Seed-driven document ranking for systematic reviews in evidence-based medicine

Lee, Eunkyung

2020

Lee, F. (2020). Seed-driven document ranking for systematic reviews in evidence-based medicine. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/146709>

<https://doi.org/10.32657/10356/146709>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 16:53:24 SGT

SEED-DRIVEN DOCUMENT RANKING FOR SYSTEMATIC REVIEWS IN EVIDENCE-BASED MEDICINE

LEE EUNKYUNG

School of Computer Science and Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Doctor of Philosophy (Ph.D)

2020

Seed-driven Document Ranking for Systematic Reviews in Evidence-based Medicine

LEE EUNKYUNG

School of Computer Science and Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Doctor of Philosophy (Ph.D)

2020

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

08 June 2020

Eunkyoung Lee

.....
Date

.....
LEE EUNKYUNG

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

23 June 2020

.....

Date

.....

SUN AIXIN

Authorship Attribution Statement

This thesis contains material from 2 paper(s) published in the following peer-reviewed journal(s) / from papers accepted at conferences in which I am listed as an author.

Chapter 3 is published as Grace E. Lee and Aixin Sun. Seed-driven Document Ranking for Systematic Reviews in Evidence-Based Medicine. SIGIR '18: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval June 2018 Pages 455–464 <https://doi.org/10.1145/3209978.3209994>

The contributions of the co-authors are as follows:

- I formulated and designed the problem, conducted the experiments, and wrote the manuscripts.
- A/Prof Sun discussed the idea with me and revised the manuscript drafts.

Chapter 6.1 is published as Grace E. Lee and Aixin Sun. A Study on Agreement in PICO Span Annotations. SIGIR'19: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval July 2019 Pages 1149–1152 <https://doi.org/10.1145/3331184.3331352>

The contributions of the co-authors are as follows:

- I formulated and designed the problem, conducted the experiments, and wrote the manuscripts.
- A/Prof Sun discussed the idea with me and edited the manuscript drafts.

Chapter 6.2 is published as Grace E. Lee and Aixin Sun. Understanding the Stability of Medical Concept Embeddings. Journal of the Association for Information Science and Technology (JASIST) 2020 <https://doi.org/10.1002/asi.24411>

The contributions of the co-authors are as follows:

- I formulated and designed the problem, conducted the experiments, and wrote the manuscripts.
- A/Prof Sun discussed the idea with me and edited the manuscript drafts.

08 June 2020

Eunkyoung Lee

.....
Date

.....
LEE EUNKYUNG

Acknowledgments

This thesis wouldn't have been possible without the support of many people. First and foremost, I would like to thank my advisor Aixin Sun for his invaluable guidance and support. The countless discussions with him have helped me not just gain substantial knowledge in my research areas, but also learn to think critically.

I also want to thank my amazing lab mates who have made my day-to-day life pleasant: Keke Huang, Phan C. Minh, Xin Zheng, Ting Lin, Jing Li, Jianglei Han, Zhe Chen, Peng Han, Yequan Wang, Yiding Liu, Xiucheng Li, Jin Yao Chin, Lucas Vinh Tran, Tuan-Anh Nguyen Pham, Junghoon Kim, Tenindra Abeywickrama, Xi Lin, and Renchi Yang. It is due to the friendly and aspiring research environment in Data Management Analytics Lab.

I would like to thank my friends from other schools in NTU: Andrew Ng, Seok Ting Lim, Pei Qi Lim, Sze Yuan Ho, Xiaomeng Xu, and Akshara Moudgalya. Our chats over lunch and coffee have brightened my days without fail.

I have to thank my lifelong friends outside of NTU: Yiseul Kim and Zai Hong Kua. I am very grateful to them for giving me the endless encouragement. They were always there for me even in challenging times.

I wouldn't be where I am today without the unconditional love and support from my parents and my older sister. They have taught me to persist and never let setbacks prevent me from doing what I believe in. Those values are what led me through my PhD and let me have fun in the journey.

Contents

Acknowledgments	v
Abstract	ix
List of Figures	xi
List of Tables	xiii
1 Introduction	2
1.1 Background	2
1.2 Overview	6
1.3 Research Contributions	8
1.4 Thesis Outline	10
2 Literature Review	11
2.1 Provision of Efficient Screening Process	11
2.1.1 Reducing Documents to Be Screened	11
2.1.2 Reducing the Number of SR Experts Needed for Screening	16
2.1.3 Increasing the Rate of Screening Documents	16
2.1.4 Screening Prioritization	17
2.2 Query by Document	18
2.3 Summary	20
3 Seed-driven Document Ranking	21
3.1 Introduction	21
3.2 Proposed Approach	22
3.2.1 Observations on Relevant Documents	23
3.2.2 Weighting Scheme and Document Ranking	25

3.3	Experiment Setup	27
3.3.1	Dataset	27
3.3.2	Baseline Methods	28
3.3.3	Evaluation Metrics	29
3.4	Results and Discussion	30
3.4.1	Overall Performance	32
3.4.2	Term Weighting Scheme and Individual SRs	34
3.4.3	Simulation of Manual Screening Process	37
3.5	Summary	39
4	Ranking by Document Matching	40
4.1	Introduction	40
4.2	Proposed Approach	42
4.2.1	Overview	42
4.2.2	Analysis: Distributions of PICO Elements	42
4.2.3	Mirror Matching	45
4.2.4	Model Discussion	47
4.3	Evaluation Settings	47
4.3.1	Dataset	47
4.3.2	Baselines	49
4.3.3	Evaluation Metrics	50
4.4	Results and Discussion	50
4.4.1	Overall Results	51
4.4.2	Analysis on M-Match	55
4.4.3	Neural Models for Screening Prioritization	58
4.5	Summary	60
5	Ranking Performance Estimation	62
5.1	Introduction	62
5.2	Related Work	63
5.3	Proposed Approach	66
5.3.1	Analysis on Ranking Quality	66

5.3.2	Proposed Measure: Topic Broadness	69
5.4	Experiments	71
5.4.1	Dataset	71
5.4.2	Baselines	71
5.4.3	Ground Truth and Evaluation Metrics.	73
5.5	Results and Discussion	73
5.5.1	Overall Results	73
5.5.2	Ablation Analysis	77
5.5.3	Impact of Seed Documents	78
5.6	Summary	80
6	Methods to Enhance Ranking	82
6.1	Agreements on PICO Span Annotations	84
6.1.1	PICO Span Annotation Dataset	85
6.1.2	PICO Span Agreement	86
6.1.3	PICO Span Recognition	91
6.1.4	Summary	92
6.2	Stability of Medical Concept Embeddings	92
6.2.1	Related Work	94
6.2.2	Analysis	97
6.2.3	Proposed Measure	102
6.2.4	Discussion	104
6.2.5	Summary	105
6.2.6	Appendix	105
7	Conclusion and Future Work	108
7.1	Conclusion	108
7.2	Future Work	110
	List of Publications	113
	References	114

Abstract

Evidence-based medicine (EBM) uses the best-available evidence in the process of patient care. Systematic reviews (SRs) are an essential part in the practice of EBM, providing the most up-to-date answers from primary evidence (published clinical studies) for a specific clinical question. SRs locate, appraise, and synthesize pertinent literature and derive answers.

Finding all relevant documents is the most critical task in systematic reviews. Missing relevant studies discredits answers in SRs. In order to find every relevant literature, SRs are strictly conducted by following four sequential steps: (i) defining a clinical question and relevance conditions, (ii) collecting potential relevant documents from digital libraries (*e.g.*, PubMed, Embase) by Boolean search, (iii) screening the retrieved documents (candidate documents) to identify relevant documents, and (iv) analyzing the relevant documents and deriving a conclusion (answer) based on them. Each step is recorded and published together in SRs.

Screening (*i.e.*, identifying relevant documents among candidate documents) is the most time-consuming process when conducting SRs. It is a manual process involving several thousands of documents, and thus typically takes a few months to complete. Many approaches have been proposed to improve the screening process using text mining and machine learning techniques. Among them, screening prioritization is the most promising approach to be implemented in practice while assisting researchers to identify all relevant documents. Given candidate documents, it provides the document rankings aiming that relevant documents are ranked at the top. Screening prioritized documents makes relevant documents be found early, and it leads to the efficient workflow in conducting SRs.

It is natural that SR experts know the existence of one or two relevant documents after defining the clinical question (*i.e.*, before the screening process). In this thesis, we propose a new approach named **seed-driven document ranking** for screening prioritization. We assume that one relevant document is given, which we call ‘seed document’ and use it as a query. The

task of seed-driven document ranking is to rank candidate documents using a seed document query. Existing approaches in screening prioritization focus on building short keyword queries using statistical information in candidate documents.

In this thesis, we extensively investigate ranking models for seed-driven document ranking. We first develop a retrieval model adapted for a seed document query. We propose ‘bag-of-clinical terms’ document representation reducing a long document query into essential clinical terms to find relevant documents. More importantly, we propose a weight function for clinical terms in a seed document. The proposed term weight function is combined with query likelihood retrieval model.

Next, we propose a new semantic ranking model. We investigate a document matching (*i.e.*, similarity) approach that ranks candidate documents according to their matching scores to a seed document. We design position-based semantic term matching, and develop a simple two-way document matching model. The semantic document matching approach further improves upon the performance of the model proposed in the previous chapter.

Thirdly, we predict the ranking performance on SRs in seed-driven document ranking. Despite the state-of-the-art performance on benchmark datasets, the performance of ranking models on individual SRs widely varies. The large performance difference is less desirable, because SR experts experience the local performance (performance on a single SR) in the manual screening, instead of the global performance (performance on multiple SRs in benchmark datasets). We hypothesize that individual SRs have a different ranking difficulty depending on its topic broadness. To this end, we propose a measure predicting the ranking difficulties of SRs.

Furthermore, we explore methods to improve the local performance especially on difficult SRs and the global performance of ranking models. We first study PICO elements in medical literature. Automatic PICO recognition helps identify essential information for the relevance of documents. We examine the boundaries of PICO span annotations and discuss how to correctly and effectively evaluate machine learning models in this task. Next we investigate medical concepts (*i.e.*, medical/clinical terms) embeddings. Our work in previous chapters has shown that medical concepts are important information for effectively representing documents. We analyze the stability of medical concept embeddings.

We conclude this thesis with the discussion on how other domain-specific applications with the similar aim on high recall can benefit from seed-driven document ranking, such as legal document search and test collection generation.

List of Figures

1.1	Pipeline of conducting a systematic review	3
2.1	Iterative process of active learning	14
3.1	Intra-similarity among relevant documents and irrelevant documents in SRs. . .	23
3.2	Distribution of clinical terms (in BOC) and words (in BOW) over the five ranges of normalized document frequency in relevant documents (best viewed in color)	24
3.3	The estimated term weight vs. document frequency of clinical terms in 50 SRs. In line graph a filled triangle is a mean value and an error range denotes variance.	34
3.4	Distribution of AvgPr of seed documents in 50 SRs. Each boxplot is a distri- bution of AvgPr values in one SR, sorted by their mean values in ascending order. Purple and blue boxplots indicate the SRs are from the train and test set in CLEF17, respectively. (best viewed in color)	35
4.1	Mirror Matching uses two-way matchings mechanism: from a query document to a candidate document and from a candidate document to a query document . Each one-way matching is based on position-based term matching.	43
4.2	Distributions of Patient, Intervention, and Outcome elements in medical docu- ments (title and abstract). Each row indicates a document. Within a row dark blocks are terms labeled with a given element.	44
4.3	WSS100 of individual SRs by M-Match against the estimated coverage of SRs .	54
4.4	Impact of different window size (λ) in position-based term matching.	56
4.5	Impact of different word embeddings on cosine similarity with AvgEm-W, WMD, and M-Match.	58
4.6	Performance of neural models on test dataset in CLEF19. Two sets of AP are presented when the train and dev datasets are comprised of different SRs. . . .	60

5.1	The lowest and the highest 5 ranking qualities of SRs in CLEF18 dataset given a ranking model. Three ranking models are presented. The pattern on the bars indicates the same SRs across the ranking models.	69
6.1	Example of span annotations in EBM-PICO dataset. This sentence (from PMID: 23549581) is annotated by 12 annotators (4 experts and 8 non-experts) and 3 different span annotations are made by them. The 3 spans have the same label P , but different boundaries.	84
6.2	Examples of correct PICO spans and predicted PICO spans in a PICO span recognition task. The example sentence is from a medical document PMID: 10940525 in EBM-PICO dataset. N indicates a token without PICO label. . . .	92
6.3	Distributions of words (left) and medical concepts (right) over frequencies in the OHSUMED dataset. The majority of concepts have low frequency.	99
6.4	Distributions of medical concepts over frequency and the stability of embeddings trained by cui2vec and NLM in heatmaps. Total number of concepts is 25,491. Each cell indicates the number of medical concepts in the corresponding bin intersected by frequency and stability.	100
6.5	Distributions of low-frequency concepts (in the first row) and commonly used concepts among low-frequency concepts (in the second row) over frequency and the stability of embeddings.	101

List of Tables

1.1	Examples of relevance conditions. In general, relevance conditions are defined under PICO categories as well as study design.	4
1.2	Percentage of relevant documents in candidate documents for 5 randomly selected SRs in CLEF 17 eHealth Task 2 dataset; and SRs with min, max, and median ratios in the dataset.	5
2.1	Summary of classifiers studied in related work to reduce the number of documents to be screened	12
2.2	Summary of features for document representations used in supervised approaches	13
2.3	Classifiers and feature representations evaluated in active learning setup	14
3.1	Performance of SDR and baseline ranking models on 30 SRs (CLEF17 test set) and 50 SRs (the entire CLEF17 dataset). The best performance among <i>all</i> models is in boldface and italics. Among SDR and baselines (<i>i.e.</i> , not including CLEF-Query and SDR+AES), the best performance is in boldface and the second best is underlined. * denotes the statistically significant difference between SDR-BOC and a baseline. † and ‡ are significance test results of SDR+AES over AES and SDR-BOC, respectively. Statistical significance is by one-tailed paired t-test with $p < 0.05$	31
3.2	Simplified relevance conditions of example SRs and their AvgPr values. The main clinical terms are manually extracted from the detailed relevance conditions (see references for the original relevance conditions).	37

3.3	Evaluation of updated ranking in simulation of manual screening process. In each iteration top 10 ranked documents are labeled and the relevant documents are appended to the seed to update the ranking. The best performance is in boldface and the second best is underlined. * denotes a statistically significant difference compared to the best, paired t-test with $p < 0.05$	38
4.1	Comparison between Mirror Matching and the baselines. The best model is in bold and the second best is underlined in each metric. * denotes statistically significant performance of the best model compared to a given model (p-value < 0.01).	52
4.2	Simulating the manual screening process. In each round, top 20 documents are labeled and the rankings of the remaining documents are updated with relevant documents.	54
4.3	AP of degraded models without key components of Mirror Matching.	55
5.1	Comparison between the estimation of SR ranking quality task and the query performance prediction task in terms of target collections. In this comparison, we use TREC Robust track in 2004 (TREC 2004) as a representative of the query performance prediction task.	64
5.2	The numbers of candidate documents and relevant documents of 72 SRs in CLEF18 dataset. The number of relevant documents are full text screening labels. The numbers of median and average are rounded to the nearest integer.	68
5.3	Spearman's rank correlation coefficients of the proposed measure and the baselines. The correlation coefficients are computed with the two lists of SR ranked by the estimated values and the actual performance (the average of AP) by a given ranking model. The best and the second best performances are in boldface and underlined, respectively. * denotes that performance of the proposed measure is statistically significant compared to the given baseline.	74

5.4	Pearson linear correlation coefficients of the proposed measure and the base- lines. The correlation coefficients are computed with the two lists of the esti- mated values and the actual performance (the average of AP) by a given ranking model. The best and the second best performances are in boldface and under- lined, respectively. * denotes that performance of the proposed measure is sta- tistically significant compared to the given baseline.	75
5.5	Impact of the two types of document similarity in the proposed measure.	78
5.6	Spearman’s rank correlation evaluation using three seed documents. The best and the second best performances are in boldface and underlined, respectively. .	79
5.7	Pearson linear correlation evaluation using three seed documents. The best and the second best performances are in boldface and underlined, respectively. . . .	80
6.1	Five examples of predicted annotations (Predicted) and their evaluations by dif- ferent agreements (<i>i.e.</i> , Exact, OB, TO). The example sentence consists of 7 to- kens, and tokens in blue denote annotated text span (the same label is assumed in all annotations). o/x indicates a correct/wrong prediction against gold anno- tation (Gold).	87
6.2	Average (standard deviation) <i>F1</i> scores estimated within MTurk workers on 5000 documents (MTurk-5000). For the 200 documents containing annotations by both MTurk workers and medical experts, the within group agreement of MTurk workers (MTurk-200) and medical experts (Expert-200) are reported. The <i>F1</i> scores are computed based on different agreements.	88
6.3	Precision, Recall, and <i>F1</i> values of cross-group annotation agreement between MTurk workers (predicted annotation) and medical experts (gold standard). The average values (standard deviation in parenthesis) of 200 documents are reported.	88
6.4	Comparison of the length of span annotations. The average (standard deviation in parenthesis) number of tokens are presented within MTurk workers on 5000 documents (MTurk-5000), and on 200 documents (MTurk-200) and medical experts on 200 documents (Expert-200).	90
6.5	Performance of BiLSTM-CRF in PICO span recognition evaluated by the ex- act and relaxed agreements. Precision/Recall/ <i>F1</i> values are estimated for each element label (P, I, O) and Micro-averaged value (Micro) for all labels.	91

6.6	Examples of input text for training concept embeddings. Medical concepts are presented in square brackets. A word embedding algorithm such as word2vec is applied.	96
6.7	Pearson correlation coefficient of the noisiness of context words with the stability of medical concepts. We present three groups of medical concepts: all concepts, concepts with low-frequency (< 100) (Low-freq), and commonly used concepts in biomedical applications among low-frequency concepts (Low-freq & Common). All correlation coefficients are statistically significant at $p < 0.001$.	103
6.8	Average (standard deviation) of the stability of medical concept embeddings when different hyperparameter values are set in the training of word embeddings	106
6.9	Pearson correlation coefficients of the noisiness of context words with the stability of medical concept embeddings. Embeddings are trained by different hyperparameter values. The strongest correlation in cui2vec and NLM is indicated as *. All correlation values are statistically significant at $p < 0.001$	107

Chapter 1

Introduction

1.1 Background

Evidence-based medicine (EBM) uses the best-available evidence in the process of patient care. Systematic reviews (SRs) are an essential part in the practice of EBM, providing the most up-to-date answers from primary evidence (published clinical studies) for a specific clinical question. SRs locate, appraise, and synthesize pertinent literature and derive the answers. An example clinical question¹ is: *Can laparoscopy (a diagnostic test) assess the resectability of pancreatic cancer?* For the better understanding of the task, we describe the process and challenges of conducting SRs in EBM.

Overview of Systematic Review. A SR is a literature survey which presents the up-to-date answer (conclusion) to a clinical question (SR topic) based on relevant studies. The validity of conclusion depends on identifying all studies relevant to its topic. To ensure that all relevant literature are included in the process of deriving a conclusion, SRs are strictly conducted by following systematic procedure. Figure 1.1 shows the overview of sequential steps in conducting a SR.

The first step is to formulate a clinical question and relevance conditions. SR experts carefully look through published literature and other SRs and determine a clinical question that needs to be answered. At the end of this step, the specific clinical question is defined and SR experts become knowing one or two relevant documents for the defined clinical question. Relevance conditions, namely eligibility criteria in SR community, are also defined in the first step.

¹The clinical question is simplified from its original version for readers to grasp the concept of systematic reviews. The original clinical question is stated in [1].

Fig. 1.1: Pipeline of conducting a systematic review

Relevance conditions are initially defined by using multiple key clinical terms often using the PICO framework². The conditions are then further elaborated from the terms and finally contain details to assess the relevance of documents. Table 1.1 shows examples of relevance conditions under PICO categories.

Once a clinical question and relevance conditions are set, the next step is to search potential relevant documents from databases (*e.g.*, PubMed, Embase) using Boolean keyword search. With the goal of finding all existing relevant documents, various combinations of keyword Boolean queries are formulated for retrieval. As a result, the number of collected documents are usually large, with low precision and high recall. We call the retrieved potential documents *candidate documents*. The list of Boolean queries are often published together with SR to ensure reproducibility. There are studies aiming to increase precision while maintaining recall in order to improve the retrieval step [2].

The process of identifying relevant documents in candidate documents is called *screening* in SR. In this process, the relevance of a document is judged by the pre-defined relevance conditions as shown in Table 1.1. When a candidate document satisfies *all relevance conditions*, it becomes a relevant document. Documents with partial or non satisfaction are labeled as irrelevant documents. The screening process is performed in two steps: abstract screening and full text screening. Abstract screening is to judge the relevance based on title and abstract text of a candidate document, and only (intermediate) relevant documents are moved to the next full text screening. Full text screening is to confirm their relevance based on the entire text of the documents. Table 1.2 lists the percentage of final relevant and candidate documents for five

²P: Patient or Problem, I: Intervention, Prognostic Factor, or Exposure, C: Comparator, O: Outcome.

Table 1.1: Examples of relevance conditions. In general, relevance conditions are defined under PICO categories as well as study design.

Patient (P)	<ul style="list-style-type: none"> • pancreatic cancer • age < 65 • no surgical history
Intervention (I)	<ul style="list-style-type: none"> • laparoscopy
Comparator (C)	<ul style="list-style-type: none"> • surgical or physical examinations
Outcome(O)	<ul style="list-style-type: none"> • the resectability of cancer cells • the stage of cancer cells • patient satisfaction
Study design	<ul style="list-style-type: none"> • randomized controlled trials

sample SRs taken from the CLEF 2017 eHealth Task 2 dataset.³ The minimum, maximum, and median percentage of 50 SRs in the dataset are also reported. Typically, fewer than 1% of candidate documents are relevant.

The last step in SR is to assess the qualities of relevant documents. Since this step is conducted on relevant documents alone, it is relatively less resource-intensive. Target information such as patient information, experiment methodologies, and results are extracted from relevant documents. Finally, SR experts appraise and synthesize the information and derive an overall conclusion of the clinical question. Improving the process of extracting target data is also an active research area [4–6].

Challenges in Screening. The vast majority of time and labor in conducting SRs is devoted to the screening step. The screening process alone takes several months to perform, because it involves the two manual screening processes (*i.e.*, abstract and full text screenings) and thousands of candidate documents to screen. Furthermore, compared to the large number of candidate documents, the number of identified relevant documents is much smaller. As reported in Table 1.2, fewer than 1% of candidate documents are relevant documents in most SRs in the CLEF17 dataset. Additionally, other studies reported that the number of relevant documents accounts for only 1.2% of the candidate documents on average [3, 7].

³According to the task overview [3], the candidate documents in CLEF 2017 eHealth Task 2 were re-retrieved by the organizers from MEDLINE database, based on the published keyword queries. The number of relevant documents is slightly smaller than that were actually used in the SRs since relevant documents (and more candidate documents) may come from other databases. In this thesis, we ignore one SR in the CLEF17 dataset which has zero relevant document.

Table 1.2: Percentage of relevant documents in candidate documents for 5 randomly selected SRs in CLEF 17 eHealth Task 2 dataset; and SRs with min, max, and median ratios in the dataset.

Sample SRs	#Rel Docs	#Candidate Docs	% of Rel Docs
SR_1	16	1,911	0.83%
SR_2	48	10,872	0.44%
SR_3	3	1,573	0.19%
SR_4	6	2,065	0.29%
SR_5	46	5,971	0.77%
Min percentage	1	12,705	0.008%
Max percentage	18	114	15.79%
Median percentage	15	2,074	0.72%

Many approaches using text mining and machine learning techniques have been proposed to make the screening step efficient. While they have shown the improvement, manual screening still remains necessary to find all relevant documents in candidate documents, concluded in a survey on improving screening process using text mining techniques [8]. Besides, the survey suggests that *screening prioritization* is the most promising approach to be adopted in practice and assist SR researchers during manual screening procedure.

Screening prioritization. Screening prioritization is a document ranking approach which provides the rankings of candidate documents where relevant documents are ranked higher than irrelevant documents. Screening ranked candidate documents leads to efficient workflow of not only the screening process, but also the overall process of conducting a SR. Abstract screening takes several months, and SR experts identify intermediate relevant documents in the early period of the abstract screening process. Thus, the following full text screening for the intermediate relevant documents is performed simultaneously, while documents with the lower rankings are still in the abstract screening. Furthermore, it allows final relevant documents to move to the last extraction step faster. With the same goal, a new task named “Technologically Assisted Reviews in Empirical Medicine” has been held for several years in CLEF eHealth competition.⁴

Existing studies on screening prioritization are classified into two approaches: supervised approach and unsupervised approach. Most early studies have investigated supervised approach. Classic machine learning models such as SVM, logistic regression, and learning-to-rank are

⁴CLEF eHealth 2017-2019 Task 2 <https://sites.google.com/site/clefehealth2017/task-2>

evaluated in multiple studies [18, 22, 33]. Later, modern deep learning models have been also attempted [20, 21]. However, the supervised approach has shown limited performance because of the document unbalance in SRs as shown in Table 1.2. More recently, in CLEF eHealth 2017-2019 Task 2, an unsupervised approach is more actively explored than supervised approach. The majority of participants have focused on formulating keyword queries to rank candidate documents and used existing unsupervised retrieval models such as BM25. To build effective queries, various techniques from simple frequency to keyword mining algorithm in candidate documents are tested. We present more in-depth discussion on existing studies in screening prioritization in Chapter 2.

1.2 Overview

In the first step of SRs, SR experts look through existing literature to determine a clinical question that is necessary to be answered from published studies. It is natural that SR experts come to know the existence of one or two relevant documents after defining the clinical question, *i.e.*, before the screening process. In this thesis, we propose a new approach named **seed-driven document ranking** for screening prioritization. We assume that one relevant document is known, which we call *seed document*, and use it as a query. As a seed document is a relevant document, it contains a full set of relevance conditions for a given clinical question. Now, our research question becomes: “*Given a seed document and candidate documents, how to rank the candidate documents such that (hidden) relevant documents appear at the top of the ranking?*”

This thesis extensively investigate ranking models for seed-driven document ranking. We develop two ranking models. One ranking model is a probabilistic retrieval model using query likelihood retrieval model and the other model is a document matching approach, which surpasses the former ranking model. Next, we study the ranking performance estimation on individual SRs. Lastly, we explore several methods that have potential to further enhance the ranking models.

First, we develop a retrieval model for a seed document query in screening prioritization. Classical retrieval models (*e.g.*, BM25 and Query likelihood model) are developed for short keyword queries in general. A long document query often becomes a challenge in retrieval models in terms of query understanding and query representation [9]. Relevance conditions

contain various medical concepts (*i.e.*, medical terminologies) as shown in Table 1.1 and they determine the relevance of documents in SRs. We show that ‘bag-of-concepts’ (BOC) is more effective than ‘bag-of-words’ (BOW) to represent documents in screening prioritization. More importantly, we propose a weight function for medical concepts in a seed document (**SDR-BOC**). The proposed weight function is combined with query likelihood retrieval model.

Secondly, we investigate a document matching (*i.e.*, similarity) model for seed-driven document ranking in which candidate documents are ranked by their matching scores to a seed. When a seed document and a candidate document contain the same term, it must be used as the same PICO element in order for the candidate to become a relevant document. For example, when a term *insulin* is used as Intervention (I) in a seed and as Outcome (O) in a candidate document, the candidate becomes an irrelevant document. Simple occurrence-based term matching is unable to match terms based on PICO elements. We first study the positions of PICO elements in abstract texts of medical literature. Our analysis reveals that they appear with positional patterns. With the aim of PICO-based term matching, we design position-based semantic term matching and propose **M-Match** as a simple and effective document matching model. The performance of M-Match outperforms the performance of the first proposed model SDR-BOC.

Despite the state-of-the-art performance on SR benchmark datasets, the performance of ranking models on individual SRs widely varies. The drastic performance difference over SRs is problematic, because SR experts experience the local performance (the performance on a single SR), instead of the global performance (the performance on multiple SRs in benchmark datasets) of a ranking model. We investigate the performance distributions of three ranking models over SRs. The analysis shows that some SRs are more difficult than other SRs for a model to achieve quality ranking results. We hypothesize the topic broadness of SR affects the ranking difficulty of SR, and propose a measure to estimate the topic broadness. The proposed measure shows clear and consistent linear relation with SR ranking difficulty and it enables the prediction of SR ranking performance.

Lastly, we explore techniques that have potential to enhance the local performance on difficult SRs and/or further increase the global performance of ranking models. We first look into PICO elements in medical literature. Automatic identification of PICO elements allows to refine the content of seed document into explicit relevance conditions. We examine the boundaries of PICO span annotations and discuss the evaluation of machine learning models for PICO recognition task. Next, we study vector representations for medical concepts. Previously, we found

that representing documents in BOC is effective in seed-driven document ranking. Motivated by this finding, we analyze the stability of medical concept embeddings and suggest ways to obtain stable representations for medical concepts.

1.3 Research Contributions

- We propose a new approach seed-driven document ranking (SDR) for screening prioritization, to assist conducting systematic reviews, with an assumption that SR experts know one relevant document. We conduct a detailed analysis of relevant documents in SRs and propose to use ‘bag of clinical terms (BOC)’ to represent documents. Above all, we estimate the importance of the clinical terms based on their distribution in candidate documents. Through extensive experiments, we demonstrate the effectiveness of SDR against baseline methods. We provide a comprehensive analysis of the ranking results and simulate the manual screening process where more relevant documents are identified through the screening process. This result is published in the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR).
- We investigate seed-driven document ranking approach in the context of document matching, and propose a two-way document matching named Mirror Matching (M-Match). We reveal that PICO elements appear with positional patterns in abstract of medical literature. M-Match uses document matching scores computed by position-based term matching with the goal of PICO-based term matching. Moreover, it adopts two-way matching scheme which aggregates two position-based matchings from seed to candidate and from candidate to seed in order to promote the exact matching between documents. We provide extensive evaluation on M-Match against retrieval models, document similarity measures, and neural IR models. We demonstrate that its superior performance in precision based metrics (*i.e.*, finding many relevant documents at the top) provides useful resources to update the ranking in manual screening.
- We investigate the ranking qualities of individual SRs. We make following observations: (i) for a SR, the changes in ranking qualities by ranking models are very small and (ii) some SRs consistently have low ranking qualities than other SRs, despite different ranking

models. Motivated by the observations, we propose the topic broadness of SRs as a factor affecting the ranking quality of SRs. We develop a measure that quantifies SR topic broadness. The extensive evaluation shows that the topic broadness is able to effectively estimate the ranking qualities of SRs and thus enables the prediction. To the best of our knowledge, we are the first to investigate the ranking quality of individual SRs. This result is under review in the Journal of the Association for Information Science and Technology (JASIST).

- We study the agreement of PICO span annotations made by human annotators, including both experts and non-experts. Agreements are estimated by a standard (exact) span agreement (i.e., matching both labels and boundaries of text spans), and two types of relaxed span agreement (i.e., matching labels without guaranteeing matching boundaries of spans). Based on the analysis, we report two observations: (i) the boundaries of PICO span annotations by individual annotators are very diverse. (ii) despite the disagreement in span boundaries, the general areas of the span annotations are broadly agreed by annotators. We claim that applying a standard agreement alone may undermine the agreement of PICO spans, and adopting both a standard and a relaxed agreements is more suitable for evaluating PICO-related tasks such as PICO recognition task. This result is reported in the 42nd International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR).
- We conduct a detailed analysis on the stability of concept embeddings in medical domain in relation to frequency in a corpus. The analysis reports the surprising high stability of low-frequency concepts: low-frequency concepts have the same high stability as high-frequency concepts, despite their fewer training instances. To develop deeper understanding of this finding, we propose a new factor, the noisiness of context words, which influences the stability of medical concept embeddings regardless of frequency. The evaluation shows the clear and consistent correlations between the proposed factor and the stability across various groups of medical concepts. Finally, we extend the evaluation into general domain and report the comparable results on the stability of word embeddings. This result is published in the Journal of the Association for Information Science and Technology (JASIST).

1.4 Thesis Outline

This thesis contains the introduction (this chapter), literature review, four main chapters, and conclusion. In Chapter 2, we provide readers with a review of related work such as literature on improving the screening and existing approaches for screening prioritization. In Chapter 3, we first propose seed-driven document ranking approach and develop a term weight function combined with a probabilistic retrieval model. Chapter 4 investigates a document matching approach for seed-driven document ranking. We propose a simple two-way document matching model. Chapter 5 studies the ranking performance (difficulty) estimation for SRs. We propose a measure to predict the ranking difficulties of individual SRs. Chapter 6 explores methods that have potential to further enhance ranking models. We discuss two studies on PICO span agreements and the stability of medical concept embeddings. Finally, Chapter 7 concludes the thesis and presents research directions for future work.

Chapter 2

Literature Review

Applying text mining techniques to improve systematic reviews has been getting attention for the last decade. There are many opportunities for text mining techniques to increase the efficiency of conducting systematic reviews such as in a retrieval step, a screening step, and an extraction step. In particular, many studies tackle a screening process, because it is the most time-consuming and labor-intensive process in systematic reviews. Our approach attempts to improve the screening step in systematic reviews and uses a seed document as a query. In this chapter, we first introduce existing work of improving screening process. Next, we present related work on query by documents.

2.1 Provision of Efficient Screening Process

To assist the manual screening process for the greater efficiency, four approaches were mainly explored: (i) reducing the number of documents to be manually screened, (ii) reducing the number of SR experts needed for screening, (iii) increasing the rate of manual screening process, and (iv) prioritizing documents to be screened [8].

2.1.1 Reducing Documents to Be Screened

As we mentioned in Chapter 1, the volume of candidate documents is large. It easily exceeds a thousand documents. The most obvious resolution for this challenge is to reduce the number of documents to be retrieved because the number of candidate documents are determined by retrieval process. Emphasis on the precision of retrieval process is possible to resolve the problem. However, it is highly likely to miss out relevant documents (lowering recall). Cochrane

Table 2.1: Summary of classifiers studied in related work to reduce the number of documents to be screened

Classifier	Literature
Support Vector Machine (SVM)	[12–14, 17, 19, 22–29]
Naive Bayes (NB)	[16, 17, 23, 25, 28–30]
K-nearest neighbor (K-NN)	[17, 29, 31]
K-means	[31, 32]
Random forest	[33]
Logistic regression	[18]
Decision tree	[23]
Ensemble	[25]
Convolutional neural network (CNN)	[20, 21]

Guidelines of conducting systematic reviews stress the need for favoring recall over precision in search process in order to avoid missing relevant documents [10, 11]. Thus, resource-efficient approaches are needed without compromising recall.

The other solution is to reduce the number of documents to screen. This approach excludes documents from the candidate document collection before starting the screening process. It is the most actively studied approach among the four approaches to increase the efficiency of screening process. It is achieved by a supervised approach and develops a classifier. To classify candidate documents of a target SR, labeled documents from other SRs are used as training data. The trained classifier explicitly makes decisions of inclusion/exclusion of candidate documents [12–16]. Many machine learning algorithms are explored to train a classifier. Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest Neighbor (K-NN) are evaluated in many studies [14, 17–19]. Recently deep learning algorithms such as convolutional neural network (CNN) were attempted in [20, 21]. They, however, show the lower performance than traditional machine learning algorithms due to the limited number of training data. The classifiers evaluated in existing studies are listed in Table 2.1. When training a classifier, under-sample/oversample strategies are used to mitigate the imbalance between relevant and irrelevant documents.

Table 2.2: Summary of features for document representations used in supervised approaches

Features	Literature
Bag-of-words (BOW)	[12–14, 16, 17, 19, 23–25, 27, 28, 33, 35]
Controlled vocabularies (UMLS, MeSH)	[12, 14, 16, 19, 24–26, 28, 30, 33, 35]
TF-IDF	[17, 23, 26, 29]
Word embedding, Paragraph embedding	[20–22]
Topic Modeling (LDA)	[27, 30]
Citation information (<i>e.g.</i> , co-citation, citation titles)	[32, 33]
Knowledge bases	[36]
Metadata (<i>e.g.</i> , publication types, journal names)	[14, 19, 28, 33]

The evaluation of classifiers focuses on how accurately it predicts true negative documents. True negative documents are the ones to be excluded and not to be screened by SR experts. Thus, they represent the workload saved by classifiers. An evaluation measure called Work Saved over Sampling (WSS) was proposed in [14] to estimate how many documents can be excluded from the manual screening. In other words, it measures how much workload can be reduced. This measure is actively used for the evaluation in systematic reviews. Regarding WSS, more details are in Chapter 3.

In addition to classifier algorithms, document representations are also a crucial component affecting the performance of supervised approaches. Many studies evaluate extensive features to effectively distinguish relevant documents from irrelevant documents. Lexical features such as bag-of-words (BOW) and part-of-speech (POS), and semantic features employing domain knowledge (*e.g.*, definition/description of medical terminology) are evaluated [17, 30]. In [34], they utilize metadata in documents such as publication types and publication venues as the elements of features. These attempts result in high-dimensional sparse vectors as well as requiring laborious feature engineering. Low-dimensional dense vector representations are also studied in [22]. They represent documents using paragraph vectors and Latent Dirichlet Allocation (LDA). Lastly word embeddings and document embeddings are used in the recent work [20, 21]. More details on document features are summarized in Table 2.2.

Active Learning. Different from the approach that employs other completed SRs to train classifiers, some studies propose an active learning (AL) setup. Given a target SR, classifiers are

Fig. 2.1: Iterative process of active learning

Table 2.3: Classifiers and feature representations evaluated in active learning setup

	Types	Literature
Classifiers	SVM	[12, 13, 22, 26, 27]
	Logistic regression	[18]
	Distance metrics (<i>e.g.</i> , cosine)	[36]
Features	BOW	[12, 13, 27]
	Controlled vocabularies	[12, 26]
	TF-IDF	[18, 26]
	Paragraph vectors	[22]
	Topic modeling	[27]
	Knowledge bases	[36]

iteratively trained with its own labeled documents [13, 14, 27]. Initially there are no labeled candidate documents for a target SR. At first, SR experts are asked to label a few number of documents (random selection), and then the labeled documents train a classifier. The classifier selects a few among the unlabeled documents and SR experts are asked again to label them. The process is iterative and the classifier is re-trained with more labeled documents for performance improvement. The process of active learning is described in Figure 2.1

Many existing studies have evaluated the active learning approach [18, 26, 27, 37]. Similarly, in active learning setting, SVM and logistic regressions are often used as classifiers and a variety of features are studied for document representations. Table 2.3 presents the classifiers and the features explored in active learning setting.

In addition to classification algorithms and document representations, selection strategies for labeling documents is another component that affects the effectiveness of active learning. As

candidate documents are extremely imbalanced in systematic reviews (see Table 1.2), choosing appropriate selection strategies is even more critical in this task. Imbalanced candidate documents are challenging, because most of labeled documents (training data) are highly likely to be irrelevant documents. Once classifiers are trained on many irrelevant documents in the early stage of iterative process, the classifiers are determined and they are hard to be changed by few relevant documents added later. To provide relevant documents as training data in the beginning, several selection strategies have been studied. Both certainty-based AL and uncertainty-based AL have been evaluated in multiple areas of SRs in [27]. Certainty-based strategy selects the most confident documents to be labeled and uncertainty-based strategy chooses the least confident documents. Based on their results certainty-based strategy is more effective than uncertainty-based strategy.

It is reported that reweighting strategies are also effective to alleviate the imbalance in training dataset [26]. The reweight strategies give the higher weight on positive (relevant) training instances than negative (irrelevant) training instances. It encourages classifiers to reflect relevant documents even though they are few and labeled later than many irrelevant documents. At the same time, it prevents classifiers from determining their performance with early labeled documents as well.

Lastly, the active learning called patient active learning (PAL) is proposed in [12] and it is specialized active learning schema for the imbalanced datasets in systematic reviews. Patient active learning is motivated from hasty generalization on classifiers. Likewise, once a classifier has seen certain amounts of training data, classifiers are generalized and rarely make a large change with later training data. For example, a hyperplane in SVM is *set* by the initial training data and in iterative process the hyperplane *adjusts* its position. It is unlikely that it ends up far from the initial location due to the dominance of irrelevant documents. PAL suggests to not start training a classifier until the certain number of relevant documents are included in training data. In this way, it discourages classifiers to be pre-determined by many irrelevant documents.

Performance of active learning varies over areas/topics of systematic reviews. Two different topics of systematic reviews are used and their performances of AL are compared in [27]. The results report that systematic reviews related public health show poor performance in AL. This is possibly because public health topics tend to have broader coverage and inconsistent terminologies. It is shown that they can benefit from word embeddings or topic modelings in active learning.

Another topic related to active learning is to find when to stop training a classifier. Several studies propose stopping strategies [18, 37, 38]. For example, in [38], two strategies (Target method and Knee method) are proposed and evaluated for e-discovery in legal areas and total recall tasks for general web documents. Later they are also tested on systematic reviews in EBM [18] and empirical results show Knee method performs well with the CLEF2017 dataset.

2.1.2 Reducing the Number of SR Experts Needed for Screening

Multiple guidelines on conducting systematic reviews recommend to make sure that every candidate document is screened by multiple SR experts and have the experts work independently. This is because SR experts may make biased judgment or mistakes by accident. Collective relevance judgments from multiple experts increase the completeness of document screening. However, while involving multiple SR experts in screening process is necessary and even required, it is very expensive to employ multiple experts with medical domain knowledge for several months.

Several studies propose to use a machine learning classifier as a SR expert or use it for supplementing human reviewers to reduce the cost of screening process [16, 17, 23, 30]. To achieve this goal, they train classifiers and the similar algorithms used in reducing documents to be screened (Table 2.1) are explored. For the evaluation of classifiers, agreements on relevance judgments by classifiers and human SR reviewers are examined. The reported results are positive. For example, in [30] a classifier reduces human's workload by 88%.

2.1.3 Increasing the Rate of Screening Documents

It takes several seconds to a few minutes for an experienced SR expert to screen a single document. The previous two approaches improve the screening process by reducing the number of documents or SR experts. Unlike them, the third approach aims to assist SR experts to make relevance judgments quickly, while the resources (*e.g.*, documents) remain unchanged. Data mining techniques with visual demonstrations are adopted to achieve this goal.

Techniques to identify similar candidate documents are developed and similar document groups are presented visually. It provides a macro view of how a large candidate collection is composed. Thus, SR experts can make fast consecutive decisions by screening similar documents, and eventually it helps speed up the overall manual screening process. Existing work

examines the document content and author connections to find clusters of similar candidate documents [31, 32, 39]. Besides, several studies focus on approaches to extract major elements from candidate documents. Keywords or key sentences are extracted or highlighted in documents to support the faster screening process [40, 41]. The third approach is the least explored approach among the four.

2.1.4 Screening Prioritization

Screening prioritization (document ranking approach) provides rankings of candidate documents where relevant documents are likely to be at the top. The prioritized candidate documents allow SR experts to have the efficient workflow of screening process. When candidate documents are ranked by their likelihood of being relevance, candidate documents with lower ranks can be screened by relatively less experienced SR experts or fewer number of SR experts [24]. More importantly, it makes the two screening steps (abstract and full text screenings) to be in parallel [8]. Screening ranked candidate documents allows relevant documents to be found as early as possible. The full text screening for early identified relevant documents is conducted at the same time, while the abstract screening for the candidate documents with low rankings are in process. It is reported in a survey that this screening prioritization approach is the one ready to be used in practice, among the four approaches [8].

In screening prioritization, both supervised and unsupervised algorithms are studied. At the early stage of screening prioritization, supervised approaches were mainly explored. SVM, logistic regression, learning-to-rank are evaluated in several studies [3, 24, 42, 43]. Similar with the approaches of reducing the documents to be screened, classifiers are trained but they do not provide explicit inclusion/exclusion decisions of documents. Instead, they provide a ranked list of documents according to its predicted scores.

Recently, unsupervised approaches were evaluated for screening prioritization. Classical retrieval models, such as BM25 and query language model, are attempted [43, 44]. Keyword queries generated in retrieval process are used as a query.

CLEF 2017-2019 eHealth Task 2. CLEF 2017 eHealth Task 2 was organized as a new task with the aim of screening prioritization [3, 45]. Participants were provided with candidate documents and their relevance labels of training SRs. Then the participants developed their methods on training data (*e.g.*, 20 SRs) and the performance of the proposed methods was evaluated on test

data (e.g., 30 SRs).s In this competition, two types of approaches were examined: (i) document ranking with no relevance feedback, and (ii) with relevance feedback such as active learning. The first option is exactly same as screening prioritization. The second option is similar with the task with reducing the number of documents to be screened using active learning. The best performance ranking (in terms of average precision) in the first category was achieved by utilizing keyword queries with cosine similarity [44]. Though it is the best result, an average precision value is lower than 0.2. It indicates the difficulty of this task. With relevance feedback, the best performance was achieved by [18]. The authors adopted continuous active learning (CAL) in SRs, and CAL was developed for technologically assisted reviews (TAR) such as e-discovery in legal area [38,46]. Their result shows that CAL can improve the screening process of systematic reviews in medical domain as well. Since CLEF 2017, CLEF 2018 and 2019 were organized with more data. Participants were provided with additional information of SRs such as Boolean keyword queries and short descriptions of relevance conditions (*i.e.*, protocols).

2.2 Query by Document

Seed-driven document ranking (SDR) assumes that one relevant document is given before screening process, and makes use of it as a query in screening prioritization. It can be seen as a *query by document* task. In this section, we present existing studies in query by document and also discuss differences between them and SDR.

Document ranking by a query document (document as query) has been studied in general and domain-specific problems. As a general retrieval problem, given web page as query, finding similar blog posts or recommending news articles have been explored [47,48]. Existing studies reduce terms to keywords and attempt to make a query document as query keywords which are more common format in information retrieval. It is motivated by that a query document contains noisy term compared to specific information need. In [47], for example, the authors first extract candidate phrases from query document and then score candidate phrases in multiple ways, *e.g.*, using statistical information or using an external knowledge base.

In domain-specific areas, patent search or academic literature search are example tasks [49, 50]. In patent search or academic literature search, finding similar documents with given query

document is a common scenario. For both problems, they also try to reduce terms in query document by identifying key phrases. In [51] a query document is incorporated with two representations. One captures an overall characteristic of the document, and the other is a representation of key phrases in document. The authors claim that utilizing two different representations of a query document complements each other and leads to performance improvement.

Several studies formulate the problem of query document as query suggestion or query diversification in patent search [50,52]. Similarly, key concepts in query document are identified while considering different aspects of documents. Based on document distribution in corpus, related keywords are identified and they are suggested as new queries. It allows users to issue short query keywords using the suggested queries.

Different from the aforementioned tasks, systematic reviews have its unique characteristics in defining relevance. Relevance in systematic reviews depends on multiple aspects of documents. In other words, relevance in systematic reviews involves explicit multiple components. The relevance can be in various expressions from key phrases to small details in documents. For instance, gender/age of subjects in clinical experiment and randomized experiment design are can be relevance conditions in SRs. They are neither frequently appeared terms nor key phrases in clinical documents. Furthermore, candidate documents are similar with each other in some sense, because they are filtered document results through the retrieval process. Therefore, we propose an approach to leverage clinical terms in query document and incorporate domain-specific characteristics of systematic reviews to estimate relevance of clinical terms in seed document (Chapter 3).

Ranking by document matching (i.e., similarity) has been less explored in document ranking by query document. This is possibly because computing pairwise similarity scores in large-scale corpora online is very costly. Several studies propose ways to improve the efficiency in pairwise similarity search problems [53–55]. However, in this task, candidate documents are typically in the scale of several thousands of documents. In the CLEF 2019 dataset, the average number of candidate documents is about 6,000. While the computational cost on a few thousand documents can be higher than that in retrieval models, it is much lower than the costs of manual screening which takes weeks to months. We present ranking by document matching in screening prioritization in Chapter 4.

2.3 Summary

In this chapter, we introduce related work in the efficient screening process with text mining techniques and studies in query by document. The related work on addressing the greater efficiency in screening process can be categorized by four types based on their goals: reducing documents to screen, reducing SR experts needed for screening, increasing the rate of screening, and improving workflow via prioritized candidate documents. It is shown that providing prioritized candidate documents are the most suitable approach to use in practice [8]. Again, the performance of state-of-the-art approach in screening prioritization is less than 0.2 in average precision. This indicates the challenge of screening prioritization task.

Second, we present existing work on query by document because our approach utilizes a seed document as a query. The existing work on query by document identifies key phrases in query document and reduces document terms to a few key phrases. However, it may not work in systematic review problems, because relevance in systematic reviews depends on explicit and detailed information including key phrases. Additionally, candidate documents share similar key phrases since they are retrieved by keyword queries through Boolean search.

Chapter 3

Seed-driven Document Ranking

3.1 Introduction

To define a clinical question and relevance conditions, SR experts carefully go through literature and existing SRs. It is natural that SR experts know the existence of one or two relevant documents after defining the clinical question, *i.e.*, before the screening process. In this chapter, we propose a solution named ‘seed-driven document ranking’ (SDR) with the assumption that one relevant document (seed document) is known.¹ Now, the research question becomes: *“Given a seed document and candidate documents, how to rank the candidate documents such that (hidden) relevant documents appear at the top of the ranking?”*

At first glance, this is a problem of query by example document. A common approach is to identify key phrases from the example document as queries to rank documents. However, the relevance conditions defined in SRs cover diverse elements in documents, and their details *e.g.*, gender/age of subjects in clinical trial and randomized experiment design. To this end, we conduct a thorough analysis of relevant documents and irrelevant documents in candidate documents. Based on the analysis we propose to represent the seed document and candidate documents using ‘bag of clinical terms’. The clinical terms are identified by using external thesaurus *i.e.*, UMLS medical thesaurus². Note that, this is different from common approaches in query by example where the key phrases are identified using statistical measures.

¹This chapter is published as Grace E. Lee and Aixin Sun. Seed-driven Document Ranking for Systematic Reviews in Evidence-based Medicine, Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 2018), 455–464, 2018.

²Unified Medical Language System <https://www.nlm.nih.gov/research/umls/index.html>

More importantly, we propose a method to estimate the weight of a term by leveraging high similarities between relevant documents in SRs. Relevant documents have higher intra-similarities than irrelevant documents, because all the relevant documents must meet *all* the relevance conditions.³ Therefore, we estimate the importance of a term based on its distribution in the candidate documents and their similarities to the seed document. In this regard, the proposed weighting scheme finds relevant documents without spending considerable effort identifying relevance conditions explicitly. We adopt and modify the query likelihood retrieval model with the weight scheme to estimate the relevance.

Recently, there are many studies on using distributed representation of words, or word embeddings, for different tasks in Information Retrieval. In our setting, a document can be easily represented based on the embeddings of its contained words. Using cosine similarity, one can get a ranking towards the seed document. In our experiments, we evaluate different document representations including bag of clinical terms, bag of words, and word embeddings, and different retrieval models including BM25 and query likelihood model. Our experimental results show that SDR outperforms all baseline methods and beat the best performance (without using relevance feedback) reported in CLEF 2017. More interestingly, we observe that relevance estimated based on word embeddings well complements SDR. The best ranking is achieved by combining the relevances estimated by SDR and by word embeddings. We also simulate the manual screening process and it demonstrates the efficacy of the proposed approach when multiple relevant documents are known.

3.2 Proposed Approach

Given a seed document d_s and a set of candidate documents $C = \{d_1, d_2, \dots, d_{|C|}\}$, $d_s \notin C$, our task of *seed-driven document ranking* is to rank the documents in C such that the relevant documents are ranked at top positions. To achieve this, we first conduct an analysis on the characteristics of relevant documents in SRs, in Section 3.2.1. Based on the observations, we present document representation scheme and term weighting method for ranking in Section 3.2.2.

³This argument can be also supported by Van Rijsbergen’s cluster hypothesis [56,57].

Fig. 3.1: Intra-similarity among relevant documents and irrelevant documents in SRs.

3.2.1 Observations on Relevant Documents

Shown in Table 1.2, systematic reviews have very skewed data distribution in terms of relevant documents among candidate documents. More specifically, only about 1% of candidate documents are relevant. Understanding the characteristics of the relevant documents is therefore the key to fully utilize the seed document and achieve the best ranking. Through exploration of the data, we make the following two observations.

Observation 3.1. *For a given SR, its relevant documents share higher pair-wise similarity than that of irrelevant documents.*

In systematic reviews, relevance judgment depends on the relevance conditions defined when formulating the clinical question. Recall that every SR has different relevance conditions and the conditions assess multiple aspects of a candidate document to determine its relevance. However, regardless of specific criteria in relevance conditions, ‘relevance’ carries the universal meaning in all SRs: *if a document meets **all relevance conditions**, its label is positive (relevant); if a document satisfies partial conditions or none, its label is negative (irrelevant).*

Positive labels on relevant documents ensure that they fully satisfy the relevance conditions. Therefore, relevant documents for a given SR share high commonalities in terms of relevance conditions, suggesting that they share great similarities among them. On the other hand, all candidate documents are retrieved based on keyword queries derived from the clinical question and relevance conditions. Therefore, every document in the candidate set is relevant to the

Fig. 3.2: Distribution of clinical terms (in BOC) and words (in BOW) over the five ranges of normalized document frequency in relevant documents (best viewed in color)

relevance conditions to some degree. To verify to what extent the relevant documents are more similar to each other, we conduct a simple test.

We calculated the average intra-similarity of relevant documents for each SR in the CLEF17 dataset. Specifically, we calculated the pair-wise cosine similarity of relevant documents using TF-IDF representation. We under sample irrelevant documents to the same number of relevant documents at random for each SR, and calculate their pair-wise similarity. We repeat this process for 10 times to estimate the intra-similarity of irrelevant documents. Plotted in Figure 3.1, on the 50 SRs in CLEF17 dataset, the intra-similarity of relevant documents and that of irrelevant documents are 0.266 and 0.109, respectively. This result suggests that: (i) relevant documents are indeed similar with each other, and (ii) it is challenging to fully rely on cosine similarity with TF-IDF representation to identify the remaining relevant documents with a seed document.

Observation 3.2 *Relevant documents for a given SR share high commonality in terms of clinical terms.*

Relevance conditions in SRs are defined in the first step of conducting SRs. As the initial step of defining relevance conditions, they are structured with several key clinical terms following the PICO model² as a basis. Relevance conditions are then further elaborated and explicitly specified to cover more details, mostly defined in clinical terms. As relevant documents must

fully satisfy all the relevance conditions, in terms of clinical terms relevant documents for a given SR are expected to share high commonality each other.

Without loss of generality, we consider a word or a phrase to be a *clinical term* if it matches an entry in the Unified Medical Language System (UMLS)². There exist many off-the-shelf tools to identify clinical terms in UMLS from medical documents, *e.g.*, NCBO BioPortal API⁴ and QuickUMLS [58].

Now, a relevant document can be represented as a ‘bag-of-clinical terms (BOC)’. We evaluate the distribution of clinical terms in relevant documents, using ‘bag-of-words (BOW)’ with stopwords removal as reference. More specifically, we are interested in the ratio of clinical terms (resp. words in BOW representation) that are shared among relevant documents. To this end, we compute the normalized document frequency (DocFreq) of each clinical term (resp. word), and bin them into five ranges: (0.0, 0.2), [0.2, 0.4), [0.4, 0.6), [0.6, 0.8), [0.8, 1.0]. A clinical term has a normalized DocFreq of 1.0 if it appears in every relevant document in a given SR, and 0.8 if it appears in 80% of relevant documents of the SR. Figure 3.2 plots the ratio of the clinical terms/words in each of the five ranges.

Observe that the vast majority of words in BOW representation are in the range of (0.0, 0.2) and very few are in the ranges of [0.6, 0.8) and [0.8, 1.0], if even noticeable. However, represented in BOC, a larger portion of clinical terms appears in higher DocFreq ranges. This comparison suggests that, ‘bag-of-clinical terms’ representation is much more effective in finding commonality between relevant documents in a SR.

Summary. Through the analysis of candidate documents and relevant documents, we show that the relevant documents share higher pair-wise similarity and bag-of-clinical terms representation is more effective in identifying commonality among relevant documents. Based on the observations, with reference to the seed document, we design a term weighting scheme to give more weight to the clinical terms that are likely to be shared among all relevant documents.

3.2.2 Weighting Scheme and Document Ranking

Based on Observation 3.2, ‘bag-of-clinical terms’ (BOC) representation is more effective in capturing the characteristics of relevant documents. From now onwards, we use BOC to represent both seed document and all candidate documents. For simplicity, we use ‘term’ to refer a ‘clinical term’.

⁴<https://bioportal.bioontology.org/>

Observation 3.1 states that relevant documents share higher pair-wise similarity. The terms that make relevant documents close to each other and distant from irrelevant documents should be promoted to have a higher weight. Without knowing all relevant documents (which is the ultimate goal), we estimate the weight of a term in seed document by measuring to what extent it separates the documents similar to the seed document and those dissimilar.

Let $t_i \in d_s$ be a term in seed document d_s . Let $D_{t_i} \subseteq C$ denote the subset of candidate documents where t_i appears in these documents, and $D_{\bar{t}_i} \subseteq C$ denote the subset of candidate documents where t_i does not appear. The conditions, $D_{t_i} \cup D_{\bar{t}_i} = C$ and $D_{t_i} \cap D_{\bar{t}_i} = \emptyset$, always hold. The weight of term t_i in d_s , denoted by $\varphi(t_i, d_s)$, is estimated by Equation 3.1.

$$\varphi(t_i, d_s) = \ln \left(1 + \frac{\delta(D_{t_i}, d_s)}{\delta(D_{\bar{t}_i}, d_s)} \right) \quad (3.1)$$

In Equation 3.1, $\delta(D_*, d_s)$ is the average similarity between d_s and all documents in D_* , defined as follows.

$$\delta(D_*, d_s) = \frac{1}{|D_*|} \sum_{d_j \in D_*} \text{sim}(d_j, d_s) \quad (3.2)$$

We measure the similarity between d_j and d_s by cosine similarity of their TF-IDF term vector representations. Note that, Equation 3.1 allows an infrequent term in the candidate collection to have a high weight if the documents where the term is appeared are very similar to the seed document.

In the following section, we present a ranking model using the query term weight to calculate the probability that a document d in a set of candidate documents C is relevant.

In SDR, we adopt the query likelihood language model with Jelinek-Mercer (JM) smoothing, as the ranking function. The original ranking function is defined in Equation 3.3 (see [59] page 126 for the derivation):

$$\text{score}(d, q) = \sum_{t_i \in d, q} c(t_i, q) \cdot \log \left(1 + \frac{1 - \lambda}{\lambda} \cdot \frac{c(t_i, d)}{L_d \cdot p(t_i | \mathbb{C})} \right) \quad (3.3)$$

where $c(t_i, q)$ and $c(t_i, d)$ denote the count of term t_i in query q and in a document d , respectively, and L_d is the length of document d in number of terms. $p(t_i | \mathbb{C})$ is the probability of term t_i in the background corpus. λ is JM smoothing parameter.

Using d_s as a query, we incorporate the term weighting $\varphi(t_i, d_s)$ into the ranking function in SDR, as follows:

$$score(d, d_s) = \sum_{t_i \in d, d_s} \varphi(t_i, d_s) \cdot c(t_i, d_s) \cdot \log \left(1 + \frac{1 - \lambda}{\lambda} \cdot \frac{c(t_i, d)}{L_d \cdot p(t_i | \mathbb{C})} \right) \quad (3.4)$$

In Equation 3.4, $p(t_i | \mathbb{C})$ is estimated by maximum likelihood estimation (MLE) on candidate document set C . We empirically set $\lambda = 0.2$ in our experiments.

When multiple seed documents become available, *e.g.*, during the actual screening process, d_s can be expanded to include additional relevant document $d_{s'}$ to form a new query and re-rank the remaining candidate documents. We also note that the proposed weight scheme and ranking function can be easily adopted to ‘bag-of-words’ representation by treating each word as a term.

3.3 Experiment Setup

We design two sets of experiments to evaluate the performance of SDR and alternative ranking models.

In the first set of experiments, we assume a *single* seed document is available for each SR, and compare the quality of candidate document ranking with different baseline ranking models. Because using different seed documents from the same SR may deliver different rankings, we average the performances obtained by *using each relevant document as the seed document* to derive the final ranking performance.

In the second set of experiments, we simulate the screening process conducted by SR experts in an iterative manner. Starting with the ranking obtained by using a single seed document, we assume SR experts manually screen a batch of top k documents to determine their relevance. The relevant documents identified in the k documents are used to expand the seed document as a new query and get a new ranking. The process iterates with the screening of another batch of top- k documents. This simulation aims to evaluate the effectiveness of SDR with multiple relevant documents available. Next, we detail the dataset, baseline methods, and evaluation metrics. The experimental results are reported and discussed in Section 3.4.

3.3.1 Dataset

We use the dataset provided by the CLEF 2017 eHealth Task 2. CLEF17 dataset consists of total 50 diagnostic test accuracy (DTA) systematic reviews. DTA SRs are known for a high level of

difficulty in finding relevant documents because of their complexities. As shown in Table 1.2, in general less than 1% of candidate documents are relevant among the 50 SRs. More detailed statistics of the dataset are reported in [3].

In CLEF17 dataset, 50 SRs are divided into two sets: 20 SRs as training set (for developing supervised models), and 30 SRs as test set. We evaluate our models on both the 30 SRs so as to compare with the best performance reported in the competition, and also the 50 SRs as our model requires no training.

The CLEF17 dataset releases only the PubMed document identification numbers (pmids). In our experiments, we obtain titles and abstracts of the clinical documents from PubMed⁵ using the provided pmids. For relevance labels, each document has two relevance labels. One is the label from abstract screening, and the other is from full text screening. We use the final relevance labels in our experiments, since these labels indicate the final relevance judgment by SR experts. In short, in our experiments, the ranking is based on titles and abstracts of the documents, and the quality of the ranking is evaluated by the final relevance labels. The same setting is applied to all the models in comparison.

There is a recently published dataset for evaluating the retrieval process in systematic reviews [7]. However, this dataset does not contain information (*e.g.*, pmids) of all candidate documents of its SRs. Hence it cannot be used in our experiments.

Clinical Term Extraction. Clinical terms were identified by using NCBO BioPortal API⁴ and QuickUMLS [58]. Both are tools that annotate free-text with clinical terms in UMLS medical thesaurus. For QuickUMLS, UMLS 2016AB version was used with default parameter settings. The extraction results from the two tools are mostly similar. To avoid missing any terms, we combined the extraction results by adding clinical terms extracted by QuickUMLS which do not exist in the results of BioPortal.

Represented in bag-of-clinical terms, the length of a document (in number of terms) is on average 15% of the same document represented in bag-of-words.

3.3.2 Baseline Methods

We compare SDR with two classic ranking models in information retrieval, namely **BM25** and the query likelihood model (**QLM**).

⁵<https://www.nlm.nih.gov/bsd/pmresources.html>

For BM25, we used the implementation in gensim v3.2.0 toolkit⁶. For QLM, we used JM smoothing (see Equation 3.3). By comparing SDR (see Equation 3.4) with QLM, we examine the impact of the proposed weighting scheme $\varphi(t_i, d_s)$ in Equation 3.1. In experiments, we also evaluate the effectiveness of ‘bag-of-clinical terms’ (**BOC**) compared to ‘bag-of-words’ (**BOW**) representation. As a result, we evaluate six rankings: three ranking models (BM25, QLM, and SDR) on two representations (BOC and BOW)

In addition, we evaluate a ranking model based on word embeddings, which we call average embedding similarity (**AES**)⁷. Word embedding is an unsupervised approach to learn continuous distributed vector representations of words. It is shown in many studies that the dense vector representation is effective to encode linguistic regularities and semantic similarities among words. In AES, a document is represented by the average embeddings of its contained words [60, 61]. A candidate document is ranked by its cosine similarity to the seed document computed on the average word embedding representation. In our implementation, we used the word embeddings pre-trained on PubMed corpus and Wikipedia.⁸

3.3.3 Evaluation Metrics

To evaluate ranking effectiveness, we report both standard retrieval evaluation measures and task-specific evaluation measures.

Standard IR Measures. For overall ranking effectiveness, we report average precision ($AvgPr$), precision at top k documents ($Pr@k$) and recall at top k documents ($Re@k$).

$Pr@k$ is the proportion of top- k ranked documents that are relevant. $Re@k$ is the proportion of relevant documents that ranked in the top- k positions. $AvgPr$ is the average of the precision values each obtained when a relevant document is retrieved. As mentioned earlier, for a given SR, every relevant document can be used as a seed document to rank candidate documents. The above three measures are computed as the average over all the values obtained after each relevant document is used as a seed document. In this sense, $AvgPr$ is the mean average precision for

⁶<https://radimrehurek.com/gensim/index.html>

⁷In recent years, contextualized word embeddings have advanced various downstream tasks. Here, word embeddings can be replaced with contextualized word embeddings such as BERT or SpanBERT. However, in this thesis the problem setting is intrinsic evaluation setting which does not fine-tune contextualized word embeddings and the performance difference between word embeddings and contextualized word embeddings is expected to be small. We provide more detailed experiment results and discussion in Section 4.4.2 in Chapter 4.)

⁸<http://bio.nlplab.org/>

a given SR. The final measures reported are the average over all SRs evaluated in a given set. We report the measures on the 30 SRs in the test set of CLEF17 and the 50 SRs in the entire dataset of CLEF17, as two sets.

LastRel% and WSS. The goal of screening in SRs is to identify *all* relevant documents from candidate documents. In CLEF17 competition the submissions are evaluated by two task-specific measures: *LastRel%* and *WSS*.

LastRel is the ranking position of the last relevant document, *i.e.*, the minimum number of documents need to be screened to get all relevant documents. *LastRel%* is the normalized value by the number of (unlabeled) candidate documents as SRs have different number of candidate documents, and the smaller, the better.

Work Saved over Sampling (*WSS*) is originally proposed in [14]. It measures how many documents in candidate documents can be removed from manual screening. Assuming manual screening starts from the top of the ranking, until the last relevant document is found. Then the remaining candidate documents can be safely ignored, hence the work saved. Specifically, $WSS = (|C| - LastRel)/|C|$. Here, $|C|$ is the number of candidate documents and LastRel is the ranking position of the last relevant document.

The difference between *LastRel%* and *WSS* emerges during the screening process. As the screening process proceeds and more documents are manually labeled, LastRel changes accordingly in the unlabeled documents. However, *WSS* remains unchanged since the number of documents after the LastRel does not change. Note that both *LastRel%* and *WSS* will change if the unlabeled documents are re-ranked from time to time during the screening process *e.g.*, based on relevance feedback or after getting more seed documents in SDR.

3.4 Results and Discussion

We report the overall ranking performances of SDR and baselines in Section 3.4.1. In Section 3.4.2, we analyze the contribution of term weighting scheme and the performance of SDR on individual SRs. Then, we simulate the manual review process and evaluate the ranking when more seed documents become available in Section 3.4.3.

Table 3.1: Performance of SDR and baseline ranking models on 30 SRs (CLEF17 test set) and 50 SRs (the entire CLEF17 dataset). The best performance among *all* models is in boldface and italics. Among SDR and baselines (*i.e.*, not including CLEF-Query and SDR+AES), the best performance is in boldface and the second best is underlined. * denotes the statistically significant difference between SDR-BOC and a baseline. † and ‡ are significance test results of SDR+AES over AES and SDR-BOC, respectively. Statistical significance is by one-tailed paired t-test with $p < 0.05$.

Dataset	Ranking Model	AvgPr	Pr@10	Pr@20	Pr@30	LastRel%	Re@10	Re@20	Re@30	WSS
30 SRs	CLEF-Query	0.18	-	-	-	46.0	-	-	-	0.54
	BM25-BOW	0.161*	0.176*	0.145*	0.126*	52.9*	0.246*	0.330*	0.385*	0.470*
	QLM-BOW	0.159*	0.165*	0.138*	0.118*	52.0*	0.245*	0.324*	0.376*	0.479*
	SDR-BOW	0.181	0.201*	0.166*	0.139*	46.7	0.257	0.353*	0.401*	0.532*
	BM25-BOC	0.213*	0.233*	0.180*	0.150*	46.5	0.261*	0.345*	0.408*	0.534*
	QLM-BOC	0.214*	0.228*	0.180*	0.150*	43.3*	0.264*	0.361*	0.415*	0.566*
	SDR-BOC	0.227	0.238	0.189	0.157	<u>39.8</u>	<u>0.273</u>	0.367	0.436	<u>0.600</u>
	AES	0.211	0.224	0.175	0.149*	38.7*	0.285*	<u>0.364</u>	<u>0.420*</u>	0.612
	SDR+AES	0.264 ^{†‡}	0.276 ^{†‡}	0.213 ^{†‡}	0.177 [†]	32.5 [†]	0.315 [†]	0.413 [†]	0.484 ^{†‡}	0.673 ^{†‡}
50 SRs	BM25-BOW	0.147*	0.179*	0.146*	0.128*	57.4	0.234	0.305	0.363	0.425*
	QLM-BOW	0.141*	0.168*	0.137*	0.119*	55.7*	0.233*	0.297*	0.343*	0.442*
	SDR-BOW	0.170*	0.205*	0.167*	0.144*	48.5	0.247	0.323	0.377	0.514
	BM25-BOC	0.164*	0.190*	0.151*	0.128*	46.4*	0.230*	0.296*	0.345*	0.535*
	QLM-BOC	0.167*	0.193*	0.156*	0.132*	43.3*	0.233*	0.307*	0.353*	0.567*
	SDR-BOC	0.178	<u>0.202</u>	<u>0.164</u>	<u>0.139</u>	39.8	<u>0.240</u>	<u>0.312</u>	<u>0.369</u>	0.601
	AES	0.147*	0.171*	0.134*	0.115*	50.5	0.238	0.294	0.333*	0.492*
	SDR+AES	0.202 ^{†‡}	0.226	0.179 ^{†‡}	0.152 ^{†‡}	37.7 ^{†‡}	0.265 ^{†‡}	0.341 ^{†‡}	0.399 ^{†‡}	0.622 ^{†‡}

3.4.1 Overall Performance

Table 3.1 reports the overall ranking performance of three ranking models (BM25, QLM, and SDR) with two document representations (BOW and BOC), on two SR sets (the 30 SRs in CLEF17 test set, and the 50 SRs in CLEF17).

Reference method: CLEF-Query. As a reference we include the results of CLEF-Query, which is the best model in terms of *AvgPr* reported in CLEF2017 eHealth Task 2 with no relevance feedback [3]. CLEF-Query is reported on 30 SRs, the CLEF17 test set.⁹ CLEF-Query utilized the query keywords which retrieved documents from PubMed to rank candidate documents [44]. Direct comparison of CLEF-Query and other models in the table is unfair because of different problem settings. Nevertheless, the *AvgPr* of 0.18 demonstrates the difficulty of the task to identify a small number of relevant documents in systematic reviews.

SDR-BOC vs. Baselines. We first compare SDR-BOC with baseline models by BM25 and QLM with BOW and BOC representations, SDR-BOW and AES. We exclude the model SDR+AES in this comparison.

On 30 SRs, for measures defined on precision, SDR-BOC achieves the best *AvgPr* and all *Pr@K* ($K \in \{10, 20, 30\}$) measures. Paired t-test results show that the improvements made by SDR-BOC on top of other baselines are statistically significant in most cases, except *AvgPr* on SDR-BOW and AES, and *Pr@10* and *Pr@20* on AES. Observe that the second best values on precision measures are achieved by BOC representations as well. On task-specific measures *LastRel%* and *WSS*, the simple ranking model AES outperformed SDR-BOC by a small margin. For measures defined on recall, the performances of SDR-BOC and AES are comparable as either best or second best performances, among all the methods in comparison.

On 50 SRs, overall SDR-BOW delivers the best performance over *Pr@K* and *Re@K*, while SDR-BOC is the best performer in terms of *AvgPr* and task-specific measures *LastRel%* and *WSS*. Word embedding based ranking, becomes less competitive on this larger dataset. We also note that, overall there is a performance drop on all measures from 30 SRs to 50 SRs.

To summarize, we show the consistently better performance of SDR ranking model over BM25 and QLM, with both BOC and BOW representations. Word embedding based ranking,

⁹For CLEF-Query, individual performances of 30 SRs were not reported in [3]. We do not have *Pr@K* and *Re@K* values and we derive the values of *LastRel%* from $(1 - WSS)$.

AES, shows good potential to be a good performer, at least on the 30 SR dataset. In terms of document representation, BOC demonstrates clear superiority over BOW for both BM25 and QLM ranking models, on both datasets. For SDR ranking model, BOC wins in terms of $AvgPr$, $LastRel\%$, and WSS and loses to BOW in terms of $Pr@K$ and $Re@K$. Although BOC does not beat BOW on all settings, we argue that BOC representation has great advantages in enabling more efficient screening process. With BOC, a candidate document is represented by a small set of clinical terms defined in UMLS. In BOC representation, the length of a document is only about 15% of that in BOW representation. Furthermore, SDR estimates a weight for each clinical term, making it easy for SR experts to interpret the important clinical terms in screening.

From these results, SDR-BOC and AES are the best performers. Interestingly, the two rankings are from two completely different spaces. This motivates us to evaluate a combined ranking model to take advantage of both. Next, we discuss the performance of the combined model: SDR-BOC + AES, or simply denoted as SDR+AES.

Combined Model: SDR+AES. Since SDR-BOC and AES leverage different aspects of documents as feature for ranking candidate documents, we combine the relevance scores computed by the two models through linear combination. Specifically, let $s(d)$ be the normalized ranking score by SDR-BOC for document d , and $e(d)$ be the normalized ranking score by AES. The ranking score computed for SDR+AES is: $\alpha \cdot s(d) + (1 - \alpha) \cdot e(d)$. We set $\alpha = 0.3$ for both the 30 SRs and 50 SRs and report the results.

Reported in Table 3.1, SDR+AES, taking advantages from both spaces, is the clear winner on all measures, on both datasets. We also conducted significance tests by comparing the results of SDR+AES against each of its components, SDR-BOC and AES. The tests show that on most measures the difference is statistically significant. That is, the two types of features in the ranking model well complement each other. We believe that although BOC representation is able to identify the important clinical terms without ambiguity, the reduction of terms (*i.e.*, to the 15% of documents' original length) inevitably results in information loss. Dense vector representation, in this case, captures such information loss and contributes to the final ranking.

Performance Differences on 30 SRs and 50 SRs. Lastly, compared to 30 SRs, performance degradation is observed on 50 SRs. To investigate the potential reason why adding the additional 20 SRs leads to poorer performance, we evaluated the nearest neighbors of relevant documents

Fig. 3.3: The estimated term weight vs. document frequency of clinical terms in 50 SRs. In line graph a filled triangle is a mean value and an error range denotes variance.

in the 30 and 20 SRs. For each relevant document, we get the five most similar documents from candidate documents by cosine similarities on TF-IDF representations of documents. In 30 SRs, on average 3.82 documents out of the 5 most similar document are irrelevant documents ($sd = 1.19$). But in the additional 20 SRs, 4.32 documents out of 5 are irrelevant documents ($sd = 0.64$). The results suggest that the 30 SRs consist of relatively easier SRs to identify the relevant documents. In other words, relevant documents in the other 20 SRs are more similar with irrelevant documents, and thus more challenging to distinguish from the irrelevant ones.

3.4.2 Term Weighting Scheme and Individual SRs

Term Weighting Scheme. The proposed seed-driven ranking model incorporates domain-specific characteristics of medical documents in SRs (*i.e.*, Observations 3.1 and 3.2). Compared with the QLM, SDR introduces an additional term weighting component in the ranking model. The term weighting scheme estimates the importance of a clinical term to identify relevant documents in candidate documents. The experimental results show the significance of this additional component as SDR significantly outperforms QLM.

Fig. 3.4: Distribution of AvgPr of seed documents in 50 SRs. Each boxplot is a distribution of AvgPr values in one SR, sorted by their mean values in ascending order. Purple and blue boxplots indicate the SRs are from the train and test set in CLEF17, respectively. (best viewed in color)

In this section, we provide a close analysis on the contribution of the term weighting scheme $\varphi(t_i, d_s)$ in Equation 3.1. We bin the clinical terms based on their normalized document frequency in SR’s relevant documents in 10 ranges as shown in the x -axis of Figure 3.3. For the terms fall into each bin, we compute the mean and variance of the weights estimated by Equation 3.1 and plot the values in Figure 3.3. The total number of terms in each bin is also presented in histogram in the figure.

Identifying the terms having higher document frequency among relevant documents and giving them higher weights are essential in improving the ranking quality. As a document frequency increases, the number of terms in the corresponding range decreases as expected. Shown in the figure, our proposed weighting scheme is extremely effective in assigning higher weights to the terms with higher document frequency. As the normalized document frequency increases, the average of term weights increases as well. This results show that the proposed weighting scheme is effective in promoting terms that appear in many relevant documents.

Performance of Individual SRs. Each SR has its unique clinical question and accordingly, relevance conditions. Depending on the SRs, relevance conditions vary in complexity and difficulty. While we have discussed the overall performance on the CLEF17 dataset (30 SRs and 50 SRs), in this following section, we present performances of individual SRs.

Figure 3.4 plots $AvgPr$ of individual SRs in SDR-BOC, sorted in ascending order. Each boxplot is an individual SR and it represents its performance distribution of using different relevant documents as seed document. The dotted gray line is the overall averaged $AvgPr$ of the 50 SRs in SDR-BOC, as presented in Table 3.1. As shown in Figure 3.4, each SR has dramatically different ranking performances. Some SRs have exceptionally high $AvgPr$ values and some only manage to achieve small values.

In the figure, we also distinguish the 20 SRs and 30 SRs that are in the train and test set of CLEF17 dataset by plotting them in purple and blue boxes respectively. The plot shows that a few SRs in the test set happen to have very high $AvgPr$ values. This also explains the poorer results on the 50 SRs in Table 3.1.

To investigate the different performances across SRs, we looked into original relevance conditions of a few example SRs. Table 3.2 reports simplified relevance conditions of two SRs with high $AvgPr$ values and two SRs with low $AvgPr$ values. Before discussing the different performances of SRs, we briefly explain the meanings of index test and reference test in SRs. Previously we introduced an example clinical question in Section 5.1: ‘Can laparoscopy (diagnostic test) assess the resectability of pancreatic cancer?’ Given the clinical question, laparoscopy is an evaluation target and the evaluation target is called index test in DTA SRs. Performance of index test is compared with another test called reference test for evaluation. We present index test and reference test as representatives of relevance conditions in Table 3.2. The first two SRs with high $AvgPr$ values, [62] and [63], have relatively simple relevance conditions. The other two SRs with low $AvgPr$ values, [64] and [65], have many more elements in their relevance conditions. More elements in relevance conditions indicate a broader range of relevance and thus it leads to higher complexities in relevance conditions. Hence, the relevance conditions reflect the difficulty of SRs and also explain different $AvgPr$ values of individual SRs.

Additionally, seed documents show varying performances within each SR. Even though every relevant document can be a seed document, their relation with other relevant documents may lead to different performances. For example, two relevant documents may cover two different reference tests listed in relevance conditions. Regarding varying performance of different seed documents, we acknowledge that it is an interesting direction to further explore. For example, investigating factors of high ranking performance in seed documents and identifying them among candidate documents can be our future work.

Table 3.2: Simplified relevance conditions of example SRs and their AvgPr values. The main clinical terms are manually extracted from the detailed relevance conditions (see references for the original relevance conditions).

SRs	Index test	Reference test	AvgPr
[62]	Capsule endoscopy	Oesophago-gastro-duodenoscopy	0.975
[63]	Non-invasive biomarkers	DMSA renal scan	0.638
[65]	Ultrasound Liver function tests	Surgical/endoscopic extraction Surgical/endoscopic negative result	0.004
[64]	Red flags (27 questions)	Plain radiographs Computed tomography (CT) Magnetic resonance imaging (MRI) Bone scan	0.001

3.4.3 Simulation of Manual Screening Process

The goal of screening process is to find *all* relevant documents. SR experts will screen individual documents in practice from the top of the ranked documents. Hence, we evaluate the ranking in the setting of manual screening process, in an iterative manner. The simulation process starts with a single seed document. We set a batch size (B) and assume that SR experts each time label a batch of documents. The obtained relevant documents from the labeled batch can be appended to the seed document to update the ranking of remaining unlabeled candidate documents. We run 3 iterations, and each iteration is based on the updated ranking result of the previous iteration.

We evaluate the three best models: SDR-BOC, AES, and SDR+AES. We use the same value of $\alpha = 0.3$ in SDR+AES and set the batch size B to be 10. That is, in each iteration, the top 10 ranked candidate documents in the current ranking will be labeled by SR experts.

Table 3.3 reports the evaluation results. Likewise, we report separated results on 30 SRs and 50 SRs. Observe from Table 3.3, SDR+AES is a clear winner on both 30 SRs and 50 SRs with all measures. The improvement over its component rankers (SDR-BOC and AES) is statistically significant in almost all comparisons. Although the second best performer is not always SDR-BOC, it consistently outperforms AES on task-specific measures *LastRel%*, *WSS*, and also *AvgPr* and *Re@K*. AES wins SDR-BOC on a few *Pr@K* measures.

We claim that as the manual screening starts, the task-specific measures such as *LastRel%* become more critical to find all relevant documents as soon as possible. Specifically, as more documents are labeled, SDR+AES and SDR-BOC keep improving the ranking of last relevant

Table 3.3: Evaluation of updated ranking in simulation of manual screening process. In each iteration top 10 ranked documents are labeled and the relevant documents are appended to the seed to update the ranking. The best performance is in boldface and the second best is underlined. * denotes a statistically significant difference compared to the best, paired t-test with $p < 0.05$.

Dataset	Iteration	Model	AvgPr	Pr@10	Pr@20	Pr@30	LastRel%	Re@10	Re@20	Re@30	WSS
30SRs	1	AES	0.205*	0.193*	0.156*	0.141*	36.0*	0.393*	0.438*	0.485*	0.635*
		SDR-BOC	<u>0.241</u>	<u>0.240*</u>	<u>0.189*</u>	<u>0.161*</u>	<u>32.4</u>	<u>0.397*</u>	<u>0.470*</u>	<u>0.516*</u>	<u>0.670*</u>
		SDR+AES	0.275	0.276	0.218	0.190	25.5	0.455	0.513	0.559	0.738
	2	AES	0.243*	0.258*	0.234*	0.221	34.7*	0.457*	0.501*	0.538*	0.649*
		SDR-BOC	<u>0.244*</u>	0.196*	0.168*	0.151*	<u>28.8*</u>	<u>0.483*</u>	<u>0.533*</u>	<u>0.562*</u>	<u>0.705*</u>
		SDR+AES	0.351	0.299	0.245	<u>0.220*</u>	22.5	0.557	0.596	0.631	0.764
50SRs	3	AES	0.222*	0.236*	0.215*	0.203*	34.1*	0.510*	0.545*	0.574*	0.657*
		SDR-BOC	<u>0.242*</u>	0.217*	0.201*	0.192*	<u>25.6*</u>	<u>0.539*</u>	<u>0.573*</u>	<u>0.606*</u>	<u>0.733*</u>
		SDR+AES	0.347	0.343	0.308	0.285	20.5	0.623	0.661	0.679	0.781
	1	AES	0.141*	0.144*	0.118*	0.107*	49.0	0.314*	0.348*	0.382*	0.508*
		SDR-BOC	0.190	0.205*	0.165*	0.142*	33.4	0.338*	0.397*	0.435*	0.663*
		SDR+AES	0.212	0.227	0.183	0.162	30.0	0.372	0.425	0.472	0.696
50SRs	2	AES	0.167*	0.187*	0.165*	0.154*	48.4*	0.365*	0.396*	0.422*	0.516*
		SDR-BOC	<u>0.191*</u>	0.172*	0.148*	0.136*	<u>30.1*</u>	<u>0.409*</u>	<u>0.452*</u>	<u>0.491*</u>	<u>0.695*</u>
		SDR+AES	0.275	0.265	0.215	0.191	26.6	0.464	0.506	0.547	0.728
	3	AES	0.151*	0.170*	0.149*	0.140*	48.2*	0.405*	0.430*	0.451*	0.520*
		SDR-BOC	<u>0.189*</u>	<u>0.181*</u>	<u>0.168*</u>	<u>0.160*</u>	<u>27.6*</u>	<u>0.460*</u>	<u>0.502*</u>	<u>0.536*</u>	<u>0.719*</u>
		SDR+AES	0.282	0.306	0.259	0.231	24.5	0.537	0.582	0.602	0.748

documents. At the end of third iteration in 50 SRs, both models manage to rank the last relevant document at around top 25% among unlabeled documents. This is very promising result. AES alone, on the other hand, achieves top 30%. Similar with the results reported in Table 3.1, poorer results are observed on 50 SRs compared to the performance obtained on 30 SRs.

In this simulation, when more relevant documents become available, we simply append them to the seed document to form a new larger seed document in SDR. For AES, we compute the average of representations of the relevant documents. We acknowledge that a more advanced approach, fully utilizing multiple relevant documents as seed documents can further improve this simulation performance. We leave this as our future work.

3.5 Summary

In this chapter, we propose seed-driven document ranking for systematic reviews. It leverages domain-specific characteristics in document representation and ranking. The superior performances of the proposed approach and the combined approach with word embedding show the effectiveness of incorporating domain-specific characteristics in screening prioritization. Furthermore, our approach has multiple advantages. Employing clinical terms reduces the total number of terms in documents and it leads to the decreased computational cost. A small number of clinical terms also helps speed up manual screening process executed by SR experts. Besides, the proposed weighting scheme can effectively promote terms that are in common in many relevant documents.

Chapter 4

Ranking by Document Matching

4.1 Introduction

Screening prioritization (*i.e.*, document ranking) is a task of ranking candidate documents according to the likelihood of satisfying relevance conditions. Relevance conditions are often defined using PICO framework (Patient, Intervention, Comparator, and Outcome). Detailed conditions are specified under each element. For example, age, gender and medical history of patients are defined in the P element.

In this chapter, we investigate a document matching (*i.e.*, similarity) approach for seed-driven document ranking. While irrelevant documents satisfy only partial or zero conditions, relevant documents satisfy all relevance conditions, so that they share the higher document similarities than irrelevant documents (detailed discussion in Chapter 3.2). Given a seed document and candidate documents, we aim to rank relevant documents higher than irrelevant documents using their matching scores to the seed.

To become a relevant document, a candidate document must be matched to a seed document in terms of PICO elements. Let us assume that a seed document and a candidate document both contain a term, *insulin*. When *insulin* is used as Outcome (O) element in the seed, it must be used as O element in the candidate document as well. If the term is used as the other element (*e.g.*, Intervention (I) element), the candidate document becomes an irrelevant document. Simple occurrence-based term matching is unable to achieve PICO-based matching.

Abstract texts in medical literature implies a sequential structure: background/purpose, method, results, and conclusion [66]. To develop deeper understanding of PICO elements, we study the positions of PICO elements in abstract texts. From the analysis using a large-scale

PICO dataset, we show PICO elements appear with positional patterns. For example, O element appears typically in the second half of abstract where result and conclusion are often discussed. Hence, when the term *insulin* is used as O element, a seed and a candidate document both likely have the term in the end of abstracts.

We propose a position-based term matching for seed-driven document ranking. It matches terms from two documents based on their similar positions. Specifically, for each term in a seed document, we match it to terms in a candidate documents which have the similar position to the seed term’s position. In this way, the proposed model aims to achieve PICO-based term matching, without explicitly identifying PICO elements in documents. Moreover, a recent study reports that it is challenging to accurately identify fine-grained PICO elements (*i.e.*, spans) in medical abstracts as they are written in a descriptive manner [67].

Furthermore, we propose to adopt a two-way matching scheme in the position-based term matching. As a final matching score, it aggregates two position-based matching scores: the matching from a seed to a candidate and the matching from a candidate to a seed. When a candidate document is a relevant document, it could be used as a seed document. In other words, given a seed document and a candidate document, they can switch the roles. And the matching score in the switched role is supposed to be as high as the matching score in the original roles. Therefore, leveraging both of the matching scores promotes two documents that are exactly mirroring each other (*i.e.*, exact match), leading to find relevant documents more effectively. We call the proposed matching model **Mirror Matching** (M-Match).

Mirror Matching is a document similarity measure which is inspired by the asymmetric scoring mechanism in retrieval models (*i.e.*, BM25) such as from a query to a document. Later, we discuss the connections and differences between M-Match and retrieval models as well as existing document similarity measures. We also verify the effectiveness of the proposed model against them. In recent few years, neural IR models have gained significant attention [68–70]. They can be used for seed-driven document ranking as a supervised document matching approach. We evaluate several neural IR models compared to M-Match and discuss their potential in seed-driven document ranking.

4.2 Proposed Approach

We study the task of seed-driven document ranking in the context of document matching. We compute matching scores (*i.e.*, similarity) between a seed document and candidate documents and rank the candidate documents by the matching scores. Existing pairwise document similarity measures are alternatives to compute matching scores.

4.2.1 Overview

Mirror Matching (M-Match) is a simple two-way position-based matching measure for a pair of documents (*i.e.*, a seed document and a candidate document). It examines (i) a candidate document whether it is the best matching of a seed document (the matching from a seed to a candidate) and (ii) a seed document whether it is the best matching of a candidate document (the matching from a candidate to a seed). These two examinations add up as the final matching score of the document pair. Each matching is asymmetric and computed by position-based term matching. Formally, given a seed document (query) Q and a candidate document D , a matching score is as follows:

$$\text{M-Match}(Q, D) = sc_{Q \rightarrow D} + sc_{D \rightarrow Q} \quad (4.1)$$

where $sc_{Q \rightarrow D}$ is a candidate document score computed by matching terms in Q to terms in D and $sc_{D \rightarrow Q}$ is a query document score computed by matching terms in D to terms in Q . The two matching scores contribute to the final matching score equally¹. Figure 4.1 presents the overview of Mirror Matching and illustrates position-based term matching in the computation of two matching scores. Before explaining the details of position-based term matching, we present the motivation of it and discuss an analysis on the positions of PICO elements.

4.2.2 Analysis: Distributions of PICO Elements

Medical scientific abstracts are carefully written by authors to provide the overview of a full scientific literature. They summarize major aspects of literature in a prescribed sequence that includes *background/purpose*, *method*, *results*, and *conclusion* [66]. We set a hypothesis that PICO words appear in positional patterns in medical scientific abstracts as the sequential components are implied in abstracts.

¹We experiment with alternative combinations, multiplication of two scores, and observe similar performance.

Fig. 4.1: Mirror Matching uses two-way matchings mechanism: from a query document to a candidate document and from a candidate document to a query document. Each one-way matching is based on position-based term matching.

The interests in the positions of PICO words are not new [71, 72]. Existing studies have detected words in PICO elements by string matching with PICO keywords and reported coarse-grained (*e.g.*, chunk-level or sentence-level) distributions of PICO elements. Different from them, we analyze fine-grained (*i.e.*, words) distributions of PICO elements. Moreover, in existing studies a string exact matching is often suffered from a vocabulary mismatch problem between PICO keywords and actual PICO words used in documents. Here we use the manually labeled PICO words, so that we include comprehensive paraphrases and synonyms of PICO in the analysis.

EBM-NLP dataset consists of large-scale PICO span annotations on 5,000 randomized controlled trials PubMed documents (title and abstract only) [73]. In this dataset, C element is combined into I element, so that every single-word token on the documents has one of the four labels: Patient, Intervention, Outcome, or None. EBM-NLP provides the annotations by individual annotators and the annotations combining all individual's annotations. We use the combined ones.

Figure 4.2 presents the distributions of P, I, and O elements. Each row is a document consisting of a sequence of tokens in which title and abstract text are concatenated. Documents

Fig. 4.2: Distributions of Patient, Intervention, and Outcome elements in medical documents (title and abstract). Each row indicates a document. Within a row dark blocks are terms labeled with a given element.

are sorted by the total number of tokens in descending order.² Different colors indicate tokens with/without a corresponding PICO element label. As documents contain the different number of tokens, token positions within a row represent relative positions of tokens in that document.

P element often appears at the beginning of documents where a title, *background* and/or *purpose* are likely to be discussed. After the beginning part in a document, its occurrence is very sporadic. Next, I element tends to appear in the very beginning of documents similar to P, but its occurrence is more spread into the first half of documents compared to P element. Besides, I element keeps appearing in the second half of document time to time. Another characteristic of I element is that it is mostly short keywords, while P element tends to be phrases. Observe that the difference in lengths of labeled consecutive tokens in P and I elements. Lastly, O element

²About 250 documents in EBM-NLP dataset have only a title, so that the number of tokens are much fewer than the rest and hard to show any patterns (see the bottom of the figure).

Algorithm 1 Mirror Matching Overview

Input: Query document $Q = \langle q_1, q_2, \dots, q_{|Q|} \rangle$,
Candidate document $D = \langle t_1, t_2, \dots, t_{|D|} \rangle$

Output: M-Match(Q, D) score between Q and D ,

```

1: initialize  $sc_{Q \rightarrow D}$  and  $sc_{D \rightarrow Q}$  to zero
2: for  $i \leftarrow 1$  to  $|Q|$  do                                 $\triangleright$  start one-way matching from  $Q$  to  $D$ 
3:   initialize  $S_{temp}$  an empty list/array
4:   for each  $j \in \text{matching\_range}(\lambda, i, |Q|, |D|)$  do
5:      $S_{temp}.append(sim(q_i, t_j))$ 
6:   end for
7:    $sc_{Q \rightarrow D} = sc_{Q \rightarrow D} + max(S_{temp})$ 
8: end for
9:  $sc_{Q \rightarrow D} = sc_{Q \rightarrow D} / |Q|$ 
10: for  $j \leftarrow 1$  to  $|D|$  do                                 $\triangleright$  start one-way matching from  $D$  to  $Q$ 
11:   initialize  $S_{temp}$  an empty list/array
12:   for each  $i \in \text{matching\_range}(\lambda, j, |D|, |Q|)$  do
13:      $S_{temp}.append(sim(q_i, t_j))$ 
14:   end for
15:    $sc_{D \rightarrow Q} = sc_{D \rightarrow Q} + max(S_{temp})$ 
16: end for
17:  $sc_{D \rightarrow Q} = sc_{D \rightarrow Q} / |D|$ 
18:  $M - Match(Q, D) = sc_{Q \rightarrow D} + sc_{D \rightarrow Q}$                  $\triangleright$  final matching score
19: return  $M - Match(Q, D)$ 

```

is mainly found in the second half of documents where *results* and *conclusion* are in abstract texts.

This analysis shows that each PICO element appears with its unique distributional pattern in abstract texts. It describes that when terms from two abstracts have similar positions, they are likely to be used as the same element. Hence, simple term matching based on their positions can have the similar impact of PICO-based term matching.

4.2.3 Mirror Matching

Algorithm 1 presents the pseudo code of Mirror Matching to compute a matching score of two documents. Let Q and D denote a seed document (query) and a candidate document, respectively. In this discussion we call a seed document query to reduce confusion with a candidate document. A query Q consists of a sequence of query terms, $\langle q_1, q_2, \dots, q_{|Q|} \rangle$, and a candidate

document D consists of a sequence of document terms $\langle t_1, t_2, \dots, t_{|D|} \rangle$. The subscription of each term indicates its position in a given document. As presented in Equation 4.1, a matching score between Q and D is the sum of two one-way scores: $sc_{Q \rightarrow D}$ and $sc_{D \rightarrow Q}$. As the same concept of position-based term matching is applied to the two scores, we explain $sc_{Q \rightarrow D}$ as a representative (line 2-9 in Algorithm 1).

The score $sc_{Q \rightarrow D}$ is the average of similarities with respect to query terms. For each term q_i , we find t_j which has a similar position to the position of q_i in Q . We then get the best semantic similarity between q_i and t_j .

To find t_j , we set a position range in D for given q_i using a window size. We use a relative window size proportional to the number of words in a document, because of the different number of words in documents. We define a hyperparameter λ that determines a relative window size. For example, if $\lambda = 0.35$, then 35% of the document length is set as a window size. Therefore, the range for similar positions in D given q_i is calculated as $(|D| \cdot i/|Q|) \pm \lambda \cdot |D|$. $(|D| \cdot i/|Q|)$ indicates q_i 's position transformed in D , and different signs denote preceding and following directions from the transformed position. The range includes the beginning and ending numbers of the interval. In Algorithm 1, `matching_range` is the procedure of finding the range for similar positions.

Next, t_j within the similar position range is matched to q_i using a cosine similarity of their word embeddings. Among the cosine similarity values a maximum value is chosen (1-max pooling) as the most informative matching signal. Word embeddings allow semantic matching, not limited to surface forms of terms. Finally $sc_{Q \rightarrow D}$ is the average of similarities obtained from each query term $q_i \in Q$. In our evaluation, we set λ to 0.35.

The computational cost of $sc_{Q \rightarrow D}$ is governed by the number of terms in Q , the number of terms in D , and λ that determines a window size. The estimation of $sc_{Q \rightarrow D}$ has the time complexity of $O(nk)$ where n is the number of terms in Q (i.e., $|Q|$) and k is the number of terms within matching positions of D .

Likewise, $sc_{D \rightarrow Q}$ is the average of similarities computed for each term in D . It is worth noting that the two one-way scores, $sc_{Q \rightarrow D}$ and $sc_{D \rightarrow Q}$, are *asymmetric* as they are computed with respect to q_i and t_j , respectively.

4.2.4 Model Discussion

We discuss the connections and differences of M-Match with ad-hoc retrieval models and document similarity measures.

Although Mirror Matching bears a similarity to ad-hoc retrieval models, there is a major difference: two-way matching. In ad-hoc retrieval, typically a query is short and a document is long. Since a given query is a primary source for relevance, retrieval models (*e.g.*, BM25) estimate document scores with respect to query terms (one-way scoring) [59]. In this problem setting, both a query and a candidate document are documents. M-Match uses the one-way scoring in both directions. The bi-direction setting demonstrates that either of query and candidate documents can be used as a query.

A difference between M-Match and common document similarity measures (*e.g.*, cosine similarity between two document representations) is the consideration of directions in a matching score. While common document similarity measures do not take into account directions, M-Match employs asymmetric matchings by directions in a matching score. Moreover, the concept of direction in document matching helps find matching document pairs. We use an example to further illustrate this point.

Let us assume that d_1 and d_2 are similar, but d_2 includes more detailed content while fully covering all content of d_1 . When finding documents matching to d_1 , d_2 can be a good matching for d_1 , as it covers all content presented in d_1 . On the other hand, when finding documents matching to d_2 , d_1 may not be a good matching for d_2 , as d_1 lacks the detailed information in d_2 . In this case, the two asymmetric scores in M-Match will give a lower score to the document pair, especially via the score from d_2 to d_1 .

4.3 Evaluation Settings

In this section, we present evaluation settings including dataset, baselines, and evaluation metrics.

4.3.1 Dataset

We conduct our evaluation on CLEF 2019 eHealth Task 2 (CLEF19) dataset. This dataset contains 79 diagnostic test accuracy (DTA) SRs, 20 Intervention, 2 Qualitative, and 1 Prognosis

SRs in CLEF eHealth 2019 Technologically Assisted Reviews (TAR) [74]. CLEF19 is currently the most comprehensive dataset for screening prioritization. We use 79 DTA SRs in the evaluation. DTA SRs are considered as a difficult type of SRs to automatically identify relevant documents because of their broad range of topics. The other types of SRs in CLEF19 are much fewer than DTA SRs, and are not included³. Training/testing data splits are defined in CLEF19 but we use all SRs since our model does not need training.

CLEF19 provides document identifiers (PMIDs) of candidate documents with their two relevance labels (from abstract screening and full document screening). We use the label from full document screening in the evaluation because these labels indicate the final relevant and non-relevant documents in SRs. From a public MEDLINE 2019 corpus, we obtain document contents (title, abstract, and MeSH keywords) using document identifiers. For the detailed statistics of the dataset, we refer to [74].

Any relevant document can be the known document in a SR. We use every relevant document as a query and rank the rest candidate documents. In CLEF19, a few SRs have relevant documents fewer than 2. In this setting, as one relevant document is used as a query, they are not included in the evaluation.

Pre-processing. In medical literature, short forms (*e.g.*, abbreviations or acronyms) of long medical terminologies are often used for effective communication. However, there is no standard rule for constructing short forms. Hence, a long medical term is written in different short forms across different documents. To increase consistency within and between documents, we identify short forms and their long forms using a simple regular expression, and transform the short forms into the long form. We also detect three kinds of numeric expressions and normalize them into INT, FLOAT, and PERCENT. Lastly, stop words are removed.

Word Embeddings. We use word embeddings locally trained on candidate documents per SRs. As candidate documents are retrieved by Boolean keyword queries derived from relevance conditions, they are a collection of similar documents. Multiple studies report that large corpora do not necessarily produce good word embeddings in biomedical domain, compared to a smaller but topic-specific collection [75, 76]. A similar finding has also been reported in

³Note that for DTA SRs, the original CLEF19 consists of 80 SRs, which are 72 SRs from CLEF18 and 8 new SRs. One new SR (id: CD011686) is already in CLEF18 and it has different relevance labels in CLEF18 and in CLEF19. We use the labels from CLEF2018.

a general domain problem [77]. Our empirical results also show that local word embeddings are more effective than global word embeddings trained on a corpus (we report details in Section 4.4.2). We use skip-gram with negative sampling implemented in gensim. The dimension of word embeddings and a window size are set to 300 and 7, respectively and the minimum word occurrence is set to 5. The default settings are used unless it is specified.

4.3.2 Baselines

The evaluation is done with ad-hoc retrieval models and document similarity measures. Candidate documents are prioritized by retrieval scores or similarity scores.

Retrieval models. Two classical retrieval models (BM25 and query likelihood model) and a retrieval model specifically designed for screening prioritization (SDR-BOC) are evaluated. **BM25** is a simple and strong retrieval model and it has continuously proved its effectiveness [78]. We use BM25 implementation in gensim. Query likelihood model (**QL**) is a language model approach and we experiment QL with JM smoothing and a smoothing parameter is empirically set to 0.2. **SDR-BOC** is a retrieval model proposed for screening prioritization [79]. It is QL combined with an additional term weighting function. As suggested in the original paper, we represent documents in bag-of-concepts (BOC) after extracting medical concepts.

Document similarity measures. We examine document similarities using lexical and semantic matching signals. The first measure is a cosine similarity with different document representations: TF-IDF, AvgEm-W, AvgEm-C, and ParaVec. **TF-IDF** is a bag-of-words document representation using the classic TF-IDF weighting scheme. **AvgEm-W** is the average of word embeddings of all words contained in a document. Likewise, **AvgEm-C** is the average of concept embeddings of all concepts contained in a document. A concept embedding is the average of word embeddings of all words contained in a medical concept. Paragraph vector (**ParaVec**) is a distributional document representation proposed by [80]. We use PV-DBOW in gensim. In addition, we tested different similarity functions such as Euclidean distance in pilot experiments, but cosine similarity showed the highest performance.

We also evaluate document similarity measures proposed by existing studies. **TF-inner** [54] computes an inner product of two bag-of-words document representations using *tf* weighting. **OK** [81] is motivated by BM25 and computes a document similarity using BM25 weights.

Word mover’s distance (**WMD**) [82] estimates a distance between two documents defined as a cost when all words in one document move to words in the other document. WMD is a symmetric distance metric and its negative value is considered as a similarity. [83] showed that WMD can be applied to ad-hoc retrieval.

AvgEm-W, AvgEm-C, and WMD use the same word embeddings as M-Match. In SDR-BOC and AvgEm-C, QuickUMLS [58] is used to extract medical concepts.

4.3.3 Evaluation Metrics

We use evaluation metrics commonly adopted in screening prioritization task: average precision, precision at k , recall at k , and work saved over sampling at recall 100%. Average precision (AP) is a metric combining a recall component to precision-focused metric by averaging all values of precision at each relevant document. Precision at k ($\text{Pr}@k$) is the ratio of relevant documents out of top k documents, and recall at k ($\text{Re}@k$) is the ratio of relevant documents in top k documents out of the total number of relevant documents k is set to 10, 20, and 30.

Work saved over sampling at recall 100% (WSS100) [14] measures the ratio of documents that are ranked lower than the last relevant document out of the total candidate documents. WSS100 estimates the workload to be reduced in the manual screening with the assumption that after the last relevant document is found the rest unlabeled documents are not screened.

For a given SR, every relevant document can be selected by users as a seed document to start the screening process. To simulate the scenario, we conduct experiments when each relevant document is used as a seed document. We compute the average over all the values obtained for a given SR. Finally, we report the mean over all SRs.

4.4 Results and Discussion

In this section, we aim to answer four research questions.

RQ1. Is M-Match more effective than the alternative models for prioritizing documents using a query document?

RQ2. How does M-Match perform with multiple relevant documents in the manual screening process?

RQ3. To what extent do the components (*e.g.*, two-way matching, position-based matching) in M-Match contribute to its performance?

RQ4. Regarding recent success of neural IR models, how do neural models perform in this task?

4.4.1 Overall Results

First, to answer **RQ1** we present the overall results of the proposed model and the baselines in Table 4.1. We also report the rank of the proposed model among all and a performance gap between the proposed model and the best baseline model in the last row. A positive/negative sign indicates win/loss of the proposed model.

M-Match outperforms all baseline models in terms of AP and $\text{Pr}@k$. Two-way matching effectively ranks relevant documents higher than non-relevant documents. WMD is the second best model in AP and $\text{Pr}@k$. While both M-Match and WMD use semantic term matching, a key difference is that M-Match matches terms using positional information to match terms in same PICO elements. Regardless positions, WMD matches terms by minimizing the cases of matching same terms multiple times and it also leads to the higher computational cost than M-Match.

Two retrieval models, BM25 and SDR-BOC, show the highest $\text{Re}@k$ values and AvgEm-C has the highest WSS100 value. Indicated in the last row of Table 4.1, M-Match follows behind these models with a marginal gap. The high performance of SDR-BOC and AvgEm-C supports that bag-of-medical concepts (BOC) is an effective representation for medical documents [79, 84, 85]. Bag-of-words (BOW) and bag-of-concepts (BOC) represent documents with their own advantages and disadvantages. BOW takes full advantage of all words in a document, despite sparsity. On the other hand, BOC relies on essential medical terminologies with risk of losing contextual information in non-medical words.

Different aspects in the proposed model are shown by the evaluation metrics. M-Match provides the overall high quality of ranking (AP) and in particular it is effective to put many relevant documents in top positions ($\text{Pr}@k$). On the other hand, the proposed model is less effective to rank *all* relevant documents at top positions ($\text{Re}@k$) and the rank of the last relevant document is lower in M-Match than in a few baseline models (WSS100).

Table 4.1: Comparison between Mirror Matching and the baselines. The best model is in bold and the second best is underlined in each metric. * denotes statistically significant performance of the best model compared to a given model (p-value < 0.01).

Models	AP	Pr@10	Pr@20	Pr@30	Re@10	Re@20	Re@30	WSS100
BM25	0.169	0.209	0.173	0.151	0.178	0.246	0.299	0.437*
QL	0.155	0.189	0.158	0.139	0.170	0.235	0.279	0.466*
SDR-BOC	0.172	0.209	0.174	0.152	0.176	0.251	0.303	0.541*
TF-IDF	0.180	0.208	0.173	0.153	0.127	0.192	0.243	0.690
AvgEm-W	0.176	0.201	0.167	0.147	0.177	0.242	0.290	0.530*
AvgEm-C	0.167	0.190	0.162	0.145	0.110*	0.175	0.228	0.692
ParaVec	0.139*	0.164*	0.141*	0.126*	0.093*	0.153*	0.198*	0.598
TF-inner	0.119*	0.136*	0.115*	0.102*	0.112	0.172	0.214	0.477*
OK	0.113*	0.113*	0.107*	0.100*	0.066*	0.125*	0.173*	0.650
WMD	0.203	0.232	0.194	0.170	0.141	0.225	0.272	0.592
M-Match	0.220	0.247	0.207	0.183	0.149	0.234	0.293	0.687
Rank (Δ)	1 (+0.017)	1 (+0.015)	1 (+0.013)	1 (+0.013)	5 (-0.029)	5 (-0.017)	3 (-0.010)	3 (-0.005)

We believe the limitation of M-Match is closely related to the coverage of relevance conditions in SRs (*i.e.*, topic coverage). When a SR covers broad topic, relevant documents can satisfy relevance conditions in multiple ways, so that not all relevant documents share high similarities. Since the proposed model is a similarity-based ranking approach, M-Match can be less effective to find the least similar relevant document (*i.e.*, the last relevant document) in SRs with broad coverage.

To verify the understanding, we estimate the coverage of SRs and analyze WSS100 of individual SRs against it. To estimate the SR coverage, we use a minimum value in all pairwise similarities between relevant documents within a SR. If a SR has broad coverage, any of relevant document pairs can have a low similarity. Then, we plot WSS100 of SRs against the coverage in Figure 4.3. Recall that WSS100 represents the portion of documents ranked after the last relevant document. The greater WSS100, the higher ranking of the last relevant document. As shown in Figure 4.3, WSS100 of individual SRs widely varies. While some SRs has WSS100 close to 1.0, some SRs only manage to 0.1. More importantly, as the SR coverage is narrow (broad), WSS100 increases (decreases). Pearson’s correlation between WSS100 and the SR coverage is $\rho = 0.83$, $p\text{-value} < 0.001$. That is, M-Match has high WSS100 on SRs with narrow topic.

We claim that the strong AP and $\text{Pr}@k$ of M-Match are highly useful in the beginning of manual screening process. The high AP and $\text{Pr}@k$ describe more relevant documents are to be found when SR experts screen a handful of top-ranked (*e.g.*, top 10) documents. When more relevant documents are identified early, they can further improve the rankings of remaining unlabeled documents.

Updates with Multiple Documents. We answer **RQ2** by simulating the manual screening process. When top 20 documents are screened, the rankings of the remaining unlabeled documents are updated by new relevant documents which are found in the top 20 documents. We iterate this process three times.

After each iteration, multiple relevant documents can be found. As the proposed model computes document-to-document matching scores, each new relevant document produces a ranked list. We combine multiple ranked lists using rankings positions since they have different ranges of scores. For example, for a candidate document, we get its ranking positions in multiple ranked lists and then the averaged ranking position is set to its updated ranking position. The

Fig. 4.3: WSS100 of individual SRs by M-Match against the estimated coverage of SRs

Table 4.2: Simulating the manual screening process. In each round, top 20 documents are labeled and the rankings of the remaining documents are updated with relevant documents.

	AP			WSS100		
round	1	2	3	1	2	3
TF-IDF	0.213	0.241	0.252	0.745	0.767	0.777
AvgEm-C	0.178	0.204	0.239	0.704	0.721	0.731
M-Match	0.259	0.262	0.279	0.719	0.728	0.732

rank-wise combination allows multiple ranked lists carry equal weights in the updated ranked list. We evaluate the updated list by AP and WSS100. We compare the proposed model against TF-IDF and AvgEm-C which showed the higher WSS100 than the proposed model in Table 4.1.

Table 4.2 presents the evaluation results over three iterations. As more relevant documents are used as a query document, all three models keep improving the rankings. M-Match shows the highest AP in all three rounds and the second highest WSS100, following TF-IDF. The results are consistent with the results shown in Table 4.1 and they affirm the strength and limitation of M-Match discussed in the prior evaluation.

Discussion. We have extensively evaluated the effectiveness of M-Match: it is effective to find many relevant documents at the top, but less effective to find the last relevant document (*i.e.*, the document the least similar to a query document). It is useful to adopt M-Match in the beginning

Table 4.3: AP of degraded models without key components of Mirror Matching.

Model	AP
M-Match w/o position	0.216
M-Match w/o two-way	0.204
M-Match w/o {two-way, position}	0.196
M-Match	0.220

of the manual screening, as it finds more relevant documents within top positions. After screening the top ranked documents by M-Match, we suggest to combine/switch M-Match to another ranking model (*e.g.*, TF-IDF) for the better ranking of the last relevant document. Alternatively, supervised approaches can be adopted since more relevant and non-relevant documents are now available.

4.4.2 Analysis on M-Match

Ablation Study. To answer **RQ3**, we investigate the effect of major components in the proposed model: two-way matching mechanism and position-based matching. Three degraded models are examined in an ablation test. **M-Match w/o position** removes positional information in finding the maximum term matching similarity. In other words, a max-pooling is applied to the entire row/column in Figure 4.1. **M-Match w/o two-way** removes two-way matching mechanism. A final matching score is $sc_{Q \rightarrow D}$, without $sc_{D \rightarrow Q}$. **M-Match w/o {two-way, position}** removes both components in Mirror Matching. A matching score is $sc_{Q \rightarrow D}$ and also it does not use positional information in max pooling.

Table 5.5 presents performances of the degraded models compared to M-Match. We observe that: (1) *M-Match w/o two-way* performs worse than *M-Match w/o position*, showing that incorporating two one-way scores has the bigger impact than position-based matching. (2) Performance of *M-Match w/o {two-way, position}* has a significant drop from M-Match. AP of *M-Match w/o {two-way, position}* is even lower than that of WMD reported in Table 4.1. These two observations suggest that both components contribute to the proposed model.

Analysis on Two-Way Matching. The key claim of two-way matching is to employ a score from candidate document’s perspective ($sc_{D \rightarrow Q}$) as well as the commonly used score from query’s

Fig. 4.4: Impact of different window size (λ) in position-based term matching.

perspective ($sc_{Q \rightarrow D}$). We study how $sc_{D \rightarrow Q}$ makes a change in matching scores of relevant documents and eventually improves the rankings.

First we find relevant and non-relevant document pairs that $sc_{Q \rightarrow D}$ of a non-relevant document has the smallest difference to $sc_{Q \rightarrow D}$ of a relevant document. That is, the two documents in pairs are ranked next to each other without two-way matching (*i.e.*, $sc_{Q \rightarrow D}$ alone is a matching score). We then compare their $sc_{D \rightarrow Q}$. We calculate the portion of pairs such that $sc_{D \rightarrow Q}$ of the relevant document is greater than that of the non-relevant document within ranked candidate documents by a query document. We then report the average portion over all queries from all SRs in CLEF19.⁴ As a result, 63.5% of the pairs have greater $sc_{D \rightarrow Q}$ of the relevant document than $sc_{D \rightarrow Q}$ of the non-relevant document. The result suggests that scores from candidate document's perspective $sc_{D \rightarrow Q}$ tend to promote relevant documents.

We note that the above analysis does not confirm the rise of relevant documents in a ranked document list as the rest of candidate documents are also involved in the total ranking. Nevertheless, this close analysis helps better understand the two-way matching in finding relevant documents.

Impact of Window Size in Position-based Matching. Positional information of terms is incorporated with a window size determined by λ . Here we study the effect of different window

⁴We also calculate the average portion via two levels: average over a SR and then average over all SRs. The resulting portion is consistent.

sizes. We experiment different values of λ from 0.05 to 0.35 with a 0.05 increment. We plot AP of M-Match in Figure 4.4. Recall that a window size indicates one side range (either preceding or following) from a given term. When $\lambda = 0.35$, for a term in the middle of document, it has a matching area of 70% of a document length. Additionally, we display AP with no positional information *i.e.*, *M-Match w/o position*.

In Figure 4.4, AP gradually increases along the increase of window size. The highest performance is observed when λ is 0.35. It is worth noting that when the model pinpoints terms with a very small window ($\lambda = 0.05$), the performance deteriorates significantly. We believe the large drop originates from the different granularity of the rhetorical structure in documents. For example, some abstracts have much longer *results* part than others. Lastly, while using positional information of terms improves performance, the performance gain is relatively small compared to the case without using it. Referring to Figure 4.2, the positional patterns of PICO elements can be seen in a collective manner among several thousands documents and the pattern in a pair of two documents may not be strong enough. Besides, using positional information is a soft way of incorporating PICO elements in matching as the PICO spans are not explicitly pre-detected [86].

Impact of Word Embeddings. The quality of word embeddings affects the performance of Mirror Matching, WMD, as well as cosine similarity over AvgEm-W and AvgEm-C. We now examine the impact of different word embeddings on AvgEm-W, WMD, and the proposed model. We evaluate two word embeddings pre-trained on large corpora: medical domain word2vec⁵ (Global word2vec) and GloVe⁶ and also present the earlier evaluation with local word embeddings trained on candidate documents (Local word2vec). As shown in Figure 4.5, in comparison of different word embeddings, locally trained word embeddings consistently show the best performance in all three models compared to the other two embeddings. Furthermore, M-Match consistently outperforms the two baselines in all different kinds of word embeddings.

We also experiment contextualized word embeddings using pre-trained BioBERT (version 1.0 PubMed 200k, PMC 270k) [87]. However, while M-Match outperforms the other models, the overall performance in all three models is much lower than distributional word representations (*e.g.*, word2vec, GloVe). To the best of our knowledge, it is unknown how to effectively

⁵ <http://bio.nplab.org/>

⁶ <https://nlp.stanford.edu/projects/glove/>

Fig. 4.5: Impact of different word embeddings on cosine similarity with AvgEm-W, WMD, and M-Match.

use BERT in a word-level intrinsic evaluation setting, instead of extrinsic evaluation settings (*i.e.*, downstream tasks). We leave this as our future work.

4.4.3 Neural Models for Screening Prioritization

We test a few neural IR models and answer the last research question **RQ4**. Similar to M-Match, neural models also find a pair of texts that match each other, but we first need to train them.

Train and Test Settings. Neural IR models typically require large amounts of training data. It is difficult to train neural models specialized for individual SRs because of too few relevant documents in a SR. We use train and test splits in CLEF19, consisting of 72 and 7 SRs, respectively⁷. For a development (dev) dataset, we randomly sample 7 SRs in the training 72 SRs. Because of the random sampling, we examine two sets of train and dev datasets with different SRs. Both evaluation results are reported.

We generate balanced positive and negative instances (document pairs) within each train/dev SR: {relevant - relevant} pairs for positive instances and {relevant - non-relevant} for negative instances. In total, the train and dev datasets contain 145,104 instances. In the test dataset, a query document is paired with every candidate document.

We use MatchZoo (version 2.1.0) toolkit [88]. MatchZoo toolkit provides a collection of representative neural IR models. We evaluate two simple neural baseline models (Dense, BiL-

⁷A development dataset is not defined in CLEF19

STM) and another two neural IR models (Duet, MV-LSTM) which share similar matching (learning) mechanisms with M-Match.

- **Dense** is a naive baseline. Word tokens are represented as one-hot vectors and they are concatenated and encoded using two layers of feedforward neural networks (FNNs).
- **BiLSTM** is bidirectional LSTMs (BiLSTM) in a siamese architecture. After embedding lookup and encoding by BiLSTM, the last hidden states from documents in a pair are concatenated. Then, two layers of FNNs are added.
- **Duet** [89] uses semantic matching as well as syntactic matching in character n -graph interactions between two documents.
- **MV-LSTM** [90] is based on a learning goal similar to M-Match. It uses term-term semantic interactions encoded by BiLSTM with positional information.

MatchZoo provides the implementations of Dense, Duet, and MV-LSTM. We implement BiLSTM using MatchZoo platform. We use a softmax activation function and cross entropy as the loss function. All models truncate a document to its first 300 tokens. We set epoch to 30 and select the best model on the dev dataset in terms of AP and report its performance on the test dataset. When training the neural models we use global word embeddings⁵ instead of local embeddings because local embeddings trained for each SR do not share embedding spaces.

Results and Discussion. Figure 4.6 presents two sets of AP on the test dataset when the train and dev datasets are comprised of different SRs. We also show the performance of M-Match on the test dataset for reference. Here, M-Match uses the same global word embeddings as the neural models. Among all four models, MV-LSTM achieves the highest performance in both evaluation sets, while it is still lower than that of Mirror Matching.

One concern we anticipate is that neural models in ad-hoc retrieval are often adopted as a re-ranker. First, documents are ranked by a simple model like BM25 and a neural model then re-ranks the top 1,000 documents, for example. We claim that the setting in screening prioritization is in fact similar to the re-ranking setup. Candidate documents in SRs are filtered documents from digital libraries by Boolean retrieval.

Different SRs in train/dev datasets significantly influence the performance of models. One set of train/dev datasets consistently achieves the higher performance than the other set of

Fig. 4.6: Performance of neural models on test dataset in CLEF19. Two sets of AP are presented when the train and dev datasets are comprised of different SRs.

train/dev datasets in all neural models. Moreover, the performance gap between the two is not negligible.

We believe that different relevance conditions (topics) in SRs are a major contributing factor to the significant performance difference. Depending on SRs in train and dev datasets, the patterns learned and validated by one set of train and dev datasets can differ from those by the other set of train and dev datasets. Furthermore, they can be different from patterns required for correct predictions on test SRs.

Another contributing factor can be the size of dataset. Due to the nature of deep neural models, a large amount of data is prerequisite for high performance. It is possible that CLEF19 dataset might not be sufficient enough to encode different patterns imposed by SRs. More data may mitigate this issue, but it requires further investigations.

Summary. We examine neural IR models for screening prioritization. The empirical results show large performance changes by different data splits. We discuss a few factors that might be related. To this end, it is hard to draw a conclusion on performance of neural models and how to obtain reliable results remains an open question.

4.5 Summary

We investigate seed-driven document ranking in the context of document matching in which candidate documents are ranked by their matching scores to the query document. We propose

two-way position-based matching named Mirror Matching (M-Match). Mirror Matching estimates a matching score of a document pair by adding up two one-way scores: (1) a candidate document score from a query document, and (2) a query document score from a candidate document. Each score is position-based term matching motivated by the positional patterns of PICO elements in abstract texts. We extensively evaluate M-Match not only for ranking documents with a single query document, but also for updating the rankings with multiple relevant documents. Moreover, by investigating the performance on individual SRs, we reveal that depending on the coverage of SRs, the proposed model shows the different levels of effectiveness. For future work, we will further examine the coverage of SRs to see whether it is possible to identify SRs that can take advantage of the strengths of M-Match.

Chapter 5

Ranking Performance Estimation

5.1 Introduction

In seed-driven document ranking, multiple ranking models have been developed. They have showed the promising results on SRs in benchmark datasets and their performance further improves when more relevant documents become available during manual screening process.

In screening prioritization it is important that a ranking model maintains high local performance (the performance on single SR) as well as high global performance (the performance on multiple SRs in benchmark datasets). SR experts experience the local performance in manual screening process, instead of the global performance of a ranking model. Despite the importance of local performance, it has been less explored.

Furthermore, while providing the rankings of documents helps achieve efficient workflow, SR experts still have little information to work with during the screening process. While SR experts manually screen prioritized documents for several months, they do not know of how good the given rankings are or whereabouts the last relevant document is. Understanding on the local performance of a ranking model helps resolve this challenge.

In this chapter, we estimate the ranking qualities of individual SRs. Our goal is to further assist SR experts by providing not only ranked documents but also the ranking quality on a given SR. Information on the ranking quality provides useful indicators for SR experts in practice. They can anticipate the time to be taken finding *all* relevant documents based on the estimated ranking quality. And it allows them to maximize their time and resources accordingly.

Ranking qualities of individual SRs are affected by the choice of ranking models. We first observe the changes of ranking qualities in SRs by three ranking models. From the analysis, we

find that the changes in each SR caused by different ranking models are very small. Moreover, when a set of SRs are compared in terms of the ranking qualities given one ranking model, the ranking quality gaps among SRs are large in all three models. That is, some SRs consistently have higher ranking qualities than other SRs regardless of ranking models. These findings imply that there might be defining characteristics in SRs that underlie their ranking qualities.

We investigate the *topic broadness* in SR as a factor which affects its ranking quality. The rationale is that when a SR has broad topic, relevant documents are diverse, so that it is difficult to find all relevant documents. Thus, the SR is likely to have low ranking quality (or high ranking difficulty¹). To this end, we propose a measure that quantify the topic broadness of SR and estimate its ranking quality using this measure. For the rest of this work, ‘ranking quality’ and ‘ranking difficulty’ are used interchangeably, yet in an inverse relationship. The high SR ranking quality means the low SR ranking difficulty.

To verify the effectiveness of the proposed measure, we conduct extensive evaluations on multiple sets of ranking qualities by different models. We analyze ranking and linear correlations between the topic broadness quantified by the proposed measure and the actual ranking qualities by average precision (AP). Our empirical results show that the topic broadness of SR has a strong correlation with the ranking quality and thus it is an effective indicator of the quality of ranked documents. This work demonstrates that it is possible to provide both ranked documents and explicit information regarding the ranking of these documents, allowing SRs experts to anticipate and plan the manual screening process.

5.2 Related Work

Query performance prediction. Query performance prediction (QPP) is a longstanding research problem in Information Retrieval, started from TREC Robust track in 2004 [91]. The task is to predict the performance of a query given a collection of documents and a search system/model. A large body of work on the QPP task is typically divided into two settings in the prediction: pre-retrieval setting and post-retrieval setting. In a pre-retrieval setting, the performance of a query is predicted without requiring a search system [92]. On the other hand, a post-retrieval setting employs information obtained from executing a search system such as

¹Because it is difficult for a ranking model to rank an SR, high ranking difficulty yields low ranking quality.

Table 5.1: Comparison between the estimation of SR ranking quality task and the query performance prediction task in terms of target collections. In this comparison, we use TREC Robust track in 2004 (TREC 2004) as a representative of the query performance prediction task.

	SR ranking quality estimation	Query performance prediction
Estimation object	Systematic review (SR)	Query
Target collection	Candidate documents	Corpus
Characteristics of target collection	Specific to a SR Collected by a Boolean retrieval Topically similar documents Small (6,000 in CLEF 2018)	Identical for all queries Taken from a long fixed time period Topically diverse documents Large (800,000 in TREC 2004)

ranking scores of documents [93] or relevance feedback [94]. Prediction in the post-retrieval setting tends to outperform that in the pre-retrieval setting, but the post-retrieval setting involves the higher computation than the pre-retrieval setting.

Several work examines the relation of QPP task with other existing tasks, such as the cluster ranking [95] and the ranking model evaluation [96]. Existing literature reports that these problems are closely related, and they can complement each other by understanding their similarities and differences.

Along the same lines, we discuss the similarities and differences between the QPP task and the SR ranking quality estimation task. Both tasks aim to estimate the performance of query/SR. As every ranking quality involves a query, one may assume that the estimated query performance by an existing QPP measure directly represents SR ranking quality. However, the estimations in the two tasks occur in different settings, to be precise different document collections (*i.e.*, target collection). In the QPP task, a target collection is a corpus which helps imitate a query's behavior in ad-hoc retrieval, so that it consists of the mixture of various documents. More importantly, every query's performance is estimated on that same target collection. In contrast, a target collection in the SR ranking quality estimation is candidate documents and they are specifically collected for a given SR. Thus, depending on SRs, the estimation is conducted on an unique target collection. Table 5.1 presents the detailed comparison between the two tasks.

Different from the SR ranking quality estimation in the screening step, the target collection in the retrieval step is a corpus which is the same as that in the QPP task. Recall that the retrieval step in SRs is to issue keyword queries to a corpus and retrieve documents. In [97]

the authors examine existing QPP measures in estimating the performance of keyword queries in the retrieval step in SRs, not for the screening step.

QPP Measures. QPP measures are mainly categorized into pre-retrieval and post-retrieval QPP measures. While post-retrieval QPP measures use statistical information computed from the execution of retrieval models, pre-retrieval QPP measure employs the information of given a query and a target collection only. Hence, they are simple and fast to compute. We explain the details of commonly adopted pre-retrieval measures [98,99].

The simplest measure is query length (**QLen**), defined as the number of non-stopwords in a query [100]. A few QPP measures are based on inverse document frequency (IDF) values of query term w , defined as follows [101].

$$IDF(w) = \log_{10} \frac{|C|}{df(w)} \quad (5.1)$$

where C is a document collection (*e.g.*, a corpus in ad-hoc retrieval) where a query and a ranking models are executed on, and $df(w)$ denotes the document frequency of w in C . Using Equation 5.1, average (**avgIDF**), sum (**sumIDF**), maximum (**maxIDF**) of terms in a query can be used as a query performance predictor [101]. Additionally, a standard deviation of IDF values (**stdIDF**) is evaluated. stdIDF is estimated with INQUERY's IDF definition [102], same as studied in [100]. stdIDF is based on the idea that if a variance of IDF values in query terms is high, a query performance is high. Note that stdIDF utilizes the distribution of IDF values as a predictor, while the rest IDF-based measures (avgIDF, sumIDF, maxIDF) use magnitudes of IDFs.

Simplified clarity score (**SCS**) is a pre-retrieval version of clarity (a post-retrieval measure, to be explained shortly). SCS estimates a divergence of the query language model from the collection language model [98].

$$SCS(q) = \sum_q P(w|q) \cdot \log_2 \frac{P(w|q)}{P(w|C)} \quad (5.2)$$

where $P(w|q)$ denotes maximum likelihood estimation of the query model based on uni-gram. It is estimated by the occurrence of query term w divided by the total number of query terms. $P(w|C)$ denotes the collection language model, computed by the relative frequency of query term w in collection C , similar to $P(w|q)$.

Average inverse collection term frequency (**avgICTF**) calculates inverse term frequency of query terms in the collection [98].

$$avgICTF(q) = \frac{\log_2 \prod_{w \in q} (\sum_{d \in C} |d|) / c(w, C)}{|q|} \quad (5.3)$$

where d is a document in C and $|d|$ and $|q|$ indicate the number of terms in d and q , respectively. $\sum_{d \in C} |d|$ calculates the total number of terms in C . $c(w, C)$ is the count of w in C .

Similarity between a collection and a query (**SCQ**) measures query performance by using term frequencies in a collection and IDFs of query terms [103]. Given a query q , SCQ is estimated as follows.

$$SCQ(q) = \sum_{w \in q} (1 + \ln c(w, C)) \cdot \ln(1 + \frac{|C|}{df(w)}) \quad (5.4)$$

As a variation of SCQ, normalized SCQ (**nSCQ**) is a SCQ value divided by the number of terms in a query q . Another variation, maximum SCQ (**maxSCQ**), is the maximum $SCQ(w)$ in query terms $w \in q$.

5.3 Proposed Approach

5.3.1 Analysis on Ranking Quality

Ranking qualities of SRs are affected by the choice of ranking models. In this section, we study changes of ranking qualities caused by ranking models. We use three promising ranking models that use different aspects of data to compute ranking scores. They have shown high performance in previous literature.

BM25 is a classic retrieval function for bag-of-words (BOW) document retrieval. It is the most well-known simple and effective baseline in retrieval tasks. It computes a document ranking score using the length of document, within-document term frequency, and inverse document frequency in a document collection. Several existing studies have shown that BM25 also provides solid baseline performance in screening prioritization [3, 104]

Seed-driven Document Ranking (**SDR**) is a ranking model specialized for screening prioritization. As we discussed in (Chapter 3), documents are represented as a bag-of-medical concepts (BOC) in SDR. When calculating ranking scores, SDR uses a weight function for medical

concepts in a query, which is combined with a probabilistic query likelihood (QL) model. When we transform documents into BOC representations, we follow the same procedure we described in Chapter 3.

We have shown that average embedding similarity (AES) is a simple and effective model to find similar documents in Chapter 3. A document is represented as the average embedding of the words that appear in the document. Different from BOW and BOC representations, word embedding alleviates the common vocabulary mismatch problem. A ranking score for a document is a cosine similarity with a query. We use publicly available word embedding trained by skip-gram word2vec on the MEDLINE/PubMed corpus². The dimension of word embedding is 200, and the window size is set to 5. The rest of the parameters are the same as the original word2vec implementation [105].

In screening prioritization, a query is not formally defined. Among several suggested resources for a query (details in Related Work), we use an example document (seed document) as a query. Using a seed document as a query does not require the additional step of creating queries, and thus the derived ranking results are easily reproducible. We use every relevant document as a query within a SR, and each document ranking is evaluated by average precision (AP). Lastly, we set the average of AP as the ranking quality of SR.

We use CLEF 2018 eHealth Task 2 TAR (CLEF18) dataset. The CLEF18 dataset consists of 72 SRs and provides document IDs (PMIDs) for each SR's candidate documents. We obtain document content (title and abstract) from the MEDLINE/PubMed corpus using PMIDs. Every candidate document has two relevance labels from the abstract screening and the full text screening. The relevant labels are binary (relevant or non-relevant). Because the full text labels are the final labels for screening, we use them to evaluate the ranking model. Table 5.2 presents the statistics of candidate documents and relevant documents in CLEF18 dataset.³

Observation 1. *The changes in ranking quality of a SR by ranking models are very small.*

For each SR, we calculate the difference of ranking qualities between two ranking models. After creating pairs from the three ranking models, we take the differences of these pairs and

²<http://bio.nlplab.org/>

³In CLEF18 dataset, a few SRs contain zero relevant documents, even though actual SRs do have relevant documents. The reason is that CLEF18 dataset is created by collecting documents in PubMed alone, and thus documents indexed in other digital libraries are not included. For details of the creation of CLEF18 dataset, we refer to [3, 104].

Table 5.2: The numbers of candidate documents and relevant documents of 72 SRs in CLEF18 dataset. The number of relevant documents are full text screening labels. The numbers of median and average are rounded to the nearest integer.

	Candidates documents	Relevant documents
Max	79,786	117
Min	64	0
Median	3,424	14
Mean	6,346	23

average them. The mean ranking quality difference in all SRs is 0.067 ($\sigma = 0.076$). This shows that despite different ranking functions, the changes of ranking qualities in SRs are minimal.

Observation 2. *Given a ranking model, the difference of ranking qualities among SRs is very large, and all three models show the large difference.*

We compare SRs in terms of their ranking qualities given a ranking model. The difference between the highest ranking quality of SR and the lowest ranking quality of SR in BM25 is 0.80. The gaps between highest and lowest ranking qualities in SDR and AES are 0.97 and 0.80, respectively. In all three ranking models, SRs have drastically different ranking qualities. Furthermore, we also present the highest and the lowest 5 ranking qualities within each ranking model in Figure 5.1. Observe the substantial gap between two groups of SRs within each model, and the large gaps are consistent in all models.

From these observations, we can come to the conclusion that even though different ranking models are used, when SRs are compared in terms of their rank qualities, their rankings (*i.e.*, their relative rank qualities) barely change. We first rank 72 SRs by their ranking qualities using two different ranking models, and then calculate a Spearman’s ranking correlation coefficient between the two lists of SRs. The coefficients in pairs of models {BM25, SDR}, {BM25, AES}, and {SDR, AES} are 0.81, 0.77, and 0.75, respectively. Indeed, all three pairs have very strong rank correlations. This results confirm that some SRs consistently have the higher ranking qualities than other SRs despite different ranking models.

Furthermore, the patterned bars in Figure 5.1 indicate the SRs that present the highest/lowest 5 ranking qualities in more than one model. On the other hand, the bars without patterns are SRs that only appear in a given ranking model, not the other two. The majority of SRs that have

Fig. 5.1: The lowest and the highest 5 ranking qualities of SRs in CLEF18 dataset given a ranking model. Three ranking models are presented. The pattern on the bars indicates the same SRs across the ranking models.

the highest/lowest ranking qualities in one model also have the highest/lowest ranking qualities in other models.

This analysis implies that there might be defining factors in SRs that underlie the ranking qualities of SRs. In other words, a SR has characteristics that may affect the document ranking of the SR to be easy (high ranking quality) or difficult (low ranking quality).

5.3.2 Proposed Measure: Topic Broadness

In this work, we investigate the *topic broadness* of SR as a factor that influences the ranking quality. The rationale is that the broad topic of SR is likely to lower the ranking quality because its relevant documents are likely to diverse (less similar to each other) and it makes difficult to find them in candidate documents. Furthermore, when a SR has broad topic, it is not straightforward to accurately encode the relevance in a query and a ranking model, compared to SRs with specific topic. Therefore, SRs with broad topic are likely to have low ranking quality (*i.e.*, high ranking difficulty).

We propose a measure that quantifies the topic broadness of SR to estimate the ranking quality. We adopt the same assumption made in [79]: a few example documents (*i.e.*, seed documents) are available before the screening step. With the assumption that two seed documents are available, the proposed measure estimates the topic broadness of SR using the two seed documents and candidate documents. Specifically, the topic broadness is computed using (i) a similarity between two seed documents and (ii) a similarity between a seed and a candidate document. Formally, we define topic broadness (TB) as follows.

$$TB(s_1, s_2, C) = -\frac{2 \cdot \text{sim}(s_1, s_2)}{\sum_{c \in C} \text{sim}(s_1, c) + \sum_{c \in C} \text{sim}(s_2, c)} \quad (5.5)$$

$$sim(d_1, d_2) = \sum_{w \in d_1, d_2} tf(w, d_1) + tf(w, d_2) \quad (5.6)$$

In Equation 5.5, s_1 and s_2 ($s_1 \neq s_2$) denote two seed documents. c is a candidate document in the target collection C ($c \in C$). Note that Equation 5.5 contains a negative sign. Hence, a great value in nominator represents small topic broadness (*i.e.*, specific topic). Besides, since TB from Equation 5.5 is the estimated topic broadness of SR, a negative value of TB can be considered as the estimated ranking quality of SR.

We consider the increase of $sim(s_1, s_2)$, a similarity between two seed documents, makes the topic broadness (ranking quality) of SR decreased (increased). When a SR has broad topic, its relevant documents including s_1 and s_2 are less similar. The relation of $sim(s_1, s_2)$ with SR ranking quality is supported by cluster hypothesis: the pairwise similarity of the relevant documents can determine the performance of a search on a particular collection [57].

Next, we set the increase of $sim(s_*, c)$, a similarity between a seed and a candidate document makes the topic broadness (ranking quality) of SR increased (decreased). Since the majority documents are non-relevant documents and only a small fraction of documents in C are relevant documents, $c \in C$ is considered a non-relevant document. High similarities between relevant and non-relevant documents affect the low ranking quality of SR as it makes difficult to differentiate relevant and non-relevant documents. Moreover, in pilot tests we found that a SR tends to have the low rank quality when its number of candidate documents is large. We leverage the size of C by using the sum of $sim(s_*, c)$ in C in TB . Lastly, the proposed measure uses the average⁴ of two similarities of candidate documents with two seeds.

In Equation 5.5, a similarity between two documents is calculated by Equation 5.6. $tf(w, d)$ denotes a frequency of word w in document d . We note that the proposed measure does not incorporate information that are obtained from executing a ranking model such as ranking scores of candidate documents or relevance feedback.

The most similar existing work to the proposed measure is [106]. The authors suggest multiple types of similarities within/across a target collection, relevant documents, and a query, in order to estimate query performance in the QPP task. We evaluate a few similarities proposed in [106] that can be applicable to the SR ranking quality estimation in experiments.

⁴We examine alternative ways (*e.g.*, the maximum of the two values) to incorporate two document similarities of each seed and candidate documents, but there was no significant difference.

5.4 Experiments

5.4.1 Dataset

The same CLEF18 dataset presented in Analysis section is used. As the proposed method assumes that two seed documents are available, SRs having less than 2 relevant documents are excluded. Among 72 SRs in CLEF18 dataset, 9 SRs have 0 or 1 relevant documents. Total 63 SRs are used in the evaluation.

Selection of two seed documents. The proposed measure assumes two relevant documents are available and uses them for estimating the topic broadness of SR. For a given SR, any two documents among relevant documents can be two seed documents. To demonstrate this scenario, we evaluate the proposed measure using all possible pairs of relevant documents in a SR, and report the averaged value over the all pairs. When many pairs are possible in a SR, we use randomly selected 30 pairs among them. Since we use the averaged value, we present the distribution over the estimated values of pairs in the Result and Discussion section and analyze the effect of different seed documents.

5.4.2 Baselines

We evaluate two sets of baselines: pre-retrieval QPP measures and alternative measures to estimate topic broadness. **Pre-retrieval QPP** As discussed in Related Work section, QPP measures are mainly classified into pre-retrieval QPP measures and post-retrieval QPP measures. The proposed measure does not use information computed by ranking models. We set only pre-retrieval QPP measures as baselines for a fair comparison. Frequently adopted pre-retrieval QPP measures are evaluated in this experiment [97] : query length (Qlen), a family of IDF-based measures (avgIDF, sumIDF, maxIDF, stdevIDF) [101], simplified clarity score (SCS) [98], average inverse collection term frequency (**avgICTF**) [98], similarity between a collection and a query (**SCQ**), and its two variations, normalized SCQ (**nSCQ**) and maximum SCQ (**maxSCQ**) [103]. The details of pre-retrieval QPP baselines are discussed in Section 5.2. They incorporate the statistics of terms in a query and the collection of documents in order to estimate the query performance. In the QPP methods, the greater the estimated value indicates the higher the query performance. In this task, we use the performance of query as the ranking quality of SR.

Each of two seed documents can be a query, so that two values are obtained for a single QPP measure. We examine the average and the maximum of two QPP estimates as the final estimation of a pre-retrieval measure. The average value results in slightly better performance compared to the maximum value, but they show no significant difference. We report the average value as the final performance of QPP baselines.

Alternatives representing topic broadness. As the second set of baselines, we suggest and examine several alternative measures that quantify the topic broadness of SR: *collSize*, *collClarity*, *numClusters*, a family of document similarities. They leverage various information from candidate documents. The assumption is that since candidate documents are specifically collected documents for a given SR, they are able to represent the topic broadness of given SR.

collSize is the size of target collection (*i.e.*, the number of candidate documents). The large number of candidate documents may represent broad topic in SR. Next, collection clarity (**collClarity**) is motivated by Clarity [101] and simplified clarity score (SCS) [98]. It measures a divergence between the candidate collection language model and a corpus (PubMed) language model as follows.

$$collClarity = \sum_{w \in C} P(w|C) \cdot \log_2 \frac{P(w|C)}{P(w|\mathbb{C})} \quad (5.7)$$

where $P(w|C)$ is estimated by maximum likelihood on the candidate collection C . $P(w|\mathbb{C})$ denotes the corpus language model and we use the uni-gram language model of PubMed corpus² $\mathbb{C}$. We also evaluate the number of clusters (**numClusters**) in the collection of candidate documents, proposed in [106]. The large number of document clusters could represent broad topic of SR. We first represent each document as the average embedding of all word embeddings² included in a given document and then find the best number of clusters a range from 2 to 100 using k -means clustering according to silhouette scores.

Lastly, we evaluate a family of pairwise document similarities among candidate documents. Likewise, a candidate document is represented as the average embedding of all word embeddings² and then we report average (*avgDocSim*), sum (*sumDocSim*), maximum (*maxDocSim*), and standard deviation (*stdevDocSim*) of the pairwise document similarities.

These alternative measures have never been examined in estimating the ranking performance of SRs. While the proposed measure and the alternatives are different in the use of seed documents, both are able to topic broadness of SRs.

5.4.3 Ground Truth and Evaluation Metrics.

We set the ground truth of the ranking quality of SR to its average precision (AP). AP is an ranking evaluation measure that takes into account both precision and recall components. Specifically, it averages all precision values at where each relevant document is found in a ranking result. In general, it indicates how good the given ranking result is. We use the ranking quality by the same three ranking models (BM25, SDR, and AES) introduced in the previous section, so that the measures are evaluated on the three sets of ground truth.

Our main focus is to estimate the relative ranking qualities of SRs, rather than the actual values of ranking qualities. The actual values of ranking qualities change (yet slightly) by ranking models, but the relative ranking qualities among SRs mostly remain unchanged (discussed in the previous section). We examine the order of SRs ranked by the estimated ranking qualities, comparing to the order of SRs ranked by ground truth. A non-parametric ranking correlation coefficient (Spearman's ρ) is computed between the two lists of SRs. A high absolute value of the coefficient presents a high accuracy of a measure in estimating the relative ranking qualities of SRs. In addition to the main focus, we also report the evaluation results in estimating the actual value of SR ranking quality, using a linear correlation coefficient (Pearson's γ). This evaluation setting is similar to that of the QPP task.

5.5 Results and Discussion

In this section, we discuss the overall evaluation results for the proposed measure and the baselines. Next, we present the detailed analysis on the proposed measure.

5.5.1 Overall Results

Tables 5.3 and 5.4 show the overall performance on Spearman's rank correlation coefficient and Pearson linear correlation coefficient, respectively. The best measure is in boldface and the second best measure is underlined. We present the ranking of the proposed method among all evaluated methods and also the performance difference between the proposed method and the best performing baseline in the last row of Tables 5.3 and 5.4. Since the evaluation metrics are correlation coefficients, the +/- signs are the direction of a correlation. We focus on the absolute values of coefficients to compare the effectiveness of measures.

Table 5.3: Spearman’s rank correlation coefficients of the proposed measure and the baselines. The correlation coefficients are computed with the two lists of SR ranked by the estimated values and the actual performance (the average of AP) by a given ranking model. The best and the second best performances are in boldface and underlined, respectively. * denotes that performance of the proposed measure is statistically significant compared to the given baseline.

Model		BM25	SDR	AES
baselines	QLen	0.181 (1.62e-01)*	0.341 (7.15e-03)*	0.200 (1.22e-01)*
	avgIDF	-0.148 (2.55e-01)*	-0.149 (2.52e-01)*	-0.256 (4.58e-02)*
	sumIDF	0.062 (6.33e-01)*	0.130 (3.18e-01)*	-0.046 (7.20e-01)*
	maxIDF	-0.351 (5.51e-03)*	-0.231 (7.40e-02)*	-0.477 (1.02e-04)*
	stdevIDF	0.430 (5.49e-04)*	0.464 (1.65e-04)*	0.492 (5.56e-05)*
	SCS	-0.057 (6.63e-01)*	-0.201 (1.20e-01)*	-0.191 (1.40e-01)*
	avgICTF	0.045 (7.31e-01)*	-0.067 (6.07e-01)*	-0.084 (5.20e-01)*
	SCQ	-0.221 (8.65e-02)*	-0.248 (5.41e-02)*	-0.348 (5.92e-03)*
	nSCQ	-0.376 (2.80e-03)*	-0.492 (5.66e-05)*	-0.478 (9.45e-05)*
	maxSCQ	-0.374 (3.00e-03)*	-0.489 (6.37e-05)*	-0.458 (2.03e-04)*
alternatives	collSize	-0.442 (3.64e-04)	<u>-0.545</u> (5.53e-06)	-0.530 (1.08e-05)*
	numClusters	-0.079 (9.68e-01)*	-0.005 (5.46e-01)*	-0.013 (9.20e-01)*
	collClarity	-0.033 (7.99e-01)*	0.113 (3.84e-01)*	0.066 (6.09e-01)*
	avgDocSim	0.090 (9.47e-01)*	-0.009 (4.93e-01)*	-0.128 (3.26e-01)*
	sumDocSim	-0.526 (3.59e-04)	-0.442 (1.32e-05)	<u>-0.568</u> (1.82e-06)
	maxDocSim	-0.232 (2.78e-01)*	-0.141 (7.26e-02)*	-0.345 (6.53e-03)*
	stdevDocSim	0.064 (4.98e-01)*	0.088 (6.24e-01)*	0.267 (3.72e-02)*
Proposed: TB		<u>-0.496</u> (4.79e-05)	-0.570 (1.67e-06)	-0.584 (7.80e-07)
Rank (Δ)		2 (0.030)	1 (0.250)	1 (0.016)

Overall, the proposed measure outperforms the most baselines in both evaluation measures (Spearman and Pearson). For the ranking quality of SDR and AES, TB shows the best performance. For BM25, it achieves the second best performance, following the best-performing baseline measure by a small gap. Moreover, the proposed measure presents a very small p -values compared to baselines.

Common characteristics in high-performing measures. Let’s look into each evaluation measure. As shown in Table 5.3, when estimating the relative rank qualities in SRs, sumDocSim, and collSize, show the best performance among the alternative measures, and they are even comparable to TB. The proposed measure and these two measures share the common element: the number of candidate documents. TB utilizes it when estimating the similarities between

Table 5.4: Pearson linear correlation coefficients of the proposed measure and the baselines. The correlation coefficients are computed with the two lists of the estimated values and the actual performance (the average of AP) by a given ranking model. The best and the second best performances are in boldface and underlined, respectively. * denotes that performance of the proposed measure is statistically significant compared to the given baseline.

Mode		BM25	SDR	AES
baselines	QLen	0.179 (1.66e-01)*	0.369 (3.47e-03)*	0.180 (1.64e-01)*
	avgIDF	-0.297 (2.01e-02)*	-0.247 (5.53e-02)*	-0.343 (6.73e-03)*
	sumIDF	-0.094 (4.70e-01)*	0.067 (6.07e-01)*	-0.154 (2.35e-01)*
	maxIDF	-0.580 (9.56e-07)*	-0.420 (7.44e-04)*	-0.626 (6.72e-08)*
	stdevIDF	0.701 (3.10e-10)	<u>0.650</u> (1.40e-08)	<u>0.703</u> (2.67e-10)*
	SCS	-0.220 (8.86e-02)*	-0.281 (2.81e-02)*	-0.272 (3.38e-02)*
	avgICTF	-0.135 (3.01e-01)*	-0.225 (8.14e-02)*	-0.183 (1.58e-01)*
	SCQ	-0.399 (1.44e-03)*	-0.327 (1.01e-02)*	-0.466 (1.53e-04)*
	nSCQ	-0.525 (1.40e-05)*	-0.525 (1.39e-05)*	-0.579 (9.87e-07)*
	maxSCQ	-0.531 (1.09e-05)*	-0.509 (2.85e-05)*	-0.583 (8.15e-07)*
alternatives	collSize	-0.254 (4.85e-02)*	-0.227 (7.90e-02)*	-0.320 (1.19e-02)*
	numClusters	-0.074 (6.29e-01)*	0.063 (5.69e-01)*	0.000 (1.00e+00)*
	collClarity	0.006 (9.62e-01)*	0.229 (7.49e-02)*	0.104 (4.23e-01)*
	avgDocSim	-0.120 (3.58e-01)*	0.084 (5.21e-01)*	-0.147 (2.59e-01)*
	sumDocSim	-0.083 (5.24e-01)*	-0.051 (6.98e-01)*	-0.201 (1.21e-01)*
	maxDocSim	-0.274 (3.25e-02)*	-0.307 (1.59e-02)*	-0.321 (1.16e-02)*
	stdevDocSim	0.135 (3.01e-01)*	-0.020 (8.81e-01)*	0.207 (1.09e-01)*
Proposed: TB		<u>-0.698</u> (3.96e-10)	-0.707 (1.97e-10)	-0.720 (6.44e-11)
Rank (Δ)		2 (0.003)	1 (0.057)	1 (0.017)

seed and candidate documents in the denominator in Equation 5.5. sumDocSim and collSize uses the number of candidate documents either indirectly or directly. The large impact of the number of candidate documents presents that the retrieval step and the screening step are closely related. The number of candidate documents in the screening step is determined in the retrieval step. According to the evaluation results, when the retrieval step collects the smaller number of candidate documents, the rank quality tends to be higher in the screening step.

We note that both sumDocSim and collSize are the alternative measure proposed by us, in order to estimate topic broadness. Their high performance supports our hypothesis that the topic broadness of SRs is a factor that underlies the ranking qualities of SRs.

As shown in Table 5.4, for estimating the absolute rank qualities of SRs, either the proposed

measure or stdevIDF achieves the highest performance. stdevIDF shows a strong correlation with the ranking quality of SRs. The high stdevIDF value indicates that terms in seed documents have various IDF values, resulting in the high ranking quality. Recall that seed documents are relevant documents of SR. This result describes that when some terms in a relevant document have high IDF values (*i.e.*, informativeness) and some have low IDF values, SR tends to have the high rank quality.

Discussion on alternative measures. Among a family of pairwise document similarity measures, sumDocSim, avgDocSim, and maxDocSim incorporate the document similarity values as an indicator directly, while stdevDocSim utilizes the distribution of them. As shown in Tables 5.3 and 5.4, the three measures present negative correlations the rank quality in most cases. It shows that when candidate documents are similar to each other, a SR tends to be difficult to achieve the high rank quality. High similarities in candidate documents mean high similarities between relevant and non-relevant documents, and thus it leads to the high ranking difficulty of SR.

numClusters does not provide meaningful correlations. We looked into the results and found that the majority of SRs has 2 or 3 as the best number of clusters. This result shows that candidate documents are very similar each other in the average embedding document representations. This is possibly because that they share many term overlaps due to Boolean retrieval in the retrieval step. At the same time, the document representation using word embeddings is not enough to effectively differentiate them in clustering method. Incorporating more distinctive document features in document representations may help identify the more fine-grained document clusters.

Discussion on pre-retrieval QPP measures. In Tables 5.3 and 5.4, most QPP measures have negative correlations with the rank qualities of SRs. This is the opposite correlation direction, when they are used for estimating the query performance. In the query performance estimation, the high value in a QPP measure represents the high query performance (positive correlation). We believe the different direction of correlations may occur by the difference of the target collections between the two tasks, as we discussed in Table 5.1. In SRs, the target collection (candidate documents) consists of topically similar documents while in QPP task, it contains topically diverse documents. Lastly, the opposite behaviors of the QPP measures confirms that even though the two tasks seem similar at first glance, they differs from each other and cannot be considered interchangeable tasks

We conduct the statistic significance test between the proposed measure and baseline measures, similar to [94, 107]. We first generate 30 sub-sample groups and each sub-sample group has 30 randomly selected SRs among 63 SRs. We compute correlation coefficients in each sub-sample group. Then, the statistical significance tests are conducted on 30 coefficient values by the proposed measure and by a baseline measure. The performance of the proposed measure is statistically significant than most baselines, except a few baseline measures. We note that the baseline measures that have superior or comparable performance to the proposed measure (*e.g.*, `collSize` and `sumDocSim`) are also the newly proposed measures in this work for estimating the topic broadness of SRs.

5.5.2 Ablation Analysis

The proposed measure incorporates two types of document similarity: similarity between two seeds $\text{sim}(s_i, s_j)$ and similarity between a seed and a candidate document $\text{sim}(s_*, c)$. We analyze the impact of each similarity type on the proposed measure. An ablation test is conducted by omitting each document similarity from the proposed measure. We examine their performances by comparing to that of the original proposed measure. One degraded measure removes $\text{sim}(s_i, s_j)$ in Equation 5.5, and the other degraded measure uses $\text{sim}(s_i, s_j)$ alone, without $\text{sim}(s_*, c)$. Both measures keep the minus sign in Equation 5.5.

Table 5.5 presents the evaluation results of the degraded measures and the proposed measure as reference. When estimating relative ranking qualities among SRs (Spearman’s rank correlation coefficient), both degraded measures decrease the estimating power of the proposed measure. It shows both types of document similarity make contribution on the effectiveness of the proposed measure. In comparison of the two similarity types, the measure without $\text{sim}(s_*, c)$ has a larger drop, while the measure without $\text{sim}(s_i, s_j)$ shows a marginal decrease. This result suggests that $\text{sim}(s_*, c)$ plays a major role in the proposed measure.

When estimating the actual values (AP) of SR ranking qualities (Pearson linear correlation coefficient), the measure without $\text{sim}(s_i, s_j)$ presents the decreased coefficients and the measure without $\text{sim}(s_*, c)$ shows the slightly increased coefficients. Although the comparable effectiveness of the measure without $\text{sim}(s_*, c)$ to the proposed measure, we argue that the performance on the direct estimates of SR ranking qualities is less significant as the actual values (AP, in this case) are different depending on ranking models. On the other hand, the relative ranking

Table 5.5: Impact of the two types of document similarity in the proposed measure.

Model	Spearman’s rank coefficient			Pearson linear coefficient		
	BM25	SDR	AES	BM25	SDR	AES
Measure _{sim(s*,c)}	-0.371	-0.316	-0.371	-0.341	-0.270	-0.364
Measure _{sim(s_i,s_j)}	-0.432	-0.522	-0.518	-0.712	-0.712	-0.725
Proposed measure	-0.496	-0.570	-0.584	-0.698	-0.707	-0.720

qualities of SRs mostly remain the same, as discussed in Observation 1 and 2. In this work, our key focus is to estimate the relative ranking qualities among SRs, evaluated by Spearman’s rank correlation coefficients.

Lastly, the greater impact of the similarity between a seed and a candidate document also demonstrates the close relation between the number of candidate documents (*i.e.*, the size of target collection) and the ranking quality of SRs. Recall that the proposed measure leverages the size of target collection by using the sum of all $sim(s_*, c)$ in C . The analysis result is consistent with the high performances of collSize and sumDocSim among baselines.

5.5.3 Impact of Seed Documents

The proposed measure assumes two relevant documents are available given a SR. As described in Chapter 4.3, we generate possible seed document pairs and evaluate the measure using them. Finally, we report the averaged performance in Tables 5.3. In this section, we examine the effect made by the choice of two seed documents in the performance. We study the variance of the values of individual pairs. A large variance means that the performance of the proposed measure is highly dependent on the choice of the two seed documents. In other words, the proposed measure may not show the good performance presented in Tables 5.3 and 5.4, depending on given two seeds. We calculate the variance within each SR. The variance in CLEF18 is $0.3e-3$ on average. The very small variance shows that the proposed measure makes stable estimates whichever two seed documents are used.

Furthermore, we study the cases when more than two relevant documents are available and show how to incorporate them in the proposed measure. We examine the proposed measure when three relevant documents are available. When more than three relevant documents are available, the idea is the same three relevant documents. Similar to the evaluation for two

Table 5.6: Spearman’s rank correlation evaluation using three seed documents. The best and the second best performances are in boldface and underlined, respectively.

Model		BM25	SDR	AES
baselines	QLen	0.202 (1.18e-01)*	0.384 (2.23e-03)*	0.186 (1.51e-01)*
	avgIDF	-0.174 (1.81e-01)*	-0.122 (3.49e-01)*	-0.288 (2.43e-02)*
	sumIDF	0.096 (4.62e-01)*	0.137 (2.92e-01)*	-0.066 (6.14e-01)*
	maxIDF	-0.365 (3.79e-03)*	-0.230 (7.42e-02)*	-0.487 (6.91e-05)*
	stdevIDF	0.395 (1.63e-03)*	0.448 (2.90e-04)*	0.456 (2.25e-04)*
	SCS	-0.153 (2.39e-01)*	-0.213 (9.91e-02)*	-0.208 (1.07e-01)*
	avgICTF	-0.058 (6.58e-01)*	-0.078 (5.48e-01)*	-0.149 (2.52e-01)*
	SCQ	-0.262 (4.17e-02)*	-0.343 (6.85e-03)*	-0.390 (1.88e-03)*
	nSCQ	-0.485 (7.51e-05)*	-0.478 (9.72e-05)*	-0.526 (1.36e-05)*
	maxSCQ	-0.372 (3.13e-03)*	-0.482 (8.38e-05)*	-0.465 (1.61e-04)*
alternatives	collSize	-0.442 (3.64e-04)	-0.545 (5.53e-06)	-0.530 (1.08e-05)*
	numClusters	-0.079 (9.68e-01)*	-0.005 (5.46e-01)*	-0.013 (9.20e-01)*
	collClarity	-0.033 (7.99e-01)*	0.113 (3.84e-01)*	0.066 (6.09e-01)*
	avgDocSim	0.090 (9.47e-01)*	-0.009 (4.93e-01)*	-0.128 (3.26e-01)*
	sumDocSim	-0.526 (3.59e-04)	-0.442 (1.32e-05)	<u>-0.568</u> (1.82e-06)
	maxDocSim	-0.232 (2.78e-01)*	-0.141 (7.26e-02)*	-0.345 (6.53e-03)*
	stdevDocSim	0.064 (4.98e-01)*	0.088 (6.24e-01)*	0.267 (3.72e-02)*
Proposed: TB		-0.574 (3.53e-05)	<u>-0.503</u> (1.31e-06)	-0.572 (1.45e-06)
Rank (Δ)		1 (0.048)	2 (0.042)	1 (0.004)

available documents, we first generate possible document triples among relevant documents in a SR. For each triple, the three document pairs are possible. Then, three values are obtained from the proposed measure by Equation 5.5. Lastly, we choose the minimum value among the three⁵. When a SR has broad topic, relevant document pairs may share low similarities. By selecting the minimum value among the three value, rather than the average value, the proposed measure is able to fully utilize the information provided by the three available documents.

We present the evaluation results in Tables 5.6 and 5.7. The overall performance shows the similar patterns in Tables 5.3 and 5.4. Three example documents provide more information about a given SR than two documents. Thus, the correlations (absolute values) in Tables 5.6 are stronger than those in Tables 5.3 and 5.4.

Another question regarding using two seed documents is: what if there is less than two

⁵Recall that the proposed measure contains a negative sign in Equation 5.5.

Table 5.7: Pearson linear correlation evaluation using three seed documents. The best and the second best performances are in boldface and underlined, respectively.

Model		BM25	SDR	AES
baselines	QLen	0.203 (1.17e-01)*	0.377 (2.71e-03)*	0.195 (1.32e-01)*
	avgIDF	-0.314 (1.38e-02)*	-0.234 (6.94e-02)*	-0.362 (4.10e-03)*
	sumIDF	-0.086 (5.10e-01)*	0.080 (5.40e-01)*	-0.161 (2.16e-01)*
	maxIDF	-0.585 (7.55e-07)*	-0.438 (4.16e-04)*	-0.631 (5.08e-08)*
	stdevIDF	<u>0.690</u> (7.95e-10)*	<u>0.653</u> (1.19e-08)	<u>0.689</u> (8.34e-10)
	SCS	-0.208 (1.07e-01)*	-0.208 (1.07e-01)*	-0.264 (4.01e-02)*
	avgICTF	-0.149 (2.52e-01)*	-0.216 (9.42e-02)*	-0.199 (1.24e-01)*
	SCQ	-0.390 (1.88e-03)*	-0.334 (8.51e-03)*	-0.463 (1.74e-04)*
	nSCQ	-0.526 (1.36e-05)*	-0.526 (1.37e-05)*	-0.581 (9.21e-07)*
	maxSCQ	-0.533 (9.88e-06)*	-0.502 (3.77e-05)*	-0.590 (5.50e-07)*
alternatives	collSize	-0.254 (4.85e-02)*	-0.227 (7.90e-02)*	-0.320 (1.19e-02)*
	numClusters	-0.074 (6.29e-01)*	0.063 (5.69e-01)*	0.000 (1.00e+00)*
	collClarity	0.006 (9.62e-01)*	0.229 (7.49e-02)*	0.104 (4.23e-01)*
	avgDocSim	-0.120 (3.58e-01)*	0.084 (5.21e-01)*	-0.147 (2.59e-01)*
	sumDocSim	-0.083 (5.24e-01)*	-0.051 (6.98e-01)*	-0.201 (1.21e-01)*
	maxDocSim	-0.274 (3.25e-02)*	-0.307 (1.59e-02)*	-0.321 (1.16e-02)*
	stdevDocSim	0.135 (3.01e-01)*	-0.020 (8.81e-01)*	0.207 (1.09e-01)*
	Proposed: TB	-0.709 (1.46e-10)	-0.710 (1.62e-10)	-0.727 (3.23e-11)
Rank (Δ)		1 (0.019)	1 (0.057)	1 (0.038)

relevant documents available and the proposed measure has a limitation for these cases. Nevertheless, the existing literature reports that it is common that SR experts have a few documents as examples when they conduct a literature survey [79].

5.6 Summary

In this work, we investigate the ranking quality estimation of SRs in screening prioritization. We first make observations on the ranking qualities with different ranking models. Our findings include that the changes in the ranking qualities of a SR by ranking models is very small, and the ranking quality differences among SRs are large. Lastly, we find that across different models, the relative rank qualities among SRs barely change, shown by strong ranking correlations. The analysis motivates us to believe that there might be defining characteristics of SRs that underlie

their ranking quality. We propose a measure that quantifies the topic broadness of SR and estimate the ranking quality of SR using this measure. Our hypothesis is that when a SR has broad topic, it is likely to have low ranking quality. Through the extensive evaluation, we show that the proposed measure effectively estimates the ranking quality of SR. We also discover the strong impact of the number of candidate documents on the ranking quality. Our work provides SR experts with not only the rankings of documents, but also the explicit estimates regarding the ranking quality. They aid SR researchers to foresee and plan the manual screening process.

Chapter 6

Methods to Enhance Ranking

So far we have studied ranking models for screening prioritization using query by document. We have developed a probabilistic ranking model with a term weight function in Chapter 3. In Chapter 4, we have proposed a semantic document matching model using positional patterns of PICO elements. It further improves the ranking quality upon the one achieved by a probabilistic ranking model in Chapter 3.

The in-depth experiment and analysis on these two ranking models and various baselines have shown a consistent pattern: depending on SRs a model performance widely varies. Motivated by the observation, we have studied to understand the ranking difficulty of SRs in Chapter 5. We have proposed a method estimating topic broadness of SR. The experiments have validated that the proposed method can predict the difficulty of a SR without relevance feedbacks.

Let us assume that a given SR is predicted to have broad topic and it is difficult to prioritize documents. Despite a ranking model with the state-of-the-art global performance, its local performance for the given SR is expected to be low. In this chapter, we ask a question how to improve local performance of a ranking model especially for difficult SRs while further strengthening the global performance?

To answer this question, we explore two approaches. First, we believe machine learning models recognizing PICO span in a document has potential to enhancing screening prioritization. We study human PICO annotations and how they reflect the performance of ML models. Second, the experiments have shown that semantic word representation (*i.e.*, word embeddings) are effective to find relevant documents in Chapter 3 and Chapter 4. We dive deeper into the

quality of word embeddings and shed a light on how to train robust word embeddings not only for frequent words/concepts but also for infrequent words/concepts.

PICO in medical literature. PICO elements are essential information of determining the relevance of documents in systematic reviews. We believe that training a machine learning model for automatic PICO span recognition and incorporating PICO spans in a ranking model have potential to raise ranking performance. To develop automatic PICO recognition models, we look into the most fundamental element influencing model performance: PICO span annotations. We investigate the agreement of PICO span annotations and understand that how the nature of PICO elements is translated into human annotations and more importantly into model performance¹ (Chapter 6.1).

Medical concepts in SRs. Medical concepts play a crucial role in systematic reviews. From the first step of conducting a SR, SR researchers start formulating a research question and relevance conditions using medical concepts as key information. As a result further elaborated relevance conditions include multiple concepts (Table 1.1). In the next step when researchers retrieve documents in the second step, these concepts are also used as a query for Boolean retrieval. Thus, candidate documents share medical concepts. In seed-drive document ranking for screening prioritization, we have shown in Chapter 3.2 medical concepts are more effective than word tokens to represent documents.

In recent years, word embeddings have shown advance in various tasks in Information Retrieval and Natural Language Processing. They are based on the assumption that words having similar meanings appear in similar contexts. Words are represented in dense and low-dimensional vector based on co-occurrence information within window size. It has been shown in many IR and NLP tasks they further improve the performance upon sparse and high-dimension vector representations. In our experiments in Chapter 1 and Chapter 4, equivalent performances have been observed. Based on the promising performance of word embeddings, we analyze the quality of word (medical concept) embeddings in terms of their stability. Our goal is to develop the deeper understanding on medical concept embeddings, which leads to robust and reliable ranking models for screening prioritization (Chapter 6.2).

¹This chapter is published as Grace E. Lee and Aixin Sun. A Study on Agreement in PICO Span Annotations, Proceedings of the 42nd International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 2019), 1149–1152, 2019.

Fig. 6.1: Example of span annotations in EBM-PICO dataset. This sentence (from PMID: 23549581) is annotated by 12 annotators (4 experts and 8 non-experts) and 3 different span annotations are made by them. The 3 spans have the same label **P**, but different boundaries.

6.1 Agreements on PICO Span Annotations

In evidence-based medicine, it is crucial for medical professionals to effectively find relevant literature since medical decisions are made based on primary evidence. Relevance of literature depends on relevance conditions and the conditions are often defined using PICO framework: **P**atient (problem, population), **I**ntervention, **C**omparator, and **O**utcome. PICO elements identified in medical literature, therefore, are critical for effective retrieval.

Existing datasets used for automatic identification of PICO elements are sentence-level annotations [108, 109]. Recently, toward more accurate and detailed identification, a large-scale PICO *span* annotation dataset (EBM-PICO) is released [73]. EBM-PICO consists of 5,000 abstracts of medical literature with PICO spans annotated by medical experts and non-experts. Annotated spans range from a single word to a long phrase.

PICO elements are presented in a *descriptive* manner in medical literature. Verbose descriptions of PICO elements make it nontrivial to decide their spans. Figure 6.1 shows an example sentence and its annotations in EBM-PICO dataset. The sentence contains information about **P** element. The three different spans are all annotated with label **P**. Even though each span has distinct boundaries, all indicate an acceptable information for **P**. Among the three, considering one span annotation correct and the rest incorrect may not be reasonable.

We study agreement in PICO span annotations made by different human annotators in EBM-PICO dataset. Specifically, we evaluate the annotation agreement using two types of measures: **exact span agreement** and **relaxed span agreement**. Exact span agreement is a standard evaluation approach for text span annotations. It evaluates both boundaries and labels of two spans. In relaxed span agreement, we analyze annotations in terms of two variants: *one-side boundary* (**OB**) agreement and *token overlap* (**TO**) agreement. OB and TO agreements evaluate

whether annotations are with a same label but allowing the start and end boundaries between two spans to mismatch.

As the annotations in EBM-PICO dataset are made by medical experts and non-experts, we estimate the agreement within each of expert and non-expert groups, and across expert and non-expert groups. Our study shows an extremely low level of exact span agreement, but a very high level of relaxed span agreements in both within and across the groups. The large improvement in relaxed span agreement indicates the general area of annotations is mostly agreed despite unmatched boundaries. Our results suggest that applying exact agreement alone may underestimate the agreement of PICO span annotations. Therefore, adopting both exact and relaxed agreements is more suitable for PICO span evaluation such as in a PICO span recognition task. Lastly, we present how the two agreements are leveraged in a PICO span recognition task.

6.1.1 PICO Span Annotation Dataset

We use EBM-PICO dataset [73] which provides PICO *span* annotations on 5,000 medical literature abstracts. Examples of annotations are shown in Figure 6.1. We study this dataset because of its fine-grained annotations and the large scale. Other existing PICO annotation datasets are sentence-level annotations [108, 109] and/or contain only a few hundreds of documents [110]. Note that, in this dataset Intervention (I) and Comparator (C) are combined as a single element I, so that each document can have three types of annotations: **P**, **I**, and **O**. Each annotation (*i.e.*, a text span as in Figure 6.1) has one of the three labels.

EBM-PICO provides annotations by individual annotators (used in Section 6.1.2), and also aggregated annotations which combine individuals' annotations (used in Section 6.1.3). We focus on the individuals' annotations to study agreements of PICO span annotations. Individual annotators include two groups: Amazon Mechanical Turk (MTurk) workers as non-experts and medical experts. Specifically, all the 5,000 documents have annotations of PIO elements by MTurk workers. For each document, annotation process is conducted by at least three MTurk workers. Among the 5,000 documents, 200 documents have annotations by medical experts.² Each of the 200 documents has at least two experts' annotations. Hence, these 200 documents have annotations by both MTurk workers and medical experts.

²We note that a few documents in the dataset have no annotations.

6.1.2 PICO Span Agreement

We measure agreement of PICO span annotations by computing $F1$. Given a pair of annotators, A and B , for a document, annotations made by A are first considered as *gold standard*, and annotations by B are considered as *predicted annotations*. We also switch A and B to calculate $F1$. Similar evaluation scheme was used in [111]. As a document has more than two annotators in EBM-PICO dataset, we average these values for all pairs of annotators, depending on the evaluation scenario (*e.g.*, within or cross non-expert and expert group evaluations). $F1$ is estimated by two kinds of agreement definitions.

Exact and Relaxed Span Agreements. In exact span agreement, two annotations agree with each other, if both have the same label and the same boundaries of text spans. By comparison, we also report two types of relaxed span agreement: one-side boundary (OB) agreement and token overlap (TO) agreement. Both OB agreement and TO agreement estimate the agreement in terms of whether two annotations are labeled as a same label, but allowing non-exact matching boundaries. For example, Table 6.1 shows 5 predicted annotations and their evaluations under the different span agreement definitions. The example sentence has 7 tokens and each cell represents a token. The cells in blue indicate a text span annotation. It is assumed that all examples in Table 6.1 have a same label.

In one-side boundary (**OB**) agreement, if either side of boundaries in predicted annotation is matched with a corresponding boundary of gold standard annotation, and there is at least one overlapped token between the two span annotations, then the predicted annotation is considered a correct prediction. In other words, a predicted span annotation can be larger or smaller than a gold standard span annotation, but the two spans share at least the same start or end boundaries. For instance, in Table 6.1, No. 1, 2, and 3 predicted annotations are correct annotations in OB agreement.

Token overlap (**TO**) agreement is defined with a more relaxed setup than one-side boundary (OB) agreement. Without considering boundaries of span annotations, if a predicted span annotation has at least one overlapped token with a gold standard span annotation, TO agreement counts the predicted annotation as a correct prediction. In Table 6.1, all five predicted annotations are correct under the TO agreement because of overlapped tokens.

Note that, the original EBM-PICO dataset paper reports token-level annotation agreement within expert annotators [73]. In their evaluation, the agreement is estimated on individual

Table 6.1: Five examples of predicted annotations (Predicted) and their evaluations by different agreements (*i.e.*, Exact, OB, TO). The example sentence consists of 7 tokens, and tokens in blue denote annotated text span (the same label is assumed in all annotations). o/x indicates a correct/wrong prediction against gold annotation (Gold).

No.	Predicted	Gold	Exact	OB	TO
1			o	o	o
2			x	o	o
3			x	o	o
4			x	x	o
5			x	x	o

tokens (token-level) and it does not consider boundaries of annotations. In our evaluation, both exact span agreement and the two relaxed span agreements are at span-level.³

Within Group Agreement. There are 2 groups of annotators (*i.e.*, MTurk workers and medical experts). In this section, we study pairwise annotation agreement *within each annotator group*.

As discussed in Section 6.1.1, there are 5000 documents annotated by MTurk workers and among them 200 documents are additionally annotated by experts. We report agreement within MTurk workers on the 5000 documents (MTurk-5000), and within medical experts on the 200 documents (Expert-200). Since these 200 documents also contain MTurk annotations, we also compute the agreement within MTurk workers on the 200 documents (MTurk-200), for a direct comparison with Expert-200.

Observation 6.1.1 *The overall exact span agreement, within MTurk worker group and also within medical expert group, is very low.*

Table 6.2 reports averaged $F1$ values (standard deviations) of pairwise agreement within MTurk workers, and within experts, by different agreement evaluations. The overall exact span agreement within each group is very low. The exact span agreement among MTurk workers is extremely low. For labels I and O, $F1$ values are even lower than 0.1. The similar low agreement is also observed among the expert annotators. The agreement for label I is slightly higher than 0.5 but for labels P and O, $F1$ values are lower than 0.4. These values show that more than a half of annotations are not agreed with other annotators, even among domain experts. We believe that the low exact span agreement is caused by the high verbosity of PICO elements. According to the exact agreement, none of the annotation pairs in Figure 6.1 are agreed.

³Token-level agreement has more evaluation instances than span-level agreement since a span consists of several tokens.

Table 6.2: Average (standard deviation) $F1$ scores estimated within MTurk workers on 5000 documents (MTurk-5000). For the 200 documents containing annotations by both MTurk workers and medical experts, the within group agreement of MTurk workers (MTurk-200) and medical experts (Expert-200) are reported. The $F1$ scores are computed based on different agreements.

	Exact span agreement			Relaxed One-side Boundary (OB)			Relaxed Token Overlap (TO)		
	MTurk-5000	MTurk-200	Expert-200	MTurk-5000	MTurk-200	Expert-200	MTurk-5000	MTurk-200	Expert-200
P	0.187 (0.136)	0.202 (0.125)	0.395 (0.201)	0.361 (0.179)	0.385 (0.157)	0.680 (0.200)	0.421 (0.190)	0.441 (0.161)	0.737 (0.189)
I	0.093 (0.085)	0.137 (0.095)	0.576 (0.301)	0.187 (0.112)	0.235 (0.120)	0.732 (0.271)	0.241 (0.123)	0.282 (0.128)	0.758 (0.269)
O	0.064 (0.053)	0.078 (0.054)	0.357 (0.167)	0.139 (0.089)	0.175 (0.093)	0.654 (0.178)	0.215 (0.121)	0.256 (0.115)	0.713 (0.169)

Table 6.3: Precision, Recall, and $F1$ values of cross-group annotation agreement between MTurk workers (predicted annotation) and medical experts (gold standard). The average values (standard deviation in parenthesis) of 200 documents are reported.

Exact span agreement				Relaxed One-side Boundary (OB)			Relaxed Token Overlap (TO)		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
P	0.266 (0.126)	0.338 (0.144)	0.275 (0.125)	0.483 (0.144)	0.624 (0.172)	0.496 (0.144)	0.537 (0.146)	0.707 (0.178)	0.553 (0.143)
I	0.243 (0.131)	0.332 (0.186)	0.257 (0.141)	0.360 (0.149)	0.570 (0.239)	0.391 (0.159)	0.408 (0.154)	0.769 (0.364)	0.457 (0.173)
O	0.147 (0.078)	0.220 (0.108)	0.159 (0.082)	0.295 (0.113)	0.450 (0.141)	0.316 (0.112)	0.364 (0.123)	0.789 (0.410)	0.412 (0.132)

Observation 6.1.2 *Annotators agree with the general areas where PIO elements appear, even though they made different choices in annotation boundaries.*

On relaxed span agreement, both MTurk workers and medical experts show largely increased $F1$ than that of exact span agreement. Specifically, on one-side boundary (OB) agreement, for MTurk worker group, $F1$ values are greater than twice of the $F1$ values estimated on exact span agreement. On token overlap (TO) agreement, for expert group, $F1$ values for all PIO elements are greater than 0.7, which is a clear indication of high level of agreement. The improved agreement in relaxed span agreement demonstrates annotators made annotations at similar areas but not necessarily with same boundaries. For instance, in Figure 6.1 some pairs of annotations are agreed depending on the OB or TO agreements, and it is different from the zero pair agreed on the exact span agreement.

To summarize, due to the verbosity of PICO elements in medical literature, annotations made by annotators have diverse boundaries. As the exact span agreement requires matching boundaries as well as labels, the low level of agreement is estimated for PICO span annotations. However, the relaxed span agreements reflect the characteristics of PICO elements and show greater agreement by allowing unmatched boundaries.

Cross Group Agreement. The finding, the low agreement in exact span agreement and the high agreement in relaxed span agreement, is observed between annotators who share similar understanding about domain knowledge (*i.e.*, MTurk-MTurk or Expert-Expert pairs). In this section, we study the annotation agreement when annotators have different levels of domain knowledge, by measuring annotation agreement between a MTurk worker and a medical expert (*i.e.*, MTurk-Expert). Furthermore, we study differences in annotations between by MTurk workers and by medical experts in terms of lengths of span annotations.

On the 200 documents having annotations by both MTurk workers and medical experts, we measure the cross-group agreement. Specifically, we consider annotations by medical experts as gold standard annotations, and then estimate Precision, Recall, and $F1$ values for annotations by MTurk workers as predicted annotations. As each document has annotations from multiple MTurk and expert annotators, values derived by all possible MTurk-Expert pairs are averaged for a document.

Table 6.3 presents Precision/Recall/ $F1$ values of cross-group agreement averaged on the 200 documents, with the three agreement types. In Table 6.3, the cross-group agreements between

Table 6.4: Comparison of the length of span annotations. The average (standard deviation in parenthesis) number of tokens are presented within MTurk workers on 5000 documents (MTurk-5000), and on 200 documents (MTurk-200) and medical experts on 200 documents (Expert-200).

Label	MTurk-5000	MTurk-200	Expert-200
P	9.355(10.406)	8.268 (9.113)	6.356(6.500)
I	4.356 (7.999)	3.322 (5.743)	1.903(2.075)
O	6.792(16.257)	6.134 (14.822)	4.379(5.347)

MTurk workers and medical experts present the similar trend shown in within-group agreement (Table 6.2). The exact agreement is low and the relaxed agreements are much higher. Based on the results of cross-group agreement as well as within group agreement, we make the third observation as follows.

Observation 6.1.3 *Our finding, the high agreement on the general areas of PICO annotations with unmatched boundaries, is consistent regardless of domain knowledge that annotators have.*

Besides, an interesting finding is that when the agreement changes from exact to relaxed, Recall increases to greater than 0.7 in the TO agreement while the improvement in precision is relatively small. This shows that the annotations by MTurk workers are a ‘superset’ of the annotations by medical experts in relaxed span agreements (high recall and low precision). Another finding is that on each agreement definition, the agreement in MTurk-Expert pair is always higher than the agreement in MTurk-MTurk pair, and lower than the agreement in Expert-Expert pair, comparing $F1$ values reported in both Tables 6.2 and 6.3. It shows that some MTurk workers are capable of annotating PICO elements similar to medical experts.

Next, we study differences in annotations made by the two groups in terms of the number of tokens in each span annotation. Table 6.4 shows the average number of tokens and its standard deviation in each label of spans for the 5000 documents by MTurk workers, and for the 200 documents by MTurk workers and by medical experts. Observe that the number of tokens in annotations by medical experts is smaller than the number of tokens in annotations by MTurk workers. We believe the difference in the length of spans is attributed to medical domain knowledge. Medical experts make more specific and short annotation spans for being able to identify essential information. Moreover, the annotations by MTurk workers show higher standard deviations than by experts.

Table 6.5: Performance of BiLSTM-CRF in PICO span recognition evaluated by the exact and relaxed agreements. Precision/Recall/ $F1$ values are estimated for each element label (P, I, O) and Micro-averaged value (Micro) for all labels.

Eval	Exact span			One-side Boundary			Token Overlap		
Label	Pre	Rec	$F1$	Pre	Rec	$F1$	Pre	Rec	$F1$
P	0.227	0.205	0.216	0.766	0.692	0.727	0.840	0.758	0.797
I	0.465	0.283	0.352	0.792	0.481	0.599	0.835	0.508	0.632
O	0.406	0.276	0.329	0.790	0.538	0.640	0.838	0.571	0.679
Micro	0.387	0.267	0.316	0.785	0.541	0.640	0.837	0.577	0.683

6.1.3 PICO Span Recognition

Our evaluation shows that the exact span agreement is low and the relaxed span agreements are much higher, among human annotators, regardless within experts, within non-experts, or cross-group. In this section, we demonstrate how the exact and relaxed span agreements can be used for the evaluation of PICO span recognition task. We also show how differently the exact and relaxed span agreements present the quality of performance.

The original EBM-PICO dataset paper also conducted a PICO recognition task. The performance is reported by token-level evaluation [73]. That is, boundaries are not evaluated since they are predetermined as a token separation, and also the evaluation instances in token-level evaluation are more than that of span evaluation. In this work, we conduct the same task as [73], but evaluate its performance on *span* evaluation, specifically, the exact, OB, and TO span agreements.

For PICO span recognition, Bi-directional LSTM-CRF (BiLSTM-CRF) model is used. We follow the same experiment settings and train/validation/test data splits of the aggregated annotations (see section 6.1.1) used in [73].⁴ Table 6.5 presents performance evaluated by the exact span agreement, and OB and TO relaxed span agreements. Low performance on the exact span agreement is observed, as it is challenging even for human annotators. Performance evaluated by the two relaxed span agreements shows significant improvement. The values are even comparable to the results reported in Tables 6.2 and 6.3.

⁴From data exploration, it is found that about 10% of tokens have multiple labels in the aggregated annotations. Before training a model, we resolve the multiple labels of tokens into a single label with the priority order of I, P, and O. We believe I element is the most important element among the three elements, followed by P and O.

Tokens	Efficacy	and	safety	of	selamectin	against	fleas	and	heartworms	in	dogs	and	cats	presented	as	veterinary	patients	in	North	America	.
Gold standard	O	N	O	N	I	N	N	N	N	N	P	P	P	P	P	P	P	P	P	P	N
Predicted label	O	O	O	N	I	N	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P

Fig. 6.2: Examples of correct PICO spans and predicted PICO spans in a PICO span recognition task. The example sentence is from a medical document PMID: 10940525 in EBM-PICO dataset. N indicates a token without PICO label.

Figure 6.2 shows examples of correct spans and predicted spans in the recognition task. There are three predicted spans (*i.e.*, O, I and P in the order). Indeed, the model correctly makes predictions on the general areas of correct spans. However, boundaries of the predicted spans are incorrect except I element prediction. Hence, by the exact span agreement, the only I predicted span is a correct prediction and the other two are incorrect predictions. On the other hand, in the OB and TO agreements, 2 and 3 out of the predicted spans are considered correct predictions, respectively.

6.1.4 Summary

We report observations made from agreements in PICO span annotations. The exact span agreement presents very low-level of agreement but the two relaxed span agreements show high-level of agreements in human annotations. The result shows that even though boundaries of PICO annotations are unmatched, the annotations are in the similar areas in general. Based on our observations, we argue that the evaluation of PICO span-related tasks shall consider not only the exact span agreement but also the relaxed span agreements, because even human annotators do not agree on exact spans due to the high verbosity of PICO elements.

6.2 Stability of Medical Concept Embeddings

Medical concepts are medical terminologies linked to Unified Medical Language System (UMLS)⁵. They are frequently used in medical domain for accurate and effective communication. Using medical concepts identified by unique IDs (*e.g.*, CUI in UMLS) helps avoid misunderstanding and allows to utilize semantic information in UMLS.

⁵UMLS is the metathesaurus in medical domain which provides an unified language system over various medical dictionaries such as ICD-10, MeSH, and SNOMED-CT. More details are in <https://www.nlm.nih.gov/research/umls/>

Word embeddings encode syntactic and semantic aspects of words into low-dimensional and dense representations. As they have shown great advance, many studies have proposed approaches to learning medical concept embeddings using various biomedical resources.

Several studies have shown the instability of word embeddings in general domain [112–115]. Whenever word embeddings are trained on the same training corpus with the same parameter setting, encoded relations among words should be consistent in each time. It is reported however that word embeddings fail to learn consistent relations, shown by using a stability measure (more details later).

The instability of embeddings poses greater problems in medical domain. Unstable medical concept embeddings result in a wide-range of negative consequences from unsuccessful document filtering to misdiagnosis. While medical concept embeddings are of importance, their stability has not been carefully investigated. In this work, we focus on medical concept embeddings and examine their stability. We use ‘word embedding’ and ‘(medical) concept embedding’ interchangeably throughout this work.

It is a well-known premise that frequency is one of the major factors contributing to quality word embeddings. As word embeddings are trained using word co-occurrence information, a high word frequency in a corpus provides sufficient training instances and thus helps obtain a quality word embedding. We conduct a detailed analysis on the relation between the stability of medical concept embeddings and concept frequency. The analysis reveals that some low-frequency concepts result in high stability as high-frequency concepts do. For example, the frequencies of two concepts, Anaesthesia procedure and Knee joint operation are 10,377 and 32 respectively in our dataset and both concepts have the same high level of stability 0.8. Our analysis results show that there are other factors that influence the stability of medical concept embeddings in addition to frequency.

Finally, we propose a new factor, *the noisiness of context words*, which affects the stability of medical concept embeddings. The rationale is: if the context words of concept are noisy, its embedding undergoes inconsistent training in various directions, leading to the low quality of embedding. We use a measure called *normalized entropy* to compute the noisiness of context words. It quantifies how flat or peaked a distribution of elements (context words in this context) is. When a word has a high frequency in a corpus, its number of context words is also high. Normalized entropy negates the effect of the total number of context words using normalization.

In this way, we measure solely the noisiness of context words without taking frequency into account.

In medical domain word2vec is widely adopted when training medical concepts embeddings. We focus on word2vec [105] as a representative among existing word embedding algorithms. In the experiments, we demonstrate the relation between the proposed factor and the stability of concept embeddings by calculating a linear correlation coefficient. The evaluation result shows moderate correlations. The correlations are clear and consistent with various groups of medical concepts. The proposed factor provides an empirical reasoning of how low-frequency concepts are able to result in high-stability embeddings, despite the small number of training instances. Besides, based on the moderate linear correlation, we make suggestions on ways to adjust the noisiness of context words. Lastly, we extend the evaluation to general-domain word embeddings and show the equivalent linear relation of the proposed factor.

6.2.1 Related Work

In this section, we first review word embedding algorithms, and then introduce training medical concept embeddings with different types of medical data. Next, we provide an overview of previous studies on the stability of word embeddings.

Word Embeddings. Word embeddings are a fundamental building block in NLP tasks. It represents syntactic and semantic meanings of a word into a low-dimensional vector of real values. There are several word embedding algorithms such as word2vec [105], GloVe [116], and FastText [117]. While details in training word embeddings are different in algorithms, they share the same hypothesis: words that occur in the same contexts tend to have similar meanings.

In this section, we focus on word2vec, particularly the skip-gram model with negative sampling [105], as it is a popular model to train embeddings in medical domain. It trains an embedding by using co-occurrences between a given word and its neighboring words (context words) that appear within a pre-defined window size. The objective of skip-gram model is to predict the context words given target word. Initially, word embeddings are randomly initialized. During the training, the skip-gram model find a set of word embeddings, which maximizes the objective function:

$$\frac{1}{T} \sum_{i=1}^T \sum_{-k \leq j \leq k, j \neq 0} \log P(w_{i+j} | w_i) \quad (6.1)$$

where k is a window size, and w_{i+j} indicates a context word for a given target word w_i in distance of j . The sign of j indicates a preceding or following direction of w_i . Whenever word embeddings are trained, the absolute values in word embeddings are different due to the random initialization of word embeddings by design. In other words, they are in different embedding spaces. However, they are supposed to encode the consistent meanings of words because they are trained on the same training corpus with the same parameter setting.

Contextual word embeddings (*e.g.*, BERT, ELMo) have advanced word representations in recent years. They necessitate computational power and large amounts of data to train and/or fine-tune thousands of parameters. In medical domain, these conditions are challenging to meet because a large scale of medical data (*e.g.*, clinical records) is often restricted to the public over privacy concerns. Thus, word embeddings are still frequently used tools to start off with. They are computationally inexpensive and fast, leading to faster prototyping and development. Moreover, word embeddings show effective performance even when a training corpus is small. Previous literature reports that using a small and topic-specific corpus is more effective than using a large and general corpus for domain-specific word embeddings [75, 76, 118].

Medical Concept Embeddings

Medical concepts (*i.e.*, medical terminologies linked to a knowledge base UMLS) are so ubiquitous that they are in diverse medical applications/resources, such as electronic health records, health insurance claims, and biomedical literature. Each medical resource provides additional information specialized in its own application, in addition to medical concepts. Previous studies have developed various approaches for effective learning of concept embeddings by leveraging extra information given by different resources.

Electronic health records (EHRs) are patients' visit records to hospitals. Each record consists of medical concepts related to given visit, as well as patient's demographic information and medical history. Records are also tagged with timestamps of visits. EHRs enable effective learning of concept embeddings by incorporating information shared by similar patients or temporal information, in addition to co-occurrences of concepts within records [119–121]. Health insurance claims are similar to EHRs and each claim includes medical concepts tagged with temporal information. However, the occurrences of claims tend to be sparse and irregular compared to EHRs. Sporadic claims make it challenging to learn the relatedness between medical concepts into embeddings. In [122], the authors have introduced grouping and shuffling techniques to mitigate the challenge.

Table 6.6: Examples of input text for training concept embeddings. Medical concepts are presented in square brackets. A word embedding algorithm such as word2vec is applied.

Original sentence	Calcium carbonate appears to be as effective as aluminum hydroxide in binding dietary phosphorus in hemodialysis patients.
cui2vec	[calcium_carbonate] appears to be as effective as [aluminum_hydroxide] in binding dietary [phosphorus] in [hemodialysis] patients.
NLM	[calcium_carbonate] [aluminum_hydroxide] [phosphorus] [hemodialysis]

Lastly, biomedical literature is in formal written language and there is a significant body of work to train concept embeddings on biomedical literature [123–125]. Unlike the previous two resources, it does not have pre-tagged medical concepts in it. To train concept embeddings, concepts must be identified first by using off-the-shelf tools such as MetaMap [126], Quick-UMLS [58]. The most simple and straightforward approach using biomedical literature is called (**cui2vec**) [127]. It considers each concept a single word and apply a word embedding algorithm to concepts and their surrounding non-medical words. Another similar approach (**NLM**) is to apply an algorithm to only concepts, eliminating non-medical words [128]. Table 6.6 presents example input texts of the two approaches. A word embedding algorithm is then applied to the input text to train concept embeddings.

Among a wide range of approaches for medical concept embeddings, in this work we use **cui2vec** and **NLM** to train medical concept embeddings. They are general enough to be adopted not only in biomedical literature but also in EHRs and insurance claims. For example, NLM can be directly applied for EHRs and insurance claims where each record/claim consists of a bag of medical concepts. Lastly, while EHRs and health insurance claims are often privately owned, biomedical literature is publicly available.

Instability of Word Embeddings

In recent years, there have been several studies reporting the instability of word embeddings in general domain. It is shown that when word embeddings are trained with the same corpus and hyperparameters but with random weight initialization in multiple times, the relations

among words, in particular the nearest neighbor words, are different across the sets of word embeddings. Thus, the word embeddings are considered unstable.

Existing studies have suggested several factors that influence the stability of word embeddings. They are classified into two groups: corpus-level factors and word-level factors. As corpus-level factors, in [114], the authors explore various values of a corpus size and different document lengths in the corpus. In [76, 113, 118], it is shown that a small and topic-specific corpus helps train more stable word embeddings than a large and general corpus.

As word-level factors, word frequency, part-of-speech (POS) tags, and concreteness of word's meaning are studied [113, 129]. In [113] the authors discover that words in some POS tags tend to have more stable embeddings than other tags. They also report that frequency is not a major factor on the stability of word embeddings. However, the authors do not further examine the reasons behind the finding.

Across corpus-level and word-level factors, most factors are related to word frequency either directly or indirectly. For example, in corpus-level factors, different sizes of corpus and topic specificity in a corpus incur the changes of word frequency. In word-level factors, part-of-speech tags are also related to word frequency, since some part-of-speech tags appear more frequently than other tags in a corpus.

6.2.2 Analysis

Frequency is one of the well-known factors that influence the quality of word embeddings. Fundamentally word embedding algorithms (*e.g.*, word2vec and GloVe) train word embeddings using co-occurrence information of words. A high frequency of a word in a corpus is equivalent to a large number of co-occurred words (*i.e.*, training instances), so that it helps produce a quality word embedding. In this section, we conduct a detailed analysis on the stability of medical concept embeddings with concept frequency in a training corpus. We first introduce the stability measure and describe the settings used in training medical concept embeddings. Then, we discuss the analysis.

6.2.2.1 Stability Measure

We use the stability measure to evaluate the quality of medical concept embeddings [113, 114]. It is initially proposed for word embedding in general domain. We first explain the stability in

the context of general-domain word embeddings. Next, we present how it is applied to medical concept embeddings.

A stability value represents how stable a word embedding is. It is the portion of overlapping words between the n nearest neighbors from different embedding spaces. Each embedding space is obtained by training word embeddings with the same training corpus and hyperparameters but with random initialization of embeddings and random sampling in a negative sampling technique, if it is used in word2vec.

Formally, given a word w and two embedding spaces, P and Q , let P_w and Q_w be the n nearest neighbors of w in P and Q based on embedding similarities, respectively. The stability value for word w 's embedding is the ratio of overlapping words in P_w and Q_w .

$$stability(w) = \frac{|P_w \cap Q_w|}{n} \quad (6.2)$$

If P_w and Q_w consist of the same set of words, $stability(w) = 1.0$. It demonstrates the embedding of w encodes consistent semantic information of w . Hence, w 's embedding is considered stable.

In our problem setting, we replace a word w with a medical concept c . The stability of concept embeddings is defined as the portion of overlapping nearest concepts from the three spaces P , Q , and R , i.e., $\frac{|P_c \cap Q_c \cap R_c|}{n}$, instead of two spaces. In doing so, the stability of medical concept embeddings is evaluated by stricter conditions, since often biomedical applications require a high level of accuracy. The number of the nearest neighbors is set to 10 ($n=10$), and the similarity between concept embeddings is calculated by cosine similarity [113].

6.2.2.2 Training Medical Concept Embeddings

There have been many studies on approaches for training medical concept embeddings with various biomedical data (details in Related Work). In this work, we examine two approaches, cui2vec [127] and NLM [128], using biomedical literature data.

Dataset. We use OHSUMED dataset to train medical concept embeddings. It consists of 348,566 abstracts sampled from a MEDLINE corpus. Though a MEDLINE corpus is much larger, multiple studies report that compared to a large and general corpus, a small and topic-specific corpus is more effective for embeddings in domain-specific applications [75, 76]. For medical concept identification, there are several tools available such as MetaMap [126], NCBO

Fig. 6.3: Distributions of words (left) and medical concepts (right) over frequencies in the OHSUMED dataset. The majority of concepts have low frequency.

BioPortal [130], PubTator [131], and QuickUMLS [58]. We use QuickUMLS because it is faster than other tools. Total 40,625 (unique) medical concepts are extracted.

The vast majority of concepts have low frequencies. Figure 6.3 shows a comparison between the frequency distributions of words (left) and the extracted concepts (right). As expected, word frequencies follow a power law distribution. The distribution of concept frequencies also shows that the very similar distribution of words. When we consider that a high frequency helps train quality word embedding, most concepts are likely to have low-quality embeddings due to their low frequency.

Finally, the original sentences are transformed with medical concepts, as shown in Table 6.6. A word embedding algorithm is applied and trains concept embeddings.

Word2vec setting. We use the skip-gram model with negative sampling implemented in gensim. The embedding size is set to 200 and the minimum frequency is set to 5. The window size for context words and the number of epochs are set to 7 and 50, respectively. The remaining hyperparameters are set to default values. After rare words are excluded using the minimum frequency threshold, total 25,491 concept embeddings are trained.

Different hyperparameter values affect the quality of word embeddings [105, 132, 133]. We explore different values for five hyperparameters: the context window size, epochs, the number of negative samples, the smoothing parameter for the negative sample distribution, and the subsampling rate. We examine the changes of embedding stability when different values are used in the training. However, the evaluation results show that they result in marginal changes and the average stabilities are very similar. We report the results at the end of Chapter 6 in Appendix.

Fig. 6.4: Distributions of medical concepts over frequency and the stability of embeddings trained by cui2vec and NLM in heatmaps. Total number of concepts is 25,491. Each cell indicates the number of medical concepts in the corresponding bin intersected by frequency and stability.

6.2.2.3 Analysis

So far, we have introduced the stability measure and the definition of stability used in this work. We have also described the setting in training concept embeddings. Now, we present a detailed analysis on the stability of concept embeddings with frequency and discuss their relations.

Figure 6.4 shows a distribution of medical concept embeddings over frequency and embedding stability, trained by cui2vec and NLM. In a subfigure, the x -axis indicates the ranges of concept frequency, and the y -axis indicates the stability from 0.0 to 1.0. Each cell denotes the number of medical concepts that belong to the corresponding frequency and stability bins. The color of cells shows the relative number of concepts.

As shown in Figure 6.4, high-frequency concepts tend to have high-stability concept embeddings. When a frequency is greater than 1,000, most concepts have stability higher than 0.5. The result confirms that indeed high frequency helps produce high stability embeddings with the sufficient training instances. Overall, cui2vec and NLM show similar distributional patterns.

We observe interesting stability distributions among low-frequency concepts. The stability of low-frequency concepts (frequency < 100) varies widely from 0.0 to 1.0. They end up with the high stability embeddings even with the much smaller number of training instances than high-frequency concepts. For example, here are two medical concepts, Anaesthesia procedure

Fig. 6.5: Distributions of low-frequency concepts (in the first row) and commonly used concepts among low-frequency concepts (in the second row) over frequency and the stability of embeddings.

(CUI: C0002903) and Knee joint operation (CUI: C0187769), with drastically different frequencies. Anaesthesia procedure has top 10% frequency (frequency: 10,377) and Knee joint operation has bottom 10% frequency (frequency: 32) in the corpus. However, both result in the very high stability 0.8 in cui2vec and NLM.

We take a close look at low-frequency concepts. We zoom into the first two columns in Figure 6.4, and present the distributions of low-frequency concepts alone in Figure 6.5. We also provide the distributions of commonly used concepts among low-frequency concepts in Figure 6.5. In medical domain, concepts of diseases, symptoms, treatments, and diagnostic tests are commonly looked up by medical professionals [134]. Among all concepts (Figure 6.4), a large portion of concepts has a frequency lower than 100, and they are also important concepts (Figure 6.5). In Figure 6.5, more than a half of concepts have the stability greater than 0.5. Some low-frequency concepts have the stability even greater than 0.8.

We compute Pearson correlation coefficient (ρ) between concept frequency and embedding stability of all concepts. The correlation coefficient is 0.189 and 0.200 in cui2vec and NLM,

respectively. Frequency shows a weak linear correlation with the embedding stability, because low-frequency concepts also have high stability as high-frequency concepts do.

Summary. In this section, we examine the stability of medical concept embeddings with frequency. High-frequency concepts have high-stability embeddings, showing that a high frequency indeed helps obtain quality embeddings with sufficient training instances. More importantly, we observe that surprisingly low-frequency concepts also have high-stability embeddings, despite the smaller number of training instances. The high-stability embeddings of low-frequency concepts are not aligned with the common understanding of the impact of frequency. This results demonstrate that there might be other factors influencing the stability of concept embeddings in addition to frequency.

6.2.3 Proposed Measure

We propose a new factor, *the noisiness of context words*, which influences the stability of concept embeddings. Our hypothesis is as follows: if a concept has noisy context words, its embedding will undergo inconsistent learning, leading to a low-quality embedding. Conversely, if a concept has coherent context words, its embedding will go through consistent training, and thus it results in a high-quality embedding. We note that the proposed factor focuses on the distribution of context words only without considering the total number of context words, which is dependent on frequency. In the following discussion, we use ‘concept’ and ‘word’ interchangeably. The idea is the same.

To measure the noisiness of context words, we use *normalized entropy*. Normalized entropy quantifies a distribution of elements (*i.e.*, context words in our context). It negates the effect of the total number of context words in the estimate using normalization. Formally, for a given word w , normalized entropy $H(w)$ is computed as follows.

$$H(w) = - \sum_{w_i \in C_w} \frac{P(w_i) \cdot \log P(w_i)}{\log |C_w|} \quad (6.3)$$

In the above equation, C_w denotes a set of context words that co-occur with word w within the window size. $P(w_i)$ is the relative frequency of context word w_i in the collection of all context words. If context words of a target word are evenly distributed (flat distribution), normalized entropy is high; if a few context words account for a large portion in the entire context words (skewed distribution), the value is low.

Table 6.7: Pearson correlation coefficient of the noisiness of context words with the stability of medical concepts. We present three groups of medical concepts: all concepts, concepts with low-frequency (< 100) (Low-freq), and commonly used concepts in biomedical applications among low-frequency concepts (Low-freq & Common). All correlation coefficients are statistically significant at $p < 0.001$.

	All concepts	Low-freq	Low-freq & Common
cui2vec	-0.593	-0.540	-0.549
NLM	-0.477	-0.370	-0.329

We evaluate the proposed factor by computing a linear correlation coefficient (Pearson correlation coefficient) with the stability of concept embeddings. Table 6.7 presents coefficient values calculated in three groups of medical concepts: all concepts, low-frequency (frequency < 100) concepts, and commonly used concepts among low-frequency concepts. For all concepts, the noisiness of context words shows a moderate linear correlation with the stability of concept embeddings. In general, the correlation is stronger in cui2vec than in NLM.

As shown in Figures 6.4 and 6.5, while high-frequency concepts tend to have high stability, low-frequency concepts show varied the embedding stabilities. To better understand the surprising stability of low-frequency concepts, we look into low-frequency concepts and conduct a focused evaluation on them.

In Table 6.7, the proposed factor consistently shows a moderate linear correlation with the stability of low-frequency concepts. The correlation in cui2vec is even comparable to the correlation estimated for all concepts. In NLM, the correlation is slightly decreased, compared to the correlation for all concepts. The similar correlations are observed for commonly used concepts as well. Lastly, the proposed factor shows much stronger correlations with the stability of embeddings, compared to frequency as shown in Figures 6.4 and 6.5. All correlation coefficients are statistically significant at the p -value smaller than 0.001.

Summary. In this section, we propose a new factor, the noisiness of context words, which influences the stability of medical concept embeddings. We use normalized entropy to estimate the proposed factor. The evaluation result shows a clear linear correlation between the proposed factor and the stability of medical concept embeddings. The correlations are consistent for all concepts, for low-frequency concepts, and for commonly used low-frequency concepts. This result supports the claim that when a concept has coherent context words, it is likely to result in

high-quality embeddings, regardless of high or low frequency in a corpus. Moreover, the proposed factor provides an empirical reasoning of the surprising high stability of low-frequency concepts.

6.2.4 Discussion

6.2.4.1 Suggestions on Improving the Stability

We have shown the negative linear correlation in the evaluation. It indicates that decreasing the noisiness of context words improves the stability of medical concept embeddings. In this section, we make suggestions on ways to adjust the noisiness of context words, while maintaining the semantic meanings of concepts implied in a training corpus.

The first approach is to utilize hierarchical relations of medical concepts from a knowledge base. All medical concepts are linked to UMLS, and UMLS provides hierarchical relations among medical concepts. Hierarchical relations are a tree-like structure and connections indicate a parent-child (IsA) relation between concepts. In a parent-child relation, while the meaning of child concept is more specific than the meaning of parent concept, their meanings are greatly overlapped with each other. For example, Knee replacement (CUI: C0086511) is a child concept of Knee joint operation (CUI: C0187769) in UMLS, and the meaning is knee joint operation for replacement. The noisiness of context words of low-stability concepts can be adjusted using the context words of parent and/or child concepts.

Another approach to decreasing the noisiness of context words is segregating a training corpus into smaller document clusters. Context words tend to be more consistent in topic-specific document corpus and topic-specific sub-collection can be identified using document clustering techniques. Clustered documents provide information representing the importance of context words, depending on which documents they appear. This information can be utilized to filter the context words.

6.2.4.2 Generalization

We have evaluated the proposed factor with concept embeddings trained with medical textual data. In this section, we conduct an experiment with embeddings trained on text from general domain. We use two datasets: Reuters-21578 (Reuters) and Wikipedia abstract (Wiki). Reuters

consists of news articles under pre-defined topical categories. Wiki is much larger than Reuters and contains various topics.

In general domain, word embeddings are more prevalent than concept/entity embeddings. We train word embeddings without concept identification, and all trained word embeddings are subject to evaluations. Likewise, the skip-gram model with negative sampling is used. After the training, every word that has a frequency greater than minimum frequency threshold (set to 5) in a dataset has a word embedding.

Pearson correlation coefficients on Reuters and Wiki are -0.412 and -0.495 , respectively. The proposed factor shows the moderate correlations with the stability of word embeddings in both datasets. The correlation values are statistically significant by $p < 0.001$. This result demonstrates that the linear relation of the proposed factor with the stability is not limited to medical-domain text, and it extends to general-domain text.

6.2.5 Summary

We analyze the stability of medical concept embeddings with respect to frequency. The analysis shows that low-frequency concepts can achieve high stability even though there is the limited number of training instances for them. Motivated by the surprising high stability of low-frequency concepts, we propose a new factor, the noisiness of context words, influencing the stability of medical concept embeddings. We measure the noisiness of context words using normalized entropy. The evaluation result shows a clear correlation between the proposed factor and the stability of medical concept embeddings. The result is consistent for all concepts, low-frequency concepts, and commonly used low-frequency concepts. This work helps understand how low-frequency concepts can result in high-stability embeddings like high-frequency concepts do.

6.2.6 Appendix

In the skip-gram model, different hyperparameters affect the quality of word embeddings. We evaluate the proposed factor on multiple sets of word embeddings that are trained with different hyperparameter values. We test five hyperparameters: window size (W), epoch (E), number of negative samples in negative sampling (N), smoothing parameter for negative sample distribution in negative sampling (M), and subsampling rate (S). Default values for W, E, N, M, and

Table 6.8: Average (standard deviation) of the stability of medical concept embeddings when different hyperparameter values are set in the training of word embeddings

Hyperparameter	W:E:N:M:S	Avg (stdev) of stability	
		cui2vec	NLM
<i>Default</i>	<i>7:50:5:0.75:0.001</i>	<i>0.595 (0.208)</i>	<i>0.627 (0.181)</i>
Window size(W)	5	0.595 (0.205)	0.636 (0.177)
	10	0.589 (0.210)	0.611 (0.187)
Epoch (E)	30	0.564 (0.226)	0.622 (0.189)
	100	0.619 (0.197)	0.622 (0.182)
Number of NS (N)	10	0.618 (0.203)	0.656 (0.177)
	15	0.625 (0.202)	0.669 (0.175)
Smoothing (M)	0.0	0.579 (0.229)	0.649 (0.179)
	1.0	0.568 (0.209)	0.611 (0.185)
Subsampling rate(S)	0.01	0.604 (0.206)	0.632 (0.181)
	0.0001	0.579 (0.208)	0.609 (0.18)

S are 7, 50, 5, 0.75, and 0.001, respectively. For each hyperparameter, two additional values are tested, while the rest hyperparameters are the default values. Likewise, we train medical concept embeddings using the two approaches, cui2vec and NLM.

Table 6.8 presents the average and the standard deviation of the stability. Marginal changes are observed in both cui2vec and NLM when different hyperparameter values are used.

Table 6.9 shows Pearson correlation coefficient of the proposed factor with the stability when embeddings are trained with different hyperparameter values. The correlation coefficients present small changes. This result makes sense that the stabilities are barely changed by the different hyperparameter values (Table 6.8)

Across the different hyperparameter values, the proposed factor shows a moderate linear correlation with the stability of medical concept embeddings. It is worth noting the correlations with the window size (W). The different values of W directly changes the noisiness of context words as the window size determines context words. When different window sizes are used in training of word embeddings, the proposed factor consistently shows the moderate correlations.

Table 6.9: Pearson correlation coefficients of the noisiness of context words with the stability of medical concept embeddings. Embeddings are trained by different hyperparameter values. The strongest correlation in cui2vec and NLM is indicated as *. All correlation values are statistically significant at $p < 0.001$.

Hyperparameters	W:E:N:M:S	Pearson coefficient	
		cui2vec	NLM
<i>Default</i>	<i>7:50:5:0.75:0.001</i>	-0.593	-0.477
Window size(W)	5	-0.577	-0.484
	10	-0.605	-0.470
Epoch (E)	30	-0.618*	-0.478
	100	-0.584	-0.501
Number of NS (N)	10	-0.564	-0.471
	15	-0.556	-0.469
Smoothing (M)	0.0	-0.519	-0.335
	1.0	-0.601	-0.506*
Subsampling rate (S)	0.01	-0.590	-0.484
	0.0001	-0.578	-0.454

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we propose a new ranking approach called seed-driven document ranking (SDR), which uses a seed document as a query in screening prioritization. While a seed document is a practical and effective resource as a query, the long document query can trouble existing retrieval models since short keyword queries are common inputs for them. Through the extensive data analysis, we have identified several critical challenges and developed novel ranking models as remedies to them. The proposed models not only address challenges, but also bring out the greater effectiveness than existing approaches.

In Chapter 1 *Introduction*, we provide the background knowledge for systematic reviews and the overview of SDR. Chapter 2 *Literature Review* introduces approaches for improving the screening process including screening prioritization and discusses existing studies in screening prioritization in particular. Next, over the four main chapters, we extensively investigate the seed-driven document ranking approach. In this chapter, we present the brief summary of the four main chapters and derive the conclusions of this thesis with the discussion on future work.

In Chapter 3 *Seed-driven Document Ranking*, we have proposed seed-driven document ranking approach for systematic reviews and established the foundation of this thesis. We provide the comprehensive analysis on relevant and irrelevant documents in SRs and demonstrate their characteristics. Specifically, we make two findings: (i) compared to irrelevant documents, relevant documents share the high intra-similarities. (ii) when documents are represented in clinical terms, the intra-similarities of relevant documents further increase. By leveraging the domain-specific knowledge, we propose a weight function for clinical terms which combines with a

query likelihood retrieval model. The superior performances of the proposed approach and the combined approach with word embedding show the effectiveness of incorporating the domain-specific understanding in the screening prioritization task.

In Chapter 4 *Ranking by Document Matching* we have studied a document matching approach for SDR in which we rank candidate documents according to their matching scores to a seed document. In order to become a relevant document, a document must be matched to a seed in terms of PICO elements. But simple occurrence-based term matching is unable to achieve this. We investigate PICO elements in medical literature and reveal the positional patterns in the PICO elements in abstract texts. To this end, we propose a document matching model named M-Match which matches terms based on their similar positions. In doing so, we seek to achieve PICO-based term matching in M-Match. Experiments demonstrate the high effectiveness of M-Match in ranking many relevant documents at top- k positions. This strength is particularly useful in the beginning of manual screening process, since it enables multiple relevant documents to be available as seed documents within a short time. They can help further update the rankings of remaining documents.

In Chapter 5 *Ranking Performance Prediction*, we investigate the performance of ranking models on SRs. In other words, we analyze performance on individual SRs (local performance) of ranking models. Our analysis finds that despite slight difference on performance by different ranking models, some SRs constantly hit the lower performance than other SRs regardless of ranking models. This finding describes that there might be inherent characteristics in SRs which make some SRs more difficult than other SRs in having high ranking performance. We propose a topic broadness as such a characteristic of SRs. The evaluation shows that the estimated high topic broadness correlates with low ranking performance. All in all, one main takeaway of this chapter is: depending on the topic broadness, some SRs are more difficult to achieve effective ranking qualities than other SRs, and these difficult SR can be identified by the topic broadness.

In Chapter 6 *Methods to Enhance Ranking*, we explore methods that have potential to enhance the ranking performance on difficult SRs and also to increase the global performance of ranking models in general. At first we study PICO span agreement. As the very first step of developing automatic PICO recognition models, we investigate the agreement of PICO span annotations, motivated by the descriptive characteristics of PICO elements. We reveal that PICO span boundaries are not agreed among annotators, even among medical experts. To this end, we

claim that models trained for the PICO recognition task be evaluated by adjusted boundary definitions, since humans do not agree on the PICO boundaries and machine learning models learn from the data created by us. Next, we study the stability of medical concept embeddings. We report that unlike the common understanding, not all infrequent medical concepts have the low stability of embeddings. Some low-frequency concepts achieve the high stability embeddings as high-frequency concepts do. We propose the noisiness of context words as a factor affecting the stability of embeddings. The evaluations demonstrate that medical concepts with a low level of the noisiness of context words achieve the high stability of embeddings, despite their low frequencies. This finding helps us better understand how to train stable vector representations for medical concepts.

In conclusion, seed-driven document ranking is a practical approach established on the understanding and consideration of the real-world aims and rules in systematic reviews. In this thesis, we propose to utilize available relevant documents as a query and focus on developing ranking models effective for document queries. While ensuring finding all relevant documents, it enables the effective screening process which results in saving important resources, such as time and money. We have walked through the multiple ranking models for SDR. We have also addressed the ranking performance on individual SRs, which is a critical aspect when a ranking model is adopted in a real-world application where local performance plays a critical role. Next, we discuss potential research directions for future work.

7.2 Future Work

Incorporating PICO elements in ranking models. PICO elements play a crucial role in systematic reviews. As relevant documents are determined by PICO elements, PICO elements are involved in the entire process of conducting a SR, from defining a clinical question (the first step) to extracting target information (the last step).

Especially, in the screening step, identifying relevant documents is to check whether PICO elements in documents are matched to PICO elements defined in relevance conditions. In SDR, this is equivalent to check whether the PICO elements in documents is the same as those in a seed document. Thus, incorporating explicit PICO elements can produce a significant gain in the ranking performance. As a supporting evidence, Chapter 4 has shown that considering

the positional patterns of PICO elements improves the ranking effectiveness, compared to the approaches that do not utilize PICO elements.

Prior to developing ways to properly employ PICO elements in ranking models, PICO elements need to be effectively and efficiently identified. Chapter 6.1 has pointed out one obstacle in recognizing PICO elements: determining the explicit boundaries of PICO elements is not straightforward. Alternatively, semantic parsing approaches can be adopted so that PICO elements are recognized based on pre-defined parsed clauses.

Developing stable neural ranking models. For the past years, neural network models have shown the remarkable advance in natural language processing and information retrieval. Comparatively, neural network models are less explored in screening prioritization. We have examined several neural IR models and reported the drastic change in their performance in Chapter 4. We have briefly discussed the potential causes and remedies. In addition, there is room for further studying neural models and developing them for screening prioritization. We present a few directions for the development of stable neural ranking models.

In Chapter 4, the neural models have shown the unstable ranking performance, depending on dataset splits. This limitation is partly due to the lack of sufficient data as discussed. However, the fundamental problem is to effectively represent the different relevance definitions in SRs. For instance, due to different relevance conditions in two SRs, a document is a relevant document for a systematic review A, while it is not for a systematic review B. When systematic review A is in training dataset and systematic review B is in test dataset, the prediction on B is likely to be incorrect.

Next, the representation space for negative instances is inconsistent. Neural network models aim to learn the difference between positive (relevant documents) and negative (irrelevant documents) instances. Depending on SRs, the relations between relevant and irrelevant documents are various. For example, in some SRs, candidate documents tend to be similar to each other so that relevant and irrelevant documents are similar as well. By contrast, in other SRs, diverse documents constitute candidate document collections, relevant and irrelevant document are quite distinct. When we use irrelevant documents from multiple SRs as negative instances, neural models need to encode varied distances between positive and negative instances.

Lastly, embeddings are the most fundamental building block in neural network models. As discussed in Chapter 6.2, word/concept embeddings are unstable when they have noisy con-

text words. The high stability of embeddings is necessary for stable and reproducible model performance.

Extension to other domain-specific problems. Finding all existing relevant documents is an critical requirement in various domain-specific problems. In this section, we introduce several domain-specific problems where 100 percent recall is a priority, similar to systematic reviews. We present existing studies in other domains and discuss how seed-driven document ranking approach is applied.

In legal domain, when new cases or patents are filed, finding similar documents plays a crucial role. For instance, when a new patent is prepared, it must be distinguished from similar existing patents. Since failures to finding similar patents can lead to undesirable consequences, achieving 100 percent of recall is a top priority. Existing literature have tackled this challenge using various approaches such as retrieval models, document classification, and information extraction [52, 135, 136]. Similar to seed-driven document ranking, the authors in [52] have proposed a retrieval model for query by document with keyword extraction techniques and showed a document query is effective in finding similar legal documents.

Similar to medical literature, legal documents have various segments or functional parts like Fact, Argument, and Final Judgment. In Chapter 4, we have proposed M-Match that incorporates the structure of documents. We believe our model or similar approaches that leverage the structural patterns in legal documents can help the effective legal document search as well.

Second, when researchers build test collections in information retrieval, it is aimed to include all relevant documents in collections for objective model evaluation. Several studies have proposed approaches for high recall [137, 138]. A recent work [138] has employed an active learning framework in collecting relevant documents, and verified that adding labeled documents into model training helps enhance the retrieval performance further. This framework can be viewed as an equivalent approach in active learning setting in screening process in SRs (discussed in Chapter 2), except retrieval and screening steps are combined.

List of Publications

International Conferences

1. **Grace E. Lee** and Aixin Sun. Seed-driven Document Ranking for Systematic Reviews in Evidence-based Medicine, Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (**SIGIR 2018**), 455–464, 2018.
2. **Grace E. Lee** and Aixin Sun. A Study on Agreement in PICO Span Annotations, Proceedings of the 42nd International ACM SIGIR Conference on Research & Development in Information Retrieval (**SIGIR 2019**), 1149–1152, 2019.

International Journals

1. **Grace E. Lee** and Aixin Sun. Understanding the Stability of Medical Concept Embeddings, Journal of the Association for Information Science and Technology (**JASIST**) 2020.

References

- [1] V. B. Allen, K. S. Gurusamy, Y. Takwoingi, A. Kalia, and B. R. Davidson, “Diagnostic accuracy of laparoscopy following computed tomography (ct) scanning for assessing the resectability with curative intent in pancreatic and periampullary cancer,” *Cochrane Database Syst Rev*, vol. 11, 2013.
- [2] H. Scells, G. Zuccon, B. Koopman, A. Deacon, L. Azzopardi, and S. Geva, “Integrating the framing of clinical questions via pico into the retrieval of medical literature for systematic reviews,” in *CIKM*, 2017, pp. 2291–2294.
- [3] E. Kanoulas, D. Li, L. Azzopardi, and R. Spijker, “Clef 2017 technologically assisted reviews in empirical medicine overview,” in *Working Notes of CLEF*, 2017.
- [4] B. C. Wallace, J. Kuiper, A. Sharma, M. B. Zhu, and I. J. Marshall, “Extracting pico sentences from clinical trial reports using supervised distant supervision,” *JMLR*, vol. 17, no. 132, pp. 1–25, 2016.
- [5] I. J. Marshall, J. Kuiper, and B. C. Wallace, “Robotreviewer: evaluation of a system for automatically assessing bias in clinical trials,” *Journal of the American Medical Informatics Association*, vol. 23, no. 1, pp. 193–201, 2015.
- [6] S. R. Jonnalagadda, P. Goyal, and M. D. Huffman, “Automating data extraction in systematic reviews: a systematic review,” *Systematic reviews*, vol. 4, no. 1, p. 78, 2015.
- [7] H. Scells, G. Zuccon, B. Koopman, A. Deacon, L. Azzopardi, and S. Geva, “A test collection for evaluating retrieval of studies for inclusion in systematic reviews,” in *SIGIR*, 2017, pp. 1237–1240.

- [8] A. ÓMara-Eves, J. Thomas, J. McNaught, M. Miwa, and S. Ananiadou, “Using text mining for study identification in systematic reviews: a systematic review of current approaches,” *Systematic reviews*, 2015.
- [9] M. Gupta, M. Bendersky *et al.*, “Information retrieval with verbose queries,” *Foundations and Trends® in Information Retrieval*, vol. 9, no. 3-4, pp. 209–354, 2015.
- [10] S. Kugley, A. Wade, J. Thomas, Q. Mahood, A.-M. K. Jørgensen, K. Hammerstrøm, and N. Sathe, “Searching for studies: A guide to information retrieval for campbell,” 2016.
- [11] S. Morton, A. Berg, L. Levit, J. Eden *et al.*, *Finding what works in health care: standards for systematic reviews*. National Academies Press, 2011.
- [12] B. C. Wallace, T. A. Trikalinos, J. Lau, C. Brodley, and C. H. Schmid, “Semi-automated screening of biomedical citations for systematic reviews,” *BMC bioinformatics*, vol. 11, no. 1, p. 55, 2010.
- [13] B. C. Wallace, K. Small, C. E. Brodley, and T. A. Trikalinos, “Active learning for biomedical citation screening,” in *KDD*, 2010, pp. 173–182.
- [14] A. M. Cohen, W. R. Hersh, K. Peterson, and P.-Y. Yen, “Reducing workload in systematic review preparation using automated citation classification,” *Journal of the American Medical Informatics Association*, pp. 206–219, 2006.
- [15] A. M. Cohen, “An effective general purpose approach for automated biomedical document classification,” in *AMIA Annual Symposium Proceedings*, vol. 2006. American Medical Informatics Association, 2006, p. 161.
- [16] O. Frunza, D. Inkpen, and S. Matwin, “Building systematic reviews using automatic text classification techniques,” in *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*. Association for Computational Linguistics, 2010, pp. 303–311.
- [17] T. Bekhuis and D. Demner-Fushman, “Screening nonrandomized studies for medical systematic reviews: a comparative study of classifiers,” *Artificial intelligence in medicine*, pp. 197–207, 2012.

REFERENCES

- [18] G. V. Cormack and M. R. Grossman, “Technology-assisted review in empirical medicine: Waterloo participation in clef ehealth 2017,” in *Working Notes of CLEF*, 2017.
- [19] A. M. Cohen, N. R. Smalheiser, M. S. McDonagh, C. Yu, C. E. Adams, J. M. Davis, and P. S. Yu, “Automated confidence ranked classification of randomized controlled trial articles: an aid to evidence-based medicine,” *Journal of the American Medical Informatics Association*, vol. 22, no. 3, pp. 707–717, 2015.
- [20] G. E. Lee, “A study of convolutional neural networks for clinical document classification in systematic reviews: Sysreview at clef ehealth 2017.”
- [21] G. Singh, I. Marshall, J. Thomas, and B. Wallace, “Identifying diagnostic test accuracy publications using a deep model.”
- [22] K. Hashimoto, G. Kontonatsios, M. Miwa, and S. Ananiadou, “Topic detection using paragraph vectors to support active learning in systematic reviews,” *Journal of biomedical informatics*, pp. 59–65, 2016.
- [23] T. Bekhuis and D. Demner-Fushman, “Towards automating the initial screening phase of a systematic review.” in *MedInfo*, 2010, pp. 146–150.
- [24] A. M. Cohen, “Optimizing feature representation for automated systematic review work prioritization,” in *AMIA annual symposium proceedings*, 2008, p. 121.
- [25] B. K. Olorisade, E. de Quincey, P. Brereton, and P. Andras, “A critical analysis of studies that address the use of text mining for citation screening in systematic reviews,” in *EASE*, 2016, p. 14.
- [26] P. Timsina, J. Liu, and O. El-Gayar, “Advanced analytics for the automation of medical systematic reviews,” *Information Systems Frontiers*, vol. 18, no. 2, pp. 237–252, 2016.
- [27] M. Miwa, J. Thomas, A. O’Mara-Eves, and S. Ananiadou, “Reducing systematic review workload through certainty-based screening,” *Journal of biomedical informatics*, pp. 242–253, 2014.

- [28] S. Choi, B. Ryu, S. Yoo, and J. Choi, “Combining relevancy and methodological quality into a single ranking for evidence-based medicine,” *Information Sciences*, vol. 214, pp. 76–90, 2012.
- [29] J. G. Adeva, J. P. Atxa, M. U. Carrillo, and E. A. Zengotitabengoa, “Automatic text classification to support systematic reviews in medicine,” *Expert Systems with Applications*, vol. 41, no. 4, pp. 1498–1508, 2014.
- [30] T. Bekhuis, E. Tseytlin, K. J. Mitchell, and D. Demner-Fushman, “Feature engineering and a proposed decision-support system for systematic reviewers of medical evidence,” *PloS one*, p. e86277, 2014.
- [31] K. R. Felizardo, N. Salleh, R. M. Martins, E. Mendes, S. G. MacDonell, and J. C. Maldonado, “Using visual text mining to support the study selection activity in systematic literature reviews,” in *ESEM*. IEEE, 2011, pp. 77–86.
- [32] K. R. Felizardo, G. F. Andery, F. V. Paulovich, R. Minghim, and J. C. Maldonado, “A visual analysis approach to validate the selection review of primary studies in systematic reviews,” *Information and Software Technology*, vol. 54, no. 10, pp. 1079–1091, 2012.
- [33] M. Khabsa, A. Elmagarmid, I. Ilyas, H. Hammady, and M. Ouzzani, “Learning to identify relevant studies for systematic reviews using random forest and external information,” *Machine Learning*, vol. 102, no. 3, pp. 465–482, 2016.
- [34] S. Choi, B. Ryu, S. Yoo, and J. Choi, “Combining relevancy and methodological quality into a single ranking for evidence-based medicine,” *Information Sciences*, pp. 76–90, 2012.
- [35] D. Martinez, S. Karimi, L. Cavedon, and T. Baldwin, “Facilitating biomedical systematic reviews using ranked text retrieval and classification,” in *ADCS*, 2008, pp. 53–60.
- [36] X. Ji, A. Ritter, and P.-Y. Yen, “Using ontology-based semantic similarity to facilitate the article screening process for systematic reviews,” *Journal of biomedical informatics*, vol. 69, pp. 33–42, 2017.

- [37] G. M. D. Nunzio, “A study of an automatic stopping strategy for technologically assisted medical reviews,” in *ECIR*, 2018, pp. 672–677.
- [38] G. V. Cormack and M. R. Grossman, “Engineering quality and reliability in technology-assisted review,” in *SIGIR*, 2016, pp. 75–84.
- [39] Y. Tsuruoka, M. Miwa, K. Hamamoto, J. Tsujii, and S. Ananiadou, “Discovering and visualizing indirect associations between biomedical concepts,” *Bioinformatics*, pp. i111–i119, 2011.
- [40] C. Blake and A. Lucic, “Automatic endpoint detection to support the systematic review process,” *Journal of biomedical informatics*, pp. 42–56, 2015.
- [41] B. C. Wallace, J. Kuiper, A. Sharma, M. B. Zhu, and I. J. Marshall, “Extracting pico sentences from clinical trial reports using supervised distant supervision,” *Journal of Machine Learning Research*, pp. 1–25, 2016.
- [42] A. M. Cohen, K. Ambert, and M. McDonagh, “Studying the potential impact of automated document classification on scheduling a systematic review update,” *BMC medical informatics and decision making*, p. 33, 2012.
- [43] H. Scells, G. Zuccon, A. Deacon, and B. Koopman, “Qut ielab at clef ehealth 2017 technology assisted reviews track: Initial experiments with learning to rank,” in *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, vol. 1866. CEUR Workshop Proceedings, 2017, pp. Paper–98.
- [44] A. Alharbi and M. Stevenson, “Ranking abstracts to identify relevant evidence for systematic reviews: The university of sheffield’s approach to clef ehealth 2017 task 2,” in *Working Notes of CLEF*, 2017.
- [45] L. Goeuriot, H. Suominen, L. Kelly, E. Kanoulas, A. Névéol, G. Zuccon, and J. Palotti, “Clef 2017 ehealth evaluation lab overview,” in *CLEF*, 2017.
- [46] M. Lease, G. V. Cormack, A. T. Nguyen, T. A. Trikalinos, and B. C. Wallace, “Systematic review is e-discovery in doctor’s clothing,” in *MedIR workshop, SIGIR*, 2016.

- [47] Y. Yang, N. Bansal, W. Dakka, P. Ipeirotis, N. Koudas, and D. Papadias, “Query by document,” in *WSDM*, 2009, pp. 34–43.
- [48] Y. Lv, T. Moon, P. Kolari, Z. Zheng, X. Wang, and Y. Chang, “Learning to model relatedness for news recommendation,” in *WWW*, 2011, pp. 57–66.
- [49] Y. Kim, J. Seo, W. B. Croft, and D. A. Smith, “Automatic suggestion of phrasal-concept queries for literature search,” *Information Processing & Management*, pp. 568–583, 2014.
- [50] Y. Kim and W. B. Croft, “Diversifying query suggestions based on query documents,” in *SIGIR*, 2014, pp. 891–894.
- [51] L. Weng, Z. Li, R. Cai, Y. Zhang, Y. Zhou, L. T. Yang, and L. Zhang, “Query by document via a decomposition-based two-level retrieval approach,” in *SIGIR*, 2011, pp. 505–514.
- [52] Y. Kim and W. B. Croft, “Improving patent search by search result diversification,” in *ICTIR*, 2015, pp. 201–210.
- [53] R. J. Bayardo, Y. Ma, and R. Srikant, “Scaling up all pairs similarity search,” in *WWW*, 2007, pp. 131–140.
- [54] J. Lin, “Brute force and indexed approaches to pairwise document similarity comparisons with mapreduce,” in *SIGIR*, 2009.
- [55] T. Elsayed, J. J. Lin, and D. W. Oard, “Pairwise document similarity in large collections with mapreduce,” in *ACL*, 2008, pp. 265–268.
- [56] D. C. Blair *et al.*, “Information retrieval, cj van rijsbergen. london: Butterworths; 1979,” *Journal of the American Society for Information Science*, vol. 30, no. 6, pp. 374–375, 1979.
- [57] E. M. Voorhees, “The cluster hypothesis revisited,” in *SIGIR*, 1985, pp. 188–196.
- [58] L. Soldaini and N. Goharian, “Quickumls: a fast, unsupervised approach for medical concept extraction,” in *MedIR workshop, SIGIR*, 2016.

REFERENCES

- [59] C. Zhai and S. Massung, *Text data management and analysis: a practical introduction to information retrieval and text mining*. Morgan & Claypool, 2016.
- [60] E. Nalisnick, B. Mitra, N. Craswell, and R. Caruana, “Improving document ranking with dual word embeddings,” in *WWW*, 2016, pp. 83–84.
- [61] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, “From word embeddings to document distances,” in *ICML*, 2015, pp. 957–966.
- [62] A. Colli, J. C. Gana, D. Turner, J. Yap, T. Adams-Webber, S. C. Ling, and G. Casazza, “Capsule endoscopy for the diagnosis of oesophageal varices in people with chronic liver disease or portal vein thrombosis,” *Cochrane Database Syst Rev*, vol. 10, 2014.
- [63] N. Shaikh, J. Borrell, J. Evron, and M. Leeflang, “Procalcitonin, c-reactive protein, and erythrocyte sedimentation rate for the diagnosis of acute pyelonephritis in children,” *Cochrane Database Syst Rev*, vol. 1, 2011.
- [64] C. M. Williams, N. Henschke, C. G. Maher, M. W. van Tulder, B. W. Koes, P. Macaskill, and L. Irwig, “Red flags to screen for vertebral fracture in patients presenting with low-back pain,” *Cochrane Database Syst Rev*, vol. 1, 2013.
- [65] K. S. Gurusamy, V. Giljaca, Y. Takwoingi, D. Higgie, G. Poropat, D. Štimac, and B. R. Davidson, “Ultrasound versus liver function tests for diagnosis of common bile duct stones,” *Cochrane Database Syst Rev*, vol. 2, 2015.
- [66] N. I. S. O. (US) and A. N. S. Institute, *Guidelines for Abstracts: An American National Standard*. Niso Press, 1997, vol. 14.
- [67] G. E. Lee and A. Sun, “A study on agreement in pico span annotations,” in *SIGIR*, 2019, pp. 1149–1152.
- [68] K. D. Onal, Y. Zhang, I. S. Altingovde, M. M. Rahman, P. Karagoz, A. Braylan, B. Dang, H.-L. Chang, H. Kim, Q. McNamara *et al.*, “Neural information retrieval: At the end of the early years,” *Information Retrieval Journal*, vol. 21, no. 2-3, pp. 111–182, 2018.

- [69] N. Craswell, W. B. Croft, M. de Rijke, J. Guo, and B. Mitra, “Neural information retrieval: introduction to the special issue,” *Information Retrieval Journal*, vol. 21, no. 2-3, pp. 107–110, 2018.
- [70] J. Guo, Y. Fan, L. Pang, L. Yang, Q. Ai, H. Zamani, C. Wu, W. B. Croft, and X. Cheng, “A deep look into neural ranking models for information retrieval,” *arXiv preprint arXiv:1903.06902*, 2019.
- [71] F. Boudin, J.-Y. Nie, and M. Dawes, “Positional language models for clinical information retrieval,” in *EMNLP*, 2010, pp. 108–115.
- [72] ———, “Clinical information retrieval using document and pico structure,” in *NAACL*, 2010, pp. 822–830.
- [73] B. Nye, J. J. Li, R. Patel, Y. Yang, I. Marshall, A. Nenkova, and B. Wallace, “A corpus with multi-level annotations of patients, interventions and outcomes to support language processing for medical literature,” in *ACL*, 2018.
- [74] E. Kanoulas, D. Li, L. Azzopardi, and R. Spijker, “CLEF 2019 technology assisted reviews in empirical medicine overview,” in *Working Notes of CLEF 2019*, 2019.
- [75] B. Chiu, G. Crichton, A. Korhonen, and S. Pyysalo, “How to train good word embeddings for biomedical nlp,” in *Proceedings of the 15th Workshop on Biomedical Natural Language Processing*, 2016, pp. 166–174.
- [76] Y. Zhu, E. Yan, and F. Wang, “Semantic relatedness and similarity of biomedical terms: examining the effects of recency, size, and section of biomedical publications on the performance of word2vec,” *BMC medical informatics and decision making*, vol. 17, no. 1, p. 95, 2017.
- [77] F. Diaz, B. Mitra, and N. Craswell, “Query expansion with locally-trained word embeddings,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2016, pp. 367–377.
- [78] W. Yang, K. Lu, P. Yang, and J. Lin, “Critically examining the “neural hype”: Weak baselines and the additivity of effectiveness gains from neural ranking models,” in *SIGIR*, 2019, pp. 1129–1132.

- [79] G. E. Lee and A. Sun, “Seed-driven document ranking for systematic reviews in evidence-based medicine,” in *SIGIR*, 2018, pp. 455–464.
- [80] Q. Le and T. Mikolov, “Distributed representations of sentences and documents,” in *ICML*, 2014, pp. 1188–1196.
- [81] J. S. Whissell and C. L. Clarke, “Effective measures for inter-document similarity,” in *CIKM*, 2013, pp. 1361–1370.
- [82] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, “From word embeddings to document distances,” in *ICML*, 2015, pp. 957–966.
- [83] J. Guo, Y. Fan, Q. Ai, and W. B. Croft, “Semantic matching by non-linear word transportation for information retrieval,” in *CIKM*, 2016, pp. 701–710.
- [84] B. Koopman, G. Zuccon, P. Bruza, L. Sitbon, and M. Lawley, “Graph-based concept weighting for medical information retrieval,” in *ADCS*, 2012, pp. 80–87.
- [85] S. Gehrmann, F. Dernoncourt, Y. Li, E. T. Carlson, J. T. Wu, J. Welt, J. Foote Jr, E. T. Moseley, D. W. Grant, P. D. Tyler *et al.*, “Comparing deep learning and concept extraction based methods for patient phenotyping from clinical narratives,” *PloS one*, vol. 13, no. 2, 2018.
- [86] A. J. Brockmeier, M. Ju, P. Przybyła, and S. Ananiadou, “Improving reference prioritisation with pico recognition,” *BMC Medical Informatics and Decision Making*, vol. 19, no. 1, p. 256, 2019.
- [87] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, “BioBERT: a pre-trained biomedical language representation model for biomedical text mining,” *Bioinformatics*, 2019.
- [88] J. Guo, Y. Fan, X. Ji, and X. Cheng, “Matchzoo: A learning, practicing, and developing system for neural text matching,” in *SIGIR*, 2019, pp. 1297–1300.
- [89] B. Mitra, F. Diaz, and N. Craswell, “Learning to match using local and distributed representations of text for web search,” in *WWW*, 2017, pp. 1291–1299.

REFERENCES

- [90] S. Wan, Y. Lan, J. Guo, J. Xu, L. Pang, and X. Cheng, “A deep architecture for semantic matching with multiple positional sentence representations,” in *AAAI*, 2016.
- [91] E. M. Voorhees, “Overview of the trec 2004 robust retrieval track,” in *Trec*, 2004.
- [92] C. Hauff, D. Hiemstra, and F. de Jong, “A survey of pre-retrieval query performance predictors,” in *CIKM*, 2008, pp. 1419–1420.
- [93] A. Shtok, O. Kurland, and D. Carmel, “Predicting query performance by query-drift estimation,” in *ICTIR*, 2009, pp. 305–312.
- [94] O. Butman, A. Shtok, O. Kurland, and D. Carmel, “Query-performance prediction using minimal relevance feedback,” in *Proceedings of the 2013 Conference on the Theory of Information Retrieval*, 2013, pp. 14–21.
- [95] O. Kurland, F. Raiber, and A. Shtok, “Query-performance prediction and cluster ranking: two sides of the same coin,” in *CIKM*, 2012, pp. 2459–2462.
- [96] S. Mizzaro, J. Mothe, K. Roitero, and M. Z. Ullah, “Query performance prediction and effectiveness evaluation without relevance judgments: Two sides of the same coin,” in *SIGIR*, 2018, pp. 1233–1236.
- [97] H. Scells, L. Azzopardi, G. Zuccon, and B. Koopman, “Query variation performance prediction for systematic reviews,” in *SIGIR*, 2018, pp. 1089–1092.
- [98] B. He and I. Ounis, “Query performance prediction,” *Information Systems*, vol. 31, no. 7, pp. 585–594, 2006.
- [99] E. M. Voorhees, “Overview of the trec 2004 robust retrieval track,” 2004.
- [100] B. He and I. Ounis, “Inferring query performance using pre-retrieval predictors,” in *International symposium on string processing and information retrieval*, 2004, pp. 43–54.
- [101] S. Cronen-Townsend, Y. Zhou, and W. B. Croft, “Predicting query performance,” in *SIGIR*, 2002, pp. 299–306.
- [102] J. Allan, L. Ballesteros, J. P. Callan, W. B. Croft, and Z. Lu, “Recent experiments with inquiry,” in *Proceedings of the 4th Text Retrieval Conference*, 1995, pp. 49–64.

REFERENCES

- [103] Y. Zhao, F. Scholer, and Y. Tsegay, “Effective pre-retrieval query performance prediction using similarity and variability evidence,” in *ECIR*, 2008, pp. 52–64.
- [104] E. Kanoulas, D. Li, L. Azzopardi, and R. Spijker, “CLEF 2018 technologically assisted reviews in empirical medicine overview,” in *CLEF*, 2018.
- [105] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in *NIPS*, 2013, pp. 3111–3119.
- [106] D. Carmel, E. Yom-Tov, A. Darlow, and D. Pelleg, “What makes a query difficult?” in *SIGIR*, 2006, pp. 390–397.
- [107] H. Roitman, “Query performance prediction using passage information,” in *SIGIR*, 2018, pp. 893–896.
- [108] S. N. Kim, D. Martinez, L. Cavedon, and L. Yencken, “Automatic classification of sentences to support evidence based medicine,” in *BMC bioinformatics*, 2011.
- [109] G. Y. Chung, “Sentence retrieval for abstracts of randomized controlled trials,” *BMC medical informatics and decision making*, 2009.
- [110] D. Demner-Fushman and J. Lin, “Answering clinical questions with knowledge-based and statistical techniques,” *Computational Linguistics*, 2007.
- [111] J. Yang, Y. Zhang, L. Li, and X. Li, “Yedda: A lightweight collaborative text span annotation tool,” in *ACL, System Demonstrations*, 2018.
- [112] M. Leszczynski, A. May, J. Zhang, S. Wu, C. R. Aberger, and C. Ré, “Understanding the downstream instability of word embeddings,” *MLSys*, 2020.
- [113] L. Wendlandt, J. K. Kummerfeld, and R. Mihalcea, “Factors influencing the surprising instability of word embeddings,” in *NAACL-HLT*, vol. 1, 2018, pp. 2092–2102.
- [114] M. Antoniak and D. Mimno, “Evaluating the stability of embedding-based word similarities,” *TACL*, vol. 6, pp. 107–119, 2018.
- [115] J. Hellrich and U. Hahn, “Bad company-neighborhoods in neural embedding spaces considered harmful,” in *COLING*, 2016, pp. 2785–2796.

- [116] J. Pennington, R. Socher, and C. Manning, “Glove: Global vectors for word representation,” in *EMNLP*, 2014, pp. 1532–1543.
- [117] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.
- [118] E. Altszyler, M. Sigman, and D. Fernández Slezak, “Corpus specificity in LSA and word2vec: The role of out-of-domain documents,” in *Proceedings of The Third Workshop on Representation Learning for NLP*. Association for Computational Linguistics, 2018, pp. 1–10.
- [119] E. Choi, M. T. Bahadori, E. Searles, C. Coffey, M. Thompson, J. Bost, J. Tejedor-Sojo, and J. Sun, “Multi-layer representation learning for medical concepts,” in *KDD*, 2016, pp. 1495–1504.
- [120] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun, “Gram: graph-based attention model for healthcare representation learning,” in *KDD*, 2017, pp. 787–795.
- [121] X. Cai, J. Gao, K. Y. Ngiam, B. C. Ooi, Y. Zhang, and X. Yuan, “Medical concept embedding with time-aware attention,” in *IJCAI*, 2018, pp. 3984–3990.
- [122] Y. Choi, C. Y.-I. Chiu, and D. Sontag, “Learning low-dimensional representations of medical concepts,” *AMIA Summits on Translational Science Proceedings*, vol. 2016, p. 41, 2016.
- [123] E. L. Mencia, G. de Melo, and J. Nam, “Medical concept embeddings via labeled background corpora,” in *Proceedings of the 10th Language Resources and Evaluation Conference (LREC 2016), Paris, France, 2016*.
- [124] D. Newman-Griffis, A. M. Lai, and E. Fosler-Lussier, “Jointly embedding entities and text with distant supervision,” in *Proceedings of The Third Workshop on Representation Learning for NLP*, 2018, pp. 195–206.

- [125] Z. Yu, T. Cohen, B. Wallace, E. Bernstam, and T. Johnson, “Retrofitting word vectors of mesh terms to improve semantic similarity measures,” in *Proceedings of the Seventh International Workshop on Health Text Mining and Information Analysis*, 2016, pp. 43–51.
- [126] A. R. Aronson, “Effective mapping of biomedical text to the umls metathesaurus: the metamap program.” in *Proceedings of the AMIA Symposium*. American Medical Informatics Association, 2001, p. 17.
- [127] A. L. Beam, B. Kompa, I. Fried, N. P. Palmer, X. Shi, T. Cai, and I. S. Kohane, “Clinical concept embeddings learned from massive sources of medical data,” *arXiv preprint arXiv:1804.01486*, 2018.
- [128] L. De Vine, G. Zuccon, B. Koopman, L. Sitbon, and P. Bruza, “Medical semantic similarity with a neural language model,” in *CIKM*, 2014, pp. 1819–1822.
- [129] B. Pierrejean and L. Tanguy, “Investigating the stability of concrete nouns in word embeddings,” in *Proceedings of the 13th International Conference on Computational Semantics-Short Papers*, 2019, pp. 65–70.
- [130] M. A. Musen, N. F. Noy, N. H. Shah, P. L. Whetzel, C. G. Chute, M.-A. Story, B. Smith, and N. team, “The national center for biomedical ontology,” *Journal of the American Medical Informatics Association*, vol. 19, no. 2, pp. 190–195, 2011.
- [131] C.-H. Wei, H.-Y. Kao, and Z. Lu, “Pubtator: a web-based text mining tool for assisting biocuration,” *Nucleic Acids Research*, vol. 41, 07 2013.
- [132] O. Levy, Y. Goldberg, and I. Dagan, “Improving distributional similarity with lessons learned from word embeddings,” *TACL*, vol. 3, pp. 211–225, 2015.
- [133] H. Caselles-Dupré, F. Lesaint, and J. Royo-Letelier, “Word2vec applied to recommendation: hyperparameters matter,” in *RecSys*, 2018, pp. 352–356.
- [134] N. Limsopatham, C. Macdonald, and I. Ounis, “Modelling relevance towards multiple inclusion criteria when ranking patients.” in *CIKM*, 2014, pp. 1639–1648.

REFERENCES

- [135] M. Surdeanu, R. Nallapati, and C. Manning, “Legal claim identification: Information extraction with hierarchically labeled data,” in *Workshop Programme*, 2010, p. 22.
- [136] R. Nallapati and C. D. Manning, “Legal docket classification: Where machine learning stumbles,” in *2008 Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, Proceedings of the Conference, 25-27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*. ACL, 2008, pp. 438–446.
- [137] E. M. Voorhees, “On building fair and reusable test collections using bandit techniques,” in *CIKM*. Association for Computing Machinery, 2018, p. 407–416.
- [138] M. M. Rahman, M. Kutlu, T. Elsayed, and M. Lease, *Efficient Test Collection Construction via Active Learning*, 2020, p. 177–184.