Entity Based Query Processing For Retrieval And Summarization In Biomedical Domain

by

JAINISHA SANKHAVARA
201521004

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY
to

DHIRUBHAI AMBANI INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGY

December, 2021

Declaration

I hereby declare that

i) the thesis comprises of my original work towards the degree of Doctor of Philosophy at Dhirubhai Ambani Institute of Information and Communication Technology and has not been submitted elsewhere for a degree,

ii) due acknowledgment has been made in the text to all the reference material used.

Jainisha Sankhavara

Certificate

This is to certify that the thesis work entitled ENTITY BASED QUERY PROCESSING FOR RETRIEVAL AND SUMMARIZATION IN BIOMEDICAL DOMAIN has been carried out by JAINISHA SANKHAVARA for the degree of Doctor of Philosophy at *Dhirubhai Ambani Institute of Information and Communication Technology* under my supervision.

Prasenjit Majumder
Thesis Supervisor

Acknowledgments

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout my research work to complete the research successfully.

I would like to express my deep and sincere gratitude to my research supervisor, Dr. Prasenjit Majumder, Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, for giving me the opportunity to do research and providing invaluable guidance throughout this research. His dynamism, vision and motivation have deeply inspired me. He has taught me the methodology to carry out the research and to present the research works as clearly as possible. It was a great privilege and honor to work and study under his guidance. I am extremely grateful for what he has offered me. I am extremely grateful to my parents for their love, prayers, caring and sacrifices for educating and preparing me for my future. I am very much thankful to my husband for his love, understanding, prayers and continuing support to complete this research work. I would like to say thanks to my friends and research colleagues, Prof. Parth Mehta, Prof. Sandip Modha, Jankhana Goswami, Dr. Milind Padalkar, and Dr. Sumukh Bansal for their constant encouragement. Finally, my thanks go to all the people who have supported me to complete the research work directly or indirectly.

Jainisha Sankhavara

Contents

Abstract **vii**

List of Tables **viii**

List of Figures **x**

1 Introduction **1**

1.1 Query processing in information retrieval and text summarization 3

1.2 Community identified challenges in biomedical information retrieval and summarization 7

1.3 Unified Medical Language System 8

1.4 Query reformulation in biomedical information retrieval systems 10

1.4.1 Query reformulation using feedback document discovery 10

1.4.2 Query reformulation using UMLS in retrieval 12

1.5 Query reformulation for biomedical text summarization 12

1.5.1 Query-focused text summarization 13

1.5.2 Query reformulation using UMLS in summarization 14

1.6 Main contributions 14

1.7 Thesis organization 15

2 Literature survey **17**

2.1 Biomedical entity identification 17

2.1.1 Biomedical Entity Identification task 18

2.1.2 Community challenges and resources for entity identification . . . 19

2.1.3 Approaches to biomedical entity identification 20

2.2 Query reformulation for biomedical document retrieval 25

2.2.1 Feedback document discovery for query expansion 25
2.2.2 Query reformulation using UMLS 26
2.3 Query reformulation for biomedical text summarization 28

3 Feedback Document Discovery based query reformulation in retrieval 30
3.1 Automatic Query Expansion With Pseudo Relevance Feedback & Relevance Feedback 30
3.2 Partial Relevance Feedback for Query Expansion 41
3.3 Feedback Documents Discovery based Query Reformulation 46
3.3.1 Feedback document discovery using classification 47
3.3.2 Feedback document discovery using classification and clustering . 48
3.3.3 Experiments 50
3.4 Feature weighting in finding feedback documents for query expansion . . 53
3.4.1 Entity based feature weighting 53
3.4.2 Experiments and Results 55
3.5 Learning To Rank 57
3.6 Query expansion using topic modeling 59
3.7 Conclusion 60

4 UMLS graph based query reformulation in retrieval 62
4.1 UMLS Concepts Based Query Reformulation 62
4.2 Query specific graph based query reformulation using UMLS 64
4.2.1 Graph creation using UMLS 65
4.2.2 Graph refinement using pseudo relevant documents 66
4.2.3 Importance value of the entities in query-specific graph 68
4.2.4 Query reformulation using weighted entities in query-specific graph 71
4.3 Experiment details 72
4.3.1 Dataset and Evaluation Metrics 72
4.3.2 Experimental Setup 73
4.4 Results 73
4.5 Conclusion 81

5 Query-focused biomedical text summarization using UMLS graph **83**

5.1 Summarization Methods with query modifications 83

5.1.1 TextRank and LexRank 83

5.1.2 Query-Sentence matching 84

5.1.3 UMLS graph based query-sentence matching 85

5.1.4 UMLS query graph based lexrank 85

5.1.5 Lexrank with Word2Vec similarity 86

5.1.6 UMLS query graph based lexrank with Word2Vec similarity . . . 87

5.2 Experiments and Results 88

5.3 Conclusion 94

6 Conclusion and future direction **95**

References **97**

Appendix A Publications **114**

Abstract

Exponential growth of biomedical literature poses different challenges in searching. To address complex information needs of the users, rigorous semantic processing of biomedical text is required. Biomedical information access emerges out as a new discipline for this reason. Traditional information access methods of matching, ranking, entity processing, entity-entity relationship processing, etc. are challenged in this domain. These are the major building blocks used to frame queries that represent complex information need in the area of biomedical and clinical information access. This thesis aims to do query processing using different IR and bioNLP techniques and to study their effects in retrieval and summarization.

Various techniques of biomedical query reformulations are carried out and compared for biomedical document retrieval. Query expansion is one query reformulation technique which was carried out using relevance feedback and pseudo relevance feedback for biomedical document retrieval. Relevance feedback approach uses information regarding actual relevant documents to the query for feedback while pseudo relevance feedback approach does not have such information and uses top retrieved documents for feedback as they are assumed to be relevant to the query. One combined approach of relevance feedback and pseudo relevance feedback has been proposed which is based on feedback document discovery and uses various classification and clustering techniques on biomedical documents to identify good document for feedback. This approach uses relevance feedback for a number of documents and tries to learn relevance for other documents for feedback. This feedback document discovery based query expansion approach shows improvement over relevance feedback based query expansion technique for biomedical document retrieval. An improved version of this feedback document discovery based query expansion approach where the features of entities are weighted based on the type of the entities and query is also

proposed which shows improvement of the document retrieval system over the previous one without feature weighting.

Automatic query expansion techniques based on feedback relies on two feedback sources: feedback documents selection and feedback terms selection. In biomedical domain, medical entities are more meaningful than surface words. Therefore the entity based processing is necessary for any application in this domain. This thesis also includes a survey on advances in biomedical entity identification which includes biomedical entity identification process, various community identified challenges in the area, various resources available, approaches for biomedical entity identification and comparison of various techniques proposed in the literature for biomedical entity identification. UMLS is one biomedical resource which brings together many health and biomedical vocabularies and standards. UMLS contains biomedical entities with categorization and their relations with semantic information. A novel query expansion technique which uses knowledge from UMLS for feedback term selection is proposed where the queries are expanded using biomedical entities. The proposed method considers UMLS entities from a query with their related entities identified by UMLS and constructs query specific graph of biomedical entities for term selection. This query reformulation approach shows improvement over pseudo relevance feedback and state-of-the-art UMLS based query reformulation approaches.

The amount of information for clinicians and clinical researchers is growing exponentially. These documents are long and number of topical documents are more. To synthesize the documents, text summarization attempts to reduce text so that the users can quickly understand relevant source information. In the biomedical domain, various summarization techniques are developed in recent years. Text summarization may be useful to medical practitioners with their information and knowledge management tasks. In this work we focus on query-focused biomedical text summarization where the summary should be related to the query. The entity-based processing is incorporated in the summarization process along with word-embedding based similarity. The aim of this work is to use query reformulation in the summarization and see how it affects the summaries, whether expanded queries help to get better summaries.

List of Tables

1.1 Community challenges for biomedical document retrieval and summarization over the years 2013-21. 8

2.1 Community challenges for biomedical entity recognition over the years 2004-17. 19

2.2 Comparison of biomedical entity identification approaches on JNLPBA04 dataset over the years 2004-2018. 22

3.1 TREC Clinical Decision Support (CDS) track DATA statistics. 31

3.2 Example queries from CDS 2015 dataset. 32

3.3 Results (MAP) of Query Expansion with PRF and RF. 34

3.4 Results (infNDCG) of Query Expansion with PRF and RF. 35

3.5 Results of PRF and RF based query expansion with different number of feedback documents on CDS 2014. 36

3.6 Results of PRF and RF based query expansion with different number of feedback documents on CDS 2015. 37

3.7 Results of PRF and RF based query expansion with different number of feedback documents on CDS 2016. 38

3.8 MAP results of Query Expansion with partial relevance feedback. 43

3.9 InfNDCG results of Query Expansion with partial relevance feedback. . . 43

3.10 Results of feedback document discovery using different classifiers on CDS 2014 dataset. 50

3.11 Results of feedback document discovery using different classification and clustering on CDS 2014 dataset. 51

3.12 Results of feedback document discovery on CDS 2015 and CDS 2016 dataset. 52

3.13 Results of feedback document discovery with feature weighting on CDS 2014. 55
3.14 Results of feedback document discovery with feature weighting on CDS 2015. 56
3.15 Results of feedback document discovery with feature weighting on CDS 2016. 56
3.16 Results of Learning to Rank with various features. 58
3.17 Results of Learning To Rank with pseudo judgements. 58
3.18 Results of query expansion using topic modeling on CDS 2014 dataset. . 59
3.19 Results of combining topic modeling with pseudo relevance feedback on CDS 2016 dataset. 60

4.1 Results of UMLS concepts based query processing. 63
4.2 InfNDCG results of UMLS query-specific graph based query reformulation using various weighting techniques. The highest results for each dataset are given in bold. 74
4.3 InfNDCG results of UMLS query-specific query-specific graph based query reformulation on CDS 2015 and CDS 2016 datasets. 75
4.4 Results of UMLS query-specific graph based query reformulation on CDS 2015. 77
4.5 Results of UMLS query-specific graph based query reformulation on CDS 2016. 77
4.6 Query category wise infNDCG results of UMLS query-specific graph based query reformulation on CDS 2015. 78
4.7 Query category wise infNDCG results of UMLS query-specific graph based query reformulation on CDS 2016. 79

5.1 ROUGE-2 Recall results of query-focused summarization on BIOASQ5. . 89
5.2 ROUGE-SU4 Recall results of query-focused summarization on BIOASQ5. 90
5.3 ROUGE-2 F-measure results of query-focused summarization on BIOASQ5. 91
5.4 ROUGE-SU4 F-measure results of query-focused summarization on BIOASQ5. 91
5.5 ROUGE-WE results of query-focused summarization on BIOASQ5. . . . 93

List of Figures

1.1 Various subdomains integrated in UMLS. 9
1.2 A subset of UMLS semantic network. 9

2.1 UMLS concepts with semantic types from UMLS semantic network. . . . 26

3.1 Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over BM25 on CDS 2014. . 39
3.2 Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over In_expC2 on CDS 2014. 39
3.3 Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over BM25 on CDS 2015. . 39
3.4 Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over In_expC2 on CDS 2015. 40
3.5 Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over BM25 on CDS 2016. . 40
3.6 Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over In_expC2 on CDS 2016. 40
3.7 Partial Relevance Feedback: Trade-off between PRF and RF. 42
3.8 No. of feedback documents vs retrieval performance(MAP, infNDCG) plot for partial relevance feedback based query expansion on CDS 2014. . . . 44
3.9 No. of feedback documents vs retrieval performance(MAP, infNDCG) plot for partial relevance feedback based query expansion on CDS 2015. . . . 44
3.10 No. of feedback documents vs retrieval performance(MAP, infNDCG) plot for partial relevance feedback based query expansion on CDS 2016. . . . 44
3.11 Effect of k on retrieval performance(MAP and infNDCG) on CDS 2014. . 45

3.12 Effect of k on retrieval performance(MAP and infNDCG) on CDS 2015. . 45
3.13 Effect of k on retrieval performance(MAP and infNDCG) on CDS 2016. . 45
3.14 Query wise performance difference between feedback document discovery and relevance feedback in terms of infNDCG. 52

4.1 A subset of initial graph constructed for query “A 78 year old male presents with frequent stools and melena”. 66
4.2 Refined graph with edge weights for query “A 78 year old male presents with frequent stools and melena". 67
4.3 Graph with node weights assigned using PageRank algorithm for query “A 78 year old male presents with frequent stools and melena”. 69
4.4 Graph with normalized weighted degree weights for query “A 78 year old male presents with frequent stools and melena”. 71
4.5 Query wise difference graph between UMLS_graph_norm_WDC+PRF and BM25+PRF for CDS 2016. 80
4.6 Distribution of relation types and semantic group types of entities in ‘diagnosis’, ‘test’ and ‘treatment’ queries of CDS 2016. 80
4.7 Distribution of relation types and semantic groups of entities in the improved queries and degraded queries of CDS 2016. 81

5.1 Query wise difference in the results of lexrank_UMLS_querygraph with baseline results lexrank and distribution of types of the queries. 92

CHAPTER 1

Introduction

Retrieving relevant information from biomedical text is a challenging area of research in information access. Thousands of articles are being added to biomedical literature each year and this large collection of publications can be useful for finding hidden biomedical knowledge by applying information retrieval (IR) and Natural Language Processing (NLP) technologies. The tasks of named entity recognition and relation and event extraction, summarization, question answering, and literature based discovery are outlined in Biomedical text mining: a survey of recent progress [129]. Biomedical text processing seeks special attention due to the characteristics of biomedical terminologies. Major challenges in the biomedical domain are handling complex, ambiguous, inconsistent medical terms and their ad-hoc abbreviations. Many medical terms are very complex. The average length of biomedical entities is much higher than other entities, which makes entity identification tasks difficult in the biomedical domain. Medical entity extraction, normalization, and relationship extraction are themselves research problems. They may help to develop better retrieval and ranking of documents for medical search systems, biomedical text summarization, biomedical text data visualization and other biomedical applications. There are various types of biomedical queries: short questions, medical case reports, medical case narratives, verbose medical queries, community questions and semi-structured queries.

Short questions:

Ex. HIV and the GI tract, recent reviews,
can radiation therapy cause a delayed pericardial effusion?

Verbose medical queries:

Ex. Describe the procedure or methods for normalization procedures that are used for microarray data,

Provide information about the role of the gene Apolipoprotein E (ApoE) in the disease Alzheimer's Disease

Semi-structured queries:

Ex. <disease>Colon cancer</disease>
<gene>KRAS (G13D), BRAF (V600E)</gene>
<demographic>52-year-old male</demographic>
<other>Type II Diabetes, Hypertension</other>

Medical case reports:

Ex. 76-year-old female with pmh of diastolic CHF, atrial fibrillation on coumadin, presenting with Hct 16.9 and shortness of breath. She had routine labs drawn yesterday at her PCP's office. Once her hematocrit came she was called and instructed to come to the ED. She is also reporting progressive shortness of breath worse with exertion over the past two weeks. She denies fevers, chills, chest pain, palpitaitons, cough, abdominal pain, constipation or diahrrea, melena, blood in her stool, dysuria, rash. She reports orthopnea.

In the ED: vitals were 98.4 131/49, 60 24 100% 2L. ekg with NSR, twi in V1, no significant change from previous. Repeat CBC showed Hct 16.1 with haptoglobin < 20, and elevated LDH to 315. In addition, her guaiac was reported as being positive.

Past medical history:

Hypertension

Atrial flutter/fibrillation, s/p cardioversion [**2797-1-27**]

Diastolic heart failure

Hysterectomy

Bilateral hip replacements

Social History:

Married for 53 years with four children. She is retired from the airport. She does not smoke or drink.

Occupation: retired from airport

Drugs: denies

Tobacco: denies any history

Alcohol: denies

Medical case narratives:

Ex. An 82 man with chronic obstructive pulmonary disease, status-post bioprosthetic atrial valve replacement for atrial stenosis, atrial fibrillation with cardioversion, right nephrectomy for renal cell carcinoma, colon cancer status-post colectomy, presents with 9 day history of productive cough, fever and dyspnea.

Community Questions:

Ex. I have some black patches on my back of the head and 2 patches on my neck area. Initially they were small but now they have become large. It concerns me a lot. What could be the reason for this problem? Please suggest me some solution.

This diverse nature of medical data introduces various challenges for biomedical information retrieval and summarization systems. The queries searched on the web are either short or verbose queries. Medical case reports and medical case narratives are the notes generated at the time of admission of the patient in the hospital. Retrieval system which can directly relate biomedical literature documents to such notes will be very much usefull to the medical practitioner to better diagnose the patient by referring to the retrieved literature articles related to patient's case report.

1.1 Query processing in information retrieval and text summarization

The classic information retrieval models have their own ways of processing queries and matching the documents. The Boolean retrieval model [115] has the Boolean queries, which include Boolean operators AND, OR, NOT and the model finds the documents where those query terms are present. The extended Boolean retrieval models also include term proximity operators through which users can specify that the two terms of the query must

occur close to each other in the document [118]. The closeness here is considered as the number of intervening words or sentences allowed.

In contrast with the Boolean retrieval model, ranked retrieval models process free text queries rather than using operators in the queries. It finds the documents which are the best match to the query by ranking them based on the similarity. The Boolean model only considers the presence or absence of query terms and retrieves a set of matching documents. There is no scale for grading or ranking those documents. However, the ranked retrieval model considers the frequency of terms in the documents, scores the documents based on the query terms present, and ranks them based on the score. The document scores are determined based on how good match a document is to a query. The most popular term weighting scheme is tf-idf which is based on term frequency and inverse document frequency [116]. The tf-idf weight to term t in document d is given by:

$$(tf\text{-}idf)_{t,d} = tf_{t,d} \times idf_t$$

This tf-idf value is higher when the term occurs frequently within a few documents which discriminates those documents from others. The value is lower when the term rarely occurs in the document, or it occurs in most of the documents. When the term t is present in all the documents, it gets the lowest tf-idf value. Thus, the scoring function is given by the sum, over all query terms, of the tf-idf weights of each query term t in document d.

$$score(q,d) = \sum_{t \in q} (tf\text{-}idf)_{t,d}$$

The most popular ranked retrieval model is vector space model [119] where the queries and the documents are represented as vectors in a common vector space. Each component of the vector corresponds to each term in the dictionary with some weight, and it loses the relative ordering of the terms in the documents. For dictionary terms that are not present in the document, the weights for those terms are zero. The weights for present terms are usually computed as tf-idf weights. However, a number of weighting schemes are there for the vectors. The standard way to compute the similarity between two documents is to get the cosine similarity of their vector representations and normalize it by the length of the documents.Considering query as a small document and representing it as a vector, the

cosine similarity between the normalized query vector and a normalized document vector can be computed as:

$$sim(q,d) = \frac{\vec{V}(q) \cdot \vec{V}(d)}{|\vec{V}(q)|\,|\vec{V}(d)|}$$

This similarity is being used as a measure to score the documents for the query. The resulting scores can then be used to select the top-matching documents. The vector space model assumes terms are statistically independent and hence it loses the order of the terms in the document.

The language modeling [131] approach to IR provides a different view for document ranking. It considers context to similar words and phrases using the surrounding terms. It builds a probabilistic language model from each document, and the documents are ranked based on the probability of generating a query from the model $p(q|M_d)$. This approach models the idea that the document whose model is likely to generate the query is a good match to the query. This is possible if the document contains the query terms often. A general query likelihood language model ranks the documents by the probability of a document $p(d/q)$ which is interpreted as the likelihood of document d relevant to query q:

$$P(d|q) = \frac{P(q|d)\ P(d)}{P(q)}$$

where $p(q)$ is the same for all the documents and $p(d)$ is uniform across all the documents. So $p(q)$ and $p(d)$ both can be ignored and the documents can be ranked by simply $p(q|d)$ i.e., the probability of generating query q from the document d. In a way, we can say that the language modeling approach models the process of query generation and ranks the documents by the probability that the query would be observed as a random sample from the respective document model. Language models require proper smoothing while calculating the probabilities for unseen words. A number of smoothing techniques are there but it is difficult to choose an effective one. Usually, language model approaches use simple unigram model, and still perform better than tf-idf and other ranking approaches. The probabilistic language modeling approach BM25 [111] is being used as a standard baseline in IR research. In our experiments also, we will use BM25 as the baseline result.

Divergence From Randomness (DFR) [5] is a nonparametric probabilistic framework of IR, based on the amount of information in the documents. The DFR models weight

the terms based on the divergence between a random term distribution and the actual term distribution. It models the idea: "The more the divergence of the within-document term-frequency from its frequency within the collection, the more the information carried by the word t in document d. In other words, the term-weight is inversely related to the probability of term-frequency within the document d obtained by a model M of randomness." [137].

$$weight(t|d) = k \times Prob_M(t \in d|Collection)$$

The above formula gives the weight for term t in document d based on divergence from randomness. M represents the model of randomness used to calculate the probability and k is defined by M.

These retrieval models usually suffer due to synonymy and polysemy present in the text. In the text data collections, the same concept is being referred using different words at different places but with the same meaning. That is known as synonymy. Sometimes, the same phrase is being used in different contexts, which have different meanings of the concept. That is known as polysemy. Synonymy and polysemy are the obstacles for any information retrieval system to get all the matching documents for queries. Retrieval models suffer due to such conditions. These problems can be addressed with the help of the query refinement process either fully automatically or by keeping the user in the loop. There are two types of query refinement processes: global methods and local methods. The global query reformulation methods do not use a query or the results returned from the query to get semantically similar terms. Global methods majorly include the corpus or external resources in the process of reformulating query. The local query reformulation methods make use of the query and the related documents matching the initial query. Relevance feedback and pseudo relevance feedback (also known as blind relevance feedback) are the local methods for reformulating the query. Relevance Feedback method involves the user in the retrieval process to improve the final result. The user gives feedback on each document of the initial set of results, whether it is relevant or non-relevant. This user feedback gets incorporated in reformulating the query to get a better representation of the information need and the system gives a revised set of results as final results. The relevance feedback can have more than one iterations also. Relevance feedback technique finds a new query vector $\vec{q}$ that maximizes similarity with user-identified relevant documents and

minimizes the similarity with user-identifies non-relevant documents. If C_r is a set of relevant documents and C_{nr} is a set of non-relevant documents, then

$$\vec{q}_{opt} = argmax_{\vec{q}}[sim(\vec{q}, C_r) - sim(\vec{q}, C_{nr})]$$

Considering cosine similarity, the optimal query vector is:

$$\vec{q}_{opt} = \frac{1}{|C_r|} \sum_{\vec{d} \in C_r} \vec{d} - \frac{1}{|C_{nr}|} \sum_{\vec{d} \in C_{nr}} \vec{d}$$

The optimal query is the vector difference between the centroids of the relevant and non-relevant documents. However, the complete set of relevant documents is not known for any real IR system. So with user query, partial knowledge of relevant and non-relevant documents, the Rocchio algorithm [75] generates a new modified query $\vec{q}_m$ as :

$$\vec{q}_m = \alpha\vec{q} + \beta\frac{1}{|D_r|} \sum_{\vec{d} \in D_r} \vec{d} - \gamma\frac{1}{|D_{nr}|} \sum_{\vec{d} \in D_{nr}} \vec{d}$$

where $\vec{q}$ is the original query vector, D_r is a set of known relevant documents, and D_{nr} is a set of known non-relevant documents. α, β and γ are the weights associated with them. For systems where only positive feedback is considered, γ will be set to 0. For most of the systems, positive feedback turns out to be more important than negative feedback. So in most IR systems, the values of β and γ are set such that $\beta > \gamma$. Relevance feedback in this way can improve precision and recall both for the IR systems.

1.2 Community identified challenges in biomedical information retrieval and summarization

Various community challenges in the area of biomedical information retrieval and biomedical text summarization are listed in Table 1.1.

Table 1.1: Community challenges for biomedical document retrieval and summarization over the years 2013-21.

Year	Challenge	Task
2003-2007	TREC Genomics Track[1]	Ad-hoc retrieval
2014-2016	TREC CDS Track[2]	Clinical Decision Support Track
2017-2020	TREC Precision Medicine Track[3]	Biomedical articles and clinical trials retrieval
2013-2016	CLEF eHealth[4]	Patient-centered Information Retrieval, Cross-lingual Information Retrieval
2014	TAC BiomedSumm Track[5]	Biomedical Summarization Track
2013-2021	BioASQ[6]	Biomedical Semantic QA (IR and summarization)

1.3 Unified Medical Language System

The Unified Medical Language System or UMLS (http://umls.nlm.nih.gov) is a set of files and software that integrates many health and biomedical vocabularies and standards. It distributes key terminology, classification, and coding standards to associated resources for effective and inter-operable biomedical information systems. UMLS can be used to enhance or develop various biomedical applications. This repository of biomedical vocabularies is developed and maintained by the US National Library of Medicine (NLM). 2 million names for some 900k concepts from more than 60 families of biomedical vocabularies are integrated in UMLS, along with 12 million relations among those concepts. UMLS metathesaurus [17] contains terms and codes from many vocabularies, including MeSH, OMIM, NCBI Taxonomy, Gene Ontology, CPT, ICD-10-CM, LOINC, RxNorm, and SNOMED-CT with their hierarchies, definitions, attributes, and relationships. Figure 1.1 shows higher-level view of UMLS composition.

[1] https://dmice.ohsu.edu/trec-gen/
[2] http://www.trec-cds.org/
[3] http://www.trec-cds.org/
[4] https://sites.google.com/site/clefehealth/
[5] https://tac.nist.gov/2014/BiomedSumm/
[6] http://www.bioasq.org/

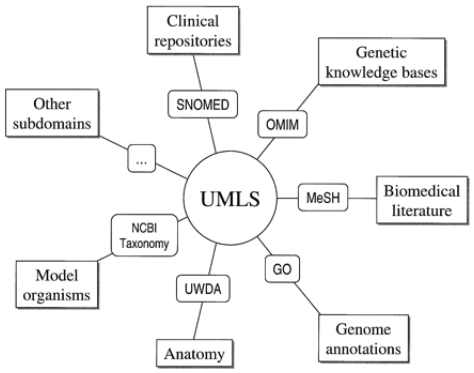

Figure 1.1: Various subdomains integrated in UMLS [19].

In addition to data, the UMLS includes tools for customizing the Metathesaurus, generating lexical variants, creating indexes, and extracting UMLS concepts from the text.

UMLS has a semantic Network that contains broad categories of semantic types used to categorize concepts of metathesaurus along with semantic relations for categorization of relations between concepts. Figure 1.2 shows a subset of UMLS semantic network.

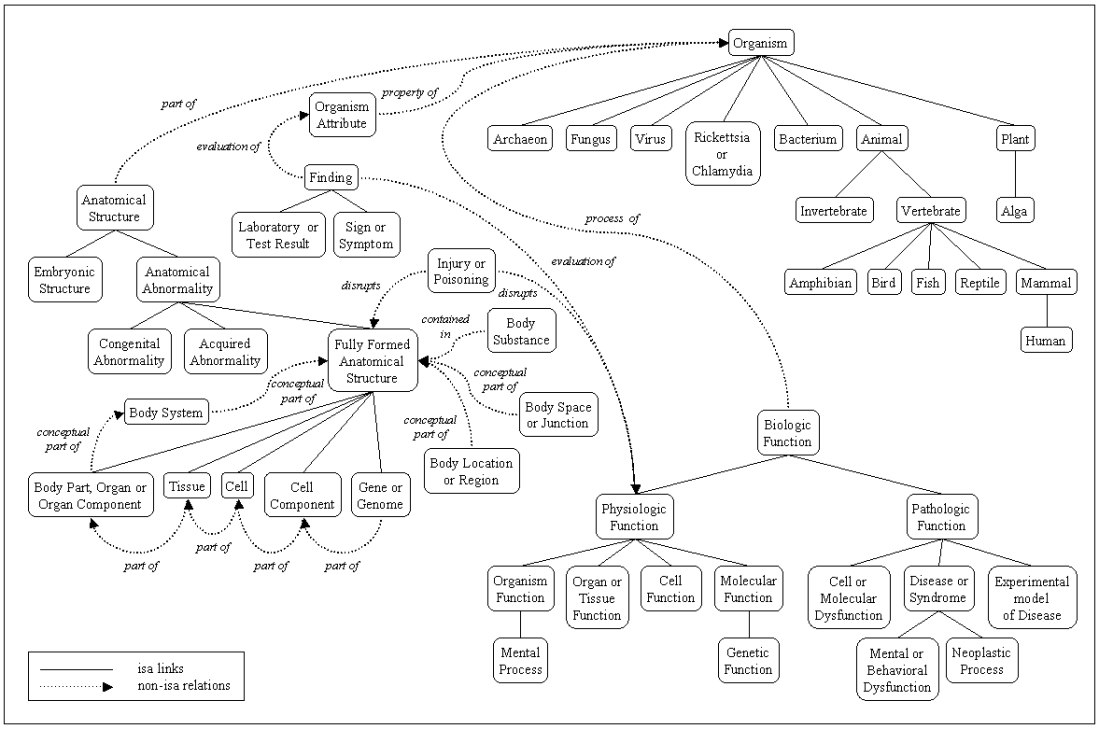

Figure 1.2: A subset of UMLS semantic network [17].

The UMLS has been useful for many applications, including decision support systems, managing patient records, information retrieval (IR), and data mining. To get effective access to knowledge in such applications, National Library of Medicine has devel-

oped a program named Metamap[7] to map the biomedical text to metathesaurus concepts. Using Metamap, users can find UMLS concepts referred in the text. MetaMap uses a knowledge-intensive approach based on symbolic, natural language processing (NLP) and computational linguistic techniques.

1.4 Query reformulation in biomedical information retrieval systems

The vast amount of available biomedical literature makes it difficult to access relevant articles for the specific information need. To facilitate this, a biomedical literature search engine called PubMed[8] is available, which does keyword-based binary matching[112] to access the MEDLINE database of abstracts and references on biomedical topics and life sciences and is maintained by the United States National Library of Medicine (NLM) at the National Institutes of Health (NIH). The biomedical document retrieval system will find the relevant documents from the literature based on medical conditions, medical history, and symptoms. It helps the medical practitioners to diagnose and treat the patient, also to the people who want to know their medical condition before visiting the medical expert. Such document retrieval system usually suffers from term mismatch problem, which is due to multiple synonyms available. Also, the term abbreviations and term inconsistencies obstacle the retrieval system in finding true relevant documents. To overcome the problems of term mismatching, query reformulation is being used in the retrieval systems.

1.4.1 Query reformulation using feedback document discovery

Query reformulation is a process of reformulating the user query to improve retrieval performance in IR systems. Query reformulation process includes adding more words to the query, removing some words from the query, weighting the words according to its importance in the query or all of it. Query reformulation aims to reduce query-document mismatch and retrieve more relevant documents for better system performance. While query expansion focuses on expanding the query using words or phrases with a similar

[7] https://metamap.nlm.nih.gov/
[8] https://www.ncbi.nlm.nih.gov/pubmed/

meaning or with some other statistical relation to the set of relevant documents in order to have better system performance. Many times in the literature, query expansion is also referred as query reformulation as it is a part of query reformulation. We also use query reformulation and query expansion interchangeably.

Automatic query expansion, which has a long history in information retrieval, can be useful in the biomedical domain. It has been seen in the literature that automatic query expansion improves the system performance as compared to no expansion of the queries [25]. There are mainly two techniques to automatic query expansion: Relevance Feedback (RF) and Pseudo Relevance Feedback (PRF). Relevance Feedback based techniques use only relevant documents as feedback documents from top retrieved documents, while Pseudo Relevance Feedback based techniques use all the top retrieved documents as feedback documents. Thus Relevance Feedback techniques require human judgements to identify relevant documents from top retrieved documents, which increases the cost of the retrieval system. The Pseudo Relevance Feedback based techniques do not require any human judgement to identify relevant feedback documents. It assumes that all the top retrieved documents are relevant and uses them as feedback documents. It has been seen that RF based techniques outperform PRF based techniques for biomedical document retrieval [120] but include the cost of human judgements. While PRF based techniques are fully automated and they do not require any expensive external inputs in the retrieval process, they still give a good improvement over original query processing.

Since Relevance Feedback is costly for a large number of top retrieved documents, we attempt to reduce the cost and try to discover good feedback documents automatically. In this thesis, we present feedback document discovery based approaches that learn to identify good feedback documents using a little human intervention. The approach uses human judgements for a small set of feedback documents, and then tries to learn to identify true relevant documents from the rest of the documents. The documents identified relevant are used for feedback, and query expansion is performed. Two approaches for feedback document discovery based on classification and clustering are presented here. Also, we compare the statistical and domain-specific representation of biomedical text in classification as well as clustering modules. In the domain-specific representation, the features are weighted based on the semantic types. We will describe the experiments

performed for feedback document discovery and show that it helps to improve the retrieval results. Domain specific feature weighting in feedback document discovery is helping in finding good feedback documents.

1.4.2 Query reformulation using UMLS in retrieval

Incorporating external biomedical knowledge sources in the retrieval systems is an another direction of research. The emergence of medical domain specific knowledge like UMLS can contribute to better understand the biomedical text from documents and queries for the retrieval system. Various approaches of information retrieval with UMLS Metathesaurus have been reported: some with decline in the results[51] and some with gain in the results[8]. Other biomedical resources like RxNorm and drug dictionary have also been used to expand queries with other related terms in order to improve the representation of a query for relevance estimation [34].

In biomedical information retrieval systems, meaningful query reformulation usually amounts to selecting the right set of entities. Here in this present thesis, we propose a new graph based query reformulation approach which selects expansion terms from the query-specific graph of related entities from UMLS. We also study the effect of using UMLS entities in query processing for clinical decision support systems. The proposed graph-based query processing technique takes advantage of UMLS knowledge resource to generate query-specific graph of related entities, and weights the entities based on the statistics from the feedback documents in the expanded query. We will show that the proposed method can give 4-5% significant improvement in the retrieval results for biomedical documents.

1.5 Query reformulation for biomedical text summarization

Text retrieval and text summarization often gets interconnected in the information systems. In the biomedical domain also, there are systems where the documents are summarized before retrieval and there are systems where the summarization is applied on the documents

after the retrieval. Kan et. al [56] has proposed the use of multidocument summarization as a post-processing step in document retrieval. They examined the use of the summary as a replacement to the standard ranked list and showed that query-based multidocument summarization systems can help retrieval system better match the needs of the searcher. Document summarization is also used in query expansion in the document retrieval process where the document summaries are used to improve term selection. [65] The retrieval documents are first summarized, the expansion terms are selected from the summaries and then retrieval is performed. This query expansion from summaries was more effective than the query expansion from whole documents.

1.5.1 Query-focused text summarization

Automatic text summarization of biomedical text is a promising method for helping clinicians and researchers to efficiently obtain and understand any topic by producing a summary from one or multiple documents. In the biomedical domain, various summarization techniques are developed in recent years. The research has focused on a hybrid technique comprising statistical, language processing, and machine learning techniques [82]. Sometimes, there may exist a query for which the user seeks information and sometimes may not. In the case of query-focused summarization, the generated summary should contain the answer to the query. It is a usual scenario that users want exact answers along with some related details for their medical related queries. Therefore, we are focusing here on query-focused biomedical multi-document summarization which will be helpful to clinicians and all other users who are seeking elaborated answers to their medical related queries.

Query focused summarization and question answering are seen to be helping each other frequently in the literature. An automatic generic document summarization system was coupled with a question-answering system QAAS where summarization system is used as a noise filter as well as summarizer for question answering system [140]. The authors show that the system has been adapted to generate customized summary depending on the specific question. Mori et al. [88] used question answering engine and integrated it with the multidocument summarization system. The sentence importance for summarization was calculated using the scores given by question answering system for responses to multiple

questions. Abstractive summarization evaluation metric based on automatic question-answering is proposed recently which checks for faithfulness by word overlap between answer generated from summary and answer generated from the source document [39] This metric has higher correlation with human faithfulness scores for highly abstractive summaries.

1.5.2 Query reformulation using UMLS in summarization

Summarization systems rely on sentence similarity measures and therefore it is important to incorporate biomedical entity knowledge in the sentence similarity measure. For query focused biomedical text summarization, we explore various techniques of based on sentence-sentence and query-sentence similarity measures. We also consider the query-specific graph based query reformulation using UMLS in the summarization process and incorporate the weights of the expanded entities in the similarity measures. We explore the word-embedding based summarization techniques where sentence-sentence similarity is calculated using the distances between words in the embedding space. Here we study the effect of query expansion and entity based similarity measures for biomedical text summarization systems.

1.6 Main contributions

In this section, we provide a brief overview of the main contributions of this thesis. Query expansion/reformulation is being actively researched upon for a long time. While Biomedical systems require domain specific knowledge manually in some or the other way, query processing in biomedical domain needs to be revisited. This thesis focuses on query reformulation techniques in biomedical applications with minimal human inputs. The primary contributions are as follows:

- We propose a partial relevance feedback technique for query expansion in biomedical document retrieval which is a combination of Relevance Feedback and Pseudo Relevance Feedback. We extensively compare the three techniques varying the the number of documents considered. We see the effect of changing the amount of manual and pseudo inputs in partial relevance feedback.

- Next we propose the feedback document discovery based query reformulation using various classification and clustering techniques. The feedback document discovery approach succeeds in reducing the cost of manual intervention in the feedback process for query expansion.

- We improvise feedback document discovery based query reformulation with feature weighting based on the type of the queries and semantic nature of the terms in documents.

- We study the impact of UMLS on query reformulation. We propose a new query-specific graph-based query reformulation method for biomedical document retrieval. The technique utilizes knowledge resource UMLS with statistical information from the feedback documents. Our proposed technique was able to outperform those query expansion techniques which used UMLS.

- At the end, we propose entity based sentence similarity measures for query-focused biomedical text summarization where we incorporate query reformulation into the existing summarization techniques.

1.7 Thesis organization

This dissertation is organized as below. We discuss the existing approaches for biomedical entity identification, query reformulation for biomedical document retrieval, and query reformulation for biomedical text summarization in the second chapter. In the third chapter, we discuss in detail two particular techniques of automatic query expansion, i.e. Relevance Feedback and Pseudo Relevance Feedback. We highlight how the nature of the two techniques affects the retrieval performance. We also present Partial Relevance Feedback method which combines both Relevance Feedback and Pseudo Relevance Feedback. We then explain the proposed feedback document discovery method for query reformulation. We define it with classification and clustering based learning module incorporated in Partial Relevance Feedback. We enhance this feedback document discovery based query reformulation method with entity based feature weighting where generic types of entities are considered to be more important. We show that feedback document discovery based query

reformulation helps to identify good feedback documents. We also show that the simple feature weighting technique can improve the system performance when incorporated with feedback document discovery. Later, we explore the other framework Learning To Rank which is a supervised approach to document ranking where query expansion is carried out. Learning To Rank framework is also used in an unsupervised way for biomedical domain. We also compare the query expansion using topic modeling with the query expansion using Pseudo Relevance Feedback and show that the feedback based query expansion is more promising.

In the fourth chapter, we discuss query reformulation methods that use UMLS. We present the query expansion using UMLS concepts and MeSH terms. We combine the query expansion technique using UMLS concepts with Pseudo Relevance Feedback as well as Relevance Feedback techniques. We show that the UMLS concepts from queries help to get better retrieval results. We present query expansion using manually identified biomedical entities from queries with Pseudo Relevance Feedback and Relevance Feedback. We then propose query-specific UMLS graph based query reformulation method which leverages biomedical entities from queries and documents. We explain the graph creation method using those entities, graph refinement method using their context and several methods of entity weighting in the graph. We compare the proposed graph based query reformulation method on biomedical datasets and show that it helps to improve the system performance.

In the fifth chapter, we discuss the existing text summarization techniques and compare them in biomedical domain. We present a query-sentence matching based summarization technique for query focused text summarization. We use query reformulation technique based on UMLS in query-focused summarization. We incorporate the weights of the biomedical entities from the reformulated queries into the similarity measures of the summarization process. We also use word-embeddings in the summarization process where the sentence-sentence similarity is defined based on the similarity between entities in the embedding space. We compare these methods on biomedical question answering dataset where the answers are generated using summarization on the biomedical text.

Finally, we conclude the thesis in the last chapter with possible directions for future work.

CHAPTER 2

Literature survey

In this chapter, we compare some of the existing techniques for biomedical entity identification. We discuss the approaches for query reformulation in biomedical document retrieval systems and the effectiveness of UMLS based query reformulation techniques. We also discuss the existing text summarization techniques for query-focused domain-specific summarization.

2.1 Biomedical entity identification

Biomedical entity identification refers to the task of identifying and classifying biomedical terms into predefined categories. It is an essential step for any biomedical natural language processing system. The biomedical entities of major concern are Genes, Proteins, Drugs, temporal expressions, Disease names, etc. As the biomedical knowledge grows, it adds new medical terms and entities to the collection of biomedical entities, making the set of biomedical entities incomplete at any point of time. Therefore, the string matching based algorithms which use the dictionary of entities for Named Entity Identification are no longer useful for biomedical domain as it is difficult to have an exhaustive dictionary. Biomedical Entity Identification task involves:

- Identifying boundaries of the entities
- Assigning a preferred class to the entity
- Getting the preferred name or concept's unique identifier of the entity

These steps are themselves individual tasks, sometimes referred as biomedical entity detection, biomedical entity classification and biomedical entity normalization respectively.

2.1.1 Biomedical Entity Identification task

IDENTIFYING BOUNDARIES OF THE ENTITIES:

Entity Identification in biomedical domain is a challenging task due to the characteristics of medical entities. Biomedical entities are sometimes very complex, for example, 'nuclear factor kappa-light-chain-enhancer of activated B cells' and 'NF-kB DNA binding with electromobility shift assay'. The average length of biomedical entities is much higher which makes the process of identifying boundaries of entities difficult.

BIO [105] is a popular scheme for identifying boundaries of entities. BIO stands for Beginning, Inside, and Outside of the entity. The starting word of the entity is marked as beginning of the entity and subsequent words of the entity are marked as inside of entity. Other non-entity words are marked as outside. BIO tagged representation of the text "nuclear factor kappa-light-chain-enhancer of activated B cells is found in almost all animal cell types" is:

```
"Nuclear/B factor/I kappa-light-chain-enhancer/I of/I
activated/I B/I cells/I is/O found/O in/O almost/O all/O
animal/O cell/O types/O"
```

CLASSIFICATION OF BIOMEDICAL ENTITIES

After identifying the boundaries of the medical entities, a proper class should be assigned to each entity. Biomedical entities are largely classified into the following categories: DNA, RNA, Protein, Cell type, Cell line, Chemicals, Genes, Species, Diseases, Treatments, etc. Physicians often use ad-hoc abbreviations for biomedical entities. They also use acronyms and abbreviations which are ambiguous like 'PSA' can be 'prostate specific antigen' or 'psoriasis arthritis' or 'poultry science association'. The meaning of all three entities is different, but they share the same abbreviation. Thus, the correct expansion of such entity is very important in order to classify the entity correctly.

BIOMEDICAL ENTITY NORMALIZATION

Biomedical terminology changes very rapidly which makes it inconsistent. For instance, 'H1N1 influenza', 'swine influenza', 'SI', 'Pig Flu' and 'Swine-Origin Influenza A H1N1

Virus', all refer to the same entity. Such different representations of the same entity should be mapped to a single entity. Due to these reasons, the biomedical entity identification task implicitly needs entity normalization as a subtask which can also address abbreviations, synonymy and polysemy. It requires external resources to map the entity to its preferred representation or concept. Such knowledge sources include ontologies (ex. Gene Ontology [30]) and semantic networks (ex. UMLS Metathesaurus [19]).

2.1.2 Community challenges and resources for entity identification

Community identified challenges in the area of biomedical entity identification and normalization are listed in Table 2.1.

Table 2.1: Community challenges for biomedical entity recognition over the years 2004-17.

Year	Challenge	Task
2004	BioCreative I	Identification of gene mentions [52, 151]
2006	BioCreative II	Gene mention tagging (GM) [130] & Gene normalization (GN) [87]
2010	BioCreative III	GN: The gene normalization task [72, 73]
2012	BioCreative IV	Chemical and Drug Named Entity Recognition (CHEMDNER) [60, 61]
2013	CLEF eHealth	Task 1: Named entity recognition and normalization of disorders [101]
2014	CLEF eHealth	Disease/Disorder Template Filling [89]
2015	BioCreative V	CHEMDNER patents [63], Chemical-disease relation (CDR) task [144]
2015	CLEF eHealth	Clinical Named Entity Recognition [92]
2016	CLEF eHealth	Multilingual Information Extraction [91]
2017	BioCreative V.5	Chemical Entity Mention recognition (CEMP) [100], Gene and Protein Related Object recognition (GPRO) [100]
2017	CLEF eHealth	Multilingual Information Extraction - ICD10 coding [93]
2019	n2c2	Open Health NLP (OHNLP) shared task on clinical concept normalization [49]

The standard evaluation measures used for Biomedical Entity Identification tasks are Precision, Recall, and F-score. Various datasets, for example, GENIA [58], AIMed [21], JNLPBA04 [59], NCBI disease corpus [37], CHEMDNER corpus [62], GENETAG [134], BioCreative GM [130], i2b2 [141] etc. are used by the community for Biomedical Entity Identification.

2.1.3 Approaches to biomedical entity identification

Biomedical Named Entity Recognition approaches mainly characterized into four groups [6] :

1. Dictionary-based approaches [103, 132, 149] that try to find names of nomenclatures in the literature.

2. Rule-based approaches [3, 66] that manually or automatically construct rules and patterns to directly match words to candidate entities in the literature.

3. Machine learning approaches [156, 108, 113] that use machine learning techniques, such as SVMs [31], CRFs [64], and neural networks to develop statistical models for biomedical entity recognition.

4. Hybrid approaches [125, 138] that merge two or more of the above approaches of Named Entity Recognition (NER).

Dictionary-based approaches use dictionary as a biomedical resource for the matching of entity occurrences directly. It identifies the biomedical entities from the text using string matching based algorithms. Dictionary-based methods utilize a comprehensive list of biomedical terms to identify biomedical entities from biomedical text. This approach highly suffers due to spelling mistakes, morphological variants of entities, and incomplete biomedical resources. To deal with such situations, spelling variations based algorithms and approximate string matching based algorithms have been proposed.

Rule-based approaches use pre-established rules based on the composition pattern of biomedical entities. These approaches need rules to identify biomedical entities; hence we need to define them properly. rule-based approaches give better performance than dictionary-based approaches.

Machine learning based approaches are becoming popular for biomedical entity identification. They use supervised statistical methods to identify entities. These methods are pre-trained on a tagged dataset and learn to identify medical entities from new data. To train these methods, we require gold-standard data created using manual intervention. Machine learning based approaches give better results than dictionary-based and rule-based approaches. There are semi-supervised methods and also methods which create training data with the use of bootstrapping. Classification-based approaches like SVM and sequential approaches like Hidden Markov Model (HMM) [104], Maximum Entropy Markov Model (MEMM) [16], Conditional Random Field (CRF) [64], and Long Short Term Memory (LSTM) [53] are very much favorable for biomedical entity identification. Sequential methods are even better than classification methods. The state-of-the-art biomedical entity recognition models are based on CRF and LSTM.

Given a word sequence $W = w_1, w_2, ..., w_n$ and its label sequence $L = l_1, l_2, ..., l_n$, the conditional probability of a linear chain CRF is given in Equation 2.1.

$$P(L/W, \lambda) = \frac{1}{z} \exp\left(\sum_{i=1}^{n} \sum_{j=1}^{m} \lambda_j f_j(l_i, l_{i-1}, w, i)\right) \tag{2.1}$$

where $f_j(l_i, l_{i-1}, w, i)$ is a feature function; l_i and l_{i-1} refer to current and previous state, respectively; z is a normalization factor shown in Equation 2.2.

$$z = \sum_{l} \exp\left(\sum_{i=1}^{n} \sum_{j=1}^{m} \lambda_j f_j(l_i, l_{i-1}, w, i)\right) \tag{2.2}$$

When any of these dictionary-based, rule-based and machine learning based approaches are combined for biomedical entity identification, they are known as hybrid approaches. Various methods proposed in the literature for biomedical entity identification in JNLPBA04 dataset are compared in Table 2.

GuoDong et al. [47] has explored various deep knowledge resources such as the name alias, the cascaded entity name, dictionary, the alias list LocusLink, abbreviation resolution and POS with SVM and achieved 72.55% F-measure for biomedical NER. Liao et al. [67] has proposed generic classifier ensemble approach using SVM based on the principle that contributing degrees of prediction classes among different classes in the same classifier are different and they also differ among different classifiers. They compared their results with

a single SVM classifier, vote-based SVM-classifier selection, HMM, MEMM and CRF and achieved a maximum F-measure of 77.85% on JNLPBA04 dataset.

Table 2.2: Comparison of biomedical entity identification approaches on JNLPBA04 dataset over the years 2004-2018.

Year	**Method**	**F-measure**
2004	HMM + SVM + deep knowledge resources [47]	72.55
2012	Generic classifier ensemble with SVM [67]	77.85
2012	SVM-CRF [156]	92.59
2013	Gimli [22]	72.23
2013	GA based feature selection for SVM and CRF [41]	75.17
2014	CRF + word representations [135]	71.39
2014	CRF + rules [106]	75.77
2015	CRF + MapReduce [136]	73.31
2015	GA based classifier-ensemble for SVM and CRF [114]	75.97
2015	Deep neutral network [150]	71.01
2016	Bidirectional LSTM (character + words) [108]	72.70
2017	BLSTM + WE + char + dropout + CRF [46]	75.87
2018	Bidirectional LM + transfer learning [113]	75.03

Zhu et al. [156] has used SVM to separate biological terms from non-biological terms and CRFs to determine the types of biological terms. Their proposed hybrid approach SVM-CRF has surprisingly achieved F-measure of 92.59% on JNLPBA04 data and 97.48% on GENIA data. An open-source tool, Gimli [22] implements a machine learning technique CRF with a rich set of features, which include morphological, orthographic, linguistic, and domain knowledge features. It also has a post-processing module that does parentheses correction and abbreviation resolution. Gimli shows 72.23% F-score on JNLPBA04 dataset. Ekbal et al. [41] have used genetic algorithm (GA) in feature selection process for SVM and CRF classifiers with stacked based ensemble to combine the classifiers. On JNLPBA04 dataset and GENETAG dataset, they achieved F-measure values of 75.17% and 94.70%, respectively. Their approach gave 1%-2% increment over best individual classifier, Majority-vote based ensemble and weighted vote-based ensemble.

Tang et al. [135] investigated and combined three different types of word representation features for Biomedical Entity Identification, including clustering-based representation, distributional representation, and word embeddings. Their system achieved F-measure 80.96% 71.39% with 3.75% and 1.39% improvement when compared with the systems using baseline features for BioCreAtIvE II GM and JNLPBA04 corpora, respectively. Raja et al. [106] have combined machine learning based approach with rule-based approach. Their generated post-processing rules were combined with CRF and achieved F-score of 75.77% on JNLPBA04 dataset.

Tang et al. [136] used a parallel optimization framework with CRF for biomedical entity identification, achieving 73.31% F-score with short training time. Saha et al. [114] used single objective optimization based classifier ensemble technique with SVM and CRF which gives F-measure values 75.97% and 95.90%, achieving increments of 1.07% and 0.57% over the individual classifiers for JNLPBA04 and GENETAG dataset, respectively.

Yao et al. [150] used a multilayer neural network to continuously learn the representation of features, achieving 71.01% F-score. Rei et al. [108] has proposed a character level neural model (bidirectional LSTM) in combination with word level model using attention mechanism and achieved 72.70% F-score as compared to F-score 70.75% for word level model. Gridach et al. [46] has achieved F-measure 75.87% using CRF on top of bidirectional LSTM in combination with pretrained word embeddings and character-level embeddings. Sachan et al. [113] has used transfer learning with bidirectional language model for biomedical entity recognition and achieved F-measure 75.03%.

Crichton et al. [32] has studied multi-task learning across 15 biomedical NER datasets using CNN with multiple output layers and observed an average improvement on multi-task learning as compared to single task learning. Ju et al. [55] proposed a neural model to identify nested entities by dynamically stacking NER layers. They used LSTM+CRF as a neural model and this dynamic model achieved F-measure 74.7% and 72.2% on GENIA and ACE2005 dataset, respectively.

Xu et al. [148] has used BiLSTM-CRF model NCBI Disease Corpus and achieved 80.22% F-score for disease named entity identification. Xu et al. [147] has proposed to combine disease dictionary using bidirectional LSTM and CRF with a dictionary attention layer for disease named entity recognition. Zeng et al. [153] showed the effect of bidirec-

tional LSTM and CRF with word embedding and character embedding for drug named entity recognition.

A comprehensive study [45] of dictionary-based concept recognition approaches for biological entities (genes, proteins, chemicals, etc) including MetaMap, NCBO Annotator, and ConceptMapper show that MetaMap has f-measure performance of 0.8 which is highest after ConceptMapper while experimenting on 8 different ontologies.

Among these dictionary-based, rule-based, machine learning based and hybrid approaches, machine learning and hybrid approaches are more popular in the research community. Combined approaches based on neural networks and sequential machine learning methods are outperforming other techniques on most biomedical entity recognition datasets.

There is a very popular program called MetaMap [7] which is developed by National Library of Medicine (NLM) to identify UMLS concepts form the text. It uses a knowledge-intensive approach based on symbolic, natural language processing (NLP) and computational linguistic techniques. Metamap has proved to be useful for many applications including decision support systems, management of patient records, information retrieval (IR) and data mining. It uses SPECIALIST minimal commitment parser and Xerox part-of-speech tagger to identify lexicons and assign syntactic tags. Later it generates variants for the lexicons with all its synonyms, acronyms, abbreviations, derivational variants, their combinations and spelling variants. Metathesaurus candidates are retrieved based on the string matching and evaluates them against the input text by calculating the strength of the mapping.

The community challenges listed in Table 2.1 focused on entity/concept recognition and normalization, separately. The challenge ShARe/CLEF eHealth 2013 Task 1 had two subtasks: Task 1a which focused on disorder mention identification/recognition based on Unified Medical Language System (UMLS) definition and Task 1b which was on disorder mention normalization to an ontology. Similarly, n2c2 shared task only focused on clinical concept normalization. By entity recognition, it means to identify existing entities in the text while the process of normalization involves mapping these entities to their normalized representation. The ShARe/CLEF eHealth 2013 Task 1b focused on mapping disorder mentions to the closest equivalent Unified Medical Language System28 (UMLS) Concept Unique Identifier (CUI) subset of SNOMED CT. While the shared task n2c2 focused

on mapping the given text span/concept mentions to the corresponding concepts in the SNOMED CT and RxNorm vocabularies from the UMLS. The shared task used the MCN corpus[74] which maps all mentions of problems, treatments, and tests in the 2010 i2b2/VA challenge data[141] to the Unified Medical Language System concepts.

2.2 Query reformulation for biomedical document retrieval

2.2.1 Feedback document discovery for query expansion

Query reformulation based on relevance feedback was first studied by Salton and Buckley [117]. Various feedback models [4, 23] show the effectiveness of query reformulation in various fields. The statistical query expansion techniques for clinical decision support systems with pseudo relevance feedback and fusion of retrieval systems have been reported in [1, 33, 36, 78, 123]. More recently, query expansion using local and global context analysis is studied by Xu and Croft [146]. For clinical decision support retrieval, an approach for pseudo relevance feedback based on proximity information is proposed in [99].

Roberts et al. [110] provides an overview and analysis of state-of-the-art biomedical literature retrieval. Oh and Jung [95] showed query expansion using external collections in biomedical information retrieval. Sankhavara et al. [124] have fused automatic and manual feedback for query expansion in biomedical information retrieval. Stokes et al. [133] discusses the success factors for effective query expansion with respect to various sources of term expansion such as corpus-based co-occurrence statistics, pseudo-relevance feedback methods, and domain-specific and domain-independent ontologies.

One attempt is to learn the truly relevant documents for feedback by using minimum human intervention. We compare partial relevance feedback approach with relevance feedback and pseudo relevance feedback. The proposed feedback document discovery approach uses human judgements for a small set of feedback documents, and then it tries to learn to identify true relevant documents from the rest of the documents. Then the documents identified relevant are used for feedback, and query expansion is performed. Two approaches for this learning based on classification and clustering are presented here. The feedback document discovery approach is modified with entity-based feature weighting

for biomedical document retrieval systems.

2.2.2 Query reformulation using UMLS

Being the largest biomedical resource, UMLS has been used in various fields of biomedical information retrieval and biomedical natural language processing systems. It has broad categories of semantic types and semantic groups. Figure 2.1 shows the partial semantic network of UMLS.

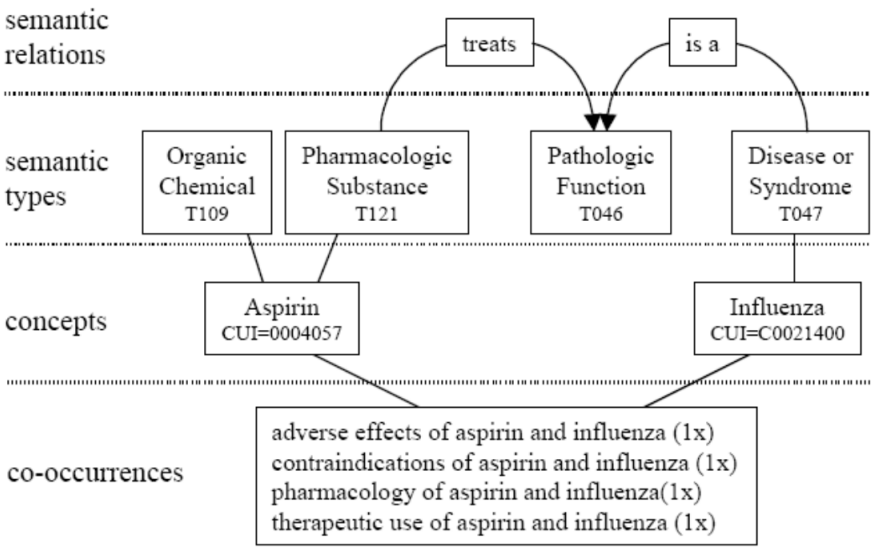

Figure 2.1: UMLS concepts with semantic types from UMLS semantic network. (Image from source[1].)

UMLS has also been used in other fields of research like medical image retrieval and medical question answering [90, 127, 139]. In the biomedical IR field, research is being done on developing conceptual relevance models based on UMLS [11, 29, 69, 143].

The UMLS thesaurus is reasonably used to solve the vocabulary mismatch problem between a query and documents. This problem arises due to insufficient medical knowledge of the users to formulate a query for their information need. To solve this problem, query reformulation techniques based on UMLS have been reported in the literature. Some techniques improve system performance while some techniques degrade it. Aronson and Rindflesch [8] reported query expansion using UMLS, where the optimum relative weights

[1] http://snu-dhpm.ac.kr/pds/files/UMLS%20Applications_%ED%95%9C%EC%8A%B9%EB%B9%88.pdf

for terms, phrases, and concepts were determined from a series of experiments and kept fixed. Hersh et al. [51] used synonym relations and hierarchical relations like parent or child from UMLS for query expansion and showed that it causes a decline in retrieval performance generally but improves it for some queries. Query expansion strategies based on UMLS metathesaurus proposed by Lu and Mu [71] show improvement in short queries while decline in results for long queries on Medline plus dataset.

Demner-Fushman et al. [35] have reviewed recently renewed interest in advanced NLP systems for clinical decision support. Balaneshin-Kordan et al. [12] show effectiveness of Markov Random Fields-based retrieval model and an optimization method for jointly weighting unigrams, ordered bigrams, and unordered bigrams of UMLS concepts. Palotti and Hanbury [98] reformulated queries based on UMLS concepts present in the queries and their triggered names and preferred names. They chose to weight concepts based on exploration manually and then used pseudo relevance feedback based query expansion. Audeh et al. [9] also used UMLS concepts and their preferred names with pseudo relevance feedback. Zhan et al. [155] used UMLS concept's variants to performed pseudo relevance feedback based query expansion and their results are not as good as query expansion using pseudo relevance feedback without using UMLS for query processing. Drosatos et al. [38] used UMLS synonyms while processing queries. Agrafiotes and Arampatzis [2] augmented queries with UMLS atoms and observed that adding larger number of words to query usually leads to decreased performance while adding a few specific words works well. Gurulingappa et al. [48] did query expansion using UMLS concepts filtered by semantic types along with pseudo relevance feedback and reported better results compared to query expansion without UMLS concepts. Wei et al. [145] used UMLS for query expansion where they considered all string variants of concepts from UMLS. Wang et al. [142] used specific types of relations from UMLS and added those concepts while processing the query. Some of these works have not compared their results with any baseline, so it is difficult to see how much they improved. Most of these previous works are limited to the participating teams of TREC CDS tracks.

A bayesian approach to incorporate different types of biomedical knowledge bases into information retrieval systems for clinical decision support in precision medicine has been proposed in [13]. This approach uses UMLS to obtain candidate query expansion

concepts from biomedical knowledge bases and outperforms state-of-the-art baselines for 2017 TREC precision medicine task. McNamee [81] has used UMLS for disease term expansion for article retrieval in precision medicine.

Martinez et al. [76] has proposed query expansion technique which uses personalized pagerank on graph representation of UMLS for electronic health record retrieval, which improved results over baselines, while our proposed method does query expansion based on query-specific graph representation from UMLS for clinical decision support systems, which makes the query reformulation process more specialized to the information need represented in the query and uses statistics from feedback documents. The literature shows that UMLS has been used in various ways to do query expansion and it is proven to be useful in their particular ways. However, there is no generalized method proposed which uses all the features of UMLS knowledge for query expansion.

Zhang and He [154] proposed query expansion using diagnosis predicted from external resources where they used UMLS for medical concepts extraction and reported better results with diagnosis prediction technique for query expansion in the document retrieval for clinical decision support systems. Apart from these, we couldn't find many works that use UMLS in query processing for clinical decision support systems.

2.3 Query reformulation for biomedical text summarization

A lot of research has been carried out in the field of biomedical text summarization. A recent survey on the research in text summarization in biomedical domain highlights that natural language processing and hybrid techniques were prominently used for summarization of multiple documents [82]. Text summarization methods using knowledge resources like UMLS have achieved a lot of interest currently in biomedical domain.

The graph-based summarization using named-entities has been presented as EntityRank algorithm, which considers information about named entities in the process of multi-document graph-based summarization [128]. Their results show that the addition of named-entity information increases the performance of graph-based summarizers in the biomedical domain. Moradi and Ghadiri [85] studied different feature selection approaches

for identifying important concepts in a biomedical text and showed that the concept based summarization method outperforms other frequency-based, domain-independent, and baseline methods.

Query based biomedical text summarization techniques which rely on external ontology knowledge resource UMLS are proposed in the literature. The ontology-based method of biomedical text summarization performed better when compared to keyword-only methods [24, 27, 42, 44, 86]. Sarker et al.[126] observed that an approach for query-focused summarization of medical text based on target-sentence-specific and target-sentence-independent statistics along with domain-specific features outperforms other baseline and benchmark summarization systems.

Text summarization approaches often rely on the similarity measure to model the text documents. Azadani and Ghadiri [10] has studied the impact of the similarity measure on the performance of the summarization methods in biomedical domain and found that exploiting both biomedical concepts and semantic types improvises the quality of summaries.

Here we propose query-specific biomedical text summarization methods, which use ontology knowledge source UMLS [19] to generate a graph of candidate biomedical entities from the query and their semantically connected entities. The importance values of the entities from the query graph are then incorporated in the similarity measure of existing summarization approaches for selecting sentences.

CHAPTER 3

Feedback Document Discovery based query reformulation in retrieval

In this chapter, we explore various query reformulation techniques in biomedical domain. We start with query expansion using pseudo relevance feedback and relevance feedback. We examine their effects and experiment with partial relevance feedback method for query expansion. Later we discuss a feedback document discovery method for query reformulation and it's effect on document retrieval. We will also include feature weighting in feedback document discovery process. A different approach of document retrieval, learning to rank, is also carried with query expansion. This chapter also includes basic experiments of query expansion using topic modeling.

3.1 Automatic Query Expansion With Pseudo Relevance Feedback & Relevance Feedback

Query Expansion (QE) is the process of reformulating a query to improve retrieval performance and efficiency of IR systems. QE is proved to be efficient in the case of document retrieval [25]. It helps to overcome vocabulary mismatch issues by expanding the user query with additional relevant terms and re-weighting all terms. Query Expansion which uses the top retrieved relevant documents is known as Relevance Feedback. It requires human judgment to identify relevant documents from top retrieved documents. In contrast, pseudo Relevance Feedback technique assumes the top retrieved documents to be relevant and uses them as feedback documents. It does not require human input at all. Here we focus on biomedical document retrieval system where biomedical literature articles are retrieved

against queries. First we see the effect of Pseudo relevance feedback and relevance feedback based query processing on the retrieval results. The Query expansion based approaches for biomedical domain gives better results as compared to retrieval without query expansion [123].

For our experiments, we have used the datasets from TREC Clinical Decision Support (CDS) track[1], which contain millions of full-text biomedical articles from PMC (PubMed Central)[2]. The medical case reports are used as queries which are case narratives of patients' medical condition. The retrieval system has to retrieve biomedical articles related to patient's medical case report from the available collection. The statistics of CDS 2014, 2015 and 2016 datasets are given in Table 3.1 below.

Table 3.1: TREC Clinical Decision Support (CDS) track DATA statistics.

Dataset	**CDS 2014**	**CDS 2015**	**CDS 2016**
#Documents	733,138	733,138	1,255,259
Collection size	47.2 GB	47.2 GB	87.8 GB
#Total terms	1,600,536,286	1,600,536,286	2,954,366,841
#Unique terms	3,689,317	3,689,317	4,564,612
#Topics	30	30	30
#Rel. docs/Topic	112	150	182
Query forms	Description, Summary	Description, Summary	Note, Description, Summary
Avg. length of Description (words)	75.8	80.4	119.9
Avg. length of Summary (in words)	24.6	20.4	33.3
Avg. length of Note (in words)	-	-	239.4
Avg. Doc length (in words)	2183	2183	2353

The topics in the datasets are medical case reports which describe information such as a patient's medical history, the patient's current symptoms, tests performed by a physician to diagnose the patient's condition, the patient's diagnosis, and finally, the steps taken by

[1]http://www.trec-cds.org/
[2]http://www.ncbi.nlm.nih.gov/pmc/

a physician to treat the patient. For each case report, there are two versions: description and summary. Description contains all medical details about the patient, while summary is a shorter version of the case report. The topics and relevance judgements address three most common generic clinical question types: 'diagnosis', 'test', and 'treatment'. Some example topic summaries from CDS 2015 dataset are shown in Table 3.2.

Table 3.2: Example queries from CDS 2015 dataset.

No.	Type	Summary
1	diagnosis	A 44-year-old man with coffee-ground emesis, tachycardia, hypoxia, hypotension and cool, clammy extremities.
11	test	A 56-year old Caucasian female presents with sensitivity to cold, fatigue, and constipation. Physical examination reveals hyporeflexia with delayed relaxation of knee and ankle reflexes, and very dry skin.
21	treatment	A 32-year-old male presents with diarrhea and foul-smelling stools. Stool smear reveals protozoan parasites.

The retrieved documents should be helpful to diagnose the patient for diagnosis type of topics, suggest test to be performed for test type of topics and for treatment type of topics, documents should suggest best treatment for the patient described in the topic. The topics for CDS 2014 and CDS 2015 are medical case narratives manually created by experts, while electronic health records of admission notes curated by physicians from the MIMIC-III data were used as topics in CDS 2016 dataset [109].

We first see the effect of standard query expansion in biomedical full-text article retrieval for patient's medical case reports. The initial experiment compares the retrieval results of standard query expansion techniques with the retrieval results without query expansion. The retrieval with original queries is done using retrieval models BM25 [111] and In_expC2 [5]. Query expansion is also experimented considering the same retrieval models. The retrieval model BM25 is a ranking function based on probabilistic retrieval framework, while In_expC2 is a probabilistic model based on Divergence From Randomness (DFR). These models are available in Terrier IR Plateform [96], developed by School of Computing

Science, University of Glasgow. We have used Terrier[3] for all the experiments on document retrieval.

Here, we consider two standard query expansion techniques: Relevance Feedback (RF) and Pseudo Relevance Feedback (PRF). The feedback based query expansion process was described in Section 1.1. The relevance feedback technique requires user's feedback on the initial retrieval results, modifies the initial query using the feedback, and re-retrieves the documents based on the modified query. On the other hand, pseudo relevance feedback does not require user's input for modifying the queries. It assumes that all the top retrieved documents are relevant and considers all of them as relevant documents while selecting expansion terms and modifying the query. The query modification process is same for both relevance feedback and pseudo relevance feedback, while the difference lies in the process of choosing feedback documents. Since relevance feedback requires domain experts and medical experts' availability is sparse and costly, the research question here is "Can we reduce human efforts in relevance feedback?". To address the question, we first examine the gap lying between relevance feedback and pseudo relevance feedback in terms of retrieval performance for biomedical domain.

The experiments of pseudo relevance feedback based query expansion and relevance feedback based query expansion are carried out on CDS 2014, CDS 2015, and CDS 2016 datasets and the results are compared with standard retrieval (without query expansion). The retrieval is done using BM25 and In_expC2 models on the summary part of the query. For pseudo relevance feedback and relevance feedback, the top 10 and top 50 retrieved documents are considered as feedback documents while modifying the queries. For the evaluation, Mean Average Precision (MAP) and Inferred Normalized Discounted Cumulative Gain (infNDCG) [152] are used as the performance measures.

Mean average precision for a set of queries is the mean of the average precision scores for each query which is given by the following:

$$MAP = \frac{\sum_{q=1}^{Q} AP(q)}{Q}$$

[3]http://terrier.org

where Q is the number of queries and AP is the average precision which is given by

$$AP = \frac{\sum_{i=1}^{n} P@i \star rel(i)}{\sum_{i=1}^{n} rel(i)}$$

where, $rel(i)$ is 1 if the item at rank i is a relevant document, zero otherwise. $P@i$ is a precision measure which corresponds to the number of relevant documents among the top i retrieved documents.

The evaluation measure infNDCG is an inferred normalized discounted cumulative gain which is calculated as the following:

$$NDCG_p = \frac{DCG_p}{IDCG_p}$$

where,

$$DCG_p = \sum_{i=1}^{p} \frac{rel(i)}{log_2(i_1)} \quad and$$

$IDCG_p$ = DCG_p of the ideal result list of documents sorted by relevance.

The higher the value of evaluation measure, the better the retrieval result of the system.

MAP and infNDCG results of the experiments are given in Table 3.3 and Table 3.4, respectively.

Table 3.3: Results (MAP) of Query Expansion with PRF and RF.

MAP	**CDS 2014**	**CDS 2015**	**CDS 2016**
BM25	0.1071	0.1147	0.0620
BM25+PRF_{10}	0.1542 (+4.71%)	0.1805 (+6.58%)	0.0769 (+1.49%)
BM25+PRF_{50}	0.1502 (+4.31%)	0.1693 (+5.46%)	0.0800 (+1.80%)
BM25+RF_{10}	0.2050 (+9.79%)	0.1941 (+7.94%)	0.0984 (+3.64%)
BM25+RF_{50}	**0.2768** (+16.97%)	**0.2283** (+11.36%)	**0.1456** (+8.36%)
In_expC2	0.1096	0.1201	0.0632
In_expC2+PRF_{10}	0.1623 (+5.27%)	0.1725 (+5.24%)	0.0754 (+1.22%)
In_expC2+PRF_{50}	0.1580 (+4.84%)	0.1752 (+5.51%)	0.0745 (+1.13%)
In_expC2+RF_{10}	0.2117 (+10.21%)	0.1895 (+6.94%)	0.0992 (+3.60%)
In_expC2+RF_{50}	**0.2587** (+14.91%)	**0.2191** (+9.90%)	**0.1275** (+6.43%)

The result improves with PRF and RF based query expansion giving statistically significant results ($p < 0.05$) as compared to no expansion. Here RF is giving 5-6% more improvement than PRF over no expansion. We argue that biomedical retrieval should be done keeping human in the loop. A small human intervention can give 6% increment to retrieval accuracy in biomedical domain. The results of relevance feedback with 50 documents are better than the results with 10 documents, while the results of pseudo relevance feedback are better with 10 feedback documents than 50 documents.

Table 3.4: Results (infNDCG) of Query Expansion with PRF and RF.

infNDCG	**CDS 2014**	**CDS 2015**	**CDS 2016**
BM25	0.1836	0.2115	0.1710
BM25+PRF_{10}	0.2522 (+6.86%)	0.2830 (+7.15%)	0.2047 (+3.37%)
BM25+PRF_{50}	0.2301 (+4.65%)	0.2658 (+5.43%)	0.2021 (+3.11%)
BM25+RF_{10}	0.3355 (+15.19%)	0.3028 (+9.13%)	0.2428 (+7.18%)
BM25+RF_{50}	**0.4186** (+23.50%)	**0.3478** (+13.63%)	**0.3094** (+13.39%)
In_expC2	0.2002	0.2132	0.1785
In_expC2+PRF_{10}	0.2724 (+7.22%)	0.2734 (+6.02%)	0.2018 (+2.33%)
In_expC2+PRF_{50}	0.2482 (+4.80%)	0.2725 (+5.93%)	0.2076 (+2.91%)
In_expC2+RF_{10}	0.3426 (+14.24%)	0.3015 (+8.83%)	0.2450 (+6.65%)
In_expC2+RF_{50}	**0.4019** (+20.17%)	**0.3390** (+12.58%)	**0.3219** (+14.34%)

To see the effect of the number of documents considered for feedback, query expansion with RF and PRF is performed varying it from 5 to 200. Considering retrieval models BM25 and In_expC2, MAP and infNDCG results for PRF and RF based query expansion with the number of feedback documents from 5 to 200 (5,10, 15, 20, 21, 30, 40, 50, 75, 100, 150 1nd 200) are shown in Table 3.5 for CDS 2014 dataset. Table 3.6 and Table 3.7 show the results of similar experiments on CDS 2015 and CDS 2016 datasets, respectively.

The results of query expansion using PRF and RF are better than the results of original queries without query expansion as shown in Table 3.5, Table 3.6 and Table 3.7 for all three datasets. As we increase the number of feedback documents, the result of RF increases. In the case of PRF, the result increases with the increment in number of feedback documents up to a certain level and starts decreasing later with more feedback documents.

Table 3.5: Results of PRF and RF based query expansion with different number of feedback documents on CDS 2014.

CDS 2014	BM25				In_expC2			
	PRF		RF		PRF		RF	
#docs	MAP	infNDCG	MAP	infNDCG	MAP	infNDCG	MAP	infNDCG
0	0.1071	0.1836	0.1071	0.1836	0.1096	0.2002	0.1096	0.2002
5	0.1513	0.2517	0.1843	0.3033	0.1484	0.2638	0.1909	0.3156
10	0.1542	0.2522	0.2050	0.3355	0.1623	0.2724	0.2117	0.3426
15	0.1442	0.2327	0.2199	0.3508	0.1596	0.2660	0.2226	0.3652
20	0.1514	0.2402	0.2346	0.3724	0.1507	0.2502	0.2378	0.3862
25	0.1518	0.2450	0.2477	0.3932	0.1509	0.2488	0.2407	0.3845
30	0.1517	0.2434	0.2494	0.4024	0.1550	0.2477	0.2421	0.3899
40	0.1533	0.2328	0.2642	0.4035	0.1554	0.2490	0.2540	0.4040
50	0.1502	0.2301	0.2768	0.4186	0.1580	0.2482	0.2587	0.4019
75	0.1498	0.2278	0.2862	0.4284	0.1543	0.2408	0.2678	0.4176
100	0.1456	0.2199	0.2911	0.4375	0.1587	0.2523	0.2737	0.4269
150	0.1425	0.2220	0.2935	0.4505	0.1519	0.2437	0.2773	0.4318
200	0.1399	0.2207	0.2968	0.4579	0.1437	0.2367	0.2788	0.4427

Table 3.6: Results of PRF and RF based query expansion with different number of feedback documents on CDS 2015.

CDS 2015	BM25				In_expC2			
	PRF		RF		PRF		RF	
#docs	MAP	infNDCG	MAP	infNDCG	MAP	infNDCG	MAP	infNDCG
0	0.1147	0.2115	0.1147	0.2115	0.1201	0.2132	0.1201	0.2132
5	0.1774	0.2905	0.1830	0.2928	0.1744	0.2793	0.1836	0.2990
10	0.1805	0.2830	0.1941	0.3028	0.1725	0.2734	0.1895	0.3015
15	0.1862	0.2972	0.2037	0.3195	0.1652	0.2583	0.1940	0.3046
20	0.1831	0.2924	0.2120	0.3298	0.1708	0.2646	0.1993	0.3159
25	0.1805	0.2865	0.2155	0.3356	0.1724	0.2659	0.2038	0.3248
30	0.1773	0.2824	0.2151	0.3334	0.1711	0.2657	0.2059	0.3231
40	0.1725	0.2676	0.2185	0.3390	0.1756	0.2749	0.2177	0.3405
50	0.1693	0.2658	0.2283	0.3478	0.1752	0.2725	0.2191	0.3390
75	0.1696	0.2666	0.2362	0.3602	0.1648	0.2572	0.2252	0.3424
100	0.1661	0.2635	0.2452	0.3704	0.1628	0.2548	0.2298	0.3502
150	0.1580	0.2555	0.2545	0.3799	0.1561	0.2476	0.2386	0.3575
200	0.1524	0.2514	0.2591	0.3836	0.1474	0.2416	0.2412	0.3628

Table 3.7: Results of PRF and RF based query expansion with different number of feedback documents on CDS 2016.

CDS 2016	BM25				In_expC2			
	PRF		RF		PRF		RF	
#docs	MAP	infNDCG	MAP	infNDCG	MAP	infNDCG	MAP	infNDCG
0	0.0620	0.1710	0.0620	0.1710	0.0632	0.1785	0.0632	0.1785
5	0.0689	0.1862	0.0785	0.2081	0.0767	0.2056	0.0917	0.2208
10	0.0769	0.2047	0.0984	0.2428	0.0754	0.2018	0.0992	0.2450
15	0.0783	0.1958	0.1138	0.2500	0.0761	0.1951	0.1056	0.2547
20	0.0789	0.2010	0.1182	0.2664	0.0710	0.1973	0.1107	0.2642
25	0.0772	0.1975	0.1271	0.2792	0.0721	0.2001	0.1164	0.2912
30	0.0752	0.1947	0.1312	0.2805	0.0746	0.2010	0.1164	0.2983
40	0.0786	0.2000	0.1408	0.2947	0.0744	0.2042	0.1205	0.3169
50	0.0800	0.2021	0.1456	0.3094	0.0745	0.2076	0.1275	0.3219
75	0.0795	0.2023	0.1556	0.3526	0.0705	0.2052	0.1260	0.3115
100	0.0817	0.2088	0.1565	0.3494	0.0717	0.2079	0.1295	0.3121
150	0.0820	0.2107	0.1637	0.3583	0.0700	0.2035	0.1292	0.3039
200	0.0788	0.2062	0.1670	0.3655	0.0707	0.1986	0.1388	0.3271

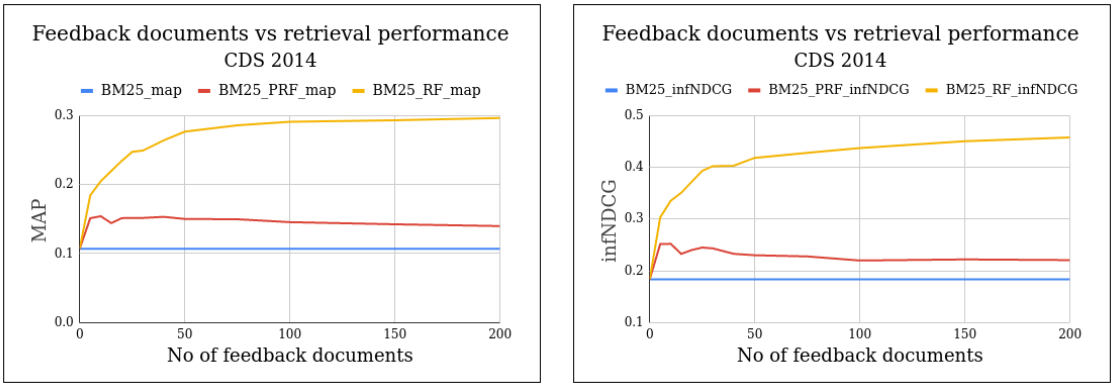

Figure 3.1: Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over BM25 on CDS 2014.

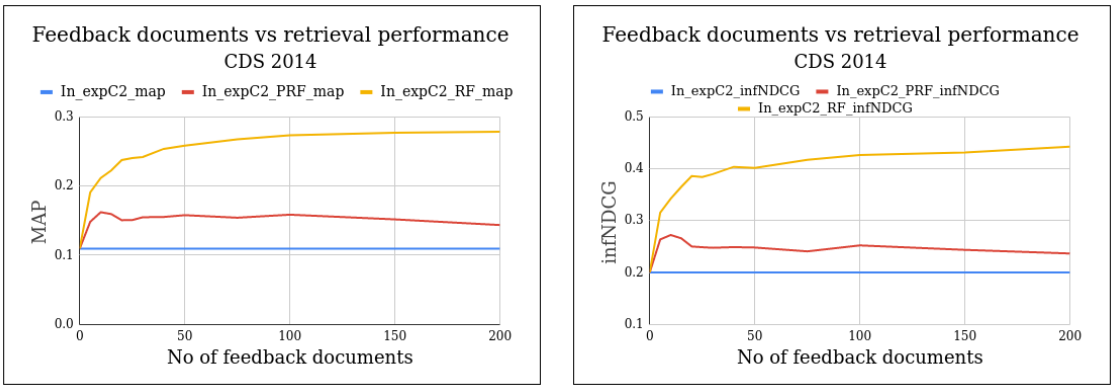

Figure 3.2: Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over In_expC2 on CDS 2014.

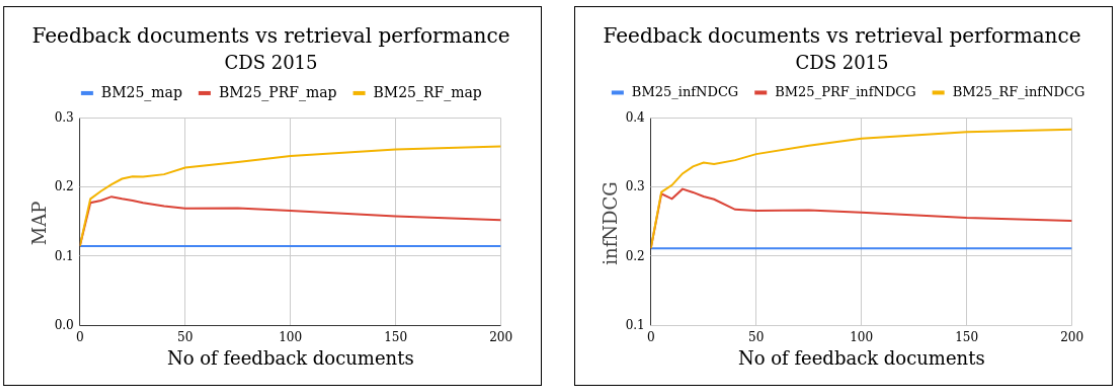

Figure 3.3: Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over BM25 on CDS 2015.

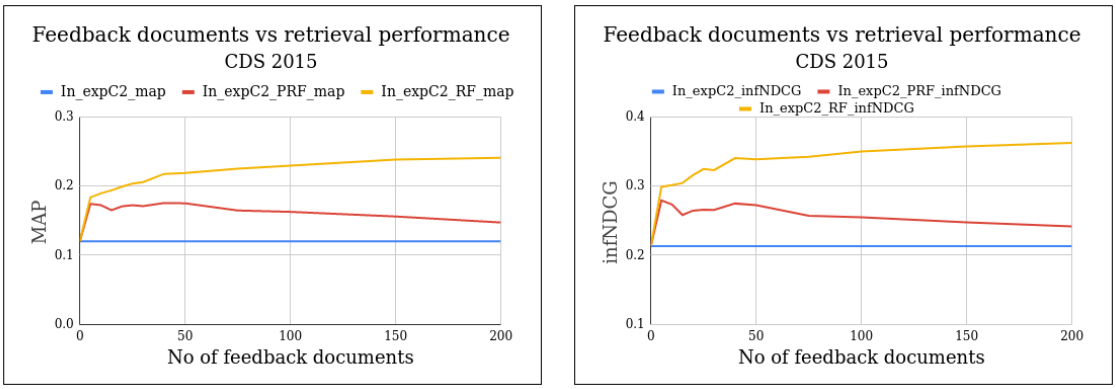

Figure 3.4: Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over In_expC2 on CDS 2015.

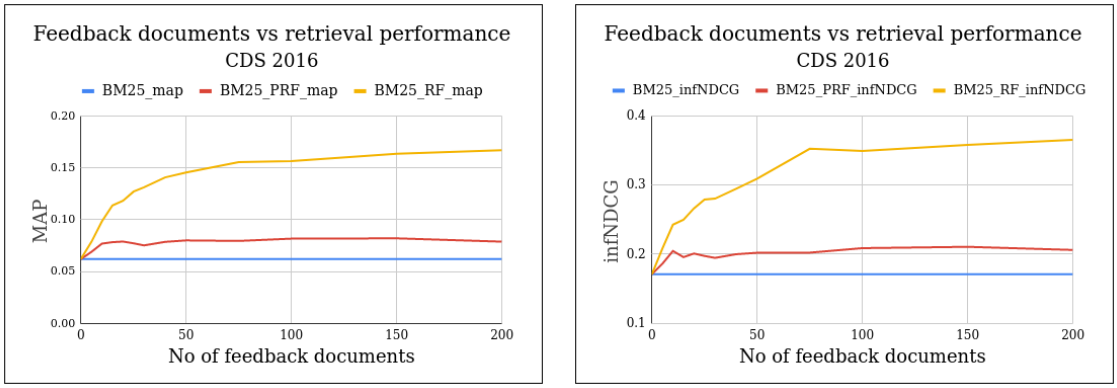

Figure 3.5: Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over BM25 on CDS 2016.

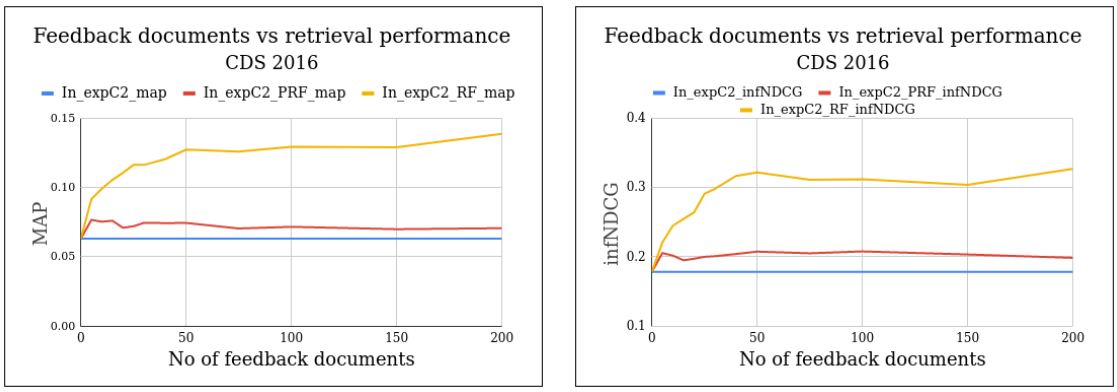

Figure 3.6: Feedback documents vs retrieval performance MAP(on left) and infNDCG(on right) of query expansion using PRF and RF over In_expC2 on CDS 2016.

The plots of these results are shown in Figures 3.1-3.6. Figure 3.1 and Figure 3.2 show feedback documents vs. retrieval performance, in terms of MAP (plot on left) and infNDCG (plot on right), for PRF and RF on CDS 2014 when using BM25 and In_expC2 retrieval models respectively. Similarly, Figure 3.3 and Figure 3.4 show the feedback documents vs. retrieval performance plots for BM25 and In_expC2, respectively, for CDS 2015 dataset. For CDS 2016, such plots are shown in Figure 3.5 and Figure 3.6.

These plots, shown in Figures 3.1-3.6, indicate that the PRF and RF are promising techniques for biomedical document retrieval. There is a significant gap between performances of PRF and RF and it is consistent across all three datasets. The research opportunity lies here to achieve the performance of RF but with less human efforts. For CDS 2014, CDS 2015, and CDS 2016 datasets, PRF is giving better results than the original queries while RF is giving the best results. In both the techniques, feedback documents matter considerably. As we increase the feedback documents, the retrieval performance using RF based query expansion increases. The more we provide manual feedback, the better we get the results. However, getting manual feedback is very costly. In case of PRF, manual feedback is not required as the feedback documents are chosen directly from top retrieved documents. So, as we increase the number of feedback documents, the retrieval performance increases initially, but later it starts decreasing with more feedback documents. The plots show that the optimal range of number of feedback documents is around 10-25 for CDS 2014, CDS 2015, and CDS 2016 datasets. After that, the performance starts decreasing, but still, it is always better than the performance of original queries (without query expansion).

3.2 Partial Relevance Feedback for Query Expansion

To get the benefits of RF with less human effort (reduced cost) and the benefits of PRF with good performance accuracy, we try to combine both and introduce a novel method partial relevance feedback (RF_p). Figure 3.7 shows the trade-off between PRF and RF.

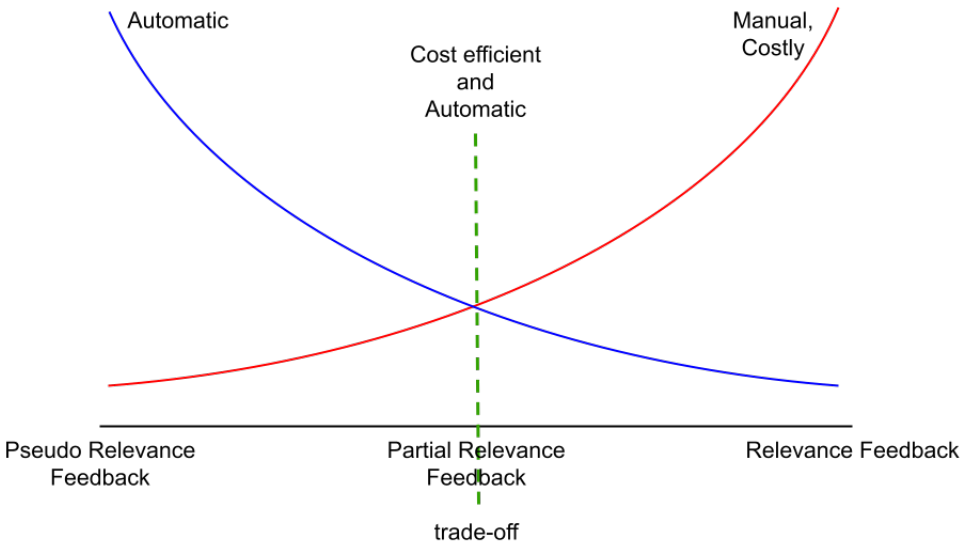

Figure 3.7: Partial Relevance Feedback: Trade-off between PRF and RF.

The partial relevance feedback technique combines feedback documents from RF and PRF both and does query expansion with the new set of feedback documents. PRF uses top retrieved documents for feedback, and RF uses user identified relevant documents for feedback, while partial relevance feedback method uses few user-identified relevant documents as well as some top retrieved documents for feedback. The top retrieved documents are divided into two parts. For the initial subset of the documents, user's input about the relevancy of the documents is considered, and only relevant documents are included in the feedback. The second subset of top retrieved documents is assumed to be relevant and used directly for feedback. Thus, partial relevance feedback is about combining PRF and RF, reducing the cost of manual intervention, and making the feedback based query expansion process more automatic. The process of partial relevance feedback considering top n retrieved documents does relevance feedback on first k documents, k$<$n, and pseudo relevance feedback on later n-k documents. Then it performs query expansion using relevant documents from first k documents and all n-k documents.

The partial relevance feedback based query expansion is performed for biomedical document retrieval on CDS datasets. The results of partial relevance feedback are compared with PRF and RF. Table 3.8 and Table 3.9 shows the result comparison on CDS 2014, CDS 2015, and CDS 2016 datasets in terms of MAP and infNDCG, respectively.

Table 3.8: MAP results of Query Expansion with partial relevance feedback.

MAP	**CDS 2014**	**CDS 2015**	**CDS 2016**
BM25	0.1071	0.1147	0.0620
BM25+PRF_{50}	0.1502	0.1693	0.0800
BM25+$RFp_{\_20\_50}$	**0.1743**	**0.1762**	**0.0874**
BM25+RF_{50}	0.2768	0.2283	0.1456
In_expC2	0.1096	0.1201	0.0632
In_expC2+PRF_{50}	0.1580	0.1752	0.0745
In_expC2+$RFp_{\_20\_50}$	**0.1809**	**0.1834**	**0.0806**
In_expC2+RF_{50}	0.2587	0.2191	0.1275

Table 3.9: InfNDCG results of Query Expansion with partial relevance feedback.

infNDCG	**CDS 2014**	**CDS 2015**	**CDS 2016**
BM25	0.1836	0.2115	0.1710
BM25+PRF_{50}	0.2301	0.2658	0.2021
BM25+$RFp_{\_20\_50}$	**0.2667**	**0.2722**	**0.2210**
BM25+RF_{50}	0.4186	0.3478	0.3094
In_expC2	0.2002	0.2132	0.1785
In_expC2+PRF_{50}	0.2482	0.2725	0.2076
In_expC2+$RFp_{\_20\_50}$	**0.2814**	**0.2823**	**0.2256**
In_expC2+RF_{50}	0.4019	0.3390	0.3219

Here, the top 50 documents are considered for feedback in all the methods. For partial relevance feedback, out of 50 documents, first 20 documents are considered with manual feedback and next 30 documents are considered with pseudo relevance. So, n=50 and k=20 is set for the experiments. On all three datasets, the results of partial relevance feedback are higher than the pseudo relevance feedback but lower than the relevance feedback which was expected to be.

The experiments of partial relevance feedback are performed varying the size of subset for which the user's input was considered. The number of documents considered for user's input was 5, 10, 15, 20, 25, 30, 40, 50, 75, and 100. These results of partial relevance

feedback, RF, PRF, and results without query expansion, using BM25 retrieval model, are plotted in Figure 3.8, Figure 3.9 and Figure 3.10 for CDS 2014, CDS 2015, and CDS 2016 datasets, respectively.

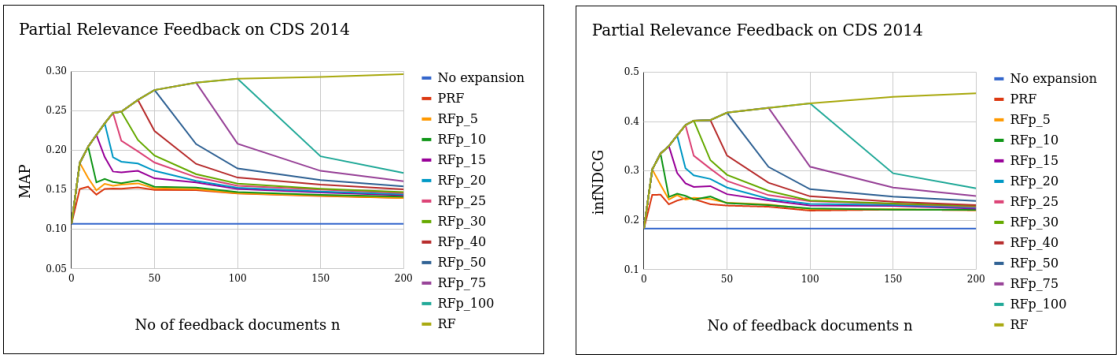

Figure 3.8: No. of feedback documents vs retrieval performance(MAP, infNDCG) plot for partial relevance feedback based query expansion on CDS 2014.

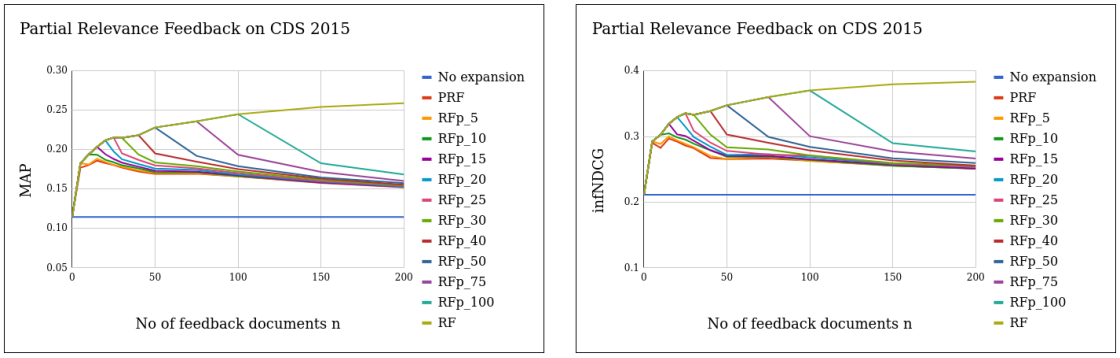

Figure 3.9: No. of feedback documents vs retrieval performance(MAP, infNDCG) plot for partial relevance feedback based query expansion on CDS 2015.

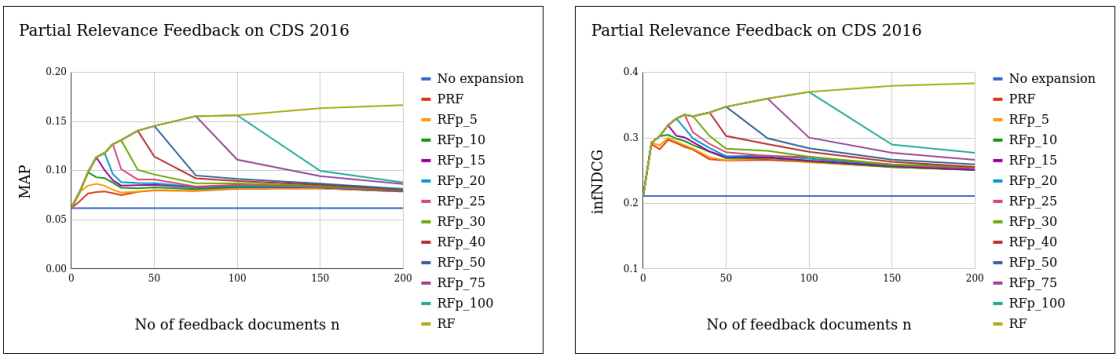

Figure 3.10: No. of feedback documents vs retrieval performance(MAP, infNDCG) plot for partial relevance feedback based query expansion on CDS 2016.

Figure 3.8 shows the plots of the number of feedback documents considered in partial relevance feedback vs. retrieval performance in terms of MAP and infNDCG for CDS 2014 dataset. Figure 3.9 and Figure 3.10 show the similar plots for CDS 2015 and CDS 2016 datasets, respectively. From all these plots, we can say that the results of partial relevance feedback are better than the results of pseudo relevance feedback.

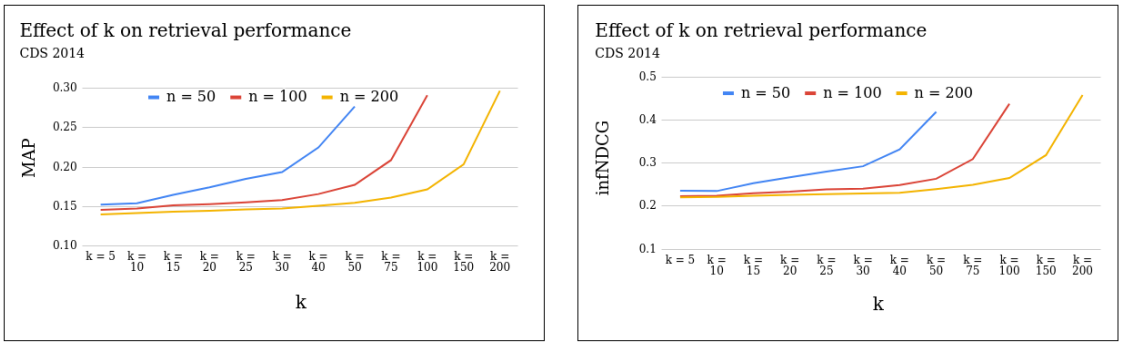

Figure 3.11: Effect of k on retrieval performance(MAP and infNDCG) on CDS 2014.

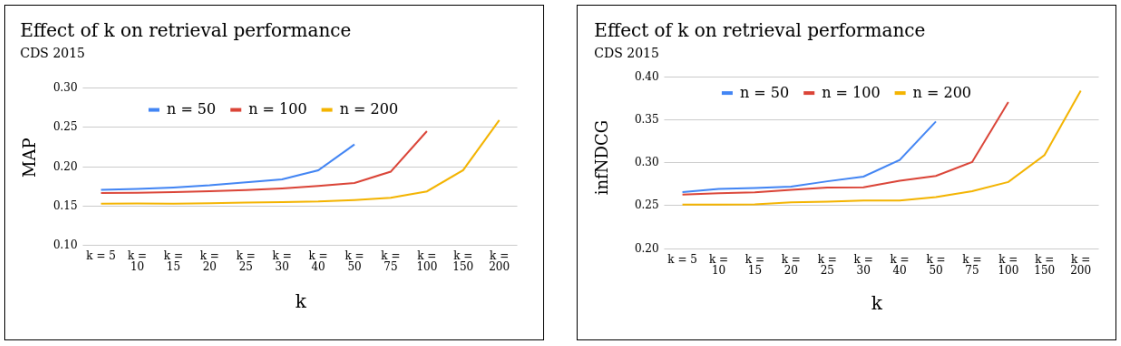

Figure 3.12: Effect of k on retrieval performance(MAP and infNDCG) on CDS 2015.

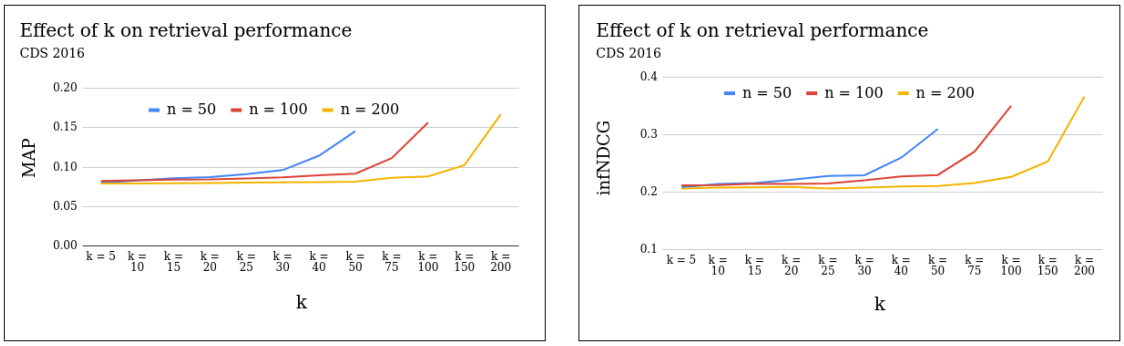

Figure 3.13: Effect of k on retrieval performance(MAP and infNDCG) on CDS 2016.

To see the effect of k, we have experimented with varying the value of k used in partial relevance feedback. Keeping the value of n fix and changing the value of k from 5 to 200, the experiments are performed on all three datasets, and the results are presented in Figure 3.11, Figure 3.12, and Figure 3.13. Figure 3.11 shows the plots of value of k vs retrieval performance in terms of MAP and infNDCG on CDS 2014 dataset, keeping n=50, 100 and 200 fix at a time. Similarly, the plots of varying k vs retrieval performance for CDS 2015 and CDS 2016 datasets are shown in Figure 3.12 and Figure 3.13, respectively. These results indicate that MAP and infNDCG increases with the value of k.

The higher the value of k, the better the performance of partial relevance feedback that means if we can incorporate more input from user in partial relevance feedback, we can get better system. For CDS 2014 dataset, we got good results if we set the value of k as one third of the value of n. For n=50, we start getting improvement from k=15. Similarly, we start getting improvement from k=30 for n=100 and from k=60 for n=200. For CDS 2015 and CDS 2016 datasets, we get the good improvements when k is half the value of n. We can see that the curves go up from k=25 for n=50, from k=50 for n=100 and from k=100 for n=200 in Figure 3.12 and Figure 3.13. That means we need approximately 30%, 50%, and 50% manual intervention in partial relevance feedback for CDS 2014, CDS 2015, and CDS 2016 datasets, respectively. Later, as we increase the the amount of manual intervention, we get more improvement but with the increased cost.

3.3 Feedback Documents Discovery based Query Reformulation

Query expansion methods largely rely on feedback documents and feedback terms. Automatic query expansion methods based on pseudo relevance feedback uses top retrieved documents as feedback documents. Those feedback documents might not be all relevant. The feedback document set might contain non-relevant docs along with truly relevant documents. The retrieval system gets hurt with these non-relevant documents in the feedback set. They become noise in the feedback system. Relevance feedback method discards non-relevant documents from feedback but the process is costly while partial relevance feedback is a combination of both, PRF and RF. Here, we propose a modified version of

partial relevance feedback method with a learning module that helps to identify relevant documents and discard the non-relevant documents for feedback.

This query reformulation approach is based on feedback document discovery which learns to identify relevant documents for query expansion from top retrieved documents [122]. The main aim is to use a small amount of human judgement and learn pseudo judgement for other documents to reformulate the queries. This approach is based on the learning method. If we have human judgements available for some of the feedback documents, then it will serve as training data for the learning module which will later be used to classify other documents as relevant or non-relevant. The documents were represented as a collection of bag-of-words, the TF-IDF scores of the words represent features, and human relevance scores provide the classes. Here, we propose two approaches of feedback document discovery. The first approach has a classification module (described in Algorithm 1), while the second approach has classification followed by clustering modules (described in Algorithm 2). The second approach is a two-step process where clustering is applied to relevant class, predicted by the classification method, to filter out more non-relevant documents. These methods only use relevant-predicted documents as feedback documents in query expansion.

3.3.1 Feedback document discovery using classification

The first proposed algorithm is based on classification. If we have human judgements available for some of the feedback documents, then it will serve as training data for classification. The documents are represented as a collection of bag-of-words, the TF-IDF scores of the words represent features, and human relevance scores provide the classes. Using these data, a classification module is trained and later used to predict the relevance of other top retrieved feedback documents. This approach of feedback document discovery using classification is given below.

Algorithm 1: Using classification

for each query Q **do**
    N = Total number of top retrieved documents to consider for feedback
    k = Number of top retrieved documents for which human judgements are available
    l = N-k; Number of top retrieved documents for which human judgements are not available
    D_N - set of N top retrieved documents $\{d_1, d_2, ..., d_N\}$
    D_k - set of k top retrieved documents for which human judgements are available $\{d_1, d_2, ..., d_k\}$
    D_l - set of l=N-k top retrieved documents for which human judgements are not available $\{d_{k+1}, d_{k+2}, ..., d_N\}$
    D_F - set of feedback documents
    $D_F = \{d_i; relevance\ of\ d_i > 0, d_i \in D_k\}$
    Train a classifier C on D_k using relevance as a class label and generate model M_c
    for each document d_j in D_l, $k+1 \leq j \leq N$ **do**
        Predict the relevance r_j of d_j using the trained model M_c
        if $r_j > 0$ **then**
            $D_F = D_F \cup \{d_j\}$

3.3.2 Feedback document discovery using classification and clustering

The second algorithm is an extension of the first algorithm. The analysis of results of the first algorithm shows that the feedback document set still contains some non-relevant docs, and they are responsible for insignificant improvement. This approach further removes non-relevant documents from the relevant document class identified by the classification approach. The idea is to perform clustering on the relevant identified documents with the number of clusters equals to two: one for actually relevant documents and another for non-relevant documents. This approach of feedback document discovery using classification

and clustering is given below.

Algorithm 2: Using classification and clustering

---

for each query Q **do**
    N = Total number of top retrieved documents to consider for feedback
    k = Number of top retrieved documents for which human judgements are available
    l = N-k; Number of top retrieved documents for which human judgements are not available
    D_N - set of N top retrieved documents $\{d_1, d_2, ..., d_N\}$
    D_k - set of k top retrieved documents for which human judgements are available $\{d_1, d_2, ..., d_k\}$
    D_l - set of l=N-k top retrieved documents for which human judgements are not available $\{d_{k+1}, d_{k+2}, ..., d_N\}$
    D_F - set of feedback documents
    $D_F = \{d_i; relevance\ of\ d_i > 0, d_i \in D_k\}$
    Train a classifier C on D_k using relevance as a class label and generate model M_c
    $D_R = \phi, D_{NR} = \phi$
    for each document d_j in D_l, $k+1 \leq j \leq N$ **do**
        Predict the relevance r_j of d_j using trained model M_c
        if $r_j > 0$ **then**
            $D_F = D_F \cup \{d_j\}$
        else
            $D_{NR} = D_{NR} \cup \{d_j\}$ \\ D_R contains relevant predicted documents from D_l
    Perform K-means clustering on D_R with k=2 (relevant docs and non-relevant docs)
    $D_F = D_F \cup \{documents\ from\ relevant\ docs\ cluster\}$

Here, K-means clustering is used with k=2. Since the convergence of K-means clustering depends on the initial choice of cluster centroids, the initial cluster centroids are chosen as the average of relevant documents' vectors and the average of non-relevant documents'

vectors from training data for relevant and non-relevant document representing classes, respectively.

3.3.3 Experiments

The query expansion considers top N retrieved documents for feedback. Here, we have considered the top 250 documents, from which a subset of top 50 documents is used in training, i.e. human judgements for top 50 documents are used in training, and the rest of 200 documents are taken for testing data. The relevance is predicted for those 200 documents and only relevant predicted documents are then used for feedback. The result of relevance feedback using top 50 documents is used as baseline. All the computed results will be compared with the baseline.

The experiments are performed using nine different classifiers for classification in the first algorithm. Table 3.10 shows the results of feedback document discovery with various classifiers in terms of MAP score for CDS 2014 dataset. Neural-Net gives the best result among all nine classifiers. Also, the result of classification with Nearest-Neighbors is comparable to the baseline.

Table 3.10: Results of feedback document discovery using different classifiers on CDS 2014 dataset.

	CDS 2014
MAP	classification
Baseline (RF_50)	0.2768
Nearest-Neighbors	**0.2761**
Linear-SVM	0.2736
RBF-SVM	0.2736
Gaussian-Process	0.2736
Decision-Tree	0.2496
Random-Forest	0.2733
Neural-Net	**0.2790**
AdaBoost	0.2618
Naive-Bayes	0.2614

The classification results are not significant with the baseline results. We investigated the results and found that the relevant classified documents in relevance class are not all actually relevant. The feedback document set still contains some irrelevant documents (misclassification). For all the 30 queries of CDS 2014, Nearest-Neighbour classified 625 documents as relevant out of all 200*30 documents. Out of 625 documents used for feedback, 244 documents were actually relevant while the other 381 documents were wrongly classified as relevant. So, these 381 irrelevant documents are noise to the system. The second approach takes this matter into consideration and further refines the feedback document set by performing 2-cluster clustering on 625 documents. Manually removing 381 irrelevant documents from feedback document set shows significant improvement over baseline. The results of manually removing false-classified documents from feedback set and feedback document discovery using classification+clustering are shown in Table 3.11.

Table 3.11: Results of feedback document discovery using different classification and clustering on CDS 2014 dataset.

	CDS 2014		
MAP	classification	classification + manually removing false relevant docs	classification + clustering
Baseline (RF_50)	0.2768	0.2768	0.2768
Nearest-Neighbors	**0.2761**	**0.2815 (p = 0.048)**	**0.2794**
Linear-SVM	0.2736	0.2760	0.2750
RBF-SVM	0.2736	0.2760	0.2750
Gaussian-Process	0.2736	0.2762	0.2753
Decision-Tree	0.2496	0.2788	0.2725
Random-Forest	0.2733	0.2760	0.2747
Neural-Net	**0.2790**	0.2808	0.2790
AdaBoost	0.2618	0.2806	0.2741
Naive-Bayes	0.2614	0.2792	0.2661

The same experiments are performed on CDS 2015 and 2016 datasets. The results of both the algorithms using six different classifiers are shown in Table 3.12.

Table 3.12: Results of feedback document discovery on CDS 2015 and CDS 2016 dataset. Bold represents highest result. ⋆ represents statistically significant result with p<0.05.

	CDS 2015		**CDS 2016**	
MAP	classification	classification + clustering	classification	classification + clustering
Baseline (RF_50)	0.2283	0.2283	0.1456	0.1456
Nearest-Neighbors	0.2234	**0.2324**⋆	0.1456	**0.1459**
Decision-Tree	0.2065	0.2218	0.1138	0.1370
Random-Forest	0.2130	0.2281	0.1450	0.1458
Neural-Net	0.2295	0.2299	0.1460	0.1466
AdaBoost	0.2092	0.2213	0.1255	0.1345
Naive-Bayes	0.2172	0.2269	0.1436	0.1468

The results of nearest-neighbors and neural-net, using both the approaches (classification and classification + clustering), are better than the baseline for CDS 2014, CDS 2015, and CDS 2016 datasets. The second algorithm with nearest-neighbors and clustering performs best for all three datasets. For CDS 2015 dataset, the result of nearest-neighbors with clustering is statistically significant as compared to baseline. For CDS 2016 dataset, both the algorithms perform similar to the baseline. Here, feedback document discovery method used manual judgements for 50 documents only. Figure 3.14 shows the query wise difference in infNDCG results between Neural net(classification + clustering) and RF_50.

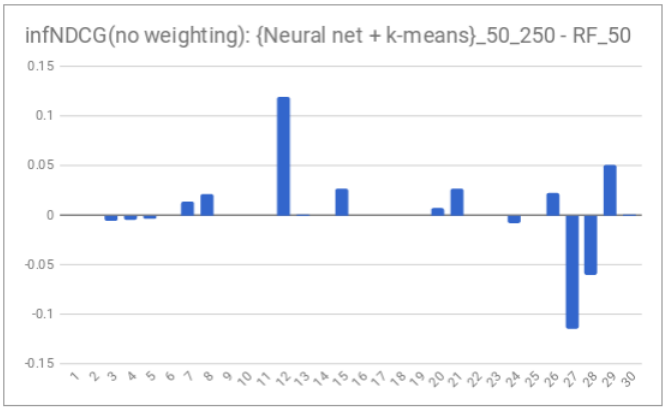

Figure 3.14: Query wise performance difference between feedback document discovery and relevance feedback in terms of infNDCG.

The positive difference means that the query expansion using feedback document discovery performs better than the query expansion using relevance feedback. The negative difference means that the query expansion using relevance feedback gives better results. Out of 30 queries of CDS 2014, two queries degrade the performance score of infNDCG, but seven queries improve while others have same scores. On an average, feedback document discovery helps in query expansion to get good feedback documents using a small amount of human intervention.

3.4 Feature weighting in finding feedback documents for query expansion

The Feedback document discovery method is reinforced with feature weighting to see the effect of biomedical features on feedback based query expansion. This proposed approach uses NLP based feature weighting technique with classification and clustering method on the documents and identifies relevant documents for feedback. The documents are represented using Term Frequency and Inverse Document Frequency (TF-IDF) features which are weighted according to the type of query and the type of the terms.

3.4.1 Entity based feature weighting

For classification and clustering in feedback document discovery, we propose a feature weighting technique which is based on the semantic type of the entities. The TF-IDF features used in feedback document discovery methods (described in Section 3.3.1 and Section 3.3.2) are weighted based on 'problem', 'test', and 'treatment' types of biomedical entities.

In this technique, we are using Clinical Named Entity Recognition system (CliNER) [18], which is an open-source natural language processing system for named entity recognition in clinical text of electronic health records. CliNER is implemented as a sequence classification task, where every token is predicted IOB-style [107] as either: Problem, Test, Treatment, or None. We have trained it on i2b2 2010 dataset [141] which includes discharge summaries from Partners Health-Care, from Beth Israel Deaconess Medical

Center, and from University of Pittsburgh Medical Center. These discharge summaries are manually annotated for concept, assertion, and relation information. The model is trained on the documents of i2b2 dataset to identify medical entities of type 'problem', 'test' and 'treatment' from them. It has precision 0.795 on 2010 i2b2 dataset. This trained model is then applied on CDS documents to identify those three types of concept entities. The features related to these entities in CDS documents are weighted, thus giving importance to these entities while learning to identify feedback documents. The proposed two approaches for feature weighting on these entities are as follows:

FW1 : The first approach does feature weighting of medical concepts based on the type of the query. There are three types of queries in the dataset: 'Diagnosis', 'Test', and 'Treatment'. For queries of a particular type, only features of the entities of the same type are weighted. The feature of term t is determined as follows:

$$f(t)_q = \begin{cases} w * TF * IDF & \text{if } q \text{ is of type diagnosis and } t \text{ is a problem type term} \\ w * TF * IDF & \text{if } q \text{ is of type test and } t \text{ is a test type term} \\ w * TF * IDF & \text{if } q \text{ is of type treatment and } t \text{ is a treatment type term} \\ TF * IDF & \text{otherwise} \end{cases}$$

For 'Diagnosis'type of queries, only 'Problem'type of entities are weighted by weight w. For 'Test'type of queries, only 'Test'type of entities are weighted by weight w. For 'Treatment'type of queries, only 'Treatment'type of entities are weighted by weight w. In this way, it gives importance to those entities which are categorically similar to the query.

FW2: The second approach does feature weighting of medical concepts irrespective of the type of query. For all the queries of type 'Diagnosis', 'Test'and 'Treatment', All the entities of types 'Problem', 'Test', and 'Treatment'are weighted by weight w. The feature of term t is determined as follows:

$$f(t)_q = \begin{cases} w * TF * IDF & \text{if } t \text{ is either problem or test or treatment type term} \\ TF * IDF & \text{otherwise} \end{cases}$$

3.4.2 Experiments and Results

The experiments of feature weighting in feedback document discovery based query expansion are performed on CDS datasets. Three years data has been used in the experiments i.e., CDS 2014, CDS 2015, and CDS 2016 dataset. The query expansion considers top N retrieved documents for feedback. Here, we have considered top 250 documents, from which the set of top 50 documents are used as training i.e., human judgements for top 50 documents are used in training, and the rest of 200 documents are taken for testing data. The relevance is predicted for those 200 documents and only relevant predicted documents are then used for feedback. The result of relevance feedback using top 50 documents is the baseline for other results.

The results of two feature weighting techniques along with the results of original queries without expansion, query expansion with relevance feedback, and query expansion with feedback document discovery without feature weighting (only using TF-IDF) for CDS 2014 dataset are given in Table 3.13 in terms of MAP and infNDCG score.

Table 3.13: Results of feedback document discovery with feature weighting on CDS 2014. Percentage improvements with respect to PRF are shown in brackets. $\star$ represents statistically significant results.

CDS 2014	**MAP**			**infNDCG**		
	TF-IDF	**FW1**	**FW2**	**TF-IDF**	**FW1**	**FW2**
Original Queries	0.1071			0.1836		
Queries+PRF_{50}	0.1502			0.2301		
Queries+RF_{50}	0.2768 (84%)			0.4186 (82%)		
Nearest neighbors	0.2761	0.2754	0.2747	0.4177	0.4161	0.4140
Nearest neighbors + k-means	0.2794	0.2778	0.2777	0.4220	0.4168	0.4195
Neural net	0.2790	0.2784	0.2787	0.4235	0.4243	0.4240
Neural net + k-means	0.2790 (86%)	0.2788	**0.2807$\star$ (87%)**	0.4218 (83%)	0.4225	**0.4269$\star$ (86%)**

The similar result comparisons for CDS 2015 and CDS 2016 dataset are given in Table

3.14 and Table 3.15, respectively.

Table 3.14: Results of feedback document discovery with feature weighting on CDS 2015. Percentage improvements with respect to PRF are shown in brackets. ⋆ represents statistically significant results.

CDS 2015	**MAP**			**infNDCG**		
	TF-IDF	**FW1**	**FW2**	**TF-IDF**	**FW1**	**FW2**
Original Queries	0.1147			0.2115		
Queries+PRF_{50}	0.1693			0.2658		
Queries+RF_{50}	0.2283 (35%)			0.3478 (31%)		
Nearest neighbors	0.2234	0.2212	0.2234	0.3518	0.3480	0.3518
Nearest neighbors + k-means	0.2244	0.2214	0.2299	0.3541	0.3519	0.3506
Neural net	0.2295	0.2297	0.2284	0.3528	0.3514	0.3492
Neural net + k-means	0.2299 (36%)	**0.2302 (36%)**	0.2301	0.3529 (33%)	0.3525	**0.3526⋆ (33%)**

Table 3.15: Results of feedback document discovery with feature weighting on CDS 2016. Percentage improvements with respect to PRF are shown in brackets. ⋆ represents statistically significant results.

CDS 2016	**MAP**			**infNDCG**		
	TF-IDF	**FW1**	**FW2**	**TF-IDF**	**FW1**	**FW2**
Original Queries	0.062			0.1710		
Queries+PRF_{50}	0.0800			0.2021		
Queries+RF_{50}	0.1456 (82%)			0.3094 (53%)		
Nearest neighbors	0.1456	0.1463	0.1458	0.3113	0.3124	0.3113
Nearest neighbors + k-means	0.1459	0.1470	0.1467	0.3127 (55%)	**0.3158⋆ (56%)**	0.3139
Neural net	0.1460	0.1467	0.1463	0.3073	0.3136	0.3143
Neural net + k-means	0.1466 (83%)	**0.1471 (84%)**	0.1458	0.3100	0.3132⋆	0.3124⋆

The results of feature weighting in feedback document discovery show improvement with neural net + k-means over relevance feedback. For CDS 2014, the results of FW2 with neural net + k-means are significantly improved than the results of relevance feedback. For CDS 2015 dataset, infNDCG result of FW2 with with neural net + k-means is significantly better than the result of relevance feedback. Feedback document discovery with feature weighting helps to identify good relevant documents for feedback which contain related biomedical entities. For example, in the query '58-year-old woman with hypertension and obesity presents with exercise-related episodic chest pain radiating to the back', the feature weighting in feedback document discovery identified important entities like 'pulmonary embolism', 'cardiac enzymes', 'stress-induced cardiomyopathy', etc.. from manually identified relevant documents and classified the other documents based on these entities. It managed to remove the documents which are containing other terms like 'procedure', 'tolerance', 'permit' etc... more often than the identified biomedical entities of type 'problem', 'test' and 'treatment'.

3.5 Learning To Rank

Learning To Rank (LTR) [70] is an application of machine learning in the construction of ranking models for information retrieval systems where retrieval problem is modeled as a ranking problem. LTR framework requires training data of queries and documents matching them together with the relevance degree of each match. The learning algorithm uses the training data and produces a ranking model which computes the relevance of documents for actual queries.

The LTR framework is applied on CDS 2014 dataset, where the features for query document pairs are considered as per the features of OHSUMED LETOR dataset [102]. These features are mainly based on TF, IDF, and their normalized versions. Since the whole document pool is too large, document pooling has been done and top K documents (retrieved by BM25) for each query are used for feature extraction. SVMRank [54] is used here as a machine learning framework.

Table 3.16 shows the results of LTR when the features are computed on Title+Abstract part of the documents and Title+Abstract+Content of the documents (i.e. full documents).

With these variations of features, the experiments are carried out on original queries, queries with UMLS concepts and queries with manually identified medical concepts.

Table 3.16: Results of Learning to Rank with various features.

infNDCG	**OHSUMED features on T, A and T+A**	**OHSUMED features on T, A and C**
Original Queries	0.0970	0.1769
Queries + UMLS	0.0833	0.1556
Queries + Manual	**0.1049 (8%)**	**0.1785 (1%)**

All these LTR experiments require human judgement for training. To overcome the need of manual judgement, pseudo judgements were also considered where out of k training documents, Top k/2 documents are considered to be relevant and other k/2 documents to be non-relevant. Table 3.17 shows the results of LTR trained using human judgements and pseudo judgements.

Table 3.17: Results of Learning To Rank with pseudo judgements.

	infNDCG
Retrieval (BM25)	0.1836
LTR using human judgements	0.1769
Pseudo LTR K=1000	0.1849
Pseudo LTR K=1500	**0.1872**
Pseudo LTR K=2000	0.1859
Pseudo LTR K=2500	0.1865
Pseudo LTR K=3000	0.1865

The results of LTR trained using pseudo judgements are better than the results of LTR trained with actual human judgements, but they are comparable to the retrieval using BM25.

3.6 Query expansion using topic modeling

Topic modeling is widely used in text mining to find semantic structures. It refers to discovering abstract topics from the text using machine learning techniques. It is an unsupervised approach based on clustering where words or phrases are grouped together according to the information content. It analyses a set of documents and identifies the groups/topics that best represents the information of unlabeled text. A set of words grouped together forms a topic. Here, we have used topic modeling for query processing, and we refer it as query expansion using topic modeling. The approach is experimented for biomedical document retrieval systems. In the process, we use top retrieved documents, apply topic modeling on them and find important topics from them. In the query expansion process, each query gets expanded with those important identified topics.

For topic modeling, we have used the topic model package MALLET[77], which has a fast and highly scalable implementation of Gibbs sampling, efficient methods for document-topic hyperparameter optimization, and tools for inferring topics for new documents given trained models. The experiments of query expansion using topic modeling are performed on CDS 2014 dataset and the results are shown in Table 3.18.

Table 3.18: Results of query expansion using topic modeling on CDS 2014 dataset.

CDS 2014	MAP	infNDCG
BM25	0.1071	0.1836
BM25+PRF_{10}	**0.1542 (44%)**	**0.2522 (37%)**
BM25+TM_{10}	0.1312	0.2172
In_expC2	0.1096	0.2002
In_expC2+PRF_{10}	**0.1623 (48%)**	**0.2724 (36%)**
In_expC2+TM_{10}	0.1438	0.2194

The results of query expansion using topic modeling are compared with the results without query expansion and pseudo relevance feedback based query expansion. The comparison shows that the results of query expansion using topic modeling are not better than or similar to the results of query expansion using PRF on CDS 2014 dataset.

The combination of two unsupervised techniques of query expansion, using topic modeling and using PRF, the experiments are performed on CDS 206 dataset [121]. The results are shown in Table 3.19.

Table 3.19: Results of combining topic modeling with pseudo relevance feedback on CDS 2016 dataset.

CDS 2016	MAP	infNDCG	infAP	R-prec	P@10
In_expC2	0.0632	0.1785	0.0203	0.1298	0.2767
In_expC2+PRF	0.0754	**0.2018 (13%)**	**0.0281 (3%)**	0.1355	**0.2833 (2%)**
In_expC2+TM	0.0642	0.1787	0.0208	0.1310	0.2700
In_expC2+TM_summ+PRF	**0.0760 (20%)**	0.1988	0.0253	**0.1416 (9%)**	0.2667
In_expC2+TM_desc+PRF	0.0576	0.1692	0.0255	0.1175	0.2900
In_expC2+TM_note+PRF	0.0631	0.1734	0.0227	0.1160	0.2100

Percentage improvement with respect to retrieval without query expansion are shown in the brackets. The results show that combining topic modeling and PRF for query expansion performs better than the individuals in terms of MAP and R-prec. While considering infNDCG, infAP and P@10, PRF performs best.

3.7 Conclusion

In this chapter, we have explored various query expansion techniques for biomedical document retrieval systems. Automatic query expansion using PRF, RF, retrieval using learning to rank, and query expansion using topic modeling are experimented, and the results are compared. We have also seen query expansion using partial relevance feedback, which is a combination of PRF and RF. Partial relevance feedback gives considerable improvement when we use RF for 30% to 50% of the feedback document set and PRF for the rest. As we increase the amount of RF, we get better results but with the heavily increased cost. The modified version of partial relevance feedback, which has a learning module to identify relevant feedback documents, is proposed as feedback document discovery

based query reformulation where different classifiers are explored. Later, this feedback document discovery based query expansion is supported with NLP based feature weighting to consider the semantics of biomedical text while processing it. With feedback document discovery and feature weighting, we can reduce the amount of manual intervention and still get the better results than the relevance feedback. In the next chapter, we will see UMLS graph based query reformulation technique for biomedical document retrieval systems, which considers conceptual and semantic properties of biomedical text using UMLS metathesaurus.

CHAPTER 4

UMLS graph based query reformulation in retrieval

Applications of biomedical domain require entity level processing instead of term level and hence require semantic processing of biomedical text. The biomedical entities tend to be complex, ambiguous, inconsistent and on an average, longer than entities in regular texts. Hence special attention to entity-level processing is required where conceptual and semantic knowledge of entities needs to be incorporated. The previous chapter explores various query expansion techniques where the queries as well as the documents are processed at term level, and the queries are expanded with related terms from related documents. In this chapter, we will explore the query expansion techniques at entity level, where the queries get expanded with conceptually and semantically related entities. For entity level processing of queries and documents, the knowledge from UMLS metathesaurus is utilized. This chapter also explores the usage and impact of UMLS for entity based query reformulation in biomedical document retrieval.

4.1 UMLS Concepts Based Query Reformulation

Biomedical domain-specific knowledge can be incorporated to the process of query reformulation in Biomedical IR system. There are knowledge-based approaches proposed in the literature [8, 34, 50]. In biomedical domain, medical concepts and entities are more informative than other terms. Moreover, medical ontologies, thesaurus and biomedical entity identifiers are available to identify medical related concepts.

With the knowledge from UMLS metathesaurus, we have modified the queries in

various ways. The following three pre-retrieval query reformulation experiments are done using it. First: The UMLS concepts are identified from the query text, and their preferred names are used with the original queries. Second: Along with the UMLS concepts, MeSH (Medical Subject Heading) entry terms are also identified and used in queries. Example: For the UMLS concept 'hypertension', the mesh entry terms are 'blood pressure, high'. MeSH is a hierarchically organized vocabulary of UMLS. Third: Medical entities are identified manually with the help of an expert and used with the queries.

Table 4.1 shows the results of these reformulated queries of CDS 2014. PRF and RF based query expansion is also carried out on each form of the queries. The highest results are shown in bold.

Table 4.1: Results of UMLS concepts based query processing.

	MAP	**infNDCF**
Original Queries	0.1071	0.1836
Queries + UMLS concepts	0.1100	0.1830
Queries + UMLS concepts + Mesh terms	0.1039	0.1749
Queries + Manual Entities	**0.1112**	**0.1860**
Original Queries + PRF_{10}	0.1542	0.2522
Queries + UMLS concepts + PRF_{10}	**0.1607**	0.2607
Queries + UMLS concepts + Mesh terms + PRF_{10}	0.1460	0.2409
Queries + Manual Entities + PRF_{10}	0.1601	**0.2634**
Original Queries + RF_{10}	0.2050	0.3355
Queries + UMLS concepts + RF_{10}	**0.2164**	**0.3423**
Queries + UMLS concepts + Mesh terms + RF_{10}	0.2052	0.3321
Queries + Manual Entities + RF_{10}	0.2112	0.3394
Original Queries + RF_{50}	0.2768	0.4186
Queries + UMLS concepts + RF_{50}	**0.2776**	**0.4232**

The results show improvement when using UMLS concepts in queries as compared to original queries. The MAP result of queries+UMLS concepts is higher than original queries and it is similar to the result of queries+manual entities. The results of queries+UMLS

concepts + PRF are higher than original queries+PRF for MAP and infNDCG. Similarly, the results of queries+UMLS concepts+RF are higher than the results of original queries+RF. Including Mesh terms in the queries along with UMLS concepts do not help to get better performance in any case, but UMLS concepts definitely give promising results. In some cases, the results of using UMLS concepts are even better than the results using manual entities.

4.2 Query specific graph based query reformulation using UMLS

A novel graph based approach for query reformulation using UMLS is described here, in which queries are expanded using biomedical entities. In biomedical information retrieval systems, meaningful query reformulation usually amounts to selecting the right set of entities. This method considers UMLS entities from a query with their related entities identified by UMLS and constructs query-specific graph of biomedical entities for term selection.

Incorporating medical knowledge by expert manually in any system leads to better performance of the system. For biomedical document retrieval systems, manually tweaking queries by an expert gives best results [12, 28, 57] which are more effective than fully automatic approaches. Another way of incorporating medical knowledge is to refine search results by medical experts and use refined results for feedback which is a partially manual feedback approach with automatic query reformulation from human identified relevant documents but still a costly scenario. On the other hand, fully automatic systems have an upper bound on performance. To achieve better feedback in automatic systems in the absence of expert knowledge, some middle ground between fully automatic and fully manual feedback needs to be identified.

This query specific graph based query reformulation approach can be characterized somewhere between manual and automatic feedback which relies on an external resource, manually prepared by medical experts, for better feedback and does query processing automatically. It uses UMLS as a substitute of medical expert intervention.

Here we study the effect of using UMLS entities in query processing for clinical

decision support systems. We present a graph based query processing technique that takes advantage of UMLS knowledge resource. This technique generates query specific graph of related entities, weights the entities and uses them to expand the query. This query reformulation approach is compared with baseline, pseudo relevance feedback based query expansion approach, and state-of-the-art UMLS based query reformulation approaches. The proposed method can be generalized to any other domain if there exists a knowledge graph or ontology for that domain.

4.2.1 Graph creation using UMLS

The proposed approach generates query specific graph using biomedical entities and relations from UMLS. It finds UMLS concepts from the query and represents them as nodes in the graph. We call them query concepts or query entities. This concepts extraction task is done using Metamap [4], which identifies UMLS metathesaurus concepts referred in the text. MetaMap uses a knowledge-intensive approach based on symbolic, natural language processing (NLP) and computational linguistic techniques to map biomedical text to UMLS metathesaurus with 86% accuracy [4]. The concepts identified by Metatmap are represented as nodes in the graph. Along with concepts, UMLS also contains relations between entities. These relations for query concepts are used to expand nodes. Each query node gets expanded by its related UMLS concepts considering all types of relations within UMLS. After the node expansion, the expanded graph contains all the related concepts as nodes and relations as edges.

Two nodes in the graph can have an edge between them if and only if those two entities have some relation in UMLS. There are various types of relations present in UMLS, and this approach considers all types of relations i.e. no manual filtering is done based on the type of relations. This makes the approach more generalized for any type of biomedical queries. There can be some isolated nodes in the graph when any query concept is not related to any other query concept, or it does not have any common related concept with another query concept. Isolated query concepts in the graph, if there exists any, will not make any difference in this query reformulation process.

For example, the query “A 78 year old male presents with frequent stools and melena” has 6 query concepts identified by UMLS: ‘year’, ‘old’, ‘male’, ‘presents’, ‘frequent stools’

and ‘melena’. These concepts are expanded with other related concepts. A subset of constructed graph for this query is shown in Figure 4.1.

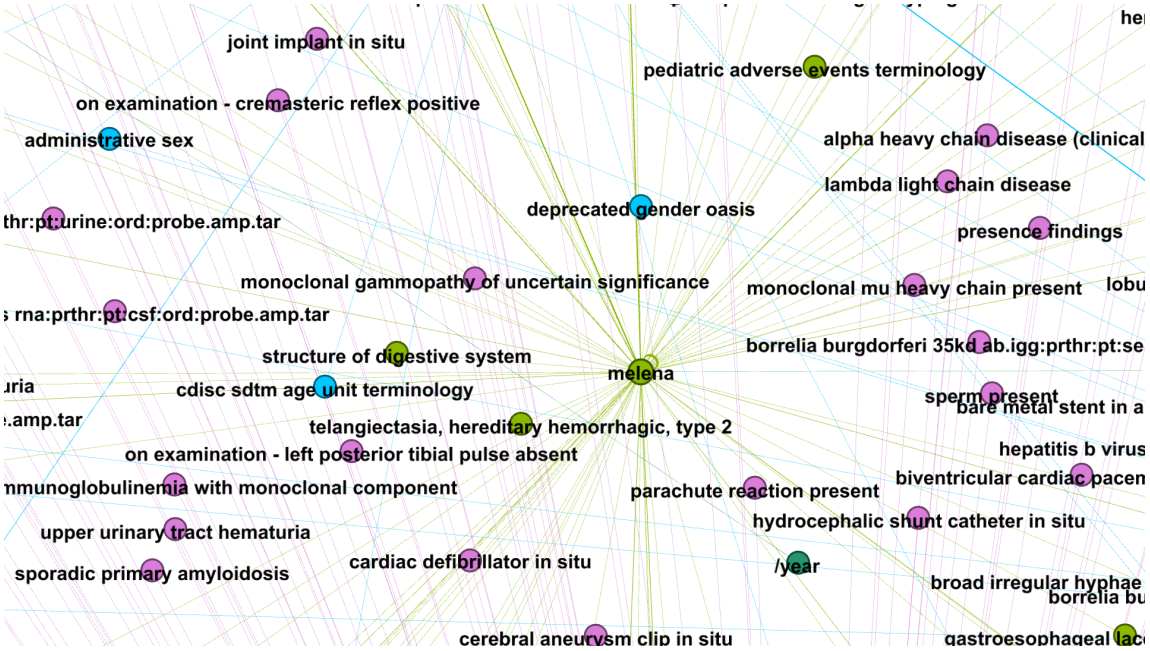

Figure 4.1: A subset of initial graph constructed for query “A 78 year old male presents with frequent stools and melena” (graph visualized using Gephi [14] software). Different colors are used just for better visualization. They do not signify anything.

4.2.2 Graph refinement using pseudo relevant documents

Once the graph is constructed using the method described in Section 4.2.1, it is further refined by assigning weights to the edges and removing some of the edges in the graph. The edge weights are calculated based on the co-occurrence value of entities in pseudo relevant documents. Pseudo-relevant documents are also parsed using Metamap to get the UMLS concepts from them. Top k retrieved documents are used as pseudo relevant documents in this process. For any edge between two entities, the number of times those two entities are co-occurring in top k retrieved documents is used as weight for that edge. The co-occurrence of two entities here is defined as the presence of both the entities in the same paragraph. If two entities are present in the same document but in different paragraphs, then they are not considered as co-occurring entities based on the assumption made here that two different paragraphs usually represent two different contexts. They might not be similar in terms of the information they refer.

The graph is further refined by removing some of the edges. The edges whose edge weights are less than a threshold (edge weight $\leq$ l) are removed from the graph. Less edge weight for an edge between two entities means those two entities rarely occur together and hence share very little or no context. Adding such entities in the query might lead to query drift which will lower the performance of the system. Figure 4.2 shows a refined graph with edge weights (edge weight $>$ threshold) for the query "A 78 year old male presents with frequent stools and melena".

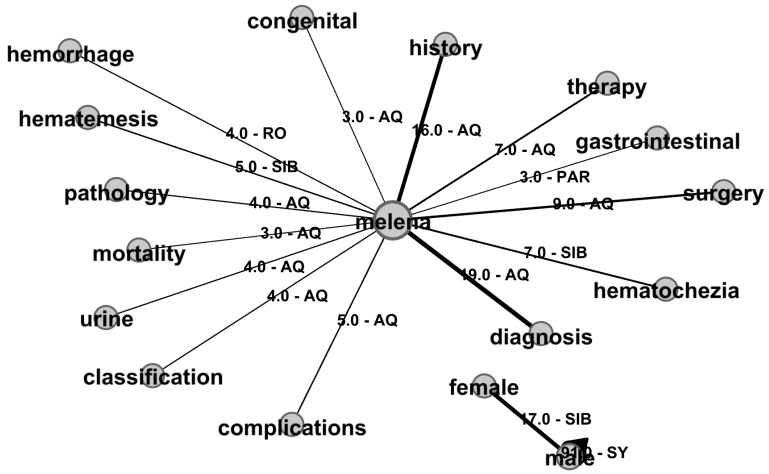

Figure 4.2: Refined graph with edge weights for query "A 78 year old male presents with frequent stools and melena" (graph visualized using Gephi software).

In the graph constructed for query "A 78 year old male presents with frequent stools and melena", the nodes 'year', 'old', 'presents' and 'frequent stools' were connected to the other nodes. But after removing the edges having edge weight less than a threshold, they got isolated, and hence they are not present in the refined graph shown in Figure 4.2. The numbers on edges show the weights of the edges, determined using the co-occurrences of two entities in the documents. The alphabetical tags on the edges are the abbreviations[1] for the type of relations between node entities in UMLS metathesaurus. For example, 5.0-SIB on edge between 'melena' and 'hematemesis' indicates that they are co-occurring 5 times in the document text, and they have a sibling type relation in UMLS metathesaurus.

[1]https://www.nlm.nih.gov/research/umls/knowledge_sources/ metathesaurus/release/abbreviations.html

4.2.3 Importance value of the entities in query-specific graph

The query graph now has weighted edges in it. The next step is to weight the nodes in the refined graph. Node weight represents the importance of that node in the graph, i.e. the importance of that entity in the graph for that particular query. We refer the generated graph as querygraph. A few techniques of node weighting are considered here.

Pagerank:

PageRank is a link analysis algorithm, originally designed to measure the importance of website pages. It assigns a numerical weighting to each element of a hyperlinked set of documents, such as the World Wide Web, to measure its relative importance within the set. The PageRank algorithm may be applied to any collection of entities that are linked internally and can be represented as a graph. The numerical weight that it assigns to any element is referred to as the PageRank of that element. The PageRank value indicates the importance of a particular node. An edge to a node counts as a vote of support. The PageRank of a node is defined recursively and depends on the number of edges to that node and PageRank metric of all nodes that are connected to it. A node that is connected to many nodes with high PageRank receives a high PageRank itself. The PageRank computations iteratively adjust approximate PageRank values to more closely reflect the theoretical true value. The PageRank algorithm on the graph of UMLS entities will iteratively compute PageRank value of each UMLS entity based on the edges that are connected to that entity. The formula for the Pagerank value of node n is given by the equation:

$$PR(n) = \sum_{i \in C(n)} \frac{PR(i)}{L(i)}$$

where $C(n)$ is the set of UMLS entities connected to node n and $L(i)$ is the number of edges on node i.

Figure 4.3 shows the graph with PageRank weights for the query “A 78 year old male presents with frequent stools and melena”.

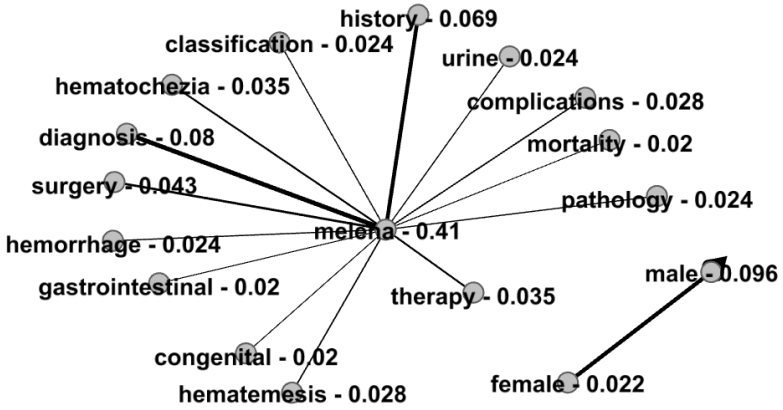

Figure 4.3: Graph with node weights assigned using PageRank algorithm for query "A 78 year old male presents with frequent stools and melena" (graph visualized using Gephi).

Degree Centrality (DC):

Degree Centrality or degree of any node is defined as the number of links/edges associated with that node. It is given as the following:

$$\begin{aligned} DC(n) &= degree(n) \\ &= number\ of\ edges\ connected\ to\ node\ n \end{aligned}$$

Sigmoid Degree Centrality (sig_DC):

This measure considers the degree centrality with sigmoid function as weights for the nodes. Sigmoid Degree Centrality is defined as the following:

$$\begin{aligned} sig\_DC(n) &= sigmoid(degree(n)) \\ &= \frac{1}{1 + e^{degree(n)}} \end{aligned}$$

Normalized Degree Centrality (norm_DC):

Normalized Degree Centrality is defined as the degree centrality of a node normalized using the degree centrality of all nodes. It is given by the formula:

$$norm\_DC(n) = \frac{degree(n)}{\sum_{i=1}^{n} degree(i)}$$

Weighted Degree Centrality (WDC):

Weighted Degree Centrality is defined as the sum of weighs of the edges incident upon a node.

$$\begin{aligned} WDC(n) &= weighted\_degree(n) \\ &= sum\ of\ weightes\ of\ edges\ connected\ to\ node\ n \end{aligned}$$

Sigmoid Weighted Degree Centrality (sig_WDC):

Sigmoid function on Weighted Degree Centrality is also considered as a node weighting measure. It is given as the following:

$$\begin{aligned} sig\_DC(n) &= sigmoid(weighted\_degree(n)) \\ &= \frac{1}{1 + e^{weighted\_degree(n)}} \end{aligned}$$

Normalized Weighted Degree Centrality (norm_WDC):

Normalized Weighted Degree Centrality measure is based on Weighted Degree Centrality, which is normalized using a logarithmic function. This measure is defined as the following:

$$norm\_WDC(n) = \frac{1}{1 + \frac{weighted\_degree(n)}{ln(weighted\_degree(n))}}$$

Figure 4.4 shows the graph with normalized weighted degree weights for the same query “A 78 year old male presents with frequent stools and melena”.

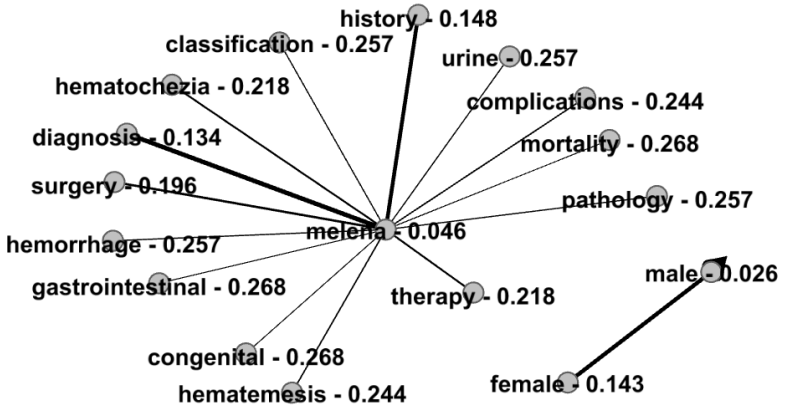

Figure 4.4: Graph with normalized weighted degree weights for query "A 78 year old male presents with frequent stools and melena" (graph visualized using Gephi software).

4.2.4 Query reformulation using weighted entities in query-specific graph

The query reformulation process uses the generated graph with node weights. Each query will have its own specific graph having the knowledge from UMLS incorporated with biomedical entities and context from top retrieved documents. The queries are reformulated by adding entities from query-specific graph along with their weights into the original query. So, the new query consists of the original query and entities with weights from its generated graph. These new queries are then expanded using pseudo relevance feedback based query expansion method.

The complete process of query reformulation using query-specific graph from UMLS is given below:

Algorithm 3: Query reformulation using UMLS graph.

for each query Q **do**
    Identify UMLS concepts from query Q and call them query concepts
    Represent these query concepts as nodes in graph
    for each query concept in graph **do**
        Find all related concepts to this query concept using UMLS
        Add these related concepts in graph as nodes
        Add connecting edges between related concepts and query concept
    for every edge in graph **do**
        assign edge_weight = co-occurrence value of two entities in top retrieved docs
    Remove the edges having low edge weight
    for each node in the graph **do**
        Assign importance weight to the node
    Reformulate the query as:
    New query Q_{New} = Q + all weighted entities from graph

4.3 Experiment details

We conducted a set of experiments using the summarization methods described in the previous section. This section describes the experimental setup.

4.3.1 Dataset and Evaluation Metrics

The retrieval experiments are performed on CDS 2014, CDS 2015 and CDS 2016 datasets. Evaluation metrics infNDCG [152] (described in Section 3.1), infAP [152], P@10 and Rprec are used here, which were used in TREC CDS 2015 and 2016 tracks. These are the standard measures for an information retrieval system to evaluate how satisfactory the search results are to the user's query intent. P@10 is precision at 10 measure, which is calculated as the following:

$$P@10 = \frac{\#no\ of\ relevant\ docs\ from\ top\ 10\ retrieved\ docs}{10}$$

Rprec measure indicates the precision at the position equal to total number of relevant documents available. It is calculated by the following formula.

$$Rprec = \frac{\#no\ of\ relevant\ docs\ from\ top\ k\ retrieved\ docs}{k}$$

where, k = total no of relevant documents available

infAP indicates inferred average precision. AP (average precision) is given by the following:

$$AP = \frac{\sum_{i=1}^{n} P@i \star rel(i)}{\#no\ of\ relevant\ docs}$$

where, rel(i) is 1 if the item at rank i is a relevant document, zero otherwise.

4.3.2 Experimental Setup

The retrieval experiments are performed on CDS 2015 and CDS 2016 datasets using Terrier[97] plateform. Queries and the documents are processed using Metamap[7] to identify UMLS concepts from them. Metamap is a software tool developed by National Library of Medicine which maps biomedical text to UMLS metathesaurus. The queries are processed using Metamap and UMLS concepts are identified. To expand the nodes of the graph with related entities and refine it based on co-occurrence values, the top 10 retrieved documents are taken into consideration and parsed using Metamap for the queries of CDS 2015 dataset, while for CDS 2016 queries, the top 100 documents were considered. The number of pseudo relevant documents used for graph refinement in both the datasets are chosen empirically. The document retrieval is done using BM25 ranking model for the summary part of queries, while Bo1 model is used in PRF.

4.4 Results

The results of the experiments of UMLS graph based query reformulation performed using various node weighting techniques are given in Table 4.2 for CDS 2014, CDS 2015 and

CDS 2016 datasets.

Table 4.2: InfNDCG results of UMLS query-specific graph based query reformulation using various weighting techniques. The highest results for each dataset are given in bold.

Method	**CDS 2014**	**CDS 2015**	**CDS 2016**
BM25	0.1398	0.1591	0.1548
UMLS_graph_no_weights	0.1249	0.1600	0.1993
UMLS_graph_pagerank	0.1375	0.1661	0.1619
UMLS_graph_DC	0.1236	0.1551	0.1814
UMLS_graph_WDC	0.1015	0.1168	0.1492
UMLS_graph_norm_DC	0.1351	0.1673	0.1654
UMLS_graph_sig_DC	0.1299	0.1613	0.1986
UMLS_graph_sig_WDC	0.1239	0.1604	0.1983
UMLS_graph_norm_WDC	0.139	0.1665	0.1907
BM25 + PRF	0.1969	0.2104	0.1916
UMLS_graph_no_weights + PRF	0.1570	0.2022	0.2359
UMLS_graph_pagerank + PRF	**0.2024**	**0.2154**	0.2154
UMLS_graph_DC + PRF	0.1556	0.1886	0.2381
UMLS_graph_WDC + PRF	0.1307	0.1480	0.1862
UMLS_graph_norm_DC + PRF	0.1961	0.2066	0.2160
UMLS_graph_sig_DC + PRF	0.1692	0.2074	**0.2424**
UMLS_graph_sig_WDC + PRF	0.1573	0.2028	0.2374
UMLS_graph_norm_WDC + PRF	0.1887	0.2114	0.2351

For CDS 2014 dataset, UMLS_graph_pagerank+PRF performed better than the BM25+PRF. For CDS 2015 dataset, all the variations of UMLS graph except UMLS_graph_DC and UMLS_graph_WDC performed better than BM25. UMLS_graph_pagerank+PRF and UMLS_graph_norm_WDC+PRF performed better than BM25+PRF. For CDS 2016, all the variations of UMLS graph except UMLS_graph_WDC, with and without PRF, helped to get better performance than BM25.

The comparison of the results of UMLS graph based query reformulation with no weights, with weights calculated by PageRank and with normalized weighted degree

weights are shown in Table 4.3 along with baseline results and state-of-the-art results for CDS 2015 and CDS 2016 datasets. Bold values represent the highest results. $\star$ represents statistically significant result with $p < 0.05$ when compared to BM25. $\star\star$ and ‡ represent statistically significant result with $p < 0.05$ and $p < 0.1$, respectively, when compared to BM25 + PRF.

Table 4.3: InfNDCG results of UMLS query-specific graph based query reformulation on CDS 2015 and CDS 2016 datasets. Value in bracket shows percentage increment from baseline result BM25.

Method	**CDS 2015**	**CDS 2016**
BM25	0.1591	0.1625
UMLS_graph_no_weights	0.1600 (+1%)	$0.1993^{\dagger}$ (+23%)
UMLS_graph_pagerank	$0.1661^{\star}$ (+4%)	0.1619
UMLS_graph_norm_WDC	$0.1665^{\star}$ (+5%)	$0.1907^{\star}$ (+17%)
BM25 + PRF	0.2104 (+32%)	0.1916 (+18%)
Best run from CDS task	0.2339 [98]	0.2265 [2]
unigram_SMDB [154]	0.2173	0.1843
UMLS_graph_no_weights + PRF	0.2022 (+27%)	**0.2359‡ (+45%)**
UMLS_graph_pagerank + PRF	**0.2154 (+35%)**	$0.2154^{\ddagger}$ (+33%)
UMLS_graph_norm_WDC + PRF	0.2114 (+33%)	**0.2351** (+45%)**

For CDS 2015 and CDS 2016 datasets, our results are compared with the best infNDCG result among the submitted runs (which makes use of UMLS) during TREC CDS 2015 and TREC CDS 2016, respectively. For comparison of the results, we have considered only those submitted runs that use UMLS for query reformulation in the standard retrieval framework.

In TREC CDS 2015, UMLS based system performing highest was Palotti and Hanbury [98] where queries were reformulated using UMLS concepts, triggered names and preferred names with fixed weights and pseudo relevance feedback based query expansion. In TREC CDS 2016, Agrafiotes and Arampatzis [2] reported the highest results using UMLS atoms in the query. These two results are directly taken from the papers and used here for comparison. In Table 4.3, Our results are also compared with results of state-of-the-art

query reformulation method using UMLS, unigram_SMDB, from Zhang and He [154] where they did query expansion based on diagnosis prediction from SemMedDB using UMLS concepts. Our method UMLS_graph_pagerank + PRF performs highest with increment 35% from baseline on CDS 2015 dataset while UMLS_graph_no_weights + PRF and UMLS_graph_norm_WDC + PRF perform highest with increment 45% from baseline on CDS 2016 dataset.

The method query-specific UMLS graph based query reformulation helps to get semantically related terms to the query terms. For example, the query 'A 78 year old male presents with frequent stools and melena' gets terms like 'ematochezia', 'gastrointestina', and 'hemorrhage' which helped to get more relevant documents after query reformulation. These terms are weighted based on the co-occurrences in the top retrieved documents which helped to fetch more relevant documents. Also the query terms 'melena' and 'male' get more weights from the query-specific graphs. The updated query after query-specific graph based query reformulation using UMLS is given below:

Initial query:
$A^1\ 78^1\ year^1\ old^1\ male^1\ presents^1\ with^1\ frequent^1\ stools^1\ and^1\ melena^1$

Updated query after pagerank (from graph shown in Figure 4.3):
$A^1\ 78^1\ year^1\ old^1\ male^1\ presents^1\ with^1\ frequent^1\ stools^1\ and^1\ melena^1\ melena^{0.41}$
$male^{0.096}\ diagnosis^{0.08}\ history^{0.069}\ surgery^{0.043}\ hematochezia^{0.035}\ therapy^{0.035}$
$complications^{0.028}\ hematemesis^{0.028}\ classification^{0.024}\ hemorrhage^{0.024}\ urine^{0.024}$
$pathology^{0.024}\ female^{0.022}\ mortality^{0.02}\ gastrointestinal^{0.02}\ congenital^{0.02}$

Updated query after normalized weighted degree (from graph shown in Figure 4.4):
$A^1\ 78^1\ year^1\ old^1\ male^1\ presents^1\ with^1\ frequent^1\ stools^1\ and^1\ melena^1\ male^{0.026}$
$melena^{0.046}\ diagnosis^{0.134}\ female^{0.143}\ history^{0.148}\ surgery^{0.196}\ therapy^{0.218}$
$hematochezia^{0.218}\ hematemesis^{0.244}\ complications^{0.244}\ pathology^{0.257}\ hemorrhage^{0.257}$
$classification^{0.257}\ urine^{0.257}\ mortality^{0.268}\ congenital^{0.268}\ gastrointestinal^{0.268}$

Evaluation results infNDCG, infAP, P@10 and Rprec on CDS 2015 and CDS 2016

datasets are shown in Table 4.4 and 4.5, respectively. Bold values represent the highest results. $\star$, $\diamond$, and † represent statistically significant results with $p < 0.05$, $p < 0.01$ and $p < 0.1$, respectively, when compared to baseline BM25. $\star\star$ and ‡ represent statistically significant result with $p < 0.05$ and $p < 0.1$, respectively, when compared to baseline BM25 + PRF.

Table 4.4: Results of UMLS query-specific graph based query reformulation on CDS 2015. Bold values represent highest results.

CDS 2015	**infNDCG**	**infAP**	**P@10**	**Rprec**
BM25	0.1591	0.0260	0.3133	0.1175
UMLS_graph_no_weights	0.1600	0.0264	0.3100	0.1230
UMLS_graph_pagerank	0.1661*	**0.0279***	**0.3300**†	0.1232
UMLS_graph_norm_WDC	**0.1665***	0.0277*	0.3267	**0.1240***
BM25 + PRF	0.2104	0.0423	**0.3600**	0.1514
UMLS_graph_no_weights + PRF	0.2022	0.0403	**0.3600**	0.1454
UMLS_graph_pagerank + PRF	**0.2154**	**0.0442**	0.3567	**0.1576**
UMLS_graph_norm_WDC + PRF	0.2114	0.0425	**0.3600**	0.1517

Table 4.5: Results of UMLS query-specific graph based query reformulation on CDS 2016. Bold values represent highest results.

CDS 2016	**infNDCG**	**infAP**	**P@10**	**Rprec**
BM25	0.1625	0.0176	0.2367	0.0970
UMLS_graph_no_weights	**0.1993**†	**0.0247**†	**0.2833**	0.1200*
UMLS_graph_pagerank	0.1619	0.0166	0.2333	0.1073*
UMLS_graph_norm_WDC	0.1907*	0.0232†	0.2767	**0.1232**$^{\diamond}$
BM25 + PRF	0.1916	0.0275	0.2967	0.1280
UMLS_graph_no_weights + PRF	**0.2359**‡	**0.0362**	**0.3633****	0.1251
UMLS_graph_pagerank + PRF	0.2154‡	0.0319‡	0.2967	0.1285
UMLS_graph_norm_WDC + PRF	0.2351**	0.0345	0.3367	**0.1331**

Statistical significance test on results shows that the results are significantly better than the baselines. UMLS graph based query reformulation helped retrieve more relevant

documents and hence all four evaluation measures show improvement.

Table 4.6: Query category wise infNDCG results of UMLS query-specific graph based query reformulation on CDS 2015. Bold values represent highest results.

CDS 2015	**all**	**Diagnosis**	**Test**	**Treatment**
BM25	0.1591	0.1469	0.1755	0.1547
UMLS_graph_no_weights	0.1600	0.1359	0.1840	0.1602
UMLS_graph_pagerank	0.1661*	0.1438	0.1874†	**0.1670***
UMLS_graph_DC	0.1551	0.1299	0.1881	0.1472
UMLS_graph_WDC	0.1168	0.1009	0.1355	0.1141
UMLS_graph_norm_DC	**0.1673**†	0.1433	**0.1884**†	**0.1702**†
UMLS_graph_sig_DC	0.1613	0.1331	0.1875†	0.1633
UMLS_graph_sig_WDC	0.1604	0.1362	0.1841	0.1610
UMLS_graph_norm_WDC	**0.1665***	**0.1515**	0.1824†	0.1657†
BM25 + PRF	0.2104	**0.2032**	0.2043	0.2236
UMLS_graph_no_weights + PRF	0.2022	0.1873	0.1944	0.2247
UMLS_graph_pagerank + PRF	**0.2154**	0.1982	**0.2189**†	**0.2292**
UMLS_graph_DC + PRF	0.1886	0.1559	0.1926	0.2172
UMLS_graph_WDC + PRF	0.1480	0.1267	0.1406	0.1767
UMLS_graph_norm_DC + PRF	0.2066	0.1965	0.1920	0.2313
UMLS_graph_sig_DC + PRF	0.2074	0.1844	0.2096	0.2284
UMLS_graph_sig_WDC + PRF	0.2028	0.1834	0.1996	0.2254
UMLS_graph_norm_WDC + PRF	0.2114	0.2009	0.2072	0.2260

For three categories of queries: 'diagnosis', 'test' and 'treatment', category wise results along with overall results on CDS 2015 and CDS 2016 datasets are shown in Table 4.6 and Table 4.7, respectively. Bold values represent the highest results. $\star$ and † represent statistically significant result with $p < 0.05$ and $p < 0.1$, respectively, when compared to baseline BM25. $\star\star$ and ‡ represent statistically significant result with $p < 0.05$ and $p < 0.1$, respectively, when compared to baseline BM25 + PRF.

Table 4.7: Query category wise infNDCG results of UMLS query-specific graph based query reformulation on CDS 2016. Bold values represent highest results.

CDS 2016	all	Diagnosis	Test	Treatment
BM25	0.1625	0.1616	0.1972	0.1287
UMLS_graph_no_weights	**0.1993**†	0.1547	0.1798	**0.2633***
UMLS_graph_pagerank	0.1619	0.1415	0.1891	0.1550†
UMLS_graph_DC	0.1814	0.1504	0.1862	0.2076
UMLS_graph_WDC	0.1492	0.1272	0.1425	0.1779
UMLS_graph_norm_DC	0.1658	0.1430	0.1882	0.1663†
UMLS_graph_sig_DC	0.1986†	0.1511	0.1885	0.2562*
UMLS_graph_sig_WDC	0.1983†	0.1574	0.1789	0.2588*
UMLS_graph_norm_WDC	**0.1907***	**0.1656**	**0.2064**	0.2001*
BM25 + PRF	0.1916	0.1802	0.2062	0.1885
UMLS_graph_no_weights + PRF	0.2359†	0.1736	0.2035	**0.3306**†
UMLS_graph_pagerank + PRF	0.2154†	**0.1915***	0.2062	0.2486
UMLS_graph_DC + PRF	0.2381†	0.1780	0.2023	0.3338†
UMLS_graph_WDC + PRF	0.1862	0.1547	0.1388	0.2651
UMLS_graph_norm_DC + PRF	0.2160	0.1649	0.2163	0.2668†
UMLS_graph_sig_DC + PRF	**0.2424**†	0.1759	0.2149	**0.3365***
UMLS_graph_sig_WDC + PRF	0.2374†	0.1736	0.2037	0.3350†
UMLS_graph_norm_WDC + PRF	**0.2351***	0.1844	**0.2238**	0.2971†

Considering all the queries of CDS 2015, UMLS_graph_norm_DC gave the highest and significant ($p < 0.1$) result, but the results of UMLS_graph_pagerank and UMLS_graph_norm-_WDC are more significant with $p < 0.05$. These three techniques, UMLS_graph_pagerank, UMLS_graph_norm_DC and UMLS_graph_norm_WDC gave statistically significant results for 'test' and 'treatment' type of queries also. In CDS 2016, all the variations using UMLS graph, except UMLS_graph_WDC and UMLS_graph_WDC+PRF, are giving better results, and most of them are statistically significant for all queries as well as 'treatment' type of queries.

Query wise infNDCG difference graph of UMLS_graph_norm_WDC+PRF with base-

line BM25+PRF for CDS 2016 is shown in Figure 4.5. From the graph, we can see that out of 30 queries, 11 queries improve, 9 queries degrade, and others remain the same. But the overall increment is three times higher than the decrements in queries. So, the overall performance increased reasonably.

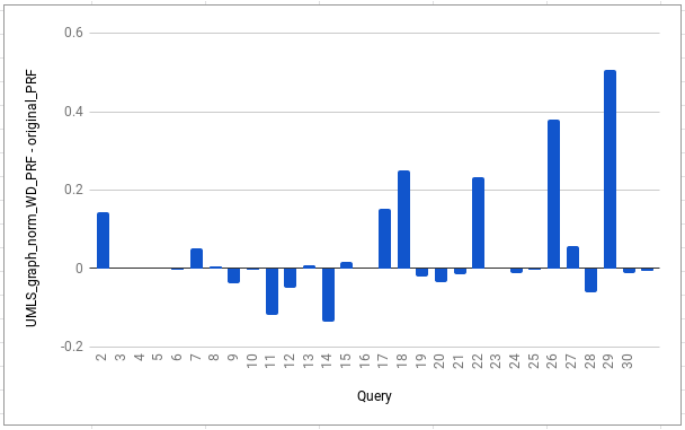

Figure 4.5: Query wise difference graph between UMLS_graph_norm_WDC+PRF and BM25+PRF for CDS 2016.

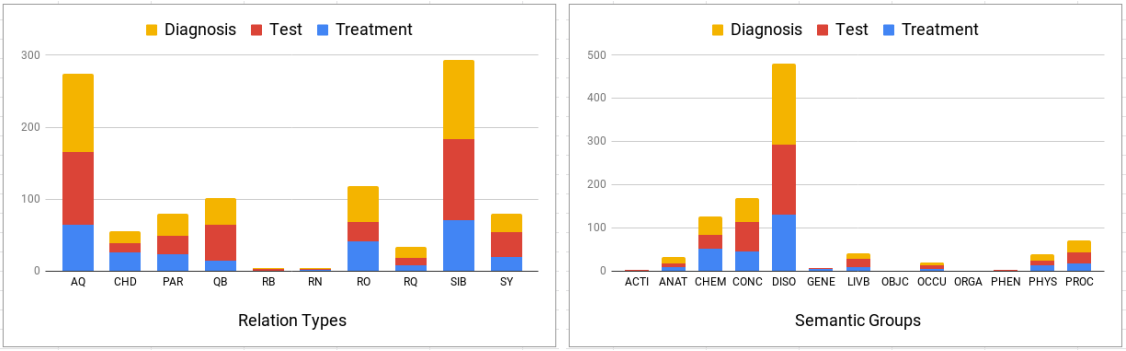

Figure 4.6: Distribution of relation types and semantic group types of entities in 'diagnosis', 'test' and 'treatment' queries of CDS 2016.

Figure 4.6 shows category wise distribution of semantic groups of entities and their relation types in the constructed graph for 'diagnosis', 'test and 'treatment' type of reformulated queries of CDS 2016. For 'diagnosis' type of queries, relations of type RQ(related and possibly synonymous) and entities of semantic types from DISO(disorders) are more in the querygraphs. For 'test' type of queries, the relations of type QB(qualified by) and SY(synonym) are more in the querygraphs, while entities of type CONC(Concepts & Ideas) are more. For 'treatment' type of queries, CHEM(Chemicals & Drugs) type of entities are

more. PAR(parent relationship) type of relations and PROC(procedures) type of entities are equally distributed in querygraphs of all three types of queries.

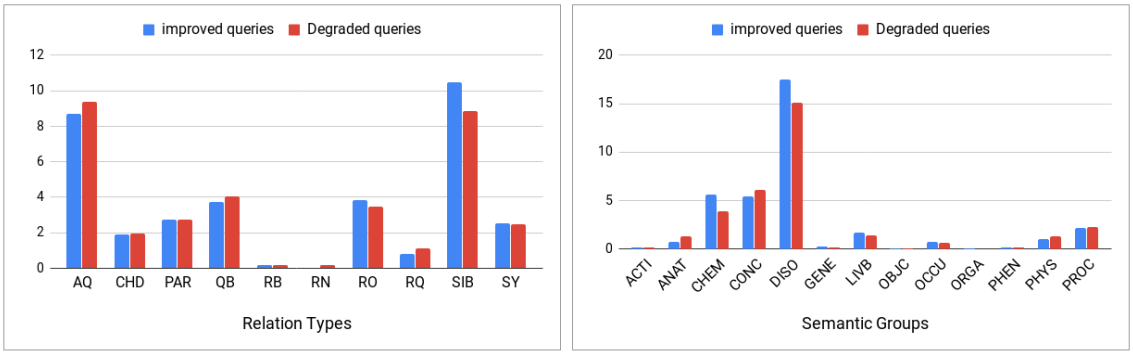

Figure 4.7: Distribution of relation types and semantic groups of entities in the improved queries and degraded queries of CDS 2016.

The distribution of relation types and semantic groups of entities in improved queries and degraded queries are shown in Figure 4.7. In the querygraphs of improved queries, SIB(sibling relationship) and RO(relationship other than synonymous, narrower, or broader) type of relations are more, while the relations of type CHD(child relationship), PAR(parent relationship), RB(broader relationship) and SY(synonymy) are same in improved as well as degraded queries. DISO(disorders), CHEM(chemicals & drugs) and LIVB(living beings) type of entities are more in the querygraphs of improved queries which may be the reason for improvement in retrieval.

4.5 Conclusion

This chapter presented entity-based query processing and reformulation techniques for biomedical document retrieval systems. The external knowledge source UMLS has been used to identify biomedical concepts from queries and query expansion is carried out with the identified UMLS concepts. Later, UMLS knowledge about the semantics of concepts and relations has been used in query-specific graph based query reformulation technique for retrieval. The method uses UMLS entities and their relations from UMLS to construct the query-specific graph and refines the graph based on the co-occurrence statistics from top retrieved documents. The entities from the graph with their weights are added to the original

queries. The experiments show that the query-specific graph based query reformulation using UMLS helped to improve retrieval results. The results are significantly better with 4%-5% improvement on CDS 2015 dataset and 23%-27% improvement on CDS 2016 dataset. The proposed method is also promising for 'test' and 'treatment' type of queries. The next chapter explores the use of these query reformulation techniques for query-focused biomedical text summarization systems. Standard techniques of text summarization are also discussed for biomedical domain, along with query-focused summarization techniques.

CHAPTER 5

Query-focused biomedical text summarization using UMLS graph

In the last two chapters, we have seen the query reformulation techniques for biomedical document retrieval systems. This chapter focuses on query-focused biomedical text summarization systems and query modification techniques for them. Query-focused summarization systems need to take query in the prime focus and generate the summaries according to it. The generated summaries should be able to answer the given query. We present various techniques of query-focused text summarization based on the sentence-sentence and query-sentence similarity measures. We also consider the query-specific graph based query reformulation method described in the previous chapter and incorporate it in the summarization process. We explore the word-embedding based summarization techniques where sentence-sentence similarity is calculated using the distances between words in the embedding space.

5.1 Summarization Methods with query modifications

This section describes standard summarization techniques, query-sentence matching based summarization method, modified query-sentence matching based summarization method using UMLS query graph, modified lexrank using UMLS query-graph and word2vec.

5.1.1 TextRank and LexRank

TextRank [79] and LexRank [43] are two standard unsupervised approaches used for text summarization. Both are graph-based ranking algorithms where a graph is constructed

using sentences as nodes. The edges in the graph are weighted based on the content overlap or the similarity between two sentences. This sentence-sentence similarity can be lexical or semantic or a combination of both. TextRank uses the similarity measure based on the number of words two sentences have in common while LexRank uses cosine similarity of TF-IDF vectors. The similarity measure used by TextRank is given by:

$$sim(s_i, s_j) = \frac{|w_k | w_k \in s_i \ \ \& \ \ w_k \in s_j|}{log(|s_i|) \times log(|s_j|)}$$

where, $|s_i|$ and $|s_i|$ are the number of words in the sentences s_i and s_i, respectively. The similarity measure used by LexRank is defined as the following:

$$sim(s_i, s_j) = \frac{\sum_{w \in s_i, s_j} (tf_{w,s_i} \times idf_w) \times (tfw, s_j \times idf_w)}{\sqrt{\sum_{w \in s_i} (tf_{w,s_i} \times idf_w)^2} \sqrt{\sum_{w \in s_j} (tf_{w,s_j} \times idf_w)^2}}$$

where tf_{w,s_i} is the term frequency of word w in sentence s_i. Term frequency is defined as the number of occurrences of the word in the sentence divided by the total number of the words in the sentence. idf_w is the total number of sentences divided by the number of sentences containing word w.

In the graph, edges were formed between the sentences having similarity greater than the threshold. In both algorithms, the sentences are ranked based on the similarities with other sentences. PageRank algorithm is applied to the resulting graph where each sentence node gets weighted based on the weights of the connected sentences. Pagerank works iteratively till the node weights remain unchanged. The summary is formed by combining the top ranking sentences, using a length cutoff to limit the size of the summary.

5.1.2 Query-Sentence matching

The Query Sentence Matching (QSM) based summarization method [20, 40] compares all the sentences with the query and takes top similar sentences to be included in the summary. The queries and all the sentences in snippets are represented by vectors of tf-idf values of words in the sentences. The similarity measure used to match query vector and sentence vector is cosine similarity. It is calculated by using the following formula:

$$sim(q,s) = \frac{\sum_{w \in q,s} (tf_{w,q} \times idf_w) \times (tf_{w,s} \times idf_w)}{\sqrt{\sum_{w \in q} (tf_{w,q} idf_w)^2} \sqrt{\sum_{w \in s} (tf_{w,s} idf_w)^2}}$$

where $tf_{w,q}$ and $tf_{w,s}$ are the term frequencies of word w in the query q and sentence s, respectively. idf_w is the inverse of the number of sentences with word w normalized by the total number of sentences.

5.1.3 UMLS graph based query-sentence matching

The UMLS graph based query-sentence matching (UMLS_querygraph_QSM) summarization method is a modified version of QSM which uses query-specific graphs generated using UMLS to get the importance of words. For each query, it generates a graph using method described in Section 4.2. This method uses concepts identified using graph based method along with weights. The weights are incorporated in the similarity measure while ranking the sentences for the summary. The UMLS query-graph based cosine similarity between a query and a sentence is calculated using the following formula:

$$sim(q,s) = \frac{\sum_{w \in q,s} (tf_{w,q} \times idf_w + W_{w,q})(tf_{w,s} \times idf_w)}{\sqrt{\sum_{w \in q} (tf_{w,q} \times idf_w + W_{w,q})^2} \sqrt{\sum_{w \in s} (tf_{w,s} \times idf_w)^2}}$$

where,

$W_{w,q}$ = importance of concept w from query-graph of q, if w is in query-graph

= 0, otherwise

$tf_{w,q}$ and $tf_{w,s}$ are the term frequencies of the word w in query q and sentence s, respectively. idf_w is the inverse of the number of sentences with word w normalized by the total number of sentences.

5.1.4 UMLS query graph based lexrank

The UMLS query-graph based lexrank (lexrank_UMLS_querygraph) is a modified version of lexrank which uses query-specific graphs generated using UMLS (as described in Section 4.2) to get the importance of words, matches sentences using weighted cosine similarity

measure, generates a graph of sentences and then applies Pagerank on the graph. The main difference with the method lexrank is UMLS query-graph weighted cosine similarity. The later processing is same as it is in lexrank. The UMLS query graph based weighted cosine similarity is given by:

$$sim(s_i, s_j) = \frac{\sum_{w \in s_i, s_j} ((tf_{w,s_i} + W_w) \times idf_w) \times ((tfw, s_j + W_w) \times idf_w)}{\sqrt{\sum_{w \in s_i} ((tf_{w,s_i} + W_w) \times idf_w)^2} \sqrt{\sum_{w \in s_j} ((tf_{w,s_j} + W_w) \times idf_w)^2}}$$

where,

W_w = importance of concept w from query-graph, if w is in query-graph

= 1, otherwise

$tf_{w,q}$ and $tf_{w,s}$ are the term frequencies of the word w in query q and sentence s, respectively. idf_w is the inverse of the number of sentences with word w normalized by the total number of sentences.

5.1.5 Lexrank with Word2Vec similarity

The summarization method lexrank with Word2Vec (lexrank_w2v) uses Word2Vec[80] in the similarity measure and uses lexrank for ranking the sentences. It incorporates Word2Vec word-embeddings based similarity between words into the sentence similarity measure. The sentences are represented as nodes and the sentence-sentence similarities represent the edges. The similarity measure is defined using weighted cosine similarity where weights are the Word2Vec similarity scores. It is given as the following:

$$sim(s_i, s_j) = \frac{\sum_{p \in s_i, q \in s_j} (tf_{p,s_i} \times idf_p) \times W2V_{p,q} \times (tfq, s_j \times idf_q)}{|S_i| \times |S_j|}$$

where,

$$|S_i| = \sqrt{\sum_{p,q \in s_i} (tf_{p,s_i} \times idf_p) \times W2V_{p,q} \times (tfq, s_i \times idf_q)} \ ,$$

$$|S_j| = \sqrt{\sum_{p,q \in s_j} (tf_{p,s_j} \times idf_p) \times W2V_{p,q} \times (tfq, s_j \times idf_q)}$$

Here, $tf_{w,s}$ is the term frequency of word w in the sentence s. $W2V_{p,q}$ is the Word2Vec similarity score of the two words p and q. idf_w is the inverse of the number of sentences with word w normalized by the total number of sentences.

With this similarity measure for edges, the sentences are ranked using PageRank and top weighted sentences are used to generate the summary.

5.1.6 UMLS query graph based lexrank with Word2Vec similarity

The summarization method UMLS query graph based lexrank with Word2Vec similarity (lexrank_w2v) combines the two methods UMLS query graph based lexrank and lexrank with Word2Vec similarity. It uses the query-specific graph generated using UMLS to weight the biomedical entities of the sentences which are semantically similar to the query. It also uses word-embedding based similarity scores of the words into the similarity measures of sentences and applies PageRank on the graph of sentences. The sentence-sentence similarity of the summarization method UMLS query graph based lexrank with Word2Vec is given below:

$$sim(s_i, s_j) = \frac{\sum_{p \in s_i, q \in s_j} ((tf_{p,s_i} + W_p) \times idf_p) \times W2V_{p,q} \times ((tf_{q,s_j} + W_q) \times idf_q)}{|S_i| \times |S_j|}$$

where,

$$|S_i| = \sqrt{\sum_{p,q \in s_i} ((tf_{p,s_i} + W_p) \times idf_p) \times W2V_{p,q} \times ((tf_{q,s_i} + W_q) \times idf_q)} \ ,$$

$$|S_j| = \sqrt{\sum_{p,q \in s_j} ((tf_{p,s_j} + W_p) \times idf_p) \times W2V_{p,q} \times ((tf_{q,s_j} + W_q) \times idf_q)}$$

$W_{w,q}$ = importance of concept w from query-graph of q, if w is in query-graph

= 0, otherwise

$W2V_{p,q}$ = Word2Vec similarity score of the two words p and q.

$tf_{w,s}$ is the term frequency of word w in the sentence s. idf_w is the inverse of the number of sentences with word w normalized by the total number of sentences.

5.2 Experiments and Results

This section describes the experiments performed for query-focused biomedical text summarization along with their results. For the experiments, the dataset of BioASQ[1] task 5b phase B challenge is used which is a benchmark dataset containing questions, in English, along with gold standard (reference) answers constructed by a team of biomedical experts. The test dataset has five different batches, each containing 100 questions. For each question, the relevant snippets are given and the ideal answer for that question need to be generated. The ideal answers are paragraph sized summaries so it's a case of multi-document summarization on relevant snippets. The experiments are performed using sumy[15] which is a standard toolkit for text summarization. The pretrained Word2Vec model trained using wikipedia articles, PUBMED absracts and PMC open-access articles [83] is used to compute semantic similarity in the methods lexrank_w2v and lexrank_UMLS_querygraph_w2v. For the evaluation of the summarization system, ROUGE [68] scores are used. We use ROUGE-2 Recall and ROUGE-SU4 Recall which were used as evaluation measures in BioASQ. We also use ROUGE-2 F-measure and ROUGE-SU4 F-measure for the evaluation.

The results of the summarization methods (described in Section 5.1) on all five test batches of BioASQ5 dataset are shown in Table 5.1 and Table 5.2 in terms of ROUGE-2 Recall and ROUGE-SU4 Recall, respectively. The results are compared with the highest result form all the participants' submitted runs (Results given on BioASQ website[2]). There were two teams who had achieved highest results. The first one was Mollá [84] who had compared the summaries generated using simple approaches like tfidf, word2vec and regression as well as deep learning approach using LSTM. He found that the results of simple approaches are better than the results of regression and LSTM on BioASQ 5b dataset. Their trivial method of selecting first n number of snippets performed better than all other methods for batch 1. For other batches of the dataset, simple method of selecting n snippets based on their similarity and ranking using tfidf and word2vec performed best. The another best performing team was Chandu et al. [26] who had experimented with different biomedical ontologies and various algorithms including agglomerative clustering, Maximum Marginal Relevance (MMR) and sentence compression.

[1] http://bioasq.org/

[2] http://participants-area.bioasq.org/oracle/results/taskB/phaseB/

Table 5.1: ROUGE-2 Recall results of query-focused summarization on BIOASQ5 dataset. Bold represents improved results. * and † represent statistical significance with $p < 0.05$ when compared to lexrank and lexrank_w2v, respectively.

	Batch 1	**Batch 2**	**Batch 3**	**Batch 4**	**Batch 5**
MQ-1 [84]	0.5470	0.5117	0.5771	0.5617	0.5184
MQ-2 [84]	0.5131	0.5351	0.6062	0.5822	0.5802
Oaqa5b [26]	-	0.1939	0.2005	0.6726	0.7064
Oaqa5b-tfidf [26]	-	0.1332	0.2010	0.6766	0.5773
Highest from BIOASQ5b	0.5470 [MQ-1]	0.5351 [MQ-2]	0.6062 [MQ-2]	0.6766 [Oaqa5b-tfidf]	0.7064 [Oaqa5b]
textrank	0.51881	0.53219	0.61788	0.61691	0.57595
QSM	0.53949	0.51927	0.58284	0.56974	0.55141
UMLS_querygraph_QSM	0.54469	0.51265	0.58951	0.57762	0.56889
lexrank	0.57164	0.56183	0.62557	0.61498	0.61600
lexrank_UMLS_querygraph	0.57934	0.55415	0.62782	0.60919	**0.63727***
lexrank_w2v	**0.63714**	**0.58883**	0.64739	**0.68106**	0.68917
lexrank_UMLS_querygraph_w2v	0.63702	0.58731	**0.65970†**	0.67914	**0.68923**

Table 5.2: ROUGE-SU4 Recall results of query-focused summarization on BIOASQ5 dataset. Bold represents improved results. † represents statistical significance with $p < 0.05$ when compared to lexrank_w2v.

	Batch 1	**Batch 2**	**Batch 3**	**Batch 4**	**Batch 5**
MQ-1 [84]	0.5599	0.5167	0.5813	0.5533	0.5189
MQ-2 [84]	0.5221	0.5384	0.6085	0.5742	0.5703
Oaqa5b [26]	-	0.1928	0.1982	0.6642	0.6962
Oaqa5b-tfidf [26]	-	0.1352	0.1993	0.6692	0.5747
Highest from BIOASQ5b	0.5599 [MQ-1]	0.5384 [MQ-2]	0.6085 [MQ-2]	0.6692 [Oaqa5b-tfidf]	0.6962 [Oaqa5b]
textrank	0.54185	0.55807	0.62477	0.63446	0.58013
QSM	0.55802	0.54324	0.59378	0.58078	0.56276
UMLS_querygraph_QSM	0.56070	0.53985	0.59884	0.59370	0.57849
lexrank	0.58874	0.58776	0.63584	0.62668	0.61692
lexrank_UMLS_querygraph	0.59513	0.57862	0.63842	0.62340	0.63599
lexrank_w2v	**0.64894**	**0.61257**	0.65070	**0.68993**	**0.68391**
lexrank_UMLS_querygraph_w2v	0.64469	0.60703	**0.66356**†	0.68865	0.68184

Table 5.3 and Table 5.4 show ROUGE-2 F-measure and ROUGE-SU4 F-measure results, respectively, for the summarization techniques on BioASQ5 dataset.

Table 5.3: ROUGE-2 F-measure results of query-focused summarization on BIOASQ5 dataset. Bold represents improved results. * represents statistical significance with $p < 0.05$ when compared to baseline lexrank. ‡ represents statistical significance with $p < 0.01$ when compared to lexrank_w2v.

	Batch 1	**Batch 2**	**Batch 3**	**Batch 4**	**Batch 5**
textrank	0.19838	0.18567	0.20888	0.24909	0.21848
QSM	0.21946	0.19911	0.21575	0.25155	0.21718
UMLS_querygraph_QSM	0.22000	0.19623	0.21736	0.25012	0.22053
lexrank	0.23048	0.20502	0.23207	0.26066	0.24558
lexrank_UMLS_querygraph	0.23244	0.20429	0.23463	0.25616	**0.25215***
lexrank_w2v	0.25814	0.22737	0.25651	0.29846	**0.28642**
lexrank_UMLS_querygraph_w2v	**0.25822**	**0.22747**	**0.26339‡**	**0.30132**	0.28630

Table 5.4: ROUGE-SU4 F-measure results of query-focused summarization on BIOASQ5 dataset. Bold represents improved results. ‡ represents statistical significance with $p < 0.01$ when compared to lexrank_w2v.

	Batch 1	**Batch 2**	**Batch 3**	**Batch 4**	**Batch 5**
textrank	0.1958	0.18035	0.20379	0.24188	0.21137
QSM	0.21542	0.19350	0.21262	0.24367	0.21170
UMLS_querygraph_QSM	0.21516	0.19172	0.21425	0.24278	0.21484
lexrank	0.22533	0.20130	0.22793	0.25181	0.23835
lexrank_UMLS_querygraph	0.22695	0.19988	0.23046	0.24801	0.24393
lexrank_w2v	**0.24993**	**0.22081**	0.24835	0.28746	**0.27606**
lexrank_UMLS_querygraph_w2v	0.24923	0.22038	**0.25472‡**	**0.29056**	0.27562

The results show that UMLS_querygraph_QSM gives improvement over QSM except for batch 2. Query modification using query-specific graph generated using UMLS helps to better match the important sentences for summaries. The method lexrank_UMLS_querygraph gives an improvement over lexrank for batch 1,3 and 5 of the dataset. For the other two batches, the results are comparable. For batch 5, improvements in ROUGE-2 Recall

and ROUGE-2 F-measure results of lexrank_UMLS_querygraph are statistically significant with respect to the results of lexrank. The method lexrank_UMLS_querygraph_w2v gives significantly better results for batch 3 as compared to lexrank_w2v. The results are statistically significant for all four evaluation measures.

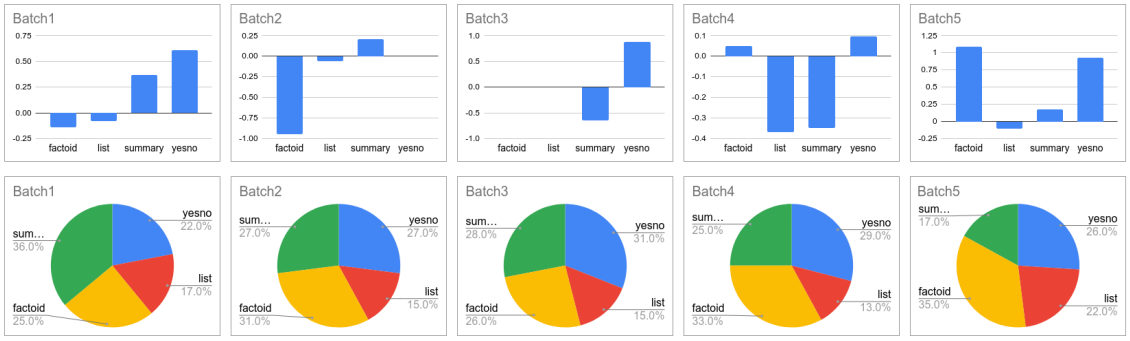

Figure 5.1: Query wise difference in the results of lexrank_UMLS_querygraph with baseline results lexrank and distribution of types of the queries.

The graphs in the first row of Figure 5.1 show the query type wise change in the results of lexrank_UMLS_querygraph as compared to lexrank for every batch of the data while the second row shows the batch wise distribution of the queries based on their types. From the graphs, we can say that the 'yesno' type of questions are getting improved in all batches (considering batch 2 where it is showing zero change: no improvement and no deterioration). The graph of batch 5 indicates that the overall result improvement of the method lexrank_UMLS_querygraph is mainly due to the improvements in 'factoid' and 'yesno' type of queries. For batch 2 and 4 where lexrank_UMLS_querygraph failed, decrements in 'factoid', 'list' and 'summary' type of queries must be the reason.

ROUGE measures surface level lexical similarity between system-generated summary and reference summary. The measures are based on the overlaps of word sequences like word pairs, n-gram which makes them unsuitable for summaries with substantial paraphrasing. The research is being carried out to improve automatic summarization evaluation. Ng and Abrecht [94] proposed ROUGE-WE measures where word embeddings are used to measure the semantic similarity of the words used in summaries instead of computing lexical similarity in ROUGE. ROUGE-WE-1 outperformed leading state-of-the-art measures. ROUGE-WE-2 is also a better evaluation measure than ROUGE-2. They also show that ROUGE-WE-SU4 takes readability into account while evaluating summaries.

Table 5.5: ROUGE-WE results of query-focused summarization on BIOASQ5 dataset. Bold represents improved results. $\star$ and † represent statistical significance with $p < 0.05$ and $p < 0.1$, respectively, when compared to lexrank_w2v.

		Batch 1	**Batch 2**	**Batch 3**	**Batch 4**	**Batch 5**
ROUGE-WE-1 F-measure scores						
ROUGE-WE-1	lexrank_w2v	0.25332	0.22886	0.24497	0.26566	0.25529
	lexrank_UMLS_querygraph_w2v	**0.25574**	**0.24109**$\star$	**0.25603**$\star$	**0.27143**	**0.26504**$\star$
ROUGE-WE-2 F-measure scores						
ROUGE-WE-2	lexrank_w2v	0.19622	0.17698	0.19784	0.21893	0.21055
	lexrank_UMLS_querygraph_w2v	**0.19878**	**0.17709**	**0.19977**	**0.22581**†	**0.21627**
ROUGE-WE-SU4 F-measure scores						
ROUGE-WE-SU4	lexrank_w2v	0.17013	0.15012	0.17243	0.18719	0.18209
	lexrank_UMLS_querygraph_w2v	**0.17611**	**0.15822**$\star$	**0.17443**	**0.19005**	**0.18741**†

Here, we also use ROUGE-WE-1, ROUGE-WE-2, and ROUGE-WE-SU4 evaluation measures to capture semantic similarities between the summaries generated using word embedding based summarization techniques and the reference summaries. Table 5.5 shows ROUGE-WE evaluation results for lexrank_w2v and lexrank_UMLS_querygraph_w2v summarization techniques. While computing ROUGE-WE scores, we used the same Word2Vec pre-trained model that was used in summarization. The results of summarization using lexrank_UMLS_querygraph_w2v are compared with the results of lexrank_w2v using ROUGE-WE-1 F-measure, ROUGE-WE-2 F-measure, and ROUGE-WE-SU4 F-measure. The method lexrank_UMLS_querygraph_w2v outperforms lexrank_w2v in all three ROUGE-WE measures for all five batches of BioASQ dataset. The weights of the query-specific graph generated using UMLS when incorporated in the similarity measures of summarization help to get better sentences for summaries and incorporating

word-embedding based similarity helps to get more semantically similar summaries.

5.3 Conclusion

For query-focused biomedical text summarization systems, various techniques have been analyzed on BioASQ dataset along with the proposed UMLS query-graph based summarization techniques. These techniques use reformulated queries by query-specific UMLS graph based method and incorporate them with the summarization methods QSM, textrank and textrank in the embedding space. The weights are determined using the statistics from the biomedical text for the candidate biomedical entities from queries and their semantically related entities identified by UMLS. These weights are then used in the similarity measures of the text summarization techniques. The experiments are performed on BioASQ 5b phaseB dataset for all 5 batches. From the comparison of their results with baselines and other top performing systems, we can conclude that UMLS query-graph based query processing is useful for query-focused biomedical text summarization also. Query-specific graph generated using UMLS when incorporated in the summarization methods either help to get better summaries with statistically significant improvement in the evaluation measure or give similar results to the original method. The result analysis based on question types shows that UMLS query graph weights of the entities, when incorporated in lexrank, helps to get better summaries for 'yesno' types of questions.

CHAPTER 6

Conclusion and future direction

This chapter presents an overview of the work that we discussed throughout the thesis and points out possible research directions. This thesis focuses on query processing for biomedical document retrieval and biomedical text summarization systems. Term selection and document selection techniques in query expansion for biomedical document retrieval and biomedical text summarization systems are explored here. A new approach of feedback document discovery for query expansion is proposed for literature document retrieval systems which is a combination of true relevance feedback and pseudo relevance feedback to optimize the cost of feedback as well as to improve the efficiency with query expansion. This approach is then improvised with feature weighting based on medical/non-medical entities for finding good feedback documents for query expansion. These approaches are showing improvement on TREC CDS datasets. The possible future research can be carried out in the direction to automatically determine the proportion of manual and learned feedback in feedback document discovery using partial feedback.

Considering the importance of the domain-specific entities in biomedical information systems, a novel method of entity based query processing using UMLS is proposed where UMLS concepts and their relations are utilized along with the statistics from the dataset to choose the expansion terms and their weights. The experiments of the query-specific UMLS graph based query reformulation when combined with pseudo relevance feedback show significant improvement in the results on TREC CDS datasets. The query category wise results are also promising and significant for all three generic clinical category types of queries. In the future, the research can be carried out to incorporate automatic learning of weights for expanded concepts in reformulated queries to enhance the retrieval system performance.

For query-focused biomedical text summarization systems, various techniques have been analyzed on BioASQ dataset along with the proposed UMLS query graph based summarization techniques. The proposed technique incorporates the weights of the candidate biomedical entities from queries and their semantically related entities identified by UMLS and statistics of biomedical text. It also considers word embedding based similarity between words while calculating the similarity between sentences. From the comparison of their results with baselines and other top performing systems results on BioASQ 5b phaseB dataset, we can conclude that UMLS query-graph based query processing is useful for query-focused biomedical text summarization also. The result analysis based on question type shows that it gives improvement for 'yesno' types of questions in every batch of the dataset. Word embedding based evaluation measures show a significant difference between summarization with query modification and without query modification on all five batches of BioASQ dataset. More sophisticated word embeddings based approaches like LSTM and BERT can be explored for query-focused biomedical text summarization in future. Collectively, we can conclude that accounting biomedical entities as features in text processing module is beneficial for biomedical information retrieval as well as biomedical text summarization systems.

References

[1] A. B. Abacha. Nlm nih at trec 2016 clinical decision support track. In *TREC*, 2016.

[2] C. Agrafiotes and A. Arampatzis. Augmenting medical queries with umls concepts via metamap. In *TREC*, 2016.

[3] R. Alfred, L. C. Leong, C. K. On, and P. Anthony. Malay named entity recognition based on rule-based approach. 2014.

[4] J. Allan. Incremental relevance feedback for information filtering. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 270–278. ACM, 1996.

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

[6] S. Ananiadou and J. McNaught. *Text mining for biology and biomedicine*. Citeseer, 2006.

[7] A. R. Aronson. Effective mapping of biomedical text to the umls metathesaurus: the metamap program. In *Proceedings of the AMIA Symposium*, page 17. American Medical Informatics Association, 2001.

[8] A. R. Aronson and T. C. Rindflesch. Query expansion using the umls metathesaurus. In *Proceedings of the AMIA Annual Fall Symposium*, page 485. American Medical Informatics Association, 1997.

[9] B. Audeh and M. Beigbeder. Emse at trec 2015 clinical decision support track. In *TREC*, 2015.

[10] M. N. Azadani and N. Ghadiri. Evaluating different similarity measures for automatic biomedical text summarization. In *International Conference on Intelligent Systems Design and Applications*, pages 305–314. Springer, 2017.

[11] A. Babashzadeh, J. Huang, and M. Daoud. Exploiting semantics for improving clinical information retrieval. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 801–804. ACM, 2013.

[12] S. Balaneshin-Kordan, A. Kotov, and R. Xisto. Wsu-ir at trec 2015 clinical decision support track: Joint weighting of explicit and latent medical query concepts from diverse sources. In *TREC*, 2015.

[13] S. Balaneshinkordan and A. Kotov. Bayesian approach to incorporating different types of biomedical knowledge bases into information retrieval systems for clinical decision support in precision medicine. *Journal of biomedical informatics*, 98:103238, 2019.

[14] M. Bastian, S. Heymann, and M. Jacomy. Gephi: an open source software for exploring and manipulating networks. In *Third international AAAI conference on weblogs and social media*, 2009.

[15] M. Belica. *Using sumy for summarization*.

[16] A. L. Berger, V. J. D. Pietra, and S. A. D. Pietra. A maximum entropy approach to natural language processing. *Computational linguistics*, 22(1):39–71, 1996.

[17] Bethesda (MD): National Library of Medicine (US); 2009 Sep-. 2, Metathesaurus.Available from: https://www.ncbi.nlm.nih.gov/books/NBK9684/. *UMLS® Reference Manual [Internet].*

[18] W. Boag, K. Wacome, T. Naumann, and A. Rumshisky. Cliner: A lightweight tool for clinical named entity recognition. *AMIA Joint Summits on Clinical Research Informatics (poster)*, 2015.

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

[20] R. Brandow, K. Mitze, and L. F. Rau. Automatic condensation of electronic publications by sentence selection. *Information Processing & Management*, 31(5):675–685, 1995.

[21] R. Bunescu, R. Ge, R. J. Kate, E. M. Marcotte, R. J. Mooney, A. K. Ramani, and Y. W. Wong. Comparative experiments on learning information extractors for proteins and their interactions. *Artificial intelligence in medicine*, 33(2):139–155, 2005.

[22] D. Campos, S. Matos, and J. L. Oliveira. Gimli: open source and high-performance biomedical name recognition. *BMC bioinformatics*, 14(1):54, 2013.

[23] G. Cao, J.-Y. Nie, J. Gao, and S. Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 243–250. ACM, 2008.

[24] Y. Cao, F. Liu, P. Simpson, L. Antieau, A. Bennett, J. J. Cimino, J. Ely, and H. Yu. Askhermes: An online question answering system for complex clinical questions. *Journal of biomedical informatics*, 44(2):277–288, 2011.

[25] C. Carpineto and G. Romano. A survey of automatic query expansion in information retrieval. *ACM Computing Surveys (CSUR)*, 44(1):1, 2012.

[26] K. Chandu, A. Naik, A. Chandrasekar, Z. Yang, N. Gupta, and E. Nyberg. Tackling biomedical text summarization: Oaqa at bioasq 5b. In *BioNLP 2017*, pages 58–66, 2017.

[27] P. Chen and R. Verma. A query-based medical information summarization system using ontology knowledge. In *19th IEEE Symposium on Computer-Based Medical Systems (CBMS'06)*, pages 37–42. IEEE, 2006.

[28] W. Chen, S. Moosavinasab, S. Rust, Y. Huang, S. M. Lin, A. Zemke, and A. Prinzbach. Evaluation of a machine learning method to rank pubmed central articles for clinical relevancy: Nch at trec 2016 clinical decision support track. In *TREC*, 2016.

[29] S. Choi, J. Choi, S. Yoo, H. Kim, and Y. Lee. Semantic concept-enriched dependence model for medical information retrieval. *Journal of biomedical informatics*, 47:18–27, 2014.

[30] G. O. Consortium. The gene ontology resource: 20 years and still going strong. *Nucleic acids research*, 47(D1):D330–D338, 2018.

[31] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[32] G. Crichton, S. Pyysalo, B. Chiu, and A. Korhonen. A neural network multi-task learning approach to biomedical named entity recognition. *BMC bioinformatics*, 18(1):368, 2017.

[33] U. De Lisboa. Novasearch at trec 2014 clinical decision support track. In *The 23rd Text REtrieval Conference (TREC 2014) Proceedings*, 2014.

[34] D. Demner-Fushman, S. Abhyankar, A. Jimeno-Yepes, R. F. Loane, B. Rance, F.-M. Lang, N. C. Ide, E. Apostolova, and A. R. Aronson. A knowledge-based approach to medical records retrieval. In *TREC*, 2011.

[35] D. Demner-Fushman, W. W. Chapman, and C. J. McDonald. What can natural language processing do for clinical decision support? *Journal of biomedical informatics*, 42(5):760–772, 2009.

[36] D. Dinh and A. Ben Abacha. CRP henri tudor at TREC 2014: Combining search results for clinical decision support. In *Proceedings of The Twenty-Third Text REtrieval Conference, TREC 2014, Gaithersburg, Maryland, USA, November 19-21, 2014*, 2014.

[37] R. I. Doğan, R. Leaman, and Z. Lu. Ncbi disease corpus: a resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 47:1–10, 2014.

[38] G. Drosatos, S. Roumeliotis, E. Kaldoudi, and A. Arampatzis. Duth at trec 2015 clinical decision support track. In *TREC*, 2015.

[39] E. Durmus, H. He, and M. Diab. Feqa: A question answering evaluation framework for faithfulness assessment in abstractive summarization. *arXiv preprint arXiv:2005.03754*, 2020.

[40] H. Edmundson. Problems in automatic abstracting. *Communications of the ACM*, 7(4):259–263, 1964.

[41] A. Ekbal and S. Saha. Stacked ensemble coupled with feature selection for biomedical entity extraction. *Knowledge-Based Systems*, 46:22–32, 2013.

[42] N. Elhadad, M.-Y. Kan, J. L. Klavans, and K. R. McKeown. Customization in a unified framework for summarizing medical literature. *Artificial intelligence in medicine*, 33(2):179–198, 2005.

[43] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479, 2004.

[44] M. Fiszman, T. C. Rindflesch, and H. Kilicoglu. Abstraction summarization for managing the biomedical research literature. In *Proceedings of the HLT-NAACL workshop on computational lexical semantics*, pages 76–83. Association for Computational Linguistics, 2004.

[45] C. Funk, W. Baumgartner, B. Garcia, C. Roeder, M. Bada, K. B. Cohen, L. E. Hunter, and K. Verspoor. Large-scale biomedical concept recognition: an evaluation of current automatic annotators and their parameters. *BMC bioinformatics*, 15(1):1–29, 2014.

[46] M. Gridach. Character-level neural network for biomedical named entity recognition. *Journal of biomedical informatics*, 70:85–91, 2017.

[47] Z. GuoDong and S. Jian. Exploring deep knowledge resources in biomedical name recognition. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 96–99. Association for Computational Linguistics, 2004.

[48] H. Gurulingappa, L. Toldo, C. Schepers, A. Bauer, and G. Megaro. Semi-supervised information retrieval system for clinical decision support. In *TREC*, 2016.

[49] S. Henry, Y. Wang, F. Shen, and O. Uzuner. The 2019 national natural language processing (nlp) clinical challenges (n2c2)/open health nlp (ohnlp) shared task on clinical concept normalization for clinical records. *Journal of the American Medical Informatics Association*, 27(10):1529–1537, 2020.

[50] W. Hersh. *Information retrieval: a health and biomedical perspective*. Springer Science & Business Media, 2008.

[51] W. Hersh, S. Price, and L. Donohoe. Assessing thesaurus-based query expansion using the umls metathesaurus. In *Proceedings of the AMIA Symposium*, page 344. American Medical Informatics Association, 2000.

[52] L. Hirschman, M. Colosimo, A. Morgan, and A. Yeh. Overview of biocreative task 1b: normalized gene lists. *BMC bioinformatics*, 6(1):S11, 2005.

[53] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[54] T. Joachims. Training linear svms in linear time. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 217–226, 2006.

[55] M. Ju, M. Miwa, and S. Ananiadou. A neural layered model for nested named entity recognition. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1446–1459, 2018.

[56] M.-Y. Kan, K. R. McKeown, and J. L. Klavans. Domain-specific informative and indicative summarization for information retrieval. In *In: Workshop on text summarization (DUC 2001*. Citeseer, 2001.

[57] S. Karimi, S. Falamaki, and V. Nguyen. Csiro at trec clinical decision support track. In *TREC*, 2016.

[58] J.-D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. Genia corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl_1):i180–i182, 2003.

[59] J.-D. Kim, T. Ohta, Y. Tsuruoka, Y. Tateisi, and N. Collier. Introduction to the bio-entity recognition task at jnlpba. In *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications*, pages 70–75. Citeseer, 2004.

[60] M. Krallinger, F. Leitner, O. Rabal, M. Vazquez, J. Oyarzabal, and A. Valencia. Overview of the chemical compound and drug name recognition (chemdner) task. In *BioCreative challenge evaluation workshop*, volume 2, page 2. Citeseer, 2013.

[61] M. Krallinger, F. Leitner, O. Rabal, M. Vazquez, J. Oyarzabal, and A. Valencia. Chemdner: The drugs and chemical names extraction challenge. *Journal of cheminformatics*, 7(1):S1, 2015.

[62] M. Krallinger, O. Rabal, F. Leitner, M. Vazquez, D. Salgado, Z. Lu, R. Leaman, Y. Lu, D. Ji, D. M. Lowe, et al. The chemdner corpus of chemicals and drugs and its annotation principles. *Journal of cheminformatics*, 7(1):S2, 2015.

[63] M. Krallinger, O. Rabal, A. Lourenço, M. P. Perez, G. P. Rodriguez, M. Vazquez, F. Leitner, J. Oyarzabal, and A. Valencia. Overview of the chemdner patents task. In *Proceedings of the fifth BioCreative challenge evaluation workshop*, pages 63–75, 2015.

[64] J. Lafferty, A. McCallum, and F. C. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289. Morgan Kaufmann Publishers Inc., 2001.

[65] A. M. Lam-Adesina and G. J. Jones. Applying summarization techniques for term selection in relevance feedback. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 1–9, 2001.

[66] W. W. Lau, C. A. Johnson, and K. G. Becker. Rule-based human gene normalization in biomedical text with confidence estimation. In *Computational Systems Bioinformatics: (Volume 6)*, pages 371–379. World Scientific, 2007.

[67] Z. Liao and Z. Zhang. A generic classifier-ensemble approach for biomedical named entity recognition. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 86–97. Springer, 2012.

[68] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[69] J. Lin and D. Demner-Fushman. The role of knowledge in conceptual retrieval: a study in the domain of clinical medicine. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 99–106. ACM, 2006.

[70] T.-Y. Liu et al. Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331, 2009.

[71] K. Lu and X. Mu. Query expansion using umls tools for health information retrieval. *Proceedings of the American Society for Information Science and Technology*, 46(1):1–16, 2009.

[72] Z. Lu, H.-Y. Kao, C.-H. Wei, M. Huang, J. Liu, C.-J. Kuo, C.-N. Hsu, R. T.-H. Tsai, H.-J. Dai, N. Okazaki, et al. The gene normalization task in biocreative iii. *BMC bioinformatics*, 12(8):S2, 2011.

[73] Z. Lu and W. J. Wilbur. Overview of biocreative iii gene normalization. In *Proceedings of the BioCreative III workshop*, pages 24–45, 2010.

[74] Y.-F. Luo, W. Sun, and A. Rumshisky. Mcn: a comprehensive corpus for medical concept normalization. *Journal of biomedical informatics*, 92:103132, 2019.

[75] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*. Cambridge University Press, 2008.

[76] D. Martinez, A. Otegi, A. Soroa, and E. Agirre. Improving search over electronic health records using umls-based query expansion through random walks. *Journal of biomedical informatics*, 51:100–106, 2014.

[77] A. K. McCallum. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

[78] P. McNamee. A domain independent approach to clinical decision support. In *TREC*, 2015.

[79] R. Mihalcea and P. Tarau. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411, 2004.

[80] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space, 2013.

[81] L. K. Milliken, S. K. Motomarry, and A. Kulkarni. Artpm: Article retrieval for precision medicine. *Journal of biomedical informatics*, page 103224, 2019.

[82] R. Mishra, J. Bian, M. Fiszman, C. R. Weir, S. Jonnalagadda, J. Mostafa, and G. Del Fiol. Text summarization in the biomedical domain: a systematic review of recent research. *Journal of biomedical informatics*, 52:457–467, 2014.

[83] S. Moen and T. S. S. Ananiadou. Distributional semantics resources for biomedical text processing. *Proceedings of LBM*, pages 39–44, 2013.

[84] D. Mollá. Macquarie university at bioasq 5b–query-based summarisation techniques for selecting the ideal answers. In *BioNLP 2017*, pages 67–75, 2017.

[85] M. Moradi and N. Ghadiri. Different approaches for identifying important concepts in probabilistic biomedical text summarization. *Artificial intelligence in medicine*, 84:101–116, 2018.

[86] L. P. Morales, A. D. Esteban, and P. Gervás. Concept-graph based biomedical automatic summarization using ontologies. In *Proceedings of the 3rd textgraphs workshop on graph-based algorithms for natural language processing*, pages 53–56. Association for Computational Linguistics, 2008.

[87] A. A. Morgan, Z. Lu, X. Wang, A. M. Cohen, J. Fluck, P. Ruch, A. Divoli, K. Fundel, R. Leaman, J. Hakenberg, et al. Overview of biocreative ii gene normalization. *Genome biology*, 9(2):S3, 2008.

[88] T. Mori, M. Nozawa, and Y. Asada. Multi-answer-focused multi-document summarization using a question-answering engine. *ACM Transactions on Asian Language Information Processing (TALIP)*, 4(3):305–320, 2005.

[89] D. L. Mowery, S. Velupillai, B. R. South, L. Christensen, D. Martinez, L. Kelly, L. Goeuriot, N. Elhadad, S. Pradhan, G. Savova, et al. Task 2: Share/clef ehealth evaluation lab 2014. In *Working Notes for CLEF 2014 Conference, Sheffield, UK, September 15-18, 2014*, volume 1180 of *CEUR Workshop Proceedings*, pages 31–42, 2014.

[90] H. Müller, A. G. S. de Herrera, J. Kalpathy-Cramer, D. Demner-Fushman, S. K. Antani, and I. Eggel. Overview of the imageclef 2012 medical image retrieval and classification tasks. In *CLEF (online working notes/labs/workshop)*, pages 1–16, 2012.

[91] A. Névéol, K. B. Cohen, C. Grouin, T. Hamon, T. Lavergne, L. Kelly, L. Goeuriot, G. Rey, A. Robert, X. Tannier, et al. Clinical information extraction at the clef ehealth evaluation lab 2016. In *CEUR workshop proceedings*, volume 1609, page 28. NIH Public Access, 2016.

[92] A. Névéol, C. Grouin, X. Tannier, T. Hamon, L. Kelly, L. Goeuriot, and P. Zweigenbaum. Clef ehealth evaluation lab 2015 task 1b: Clinical named entity recognition. In *CLEF (Working Notes)*, 2015.

[93] A. Névéol, A. Robert, R. Anderson, K. B. Cohen, C. Grouin, T. Lavergne, G. Rey, C. Rondet, and P. Zweigenbaum. Clef ehealth 2017 multilingual information extraction task overview: Icd10 coding of death certificates in english and french. In *CLEF (Working Notes)*, 2017.

[94] J.-P. Ng and V. Abrecht. Better summarization evaluation with word embeddings for rouge. *arXiv preprint arXiv:1508.06034*, 2015.

[95] H.-S. Oh and Y. Jung. Cluster-based query expansion using external collections in medical information retrieval. *Journal of biomedical informatics*, 58:70–79, 2015.

[96] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and D. Johnson. Terrier information retrieval platform. In *European Conference on Information Retrieval*, pages 517–519, 2005.

[97] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and C. Lioma. Terrier: A high performance and scalable information retrieval platform. In *Proceedings of the OSIR Workshop*, pages 18–25, 2006.

[98] J. Palotti and A. Hanbury. Tuw@ trec clinical decision support track 2015. In *TREC*, 2015.

[99] M. Pan, Y. Zhang, T. He, and X. Jiang. An enhanced hal-based pseudo relevance feedback model in clinical decision support retrieval. In *International Conference on Intelligent Computing*, pages 93–99. Springer, 2018.

[100] M. Pérez-Pérez, O. Rabal, G. Pérez-Rodríguez, M. Vazquez, F. Fdez-Riverola, J. Oyarzabal, A. Valencia, A. Lourenço, and M. Krallinger. Evaluation of chemical and gene/protein entity recognition systems at biocreative v. 5: the cemp and gpro patents tracks. In *Proceedings of the BioCreative V.5 Challenge Evaluation Workshop)*, 2017.

[101] S. Pradhan, N. Elhadad, B. R. South, D. Martinez, L. M. Christensen, A. Vogel, H. Suominen, W. W. Chapman, and G. K. Savova. Task 1: Share/clef ehealth evaluation lab 2013. In *CLEF (Working Notes)*, 2013.

[102] T. Qin, T.-Y. Liu, J. Xu, and H. Li. Letor: A benchmark collection for research on learning to rank for information retrieval. *Information Retrieval*, 13(4):346–374, 2010.

[103] A. P. Quimbaya, A. S. Múnera, R. A. G. Rivera, J. C. D. Rodríguez, O. M. M. Velandia, A. A. G. Peña, and C. Labbé. Named entity recognition over electronic health records through a combined dictionary-based approach. *Procedia Computer Science*, 100:55–61, 2016.

[104] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[105] H. U. Rahman, N. Chowk, T. Hahn, and R. Segall. Disease named entity recognition using conditional random fields. In *Proceedings of the 7th International Symposium on Semantic Mining in Biomedicine*, 2016.

[106] K. Raja, S. Subramani, and J. Natarajan. A hybrid named entity tagger for tagging human proteins/genes. *International journal of data mining and bioinformatics*, 10(3):315–328, 2014.

[107] L. A. Ramshaw and M. P. Marcus. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer, 1999.

[108] M. Rei, G. K. Crichton, and S. Pyysalo. Attending to characters in neural sequence labeling models. *arXiv preprint arXiv:1611.04361*, 2016.

[109] K. Roberts, D. Demner-Fushman, E. M. Voorhees, and W. R. Hersh. Overview of the TREC 2016 clinical decision support track. In E. M. Voorhees and A. Ellis, editors, *Proceedings of The Twenty-Fifth Text REtrieval Conference, TREC 2016, Gaithersburg, Maryland, USA, November 15-18, 2016*, volume Special Publication 500-321. National Institute of Standards and Technology (NIST), 2016.

[110] K. Roberts, M. Simpson, D. Demner-Fushman, E. Voorhees, and W. Hersh. State-of-the-art in biomedical literature retrieval for clinical cases: a survey of the trec 2014 cds track. *Information Retrieval Journal*, 19(1-2):113–148, 2016.

[111] S. Robertson, H. Zaragoza, et al. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389, 2009.

[112] S. E. Robertson and K. S. Jones. Relevance weighting of search terms. *Journal of the Association for Information Science and Technology*, 27(3):129–146, 1976.

[113] D. S. Sachan, P. Xie, M. Sachan, and E. P. Xing. Effective use of bidirectional language modeling for transfer learning in biomedical named entity recognition. In *Machine Learning for Healthcare Conference*, pages 383–402, 2018.

[114] S. Saha, A. Ekbal, and U. K. Sikdar. Named entity recognition and classification in biomedical text using classifier ensemble. *International journal of data mining and bioinformatics*, 11(4):365–391, 2015.

[115] G. Salton. *Automatic Information Organization and Retrieval.* McGraw Hill Text, 1968.

[116] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

[117] G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of the American society for information science*, 41(4):288–297, 1990.

[118] G. Salton, E. A. Fox, and H. Wu. Extended boolean information retrieval. *Communications of the ACM*, 26(11):1022–1036, 1983.

[119] G. Salton, A. Wong, and C.-S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[120] J. Sankhavara. Biomedical document retrieval for clinical decision support system. In *Proceedings of ACL 2018, Student Research Workshop*, pages 84–90, 2018.

[121] J. Sankhavara and P. Majumder. Team da iict at clinical decision support track in trec 2016: Topic modeling for query expansion. In *TREC*, 2016.

[122] J. Sankhavara and P. Majumder. Biomedical information retrieval. In *FIRE (Working Notes)*, pages 154–157, 2017.

[123] J. Sankhavara, F. Thakrar, P. Majumder, and S. Sarkar. Fusing manual and machine feedback in biomedical domain. In *Proceedings of The Twenty-Third Text REtrieval Conference, TREC 2014, Gaithersburg, Maryland, USA, November 19-21, 2014*, 2014.

[124] J. Sankhavara, F. Thakrar, S. Sarkar, and P. Majumder. Fusing manual and machine feedback in biomedical domain. Technical report, DHIRUBHAI AMBANI INST OF INFORMATION AND COMMUNICATION TECHNOLOGY ..., 2014.

[125] Y. Sari, M. F. Hassan, and N. Zamin. Rule-based pattern extractor and named entity recognition: A hybrid approach. In *2010 International Symposium on Information Technology*, volume 2, pages 563–568. IEEE, 2010.

[126] A. Sarker, D. Mollá, and C. Paris. An approach for query-focused text summarisation for evidence based medicine. In *Conference on Artificial Intelligence in Medicine in Europe*, pages 295–304. Springer, 2013.

[127] M. Sarrouti and S. O. El Alaoui. A passage retrieval method based on probabilistic information retrieval model and umls concepts in biomedical question answering. *Journal of biomedical informatics*, 68:96–103, 2017.

[128] F. Schulze and M. Neves. Entity-supported summarization of biomedical abstracts. In *Proceedings of the Fifth Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM2016)*, pages 40–49, 2016.

[129] M. S. Simpson and D. Demner-Fushman. Biomedical text mining: a survey of recent progress. In *Mining text data*, pages 465–517. Springer, 2012.

[130] L. Smith, L. K. Tanabe, R. J. nee Ando, C.-J. Kuo, I.-F. Chung, C.-N. Hsu, Y.-S. Lin, R. Klinger, C. M. Friedrich, K. Ganchev, et al. Overview of biocreative ii gene mention recognition. *Genome biology*, 9(2):S2, 2008.

[131] F. Song and W. B. Croft. A general language model for information retrieval. In *Proceedings of the eighth international conference on Information and knowledge management*, pages 316–321, 1999.

[132] M. Song, H. Yu, and W.-S. Han. Developing a hybrid dictionary-based bio-entity recognition technique. *BMC medical informatics and decision making*, 15(1):1–8, 2015.

[133] N. Stokes, Y. Li, L. Cavedon, and J. Zobel. Exploring criteria for successful query expansion in the genomic domain. *Information retrieval*, 12(1):17–50, 2009.

[134] L. Tanabe, N. Xie, L. H. Thom, W. Matten, and W. J. Wilbur. Genetag: a tagged corpus for gene/protein named entity recognition. *BMC bioinformatics*, 6(1):S3, 2005.

[135] B. Tang, H. Cao, X. Wang, Q. Chen, and H. Xu. Evaluating word representation features in biomedical named entity recognition tasks. *BioMed research international*, 2014, 2014.

[136] Z. Tang, L. Jiang, L. Yang, K. Li, and K. Li. Crfs based parallel biomedical named entity recognition algorithm employing mapreduce framework. *Cluster Computing*, 18(2):493–505, 2015.

[137] Terrier. University of glasgow. divergence from randomness (dfr) framework. http://terrier.org/docs/v2.2.1/dfr_description.html. Accessed: 2021-03-18.

[138] M. Tiftikci, A. Özgür, Y. He, and J. Hur. Machine learning-based identification and rule-based normalization of adverse drug reactions in drug labels. *BMC bioinformatics*, 20(21):1–9, 2019.

[139] M. Torjmen-Khemakhem and K. Gasmi. Document/query expansion based on selecting significant concepts for context based retrieval of medical images. *Journal of biomedical informatics*, 95:103210, 2019.

[140] J.-M. Torres-Moreno, P.-L. St-Onge, M. Gagnon, M. El-Beze, and P. Bellot. Automatic summarization system coupled with a question-answering system (qaas). *arXiv preprint arXiv:0905.2990*, 2009.

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

[142] Y. Wang and H. Fang. Exploring the query expansion methods for concept based representation. In *TREC*, 2014.

[143] Y. Wang, X. Liu, and H. Fang. A study of concept-based weighting regularization for medical records search. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 603–612, 2014.

[144] C.-H. Wei, Y. Peng, R. Leaman, A. P. Davis, C. J. Mattingly, J. Li, T. C. Wiegers, and Z. Lu. Overview of the biocreative v chemical disease relation (cdr) task. In *Proceedings of the fifth BioCreative challenge evaluation workshop*, volume 14, 2015.

[145] Y. Wei, C. Hsu, A. Thomas, and J. F. McCarthy. Atigeo at trec 2014 clinical decision support task. In *TREC*, 2014.

[146] J. Xu and W. B. Croft. Quary expansion using local and global document analysis. In *Acm sigir forum*, volume 51, pages 168–175. ACM, 2017.

[147] K. Xu, Z. Yang, P. Kang, Q. Wang, and W. Liu. Document-level attention-based bilstm-crf incorporating disease dictionary for disease named entity recognition. *Computers in biology and medicine*, 108:122–132, 2019.

[148] K. Xu, Z. Zhou, T. Hao, and W. Liu. A bidirectional lstm and conditional random fields approach to medical named entity recognition. In *International Conference on Advanced Intelligent Systems and Informatics*, pages 355–365. Springer, 2017.

[149] Z. Yang, H. Lin, and Y. Li. Exploiting the performance of dictionary-based bio-entity name recognition in biomedical literature. *Computational biology and chemistry*, 32(4):287–291, 2008.

[150] L. Yao, H. Liu, Y. Liu, X. Li, and M. W. Anwar. Biomedical named entity recognition based on deep neutral network. *Int. J. Hybrid Inf. Technol*, 8(8):279–288, 2015.

[151] A. Yeh, A. Morgan, M. Colosimo, and L. Hirschman. Biocreative task 1a: gene mention finding evaluation. *BMC bioinformatics*, 6(1):S2, 2005.

[152] E. Yilmaz, E. Kanoulas, and J. 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*, pages 603–610, 2008.

[153] D. Zeng, C. Sun, L. Lin, and B. Liu. Lstm-crf for drug-named entity recognition. *Entropy*, 19(6):283, 2017.

[154] D. Zhang and D. He. Enhancing clinical decision support systems with public knowledge bases. *Data and Information Management*, 1(1):49–60, 2017.

[155] S. Zhang, W. Fan, and B. He. Cbia vt at trec 2015 clinical decision support track-exploring relevance feedback and query expansion in biomedical information retrieval. In *TREC*, 2015.

[156] F. Zhu and B. Shen. Combined svm-crfs for biological named entity recognition with maximal bidirectional squeezing. *PloS one*, 7(6):e39230, 2012.

CHAPTER A

Publications

Journal

- J. Sankhavara, R. Dave, B. Dave, and P. Majumder. Query specific graph-based query reformulation using umls for clinical information access. Journal of Biomedical Informatics, 108:103493, 2020.
- J. Sankhavara. Feature weighting in finding feedback documents for query expansion in biomedical document retrieval. SN Computer Science, 1(2):1–7, 2020.

Conference and Workshop

- J. Sankhavara and P. Majumder. Query-focused biomedical text summarization in BioASQ 8B. CLEF (Working Notes) 2020.
- J. Sankhavara and P. Majumder. Advances in biomedical entity identification: A survey. In Biotechnology and Biological Sciences: Proceedings of the 3rd International Conference of Biotechnology and Biological Sciences (BIOSPECTRUM 2019), August 8-10, 2019, Kolkata, India, page 114. CRC Press, 2019.
- J. Sankhavara. Biomedical document retrieval for clinical decision support system. In Proceedings of ACL 2018, Student Research Workshop, pages 84–90, 2018.
- J. Sankhavara and P. Majumder. Biomedical information retrieval. In FIRE (Working Notes), pages 154–157, 2017.

- J. Sankhavara and P. Majumder. Team da_iict at clinical decision support track in trec 2016: Topic modeling for query expansion. In TREC, 2016.