldaini et al. "Retrieving medical literature for clinical decision support". In: *European Conference on Information Retrieval*. Springer. 2015, pp. 538–549.

[133] Amanda Spink et al. "A study of medical and health queries to web search engines". In: *Health Information & Libraries Journal* 21.1 (2004), pp. 44–51.

[134] P Srinivasan. "Optimal document-indexing vocabulary for MEDLINE". In: *Information Processing & Management* 32.5 (1996), pp. 503–514.

[135] Padmini Srinivasan. "Query expansion and MEDLINE". In: *Information Processing & Management* 32.4 (1996), pp. 431–443.

[136] Nitish Srivastava et al. "Dropout: a simple way to prevent neural networks from overfitting." In: *JMLR* (2014).

[137] Isabelle Stanton, Samuel Ieong, and Nina Mishra. "Circumlocution in diagnostic medical queries". In: *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. ACM. 2014, pp. 133–142.

[138] Nicola Stokes et al. “Exploring criteria for successful query expansion in the genomic domain”. In: *Information retrieval* 12.1 (2009), pp. 17–50.

[139] Karthik Subbian and Prem Melville. “Supervised Rank Aggregation for Predicting influencers in twitter”. In: *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*. IEEE. 2011, pp. 661–665.

[140] Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. “Yago: A large ontology from Wikipedia and wordnet”. In: *Web Semantics: Science, Services and Agents on the World Wide Web* 6.3 (2008), pp. 203–217.

[141] Niek Tax, Sander Bockting, and Djoerd Hiemstra. “A cross-benchmark comparison of 87 learning to rank methods”. In: *Inf. Proc. Manag.* (2015).

[142] Carol Tenopir et al. “Reading patterns and preferences of pediatricians”. In: *Journal of the Medical Library Association* (2007).

[143] Ornuma Thesprasith and Chuleerat Jaruskulchai. “Query expansion using medical subject headings terms in the biomedical documents”. In: *Asian Conference on Intelligent Information and Database Systems*. Springer. 2014, pp. 93–102.

[144] Elaine G Toms and Celeste Latter. “How consumers search for health information”. In: *Health Informatics Journal* 13.3 (2007), pp. 223–235.

[145] Ozlem Uzuner, Yuan Luo, and Peter Szolovits. “Evaluating the state-of-the-art in automatic de-identification”. en. In: *J. Am. Med. Inform. Assoc.* 14.5 (Sept. 2007), pp. 550–563.

[146] Ozlem Uzuner et al. “Identifying patient smoking status from medical discharge records”. en. In: *J. Am. Med. Inform. Assoc.* 15.1 (Jan. 2008), pp. 14–24.

[147] Özlem Uzuner et al. “2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text”. In: *Journal of the American Medical Informatics Association* 18.5 (2011).

[148] Hamed Valizadegan et al. “Learning to rank by optimizing nDCG measure”. In: *NIPS*. 2009.

[149] Christina R Vargas et al. “Readability of online patient resources for the operative treatment of breast cancer”. In: *Surgery* 156.2 (2014), pp. 311–318.

[150] Ellen M Voorhees and William R Hersh. “Overview of the TREC 2012 Medical Records Track.” In: *TREC*. 2012.

[151] Ellen M Voorhees and Richard M Tong. “Overview of the TREC 2011 Medical Records Track.” In: *TREC*. 2011.

[152] Liupu Wang et al. “Using Internet search engines to obtain medical information: a comparative study”. In: *Journal of medical Internet research* 14.3 (2012), e74.

[153] Ryen W. White, Susan Dumais, and Jaime Teevan. “How Medical Expertise Influences Web Search Interaction”. In: *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR ’08. Singapore, Singapore: ACM, 2008, pp. 791–792.

[154] Ryen W White and Eric Horvitz. “Cyberchondria: studies of the escalation of medical concerns in web search”. In: *ACM Transactions on Information Systems (TOIS)* 27.4 (2009), p. 23.

[155] Qiang Wu et al. “Adapting boosting for information retrieval measures”. In: *IR Journal* (2010).

[156] Jun Xu and Hang Li. “Adarank: a boosting algorithm for information retrieval”. In: *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM. 2007, pp. 391–398.

[157] Tan Xu, Paul McNamee, and Douglas W Oard. “HLTCOE at TREC 2014: Microblog and Clinical Decision Support”. In: (2014).

[158] Yang Xu, Fan Ding, and Bin Wang. "Entity-based Query Reformulation Using Wikipedia". In: *Proceedings of the 17th ACM Conference on Information and Knowledge Management*. CIKM '08. Napa Valley, California, USA: ACM, 2008, pp. 1441–1442.

[159] Andrew Yates and Nazli Goharian. "ADRTrace: Detecting Expected and Unexpected Adverse Drug Reactions from User Reviews on Social Media Sites". In: *Proceedings of the 35th European conference on Advances in Information Retrieval (ECIR'13)*. 2013.

[160] Andrew Yates and Nazli Goharian. "ADRTrace: detecting expected and unexpected adverse drug reactions from user reviews on social media sites". In: *Advances in Information Retrieval*. Springer, 2013, pp. 816–819.

[161] Andrew Yates, Nazli Goharian, and Ophir Frieder. "Extracting Adverse Drug Reactions from Social Media." In: *AAAI*. 2015, pp. 2460–2467.

[162] Andrew Yates, Nazli Goharian, and Ophir Frieder. "Learning the Relationships between Drug, Symptom, and Medical Condition Mentions in Social Media." In: *ICWSM*. 2016, pp. 739–742.

[163] Andrew Yates, Nazli Goharian, and Ophir Frieder. "Relevance-ranked domain-specific synonym discovery". In: *Advances in Information Retrieval*. Springer, 2014, pp. 124–135.

[164] Emine Yilmaz, Evangelos Kanoulas, and Javed A Aslam. "A simple and efficient sampling method for estimating AP and NDCG". In: *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM. 2008, pp. 603–610.

[165] Elad Yom-Tov, Luis Fernandez-Luque, and L Luque. "Information is in the eye of the beholder: Seeking information on the MMR vaccine through an internet search

engine". In: *Proceedings of the 2014 conference of the American Medical Informatics Association*. 2014.

[166] Elad Yom-Tov and Evgeniy Gabrilovich. "Postmarket drug surveillance without trial costs: discovery of adverse drug reactions through large-scale analysis of web search queries". In: *Journal of medical Internet research* 15.6 (2013), e124.

[167] Elad Yom-Tov, Ryen W White, and Eric Horvitz. "Seeking insights about cycling mood disorders via anonymized search logs". In: *Journal of medical Internet research* 16.2 (2014), e65.

[168] Elad Yom-Tov et al. "Automatic identification of Web-based risk markers for health events". In: *Journal of medical Internet research* 17.1 (2015).

[169] Elad Yom-Tov et al. "Differences in physical status, mental state and online behavior of people in pro-anorexia web communities". In: *Eating behaviors* 22 (2016), pp. 109–112.

[170] Elad Yom-Tov et al. "Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval". In: *Proceedings of SIGIR '05*. ACM. 2005, pp. 512–519.

[171] Elad Yom-Tov et al. "The Effect of Limited Health Literacy on How Internet Users Learn About Diabetes". In: *Journal of Health Communication* 21.10 (2016), pp. 1107–1114.

[172] H Peyton Young and Arthur Levenglick. "A Consistent Extension of Condorcet's Election Principle". In: *SIAM Journal on Applied Mathematics* 35.2 (1978), pp. 285–300.

[173] Qing T Zeng et al. "Exploring lexical forms: first-generation consumer health vocabularies". In: *AMIA Annual Symposium*. 2006.

[174] Qing T Zeng et al. “Positive attitudes and failed queries: an exploration of the conundrums of consumer health information retrieval”. In: *International journal of medical informatics* 73.1 (2004), pp. 45–55.

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

[176] Hualong Zhang and Liting Liu. “NKU at TREC 2016: Clinical Decision Support Track”. In: (2017).

[177] Yan Zhang. “Beyond quality and accessibility: Source selection in consumer health information searching”. In: *Journal of the Association for Information Science and Technology* 65.5 (2014), pp. 911–927.

[178] ZH Zheng et al. “Applying Probabilistic Thematic Clustering for Classification in the TREC 2005 Genomics Track.” In: *TREC*. 2005.

[179] Dongqing Zhu et al. “Using large clinical corpora for query expansion in text-based cohort identification”. In: *Journal of biomedical informatics* 49 (2014), pp. 275–281.

[180] Guido Zuccon, Bevan Koopman, and João Palotti. “Diagnose This If You Can”. English. In: *Advances in Information Retrieval*. Ed. by Allan Hanbury et al. Vol. 9022. Lecture Notes in Computer Science. Springer International Publishing, 2015, pp. 562–567.

[181] Guido Zuccon et al. “The IR task at the CLEF eHealth evaluation lab 2015 User-centred Health Information Retrieval”. In: *CLEF 2016 Evaluation Labs and Workshop: Online Working Notes*. CLEF. CEUR-WS, Sept. 2016.

[182] Guido Zuccon et al. “The IR task at the CLEF eHealth Evaluation Lab 2016: User-centred health information retrieval”. In: CLEF. 2016.