Causality-driven Ad-hoc Information Retrieval

Suchana Datta (19208875)

This thesis is submitted to University College Dublin in fulfilment of the requirements for the degree of Doctor of Philosophy in Computer Science

School of Computer Science

Head of School: Assoc. Prof. Neil Hurley

Principle Supervisor: Assoc. Prof. Derek Greene
Co-supervisor: Dr. Debasis Ganguly

RSP Chair: Assoc. Prof. Catherine Mooney
RSP Advisor I: Prof. Nial Friel
RSP Advisor II: Dr. Deepak Ajwani

April 2024

Statement of Authorship

I hereby certify that the submitted work is my own work, was completed while registered as a candidate for the degree stated on the Title Page, and I have not obtained a degree elsewhere on the basis of the research presented in this submitted work.

Signature _____

ABSTRACT

Traditional information retrieval systems are primarily focused on finding topically-relevant documents, which are descriptive of a particular query concept. However, when working with sources such as collections of news articles, users frequently seek not only those documents that describe a news event but also documents that explain the chain of events that could have contributed to the occurrence of that event. These associations might be complex, involving a number of causal factors. Motivated by this information need, we formulate the task of *causal information retrieval*. First, we offer a comprehensive review of the existing literature on causality-related research, explaining how the proposed task differs from standard retrieval problems. Following this, we conduct empirical experiments to assess the effectiveness of popular existing retrieval methods to retrieve causally-relevant documents. Our findings illustrate that conventional methods are not suitable for this task, highlighting that causal information retrieval remains an open challenge that merits further research and exploration. To the best of our knowledge, the study of causal information retrieval, especially the extraction of information indicating causality directly from the documents, is a novel area of research. Consequently, there currently exists no off-the-shelf benchmark dataset for evaluating such systems. This thesis contributes a new dataset specifically tailored for causal information retrieval, which is made available to the community to support further research.

Additionally, in this thesis, we contend that while causally relevant documents would have partial term overlap with the ones that are topically relevant for a query, it is anticipated that a substantial portion of these documents will employ a distinct set of terms to describe various potential causes that could result in specific effects. To address this issue, we propose an unsupervised feedback model to estimate a distribution of terms that are relatively infrequent but are associated with high weights in the topically-relevant distribution, indicating potential causal relevance. Our experiments reveal that this feedback model proves to be significantly more effective than conventional IR models and several other baseline heuristics related to causality.

As a further contribution of this thesis, we introduce a supervised approach to enhance retrieval effectiveness in the context of causality. The fundamental idea here is to analyze input queries and estimate their specificity to the collection, enabling us to

determine whether or not to apply feedback in order to retrieve more causally relevant content towards top ranks. We introduce two such supervised query performance estimation models and demonstrate that these approaches yield significant performance improvements on a range of benchmark IR datasets. The effectiveness of the proposed query performance estimation models serves as motivation for the selective feedback model for causal information extraction. We illustrate how the intermediate decision of whether or not to apply query performance prediction ultimately results in an increase in downstream effectiveness.

ACKNOWLEDGEMENTS

First of all, I would like to extend my heartfelt gratitude to my supervisors, Prof. Derek Greene, University College Dublin and Dr. Debasis Ganguly, University of Glasgow for their time and valuable suggestions which has guided me in proceeding with this challenging PhD work successfully. Their help in suggesting me the relevant resources and course works from time to time is deeply appreciated. They went above and beyond to support my research and keep me motivated during the COVID pandemic, through my thick and thin. I will always remember their motivating words when we were having a series of failures. I am thankful to Science Foundation Ireland (SFI) for funding this research work without which the work itself would not have existed.

I am immensely grateful to Prof. Mandar Mitra, Indian Statistical Institute (ISI), Kolkata who inspired me to pursue information retrieval (IR) as my PhD research topic. His unaccountable knowledge and passion for IR made me love this subject and he has always been an excellent guide as well as a colleague. I extend my sincere gratitude to my IR lab senior at ISI, Dr. Dwaipayan Roy for his polite response to my naive questions on general IR topics during my ISI days. He also introduced me to Lucene which has been used extensively in this thesis. I express my gratitude to my fantastic mentor and colleague Dr. Sean MacAvaney who taught me IR from neural perspective. I am thankful to all my friends and colleagues at Insight for their unconditional support whenever needed. A special thanks to Rosemary Deevy for her excellent support with all my administrative queries and paperwork at the earliest.

Finally, I would like to thank the persons for whom working for a PhD had become a reality. I have no words to express my gratitude to my father, Manindra Kumar Datta and my mother, Rama Datta whose lessons and blessings have made me what I am today. I am grateful to my brother, Srijon Datta, my husband, Dr. Sudipta Lal Basu and my mother-in-law, Shila Basu who have always been the source of inspiration. Without their support, I could have never been able to complete this journey. A special thanks to Sudipta for his patience and for the support with household duties when I was busy doing thesis works.

LIST OF PUBLICATIONS

Core Publications

- Suchana Datta, Sean MacAvaney, Debasis Ganguly, and Derek Greene. "A 'Pointwise-Query, Listwise-Document' based Query Performance Prediction Approach." *In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'22*, pp. 2148-2153. 2022. [Chapter 6 is based on this paper.]

- Suchana Datta, Debasis Ganguly, Derek Greene, and Mandar Mitra. "Deep-qpp: A pairwise interaction-based deep learning model for supervised query performance prediction." *In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, WSDM'22*, pp. 201-209. 2022. [Chapter 6 is based on this paper.]

- Suchana Datta, Debasis Ganguly, Dwaipayan Roy, Francesca Bonin, Charles Jochim, and Mandar Mitra. "Retrieving potential causes from a query event." *In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'20*, pp. 1689-1692. 2020. [Chapter 5 is based on this paper.]

- Suchana Datta, Derek Greene, Debasis Ganguly, Dwaipayan Roy, and Mandar Mitra. "Where's the Why? In Search of Chains of Causes for Query Events." *In Proceedings of The 28th Irish Conference on Artificial Intelligence and Cognitive Science, AICS'20*, pp. 109-120. 2020. [Chapter 3 is based on this paper.]

- Suchana Datta, Debasis Ganguly, Sean MacAvaney, and Derek Greene. "A Deep Learning Approach for Selective Relevance Feedback". In: Goharian, N., et al. Advances in Information Retrieval. ECIR 2024. Lecture

Notes in Computer Science, vol 14609. Springer, Cham. [Chapter 7 is based on this paper.]

Other Publications

- Suchana Datta, Debasis Ganguly, Mandar Mitra, and Derek Greene. "A relative information gain-based query performance prediction framework with generated query variants." *ACM Transactions on Information Systems 41*, no. 2 (2022): 1-31.
- Debasis Ganguly, Suchana Datta, Mandar Mitra, and Derek Greene. "An analysis of variations in the effectiveness of query performance prediction." *In European Conference on Information Retrieval, ECIR'22*, pp. 215-229, 2022.
- Ashutosh Singh, Debasis Ganguly, Suchana Datta, and Craig McDonald. "Unsupervised Query Performance Prediction for Neural Models with Pairwise Rank Preferences." *In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'23*, pp. 2486-2490. 2023.
- Suchana Datta, Debasis Ganguly, Derek Greene, and Mandar Mitra. "On the Feasibility and Robustness of Pointwise Evaluation of Query Performance Prediction." *In Proceedings of the The QPP++ 2023: Query Performance Prediction and Its Evaluation in New Tasks Workshop co-located with The 45th European Conference on Information Retrieval, ECIR'23*, pp. 1-6. 2023.
- Suchana Datta, Debasis Ganguly, Josiane Mothe and Md Zia Ullah. "Combining Word Embedding Interactions and LETOR Feature Evidences for Supervised QPP." *In Proceedings of the The QPP++ 2023: Query Performance Prediction and Its Evaluation in New Tasks Workshop co-located with The 45th European Conference on Information Retrieval, ECIR'23*, pp. 7-12. 2023.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER

ONE

INTRODUCTION

Faced with any situation or event, it is a fundamental part of human nature to ask 'why?' and 'when?', as we attempt to understand the context in which we find ourselves. The same can be said when we seek to analyze the complex nature of events in modern society. For instance, we may want to find out *why* was the UAE-Israel peace accord signed, for a potential analysis of its consequences. In the existing literature, the study of cause-effect relations has focused on analyzing the inter-relationships among different phenomena, in terms of causes and their effects (Asghar, 2016), as humans often perceive the present reality as a chain of causes and consequences. Sometimes these associations are immediately evident to us, such as, seismic plate shift *causes* earthquake. However, in other cases, these relations can be far more subtle and complex, involving a combination of a number of causal factors that might have led to an observed event, which in turn might have been caused by other factors and so forth. Returning to the example of the UAE-Israel peace accord, while the immediate causes may include factors like Israel's settlement plan or Trump's diplomatic strategy (BBC Middle East editor, 2020), there may also be deeper-rooted causes dating back further in history, such as the pursuit of global recognition and efforts to improve relations with the Middle East (Frank Gardner, 2020).

Generally speaking, the study of *cause-effect* relationship has a long history mostly in terms of its psychological and cognitive aspect (Koriat *et al.*, 2006; Kuo *et al.*, 2019), where the concept of cause-and-effect associations is established as fundamental to recognizing various facts and phenomena. Human psychology refers a cause-effect relationship as one factor (the *cause*) triggers an outcome (the *effect*). For example, a student performed well in an exam (effect) because she worked hard throughout the course (cause). Now the crucial step is to establish the causality as far as human psychology is concerned, i.e.,

to demonstrate an association between the cause and its effect. Psychologists usually start by asking a simple question: 'Is there an association between dependent (effect) and independent (cause) variables?'. In contrast, the goal of this research is to study fact-based causality, where the causation is explored from an information retrieval (IR) perspective that involves retrieving potential triggering list of causes in response to an user's query which is an expression of an event occurred in the past.

From an IR perspective, these cause-and-effect inquiries can be viewed as analogous to the interactions between a user and a search engine, where the user might input an effect as a query to the search system and expect to receive a collection of triggering causes in response. Unlike traditional search responses (i.e., capturing term overlaps between query and documents), finding this specific type of cause-effect information might be challenging for a standard search engine. Indeed, the research literature emphasizes that, in most situations, there are no definitive rules around how cause-effect relations should be structured (Hashimoto *et al.*, 2015; Riaz & Girju, 2014). When dealing with intricate causal relationships, such as news events and their consequences, explicitly enumerating a list of causes (in the form of short text segments) often becomes difficult. This is primarily because, in most instances, the causes leading to an event involve subjectivity, and these causal factors are spread across a number of multi-topic documents (Kiciman, 2018; Kiciman & Thelin, 2018; Datta *et al.*, 2020).

As noted above, traditional retrieval systems typically concentrate on matching terms between documents and a user query. However, such techniques may not be adequate for the situation where a user's search is intended to reveal the causes which led to a specific event. In this context, a user might consider a straightforward approach, such as including terms like 'why', 'causes' or 'reasons' as additional query terms. However, this approach often proves ineffective in practice for identifying causal links as the nature of causal relevance is likely to be different from that of its topical relevance. Later in this thesis, we further detail how the nature of causal relevance differs from that of traditional topical relevance (see Chapter 3).

In this thesis, our exploration of causality highlights that while causally relevant documents may share some term overlap with those topically relevant to a query, it is expected that most of these documents will employ a different set of terms to describe various potential causes and their effects. This is an aspect that traditional search systems typically struggle to address. Therefore, to ad-

dress this gap in the IR literature, we aim to investigate the novel problem of *causal search*, where user's search intention is exclusively to know 'why?' and proposes potential solutions to mitigate the gap.

1.1 Motivation

Finding causal information involves multiple query reformulations. In traditional search systems, a user is required to specify the information need in the form of a search query. In some cases, these queries are clear reflections of the user's requirements. However, more often, human-generated queries are not particularly specific in terms of the user's search intent. Based on the input query, an IR system retrieves the top-most similar documents, where the similarity between a document and a query is measured with the help of an underlying scoring function of a retrieval model, such as BM25 and LM-JM. (Hiemstra, 2001; Zhai & Lafferty, 2001; Ponte & Croft, 1998; Robertson & Zaragoza, 2009). The user then typically inspects the retrieved documents, seeking instances which satisfy their information need.

It is important to note that traditional IR systems do not take a user's search intent into account while retrieving search results for the user. Therefore, the usefulness of the output of an ad-hoc IR system, in the form of a ranked list of documents, is likely to be limited in situations when either: i) decision makers need to formulate policies to mitigate a current event that requires attention (e.g. drop in the value of British pound); ii) policy-making regarding societal benefits (e.g. formulating government policies to reduce housing crisis by analyzing the most likely causes). In such situations, the user of a traditional search system is required to carefully analyze the relevant documents (likely to describe the main event expressed in the query itself) and most likely will have to reformulate queries in order to retrieve documents related to the potential *causes leading* to the (query) event.

Subtle nature of cause-effect relations in news articles. There is a long history of research on finding causal relations in a form of textual entailment, where cause-effect relations hold clear conjunctives (for example, 'because', 'due to', 'leads to' and so on) (Asghar, 2016). In contrast, we are interested in cases where cause-effect relations are rather complex without having any direct associations, as is observed for news articles (refer back to the example

of UAE-Israel peace accord signed at the beginning of this chapter). For news articles, an important point to note is that the causes leading to an event are rather often subtle in nature instead of being explicit (Datta *et al.*, 2020). Moreover, more often than not, an event is triggered by a series of causes spread over a considerable period of time (Datta *et al.*, 2020). Consequently, it is often difficult to find news documents that would 'single out' the cause of an event to be one specific event in the past. This in turn means that making the initial query more specific by adding cause-related keywords, such as 'pound drop causes' or 'pound drop reasons' etc., and then using a traditional IR system is unlikely to retrieve relevant information, because such information is not explicitly reported in news articles. However, such information may be discovered by analyzing a number of documents and associating the latent relationships between their terms. Therefore, the user of a traditional IR system must spend considerable effort in reformulating queries in order to retrieve causally relevant documents.

User needs prior knowledge to capture and verify causal relevance. To illustrate the problem, consider a scenario where a user would like to find potential causes leading to the event 'drop of the British pound', without having prior knowledge of the possible reasons. Thus, the search intention is to *explore*, rather than to *recall* or confirm previously known information. In this situation, the user first needs to submit a query related to the event (e.g. 'pound value drop'). The documents retrieved at the top ranks by a traditional search systems will be related to the topic itself, since these documents are expected to contain terms that are representative of the relevance to the information need (e.g. recent news reporting the drop in the value of the pound). Since such top-ranked documents are unlikely to be *causally relevant* to the information need (i.e., they will not list the likely causes leading to the query event), the user must then manually reformulate their query by including terms that are representative of the likely causes (e.g. concepts such as 'Brexit delay' or 'negotiation difficulties between EU and UK'). However, this becomes infeasible when the user is unaware of these causes in advance.

Recent AI tools may not be adequate for causality. With the fast growing availability of AI tools, such as ChatGPT[1] from OpenAI, a potential question might arise around whether finding information via ranked lists of doc-

[1] `https://openai.com/blog/chatgpt`

uments still relevant in today's world of ChatGPT and its successors. These instruction-tuned large language models are typically trained on huge datasets incorporating large language models (Zhao *et al.*, 2023) via zero shot or few shot learning. Our understanding of the precise logic or depth of knowledge underlying the responses generated by these models is limited. Consequently, their responses can be superficial, particularly in cases like ours involving significant subjectivity, and where the association between the query (i.e., any event) and the answer (i.e., prevalent causes of that event) is not explicit. In the worst case, users might encounter 'hallucinations' in the responses generated by these bots (Guo *et al.*, 2023). Therefore, AI-generated answers in the context of causality will likely require thorough validation to prevent such hallucinations.

Driven by this motivation, this thesis focuses on creating effective retrieval models that support **causality-based relevance**. These models aim to go beyond traditional topical relevance, reducing the need for manual query reformulation and providing accurate information in response to users' causal information needs. Broadly speaking, we investigate the existing gap in the information retrieval literature in terms of causal IR as outlined earlier, by addressing a number of associated research questions and challenges, as detailed in Section 1.2. Our key objective is to develop an end-to-end causal retrieval system, (**C**ausal **I**nformation **R**etrieval **S**ystem), to mitigate this gap, thereby exploring a novel direction of research which augments a classic ranked list of topically-relevant search results with a list of *causally-relevant* results.

1.2 Research Questions and Key Contributions

The central research challenge this thesis seeks to address is how to identify and extract segments of text from a document that indicate potential causes in response to a user query, which is formulated based on an event that could have a series of underlying prevalent causes. Particularly, this thesis contributes to the novel field of causal IR research by developing retrieval models using both unsupervised and supervised approaches that focus primarily on causality-based relevance. The specific research questions addressed in the respective chapters of this thesis are outlined in the following sections.

1.2.1 Standard Information Retrieval and Causality

Traditional information retrieval systems are primarily focused on finding topically-relevant documents, which are descriptive of a particular query concept. Such systems mainly concentrate on matching terms between documents and a user query, applying *topical relevance* to meet the user's *information need*. However, this might not address situations in which a user's search is intended to uncover the original causes that led to a specific event. More specifically, when dealing with sources such as collections of news articles, users often seek to identify not only documents that depict a news event but also those that elucidate the sequence of events that could have potentially culminated in the occurrence of that event. These connections can be intricate, encompassing a multitude of causal factors.

In the existing literature, the exploration of causal relations occurs either at the sentence level or within a single document (Asghar, 2016). Some methods incorporate prior knowledge about causal events, while others rely on predefined lexical, syntactic, or morphological relationships. However, these techniques are insufficient when it comes to addressing the nuanced causes and effects present within extensive document collections, a gap that we aim to bridge through the use of retrieval models. This thesis investigates this significant gap in the IR literature, and thus we formulate our first research question as follows.

> **RQ-1:** Is a traditional search system sufficient for identifying causally-relevant information, or is there a need to introduce a new research paradigm, namely, *causal information retrieval*?

Contributions. Chapter 3 and **4** are based on our investigations and observations with respect to **RQ-1**. The key contributions here are as follows:

- We create an initial pilot dataset for the novel causal document retrieval task that enumerates a list of cause indicative documents in response to an user's query.
- A set of rigorous experiments are conducted on the pilot dataset, illustrating that standard retrieval models do not suffice causality because of the subtle nature of causally-relevant documents with respect to its query events.

- We propose a new recursive causal retrieval framework design to perform in-depth exploration to find a chain of likely causes for a query.
- We introduce a newly-annotated, fine-grained dataset specifically designed to meet the needs of the retrieval framework, namely, identifying precise pieces of information within causally relevant documents.

1.2.2 Unsupervised Causal Retrieval

To answer RQ-1, in Chapter 3 we empirically demonstrate that standard IR models are inadequate for the task of causal retrieval. Therefore, it becomes clear that we need to introduce a new IR paradigm to capture the list of causes behind the input query. Ideally, a causal retrieval system should identify a series of triggering causes, likely distributed across multiple topically-relevant documents, and present them to the user as short text snippets. Furthermore, certain text segments from the list of triggering causes may have further causal needs that warrant further exploration.

Referring back to the example of 'UAE-Israel peace accord signed', the immediate causal factors might include Israel's settlement plan or Trump's diplomatic strategy (BBC Middle East editor, 2020). However, upon deeper examination of the antecedent causes of Israel's settlement plan, we may identify additional prominent factors, such as acquiring global recognition and improving relations with the Middle East. Thus, navigational search activities play an important role in uncovering causal relationships, where the success of the process is solely dependent on the quality of the initial set of retrieved results. This is because a collection of relevant text segments is essential in guiding a user towards uncovering more meaningful effect events and their underlying causes.

In answer to the RQ-1, we empirically show that causal and topical documents share a partial term overlap, such that, in almost all the cases, infrequent terms occurring in the pseudo-relevant document set are actually the carrier of causal information. Therefore, a simple yet effective approach is to expand queries with a list of *potential* causally-relevant terms. Here the causal indicative terms are to be captured by using simple heuristics applied to the collection. One effective way of expanding queries is to apply relevance feedback (Lavrenko & Croft, 2001) which assigns a probability score to the constituent terms of the top result list in an unsupervised way. Therefore, our first causality-based IR model is formalized based on the idea of traditional

relevance feedback. Furthermore, another reason for starting our initial investigation in an unsupervised manner is that the pilot dataset that we introduce later in Chapter 2 is unlikely to be adequate for most of the data-hungry neural re-rankers. Keeping this in mind, we formulate our second research question as below.

> **RQ-2:** Can we develop a system that, without any supervision, generates a list of potential causes, represented as short text segments, in response to any given causal query?

Contributions. Building on the foundations laid in previous chapters, Chapter 5 addresses **RQ-2** as follows:

- We propose an unsupervised feedback model to estimate a distribution of terms which are relatively infrequent but associated with high weights in the *topically relevant* distribution, leading to potential causal relevance.
- Through detailed experiments on both ad-hoc IR datasets and our newly-created causal dataset, we show that this feedback model is substantially more effective than traditional IR models and several other causality heuristic baselines.

1.2.3 Causal Retrieval Model with Supervision

Broadly speaking, any standard causal retrieval system can be considered as a black box which takes an event in the form of a query as input and yields a set of causes (i.e., text segments) in turn that have eventually led to the input event. However, the question arises, how does that black box distinguish an *effect* event that may have some causal links, given any piece of text as input? More specifically, even before finding out the list of text segments carrying causal information for any given query, the first challenge appears to be estimating how likely a given input query to contain an potential effect. In other words, for any standard causal retrieval system, we must decide if the input query is adequate to retrieve potential causally relevant documents from the collection, or might need reformulations to capture the causal trail. Specifically, to estimate the likelihood of an input query to be a causal one, a standard causal retrieval framework is expected to first make predictions on the performance of the query, and based on that prediction (i.e., if the query is specific to

the collection or not) the query must be re-framed. It bears repeating that the focus of this thesis is not on straightforward cause-effect relationships, like a seismic plate shift causing an earthquake or an ice jam leading to a flood. Instead, we direct our research to explore chains of cause-effect relations in news stories where the causal information tends to be complex and potentially open to interpretation.

For instance, the information need underlying the two queries 'Abraham Peace Accord' and 'Abraham Peace Accord signed' are likely to differ. The first query is highly specific, seeking documents pertaining directly to the contents of the treaty itself (e.g. the date of the event, the entities involved, and the specific agreements). This, however, falls outside the scope of our research. In contrast, the second query aims to reveal the predominant causes that lead to the agreement, such as Trump's diplomatic strategy, Israel's settlement plan, or efforts to reduce Israel's regional isolation. Our research centrally focuses on catering to this kind of causal query or event. Therefore, from the perspective of causal IR, it becomes important to assess the specificity of any given input query. Specifically, we must estimate whether the original query, in its current form, can adequately capture the underlying causal factors or if it necessitates additional information to uncover the causal trail.

In addressing RQ-2, we have developed a causal information retrieval model based on relevance feedback, which employs the query expansion technique without first assessing whether the original query captures causal relevance in its existing form. Later in Chapter 5, we observe that this kind of blind feedback approach often penalizes queries and introduces query-drifts by deviating it from the initial information need. Therefore, we revisit the concept of selective feedback for improved retrieval effectiveness. Our intuition is that a system will learn whether or not to apply feedback when it is trained with some labeled samples with strong supervision. Hence, this consideration leads us to formulate our third research question.

RQ-3: Can we develop a supervised decision-making pipeline capable of determining when query reformulation is necessary in order to capture causal relevance?

Contributions. Both Chapter 6 and 7 concentrate on developing the decision-making system required to address **RQ-3**. The principal contributions of that work are listed below:

- We develop a data-driven end-to-end convolutional neural framework designed to predict query specificity in ad-hoc retrieval.
- A novel end-to-end neural cross-encoder-based approach is proposed that estimates the specificity of an input query, which is validated across a variety of benchmark IR datasets.
- We propose a new deep-learning framework for the decision-making pipeline that leverages the idea of our proposed data-driven convolutional and cross-encoder-based query estimators.
- We demonstrate the effectiveness of the model on a comprehensive set of experiments on standard benchmark datasets and our newly-proposed causal dataset.

1.3 Thesis Outline

The rest of the thesis is organized as follows:

- Chapter 2 provides a comprehensive literature review on causality-based research to date, emphasizing how this PhD study differs from existing literature in the field.
- Chapter 3 illustrates how the notion of causality differs from its topical counterpart, with extensive experiments and analysis. Based on these findings, we introduce a general framework for causal retrieval.
- In Chapter 4, we present a new dataset for the evaluation of causal retrieval models, with detailed information regarding the data annotation process and characteristics of the dataset.
- In Chapter 5, we introduce a novel unsupervised relevance feedback-based approach for causal information retrieval. We describe the end-to-end architecture of the proposed model, highlighting its notable performance on data with a focus on causality-focused data and offering a thorough analysis of the model's retrieval effectiveness.
- In Chapters 6 and 7, the goal is to enhance the retrieval effectiveness of the original unsupervised model proposed in Chapter 5 through the incorporation of supervision. The central idea is to analyze input queries

and gauge their specificity to the collection, determining whether feedback should be applied to capture a greater number of relevant documents while reducing query drift. Chapter 6 proposes two such supervised query estimation models and demonstrates their marked performance across various benchmark IR datasets. Building on the effectiveness of these query performance estimation models, Chapter 7 explores an selective feedback model for causal information extraction. We describe the selective feedback pipeline in depth, demonstrating its superior retrieval effectiveness compared to the original unsupervised model.

- Finally, Chapter 8 concludes the thesis, highlighting the main findings and exploring potential directions for future research.

CHAPTER

TWO

BACKGROUND

2.1 Introduction

The general notion of causality has been extensively studied from various perspectives in the natural language processing community (Hashimoto *et al.*, 2012, 2015; Asghar, 2016). For instance, causal inference typically involves an entailment between a known set of effects and a set of possible causes. In other words, the goal is to derive the genuine causes from a given set of effects. These methods have practical applications, such as in the medical domain, where they can be used to automatically simulate random control trials (Austin, 2011).

Methods such as text classification have previously been applied for causal inference (Wood-Doughty *et al.*, 2018). Asghar (2016) provides a summary of approaches that use text mining to extract causes and effects. A number of existing approaches extract cause-effect patterns using lexical, syntactic, and more recently, semantic relations (Blanco *et al.*, 2008; Chang & Choi, 2005; Hashimoto *et al.*, 2014; Radinsky *et al.*, 2012). These are primarily taken from either headlines or single sentences. The cause-effect pattern approach was extended by Zhao *et al.* (2017), where a set of patterns were initially used to create a network of causes and effects, and then a relational embedding method (similar to TransE developed by Bordes *et al.* (2013)) was used to jointly embed causes and effects.

The aforementioned studies are particularly relevant for finding explicitly-mentioned causes. These are specified via typical patterns within a single sentence, such as 'X leads to Y', where the system is provided with sufficient information to learn underlying cause-effect relations from these patterns, which can used for future predictions. However, the notion of causality that we wish

to address in this thesis is more subtle and subjective. In our scenario, the system also has limited information from which to learn or capture reliable cause-effect patterns. As originally stated in Section 1.1, our focus will mostly be on cases where there are no direct evidences of causality relations between a query event and its causally-relevant precursors.

A key difference between our research and existing work is that we aim to retrieve causal information in the form of document excerpts (note that it could be the entire document in certain cases), the scale of which is typically much larger than sentence-level causality, such as 'heavy rain *causes* flood'. The work by Kiciman (2018) and Kiciman & Thelin (2018) addresses causal inference in IR, albeit in the reverse direction, where a query describes a cause and the results provide a list of possible effects. The authors focused on social media data and targeted potential future effects, while our work targets past causes with a focus on news events.

All of the above underscores that our perspective on addressing the challenges of Causal Information Retrieval is novel, and this research direction holds promise for fulfilling real-world user information requirements. We are particularly interested in capturing document-level causal information, rather than working at the sentence level. Next we provide a high-level overview of various existing approaches designed to capture cause-effect associations, which helped us to frame the problem of causal information retrieval. We have comprehensively compared existing techniques and grouped them into seven distinct categories as follows to emphasize the novelty of our causality research in this thesis.

2.2 Causal Relation Extraction

As deep neural architectures have gained popularity, the study of causation has increasingly shifted towards understanding counterfactuals, exploring questions like 'what might have happened under different circumstances?' (e.g. 'would I be more healthy if I had not smoked for the last two years?'). However, traditional research on causality primarily focused on pinpointing the semantic relations between a cause and its subsequent effect. For example, some work concentrated on extracting relevant *noun-verb* associations or lexical patterns from texts (Riaz & Girju, 2014; Tanaka *et al.*, 2012). Other studies have leveraged cue phrases and word-pair probabilities to examine the causal

dimensions within documents (Chang & Choi, 2006).

While sentence-level entailment (e.g. a statement such as 'sedentary lifestyle causes childhood obesity') has been harnessed to capture causal characteristics (Inui & Okumura, 2005), other authors have investigated the causal relations between two queries (Sun *et al.*, 2007), which eventually has lead to the idea of using event pairs (Beamer & Girju, 2009). Studies have shown that, if we can construct a map that connects an event in one query to another event in some other query, then the Granger Causality Test[1] Granger (2001) can effectively re-rank causal associations. This has lead to the idea of using event pairs. For instance, Beamer & Girju (2009) introduced a measure 'causal potential', which aimed to encapsulate the causality between temporally adjacent events, like, *wear* $\rightarrow$ *tailor* (e.g. 'wears a tailored jacket'). Later, Do *et al.* (2011) attempted to identify causality within texts by predicting event causality, i.e causality between event pairs, triggered by *noun-noun*, *verb-verb* or *verb-noun* pairs, and with the help of discourse relations (e.g. 'police arrested him' *because* 'he killed someone').

A novel conceptual map that steers through different causal analysis problems was presented in Lattimore & Ong (2018). Observations show that causal effect estimation for continuous variables are more complex than for discrete random variables. Based on the assumption that correlation does not imply causation, the authors introduced four schools of causality pertaining to three fundamental aspects of causation (i.e association, intervention and counterfactuals), as highlighted in Pearl & Mackenzie (2018). Modeling with Causal Bayesian networks (by definition, a link $y_i \rightarrow y_j$ implies y_i causes y_j, i.e. an intervention to change the value of y_i might affect y_j, but the reverse does not apply), Counterfactuals, Structured equation model (by definition, it represents a model with N variables that acts as a set of N simultaneous equations, where each variable is a dependent variable in one equation. For causality, it depicts causal assumptions about the associations between variables.) and Granger causality inspired them to build a unified architecture that leverages causal dimensions within sentences.

This thesis also focuses on studying the underlying relationships between event pairs that are causally connected, i.e., our interest revolves around one of the fundamental aspects of causation, 'association' as explained in

[1]Granger causality is a way to investigate causality between two variables in a time series. It provides a probabilistic account of causality by using empirical datasets to find patterns of correlation.

Pearl & Mackenzie (2018). In our case, the assumption is that for any effect event embedded in the input query, might have a number of cause-effect pairs spread across the collection. The main difference is that all the aforementioned approaches are concerned with sentential cause-effect relation extraction, whereas we investigate causality for a given query spanning across a document collection.

2.3 Graph-based Approaches

Graphs can provide a convenient way to encode and visualize cause-effect relations. Some authors have proposed a non-parametric graph-based framework to trace causal inferences. For instance, Pearl (1995) showed that causal diagrams can be constructed based on the assumptions of the in-domain causal influences. The resulting causal graph can then be queried to produce mathematical expressions for finding cause-effects for observed distributions to validate if available suppositions are adequate for identifying causal effects or not. This emphasizes that there must be a number of firm quantifying assumptions beforehand, which is unlikely in our case as we assume that the user is not aware of the causes of an event in advance.

Previous works have used Directed Acyclic Graphs (DAGs) to represent causal relations (Dawid, 2010; Pearl & Paz, 2022), with a later shift focus to Bayesian Networks (Zhang, 2008). Studies have shown that, with the help of probabilistic causality (Suppes, 1970; Richard & Peter, 2008), links can be captured between such causally connected concepts. In some cases this can be done via Markov chain conditions (Richardson & Spirtes, 2000). Another method employed a graphical approach, extracting causal associations by employing causal Bayesian networks. These were represented as Ancestral Graphs, as detailed by Zhang (2008). This approach aimed to tackle causal reasoning challenges that arise when multiple equivalent classes of causal diagrams are available from (partial) ancestral graphs. The authors modeled causality with non-linear causal graphs where nonlinear effect of the causes and inner noise effects are taken care of in graph-based approaches.

In other work, Rink *et al.* (2010) proposed using graphs to solve event-pair causality relations as encoded in text (e.g. 'we *recognized* the problem and *took* care of it'). They extracted various graph patterns at the sentence level, taking the form of subgraphs from each sentence-graph. These graph patterns were

then employed as binary classifiers to distinguish between causes and effects. In general, graph-based techniques have primarily focused on the extraction of event pairs from text and the study of their patterns via probabilistic measures which motivates us to come up with a recursive causal extraction model as depicted in the next chapter of this thesis (see Figure 3.3). For a given user input query, the cause-effect pairs at each successive steps are likely to be considered as a strongly connected nodes of a graph generated from a collection.

2.4 Causal Knowledge Bases

Research on causality that makes use of domain-independent knowledge was first introduced in the late 1990s and continues today. As knowledge-based causality developed gradually, researchers attempted to explore automatic causal relation acquisition, specifically via common cause-effect propositions (Kaplan & Berry-Rogghe, 1991). The goal here is to exploit semantic property of predicates (Hashimoto *et al.*, 2012) which efficiently find contradictory pairs (e.g. 'destroy cancer' $\perp$ 'develop cancer'). Zhao *et al.* (2017) expanded the knowledge-base pattern approach by initially using a set of patterns to establish a network of causes and effects, forming the basis for a relational embedding method.

Kaplan & Berry-Rogghe (1991) developed a knowledge-based causal relation acquisition system, named TAKT, to explore the automatic understanding of expository text. The system processes a set of propositions, represented as causal chains of cause-event and effect-event from input text sentences, yielding domain-independent causal knowledge. This technique handles some common cause-effect propositions from text that could be used in specific cases (e.g. 'when a cloud forms the water vapor condenses into water').

Furthermore, an automated system to learn expressions relating to cause-effect correspondence was described by Kozareva (2012). By using bootstrapping the authors developed a learning database that is capable of mapping causal patterns like, {*bacteria, worms, germs*} to some effect patterns like, {*diseases, damage, contamination*} from the web using a recursive pattern '**and virus cause**'. The cause-effect pattern approach was extended by Zhao *et al.* (2017), where a set of patterns was initially used to create a network of causes and effects, and then a relational embedding method is used to jointly embed causes and effects and thus system becomes well-versed with the domain knowledge.

However, such an approach does not fit in cases where causally relevant documents do not share the same embedding space with the query as in our case. In other words, the cases where the cause and effect events do not have any direct relations rather share nuanced associations, such as, the association between 'why was Brexit happened?' and the underlying effect of 'dropping the value of British Pound'. Chapter 3 conducts a thorough investigation around how topical and causal document embeddings share a very small partial overlap and via detailed experiments and analysis we show that how the subtle cause-effect relations make the task a challenging and novel one.

2.5 Document Classification

Causality has also been shown to be relevant in the context of document classification, where the relationship between features and classes is often complex. Paul (2017) sought to answer the question of 'which term features *cause* documents to have the class labels that they do?', and developed a *propensity score matching* technique for selecting important features. Conversely, if a term 'horrible' is added to a movie review, it certainly points to a negative sentiment, while the term 'said' does not.

Work by Wood-Doughty *et al.* (2018) considered the causal inference task as a classification problem, and by using logistic regression, they illustrated how to analyze causality a variety of datasets. The authors took into account factors such as missing data and measurement errors, which often hinder downstream causal analysis. To facilitate causal inference, they explored methods which can integrate text classifiers with the investigation of causal inference. They designed their classification model to account for two key factors: missing data and measurement errors. Their approach highlighted how modeling assumptions can introduce biases and obstacles to subsequent causal analysis.

In light of the concept of document classification via feature analysis, we formulate our causal retrieval problem as a classification task and in line to this we propose a causal document retrieval model that segregates the causal documents from that of its topical ones via analyzing term heuristics and features extracted from the collection. We detail the model in Chapter 5.

2.6 Future Scenario Prediction

Researchers working on contingency discourse tasks in NLP, specifically new event prediction, have regarded causal relation extraction from text data as being particularly challenging (Radinsky *et al.*, 2012). Radinsky & Horvitz (2013) initiated this research with the automatic compilation and generalization of a sequence of events from different web corpora. However, others have argued that, in order to address causality, either two of the events in the consecutive sentences must hold an inter-sentential contingent relation (Riaz & Girju, 2010) or there should be a pre-trained event-causality chaining database generated from web data (Hashimoto *et al.*, 2014). Therefore, future scenario prediction problems require prior event knowledge, which is unlikely to be available in our case, since users will typically have no prior knowledge about the plausible causes of a query event.

Riaz & Girju (2010) showed that two novel measures, *Effect Control Dependency* (ECD) and *Effect Control Ratio* (ECR), were effective in identifying cause-effect pairs from online news articles. This was achieved by looking at both 'intra-sentential' and 'inter-sentential' texts, without relying heavily on prior contextual information. Specifically, the authors considered context as consisting of two events: the cause (independent) event (X), and the effect (dependent) event (Y), which holds a contingent relation in between, such that $X \rightarrow Y$. Consider the example: 'Katrina $[hit]$ Florida late last week. Since Friday, Dallas-based Southwest airlines $[cancelled]$ more than 250 flights.' Here two of the events in the consecutive sentences hold an inter-sentential contingent relation. However, in the context of our retrieval model, contingent relations might not apply, as there will be cases where query events are not mentioned in the relevant causal documents at all.

Aside from dealing with semantics, context and association features of web data (such as, *'conduct slash-and-burn agriculture'* $\rightarrow$ *'exacerbate desertification'*), Hashimoto *et al.* (2014) exploits future scenario generation by chaining event-causality using causal-compatibility (e.g. *'conduct slash-and-burn agriculture'* $\rightarrow$ *'exacerbate desertification'* $\rightarrow$ *'increase Asian dust (from China)'* $\rightarrow$ *'asthma gets worse'*). This chaining architecture is relatively novel, although previous work by Radinsky & Horvitz (2013) did address contingency discourse by automatically compiling and generalizing a sequence of events from various web corpora.

Radinsky *et al.* (2012) showed that future plausible news event prediction of-

ten involves causality. The authors introduced an intelligent system, called Pundit, capable of modelling and potentially predicting future events For instance, for an event '*Magnitude 6.5 earthquake rocks the Solomon Islands*', 'Pundit' predicts '*Tsunami-warning will be issued in the Pacific Ocean*'. It used the event knowledge database in which '*Tsunami warning issued for Indian Ocean*' after '*7.6 earthquake strikes island near India*' was one of the cause-effect sentence pair which is applicable in only cases. This means that contingency discourse and future scenario prediction both require prior event knowledge, which is not the case we aim to address. In our case we focus on capturing triggering causes behind any fact-based informational causal query where users are unlikely to have access to this prior knowledge. The literature study in this section again underlines the novelty and challenges around the task that we aim to solve in this thesis.

2.7 Question-Answering

The NLP literature highlights that question-answering (QA) systems exploit the inherent nature of causality by disambiguating the pervasive nature of causal relations (Girju, 2003). This can help to identify *inter* and *intra*-sentential causal links between terms and clauses to answer 'why' questions (Oh *et al.*, 2013). Recently, a decision support system was proposed by Kiciman & Thelin (2018) to foresee the consequences of queries like, '*Should I join the military*?' or '*Should I move to California*?'. Another group of researchers focused on a new variant of QA, referred to as common sense causality identification (Gordon *et al.*, 2011, 2012). This technique helped to disambiguate discourse relations and reasoning with sentence proximity by making use of knowledge bases.

More recently, question-answering systems have begun to exploit causal associations. For instance, Kiciman & Thelin (2018) proposed a method to explore 'expectations' in terms of online search by incorporating causality. However, the study of causality in question-answering began earlier with work by Girju (2003) who proposed an automated system capable of capturing lexico-syntactic patterns in the form of simple (e.g. cause, lead to, bring about, generate, make, force, allow), resultative (e.g. kill, melt, dry) and instrumental causative (e.g. poison, hang, punch, clean), that are useful to exhibit '*causation*' in English texts. The authors showed that their system is capable of disambiguating the pervasive nature of causal relations and proves to be coherent to 'explicit', 'ambiguous', and 'implicit' causal questions.

Other researchers have investigated the significance of *inter* and *intra*-sentential causal links between terms and clauses analyzing syntactic and morphological features for re-ranking candidate answers to 'why'-questions (Oh *et al.*, 2013). As a concrete example, {the ocean's water mass is displaced and, much like throwing a stone into a pond, waves are generated}$_{cause}$ and {Tsunamis that can cause large coastal inundation are generated}$_{effect}$ is a reasonable answer to the question, '*Why are tsunamis generated?*'. Thus, QA approaches involve either lexical or syntactic patterns generation; or morphological features extraction between cause and effect. Therefore, this does not fit into tasks where causal documents are unlikely to have any explicit pattern matching with the query event.

2.8 Deep Causal Relations

In recent years, causality has been incorporated into standard CNN models (Narendra *et al.*, 2018), and has also been used to furnish a general abstraction over deep unsupervised learning methods (Raina *et al.*, 2009). Work by Harradon *et al.* (2018) focused on the salient concepts extracted from a target CNN network, which further helped to estimate the information captured by activations in the target network. Conversely, Li & Mao (2019) proposed the use of knowledge-based CNNs to identify causal relations from natural language text. Since 2018, with the extensive adoption of neural architectures, causality-related research has moved in a new direction. A representative example is the '*Structured Causal Model*' (Narendra *et al.*, 2018), which applies causal inference practices as part of a general framework to reason over classical CNN models. Other researchers use causality to provide a general abstraction over DNN (Raina *et al.*, 2009; Zhou *et al.*, 2015; Selvaraju *et al.*, 2016) model in a way that it can allow some arbitrary causal interventions and can answer related queries, such as- *What is the impact of the n-th filter on the m-th layer on the model's predictions?* In the same year, another group of researchers (Harradon *et al.*, 2018) employed a similar strategy, building a 'Bayesian Causal Model' that works with the salient concepts extracted from a target CNN network using 'auto-encoders' trained with a novel '*deep*' loss leveraging increased flexibility. These auto-encoders helps to estimate the information captured by activations in a target network. Li & Mao (2019) take an alternative approach, making use of knowledge-based CNNs to extract causal relations from natural language text.

Work by Kiciman (2018) and Kiciman & Thelin (2018) addresses causal infer-

ence in IR, albeit in the reverse direction, where a query describes a cause (e.g., current situation or proposed action). In this case, the results yield a list of possible effects. This approach leverages social media data and targets future effects, while we target past causes focusing on news events. We also make use of traditional term (re-)weighting techniques to reshape the retrieval algorithm subjected to the query event, which contributes towards addressing the underlying nature of cause-effect event pairs spanned across the collection. For more details on this approach, see Chapter 5.

2.9 Summary

In this chapter we provided a comprehensive literature review on causality-based research to date, emphasizing how this PhD study differs from existing literature in the field. It is worth noting that, while there is a long history of diverse work in that area, the techniques we enumerated consider causal relations either at the sentence level or within a single document. In some cases, these methods require prior knowledge about causal events, while in other cases they require some predefined lexical, syntactic, or morphological relations. However, these techniques do not cover the nuanced causes and effects in larger document collections, such as those we seek to capture via retrieval models. Therefore, to address the central research questions initially introduced in Section 1.2, in the next chapter we propose a general end-to-end architecture for causal information retrieval.

CHAPTER

THREE

STANDARD IR AND CAUSALITY

3.1 Introduction

Traditional information retrieval systems are primarily focused on finding topically-relevant documents, which are descriptive of a particular query concept. Such systems mainly concentrate on matching terms between documents and a user query, i.e., they apply *topical relevance* to address the user's *information need*. However, this might not address situations in which a user's search is intended to uncover the original causes that led to a specific event. More specifically, when dealing with sources such as collections of news articles, users often seek to identify not only documents that depict a news event but also those that elucidate the sequence of events that could have potentially culminated in the occurrence of that event. These connections can be intricate, encompassing a multitude of causal factors.

The techniques described previously in Chapter 2 consider causal relations either at the sentence level or within a single document. In certain cases, these methods incorporate prior knowledge about causal events, while in others, they rely on predefined lexical, syntactic, or morphological relationships. Nevertheless, these techniques are likely to fall short of addressing more nuanced causes and effects within extensive document collections, which is precisely what we aim to capture using retrieval models. Therefore, in this chapter we investigate this gap in the information retrieval literature, by addressing **RQ-1** in Chapter 1, i.e., whether a traditional search system is adequate for the requirements of identifying causally-relevant information or a new research paradigm to be introduced.

In order to answer the above research question, we introduce a theoretical model of causality from an information retrieval perspective. In Section 3.2,

we explain the distinctions between the proposed task and conventional retrieval problems. This chapter also aims to empirically investigate the ability of popular retrieval methods to successfully retrieve causally-relevant documents. In Section 3.3 we describe a general causal IR workflow. Beginning from Section 3.4, we investigate the extent to which the requirements of causal search diverge from those of topical search. We do this by analyzing the performance of various standard retrieval models on a pilot benchmark dataset with causal annotations.

3.2 Why Do We Need a Causal Retrieval Model?

In practice, information retrieval tasks are addressed by making use of term overlaps between a query and documents, where the notion of relevance varies depending on the task specifications. As an example of this, consider the query 'Why were the American military officers at Abu Ghraib prison accused?', and a set of sample top-ranked document excerpts for this query (see Table 3.1). If the goal is to retrieve documents pertaining to the topic itself, then any document detailing accusations against US military officers, offensive treatment of detainees, leaked images of their torture, or related actions undertaken by the US government might be considered as relevant. For example, four of the documents listed in Table 3.1 could be considered relevant, and using term overlap for retrieval fulfills the task.

On the other hand, if the task pertains to identifying causally-relevant documents recursively (i.e., $query_{event} \leftarrow cause_{event} \leftarrow cause_{event} \leftarrow ...$) for the same query, the notion of relevance would now be concentrated on *'why US military officers are accused'* and the chain of further precursory causal events at different levels. In that case, documents corresponding to reports on officers' torture stories, detainees statements accusing officers, or evidence published on newspapers might be likely to meet the requirements of the task at a given level i and for the next level $i+1$, we would be finding further prevalent causes given the effect event at level i. Hence, out of the documents listed in Table 3.1 and labeled as 'causal', only two appear to exhibit causal relevance to the query specified above.

The question arises as to whether term overlap between the query and documents is sufficient to fulfill the requirements of this task, or if alternative strategies are necessary. We delve into this question in the latter part of this

Table 3.1: Document excerpts taken from the FIRE collection (Palchowdhury. *et al.*, 2011), for a query seeking information on accusations related to Abu Ghraib prison.

	Query - Accused American military officers in Abu Ghraib prison
Topical	The US is investigating a series of allegations of abuse, including sexual humiliation, of prisoners by the US military in Iraqs Abu Ghraib jail...
RelDoc: 1	The first American military intelligence soldier to be court-martialled over the Abu Ghraib abuse scandal was sentenced today to eight months in jail...
RelDoc: 2	The torture in Abu Ghraib prison reflects the breakdown in the chain of command in the US military... ...abuse is everywhere routine. One cornerstone of this new US policy seems to be to outsource the task of interrogating....where torture is routine like Syria or Egypt...
Causal	....a female US soldier dragging an Iraqi detainee on the prison floor like a dog on a leash, one end of which is shown tied to the mans neck...
RelDoc: 1	....one detainee handcuffed to a bunk bed in Baghdads Abu Ghraib prison, his arms pulled so wide apart that his back is arched...
RelDoc: 2	....they were savagely beaten and repeatedly humiliated by American soldiers working on the night shift at Tier 1A in Abu Ghraib during the holy month of Ramazan,.... ...they were pressed to denounce Islam or were force-fed pork and liquor...They forced us to walk like dogs on our hands and knees...hitting us hard on our face and chest...

chapter. Moreover, events that are eventually reported by news media are often triggered by a series of causes spread over an extended period of time. Consequently, making the initial query more specific by adding cause-related keywords, such as 'American military officers accusation causes (or reasons)', and then using a traditional IR system appears unlikely to retrieve relevant information, since details regarding the causes of the event might not be explicitly reported in news articles. However, such causality-specific information could be discovered by analyzing a number of documents and associating the latent relationships between their terms, along with the chain of triggering causes. Once more, we investigate this matter later in this chapter.

3.3 Causal IR Model Workflow

For a causal retrieval model, we assume that the user is searching for cause-related information and there exists an agent or system to assist the user. Given a query event $\mathcal{Q} = \{q_0, q_1, .., q_n\}$, where, $q_0, q_1, .., q_n$ are the query terms, the user seeks documents containing causal information related to the query, and the search is performed over a fixed document collection C. A causal retrieval model will therefore aim to present causally-connected information in a recur-

sive fashion. That is, given an event, it finds possible causes for that event. It is worth mentioning here that such causally-connected information may have several forms, such as, it can be a sentence, a paragraph or even an entire document. Furthermore, given those causal documents which might contain any event that has further causal information need, the system then finds what might have caused those second level causal events, successively (see Figure 3.3). Here each succession represents one level in the chain of causes. We now formally describe the complete retrieval process.

We assume that a n-term query $\mathcal{Q}$ can be represented as the 0^{th} event at level-0 (i.e., no retrieval is performed yet), which we denote as $D^0_{(0,0)}$. At the next level (i.e. level-1), $D^0_{(0,0)}$ acts as a potential query and the system displays a set of top ranked k documents to the user, denoted as $\mathcal{D}^{(1)} = \{D_1, D_2, ..., D_k\}$. Here consider each document $D_j \in D$ might contain one or multiple potential event in it which might have preceding causes. Again we mention that each of these $D_j \in D$ can be a single sentence, a paragraph or an entire article containing zero to multiple causal event/s. Thus, we constitute a document D_i at level-1 as D_i= $\{D^1_{(j,1)}, D^1_{(j,2)},..., D^1_{(j,i)},...., D^1_{(j,n(D^1_j))}\}$, where $D^1_{(j,i)}$ denotes the i^{th} event identified at level-1 from the document retrieved at j^{th} rank.

Assume that at level-1, $D^1_{(j,i)}$ is recognized as a potential event which has precursory chain of causes. Consequently, $D^1_{(j,i)}$ will act as a query at level-1 and

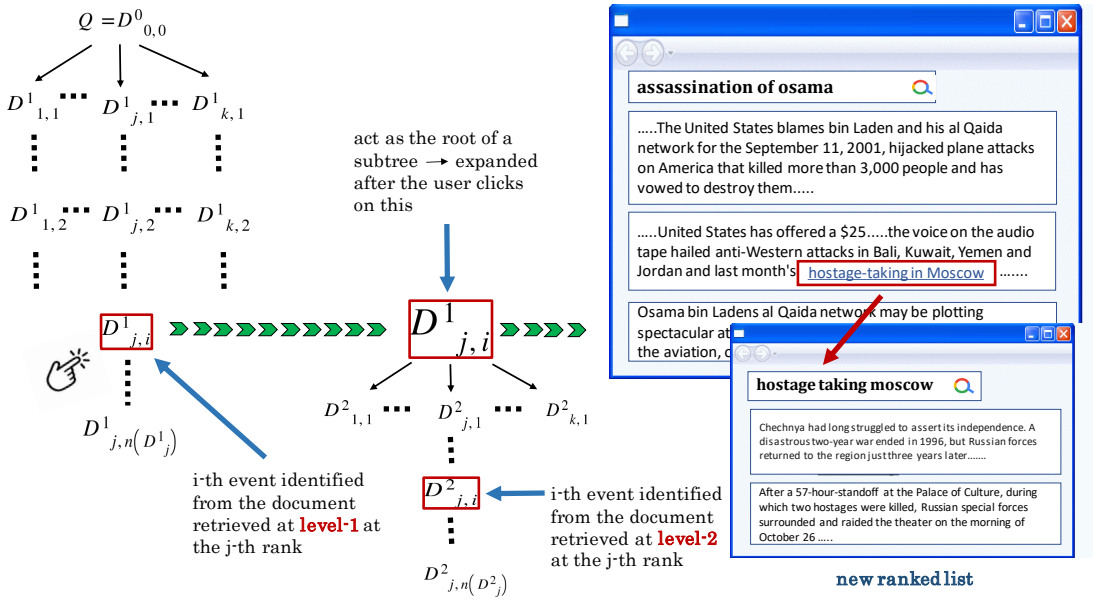


Figure 3.1: General workflow of a user's experience in a hypothetical interactive causality search interface for an input query.

retrieve a further set of k causally-relevant documents, which will be treated as level-2. In this situation, $D^1_{(j,i)}$ could be displayed to the user as hyperlink, which expands to a new set of ranked list once clicked by the user. As shown in Figure 3.3, the candidate causal event $D^1_{(j,i)}$ is considered as root of the sub-tree at level-2 and it further expands to a new ranked list of documents $\mathcal{D}^{(2)} = \{D_1, D_2, ..., D_k\}$. Thus, we repeat the same steps as at level-1 and the process continues recursively.

Evidently, at each level of this process, the main challenge involves retrieving the top-ranked causally-relevant document pertaining to the event. Therefore, in the next section we investigate the problem analytically to find the answer to our first research question – is a traditional search system adequate for the requirements of the causal information retrieval task?

3.4 Problem Investigation

If we observe the term overlaps of topical and causal documents in Table 3.1 for a given query event, the two sets of relevant documents (topical and causal) will have only a partial term overlap. With the help of a pseudo-relevance feedback technique, one might make use of high term sampling probabilities for terms that are infrequent in the pseudo-relevant document set to identify causal documents. However, prioritizing infrequent terms might always not helpful, especially in cases where the query is quite broad, such as 'Assassination of Osama bin Laden'. We illustrate this situation in Figure 3.4, where it is clear that many terms, such as *Bush, Iran, SEALs*, and *typhoid* are quite infrequent. However, these terms might not lead us to the actual causes of the event.

Therefore, to investigate the nature of causally-relevant documents and how they are coupled with that of topical one, we first conduct a number of experiments on a pilot causality dataset[1]. This represents a subset of a collection $303,291$ news articles retrieved from The Telegraph India[2]. There are 25 topics which have a causal information need and annotated relevance judgments, each related to a different news event. In the next section, we describe the dataset in detail. We measure the cosine similarity between the two associated relevance judgment sets (i.e., topical and causal) based on their term associations, as depicted in Figure 3.4. We observe that news events which

[1]Available at `https://cair-miners.github.io/CAIR-2020-website/`
[2]`https://www.telegraphindia.com`

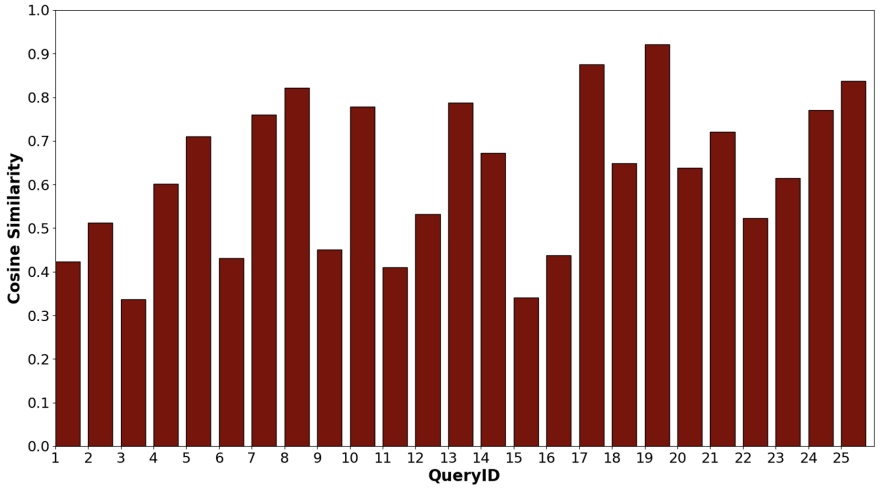


Figure 3.2: Per-query topical-causal relations in terms of relevant documents. Each bar shows the cosine similarity between topical and causal documents for any given topic in the pilot dataset.

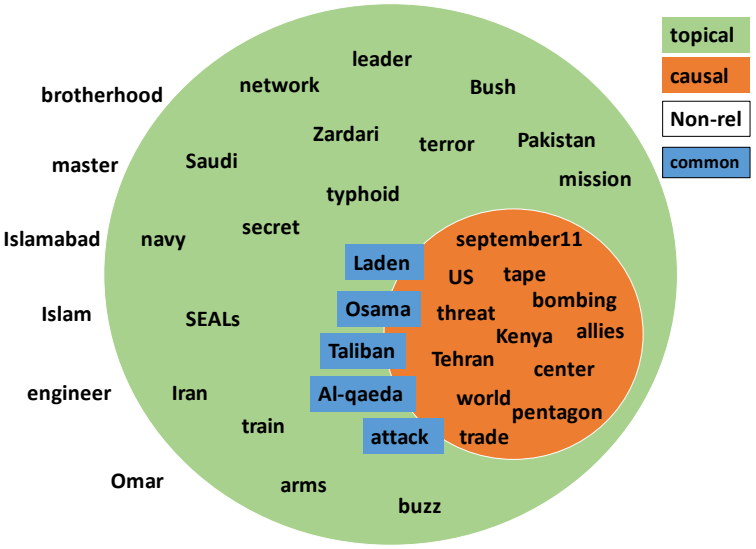


Figure 3.3: Per-query topical-causal relations with respect to terms in the top ranked-documents. Here the term associations are related to Osama bin Laden's assassination.

might have been triggered by multiple causes, such as *'Assassination of Osama bin Laden'* (topic-1 in the dataset) or involve prominent figures or organizations that are often reported in news articles, such as *'Maharashtra chief minister resigned'* (topic-3 in the dataset), have poor similarity between both set of documents. This reflects the fact that the causal results for this event have a small term overlap with the topical set. In contrast, the similarity value increases substantially if events have either a smaller number of causal factors, such as *'Carphone Warehouse terminated deal with Channel 4'* (see topic-19 in the dataset), or are related to less significant entities, for example *'Court blocks Facebook in*

Pakistan'. Such cases exhibit considerable term overlap, which we validate with retrieval experiments later in this paper. Furthermore, we explore this association with a couple of experiments and discuss our observations in the following subsections.

3.5 Pilot Causal Retrieval Dataset

Since the task of causal retrieval itself is novel, there currently exists no ready-made dataset tailored for it. This section outlines the process of creating an initial pilot dataset for the task, named as pilot causal retrieval dataset (PCRD). This dataset has been made available to the IR community (web, 2021a) with the intention of encouraging additional research focused on causality. Specifically, for our experiments, we use the English ad-hoc IR collection of the FIRE evaluation forum (Palchowdhury. *et al.*, 2011) as the target document collection. This test collection is comprised of news articles retrieved from The Telegraph India, published over a period of 10 years (2001–2011). The crawled content is structured using XML markup and organized into distinct categories or domains, including 'sports,' 'business,' and more. The entire collection comprises $303,291$ documents. Table 3.2 provides an overview of our pilot PCRD dataset.

3.5.1 Queries

As queries (topics), we used our prior knowledge about the news events in the collection to curate a fixed set of events, such that for each event it can be reasoned that a number of factors could have been responsible in leading towards it. We exclude those cases where the factors are either too obvious (e.g. mentioned in the same document also describing the query event) or the number of such factors is too small in number (i.e., ≤ 1). We also ensured that each topic is representative of an event that occurred during the period covered by the target collection, i.e. between 2001-2011.

We compiled a total of 25 queries for our study. Each query comprises of a *title* (a small number of keywords), a *description* (a well-formed sentence describing the information need in more detail), and a *narrative* (a paragraph describing the *causation*-based relevance criteria).

Table 3.2: Summary of the data used in our experiments having two types of relevance judgements, topical and causal. The columns '$|\bar{Q}|$' and '$\#\bar{R}el$' denote average number of query terms and average number of relevant documents, respectively.

Collection	#Docs	#Topics	Rel Set	$\lvert\bar{Q}\rvert$	#Rel	$\#\bar{R}el$
FIRE ad-hoc English	303,291	25	Topical	7	1,064	42.56
			Causal		578	23.12

3.5.2 Relevance Assessments

A *pool* of documents for manual relevance assessments in standard topical retrieval is usually constructed by combining the top-ranked documents identified by a number of different systems (Voorhees & Harman, 1999). In the context of causal retrieval, employing this conventional approach to construct the pool is likely to yield poor results due to two main reasons. Firstly, unlike topical IR, there is a lack of empirically established models for causal IR (in fact, the purpose of developing the manual assessments is to establish one). Secondly, relying solely on the proposed causal relevance model and standard topical relevance IR models (e.g., BM25, LM, etc.) does not guarantee the inclusion of genuinely relevant documents in the pool.

To alleviate this issue, we treated causation finding for a topic as an exploratory task involving a series of query formulations and reformulations. To aid our exploration, we used an interactive system that allowed *bookmarking* documents for future use (e.g., start exploring along a particular aspect of a potential cause of the main topic). At the end of the exploratory task, these bookmarked documents, being indicative of potentially relevant documents to establish the causal links with the query topic, were added to an assessment pool. Further, documents top-100 retrieved with standard IR and feedback models (specifically LM, BM25, and RLM) were added to this pool. Documents from this pool were then assessed with binary causal relevance judgments using both prior knowledge on the topic and the knowledge gained during the exploratory session.

3.6 Initial Experiments

3.6.1 Methods Investigated

Considering our objective to explore the concept of causal relevance for query events, we assess the effectiveness of various standard retrieval models to determine their ability to fulfill the requirements of causality. Firstly, we employ a retrieval framework with the BM25 ranking function to see if query term overlaps with the document could capture causes or not. We named this method 'BM25' as reported in Table 3.3[3]. Next, we evaluated how classical language retrieval models, specifically a linear smoothed language model performed with: (i) Jelinek-Mercer smoothing; (ii) Dirichlet smoothing (Zhai & Lafferty, 2001). We refer to these methods as 'LM-JM' and 'LM-DIR', respectively. Appendix A.2 details all the baseline retrieval models used in this thesis.

It is evident that there are specific representative terms for each query event which result in the difference between its corresponding topical and causal document sets. Usually query narrations are good resources for those representative terms as they clearly express information need for the associated task. Therefore, the next method that we investigate is 'BM25-TN' (i.e. search using Title along with Narration and rank by BM25), where we use *topic narrations* as queries, which in turn leads us to a causally-relevant document set.

Based on the intuition that terms close to the query event in an N-dimensional word vector space might be useful to capture causes, we examine whether query reformulation with *word2vec* word vectors can capture causality. For background information on the generation of word vectors, please refer to Appendix A.3. We make use of a pre-trained model, built on the Telegraph collection described previously, to help us to learn query-term associations. Once trained, this model can recommend related terms that are similar to the query terms, which might potentially be causally relevant. Thus we selected m nearby candidate terms for expanding the query to identify causal documents from the target collection, ranking them using BM25 (referred to as 'BM25-W2V').

Finally, we explored the method 'BM25-CS' (Causality Specific), where we make the query more specific to the causal information need. We consider that a user might build queries including one or more causality-indicative terms. For instance, 'Assassination of Osama bin Laden *causes* (or *reasons*)'

[3]Code available at `https://github.com/suchanadatta/AICS-2020.git`

Table 3.3: Comparison of retrieval effectiveness of various standard retrieval models both in topical and causal perspectives with respect to two different notion of relevance, topical relevance (left group) and the causal relevance (right group) using standard retrieval evaluation metrics.

	Topical				Causal			
	MAP	Recall	nDCG	P@5	MAP	Recall	nDCG	P@5
BM25	0.6400	0.9125	0.8181	0.9440	0.4690	0.7846	0.7581	0.5840
LM-JM	0.6410	0.8917	0.8148	0.9520	0.4423	0.7825	0.7411	0.5360
LM-DIR	0.6304	0.8846	0.8133	0.9040	0.4635	0.7817	0.7542	0.5840
BM25-TN	0.5774	0.8130	0.8062	0.9200	0.5272	0.9310	0.8043	0.7600
BM25-W2V	0.5390	0.7627	0.7691	0.9131	0.4410	0.6900	0.7382	0.5273
BM25-CS	0.2149	0.4829	0.4805	0.5200	0.1803	0.6170	0.4806	0.3120

might sound more reasonable than 'Assassination of Osama bin Laden', if the search intention is to to find the causes of the event. Therefore, we made use of a subset of 25 synonyms for the term 'cause' to formulate more causality-specified queries on which to search. This set includes terms such as: {*induce, lead, produce, provoke, compel, elicit, evoke, incite, introduce, kickoff, kindle, motivate, reason*}.

3.6.2 Parameter Settings

Parameter tuning not only helps us to identify the performing method, but also ensures that it provides a general setup for learning parameters in best possible way. The parameters associated with BM25, specifically k_1 (used for term frequency scaling) and b (term frequency normalization by document length), were varied in range of $[0.1, 1.5]$ and $[0.1, 0.9]$ respectively in steps of 0.1. We also tuned λ for the method LM-JM in the range $[0.1, 0.9]$ (varied in steps of 0.1), and μ for LM-DIR in $[500, 2000]$ (varied in steps of 100). Additionally, we varied the number of candidate expansion terms chosen by BM25-W2V from 50 to 200, varying in steps of 10. Table 3.3 illustrates the optimal results achieved by optimizing parameters using grid search.

In order to compare retrieval effectiveness across different approaches, we report mean average precision (MAP) along with normalized discounted cumulative gain (nDCG). We also report the comparative retrieval effectiveness with respect to recall at top 1000 ranked list and precision at first 5 retrieved result. More information about these metrics can be found in Appendix A.4.

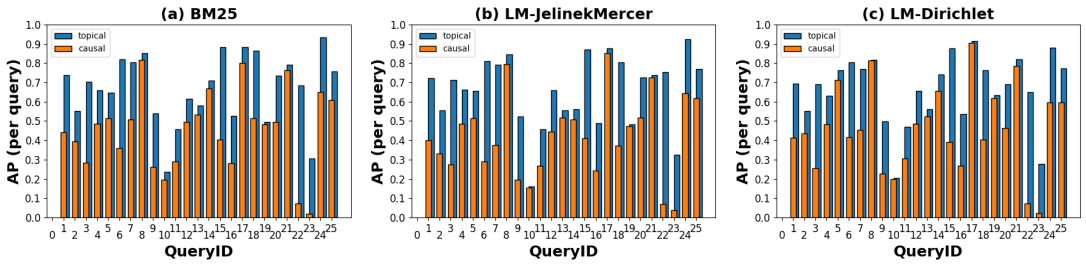


Figure 3.4: Comparison of AP scores per query for standard retrieval models.

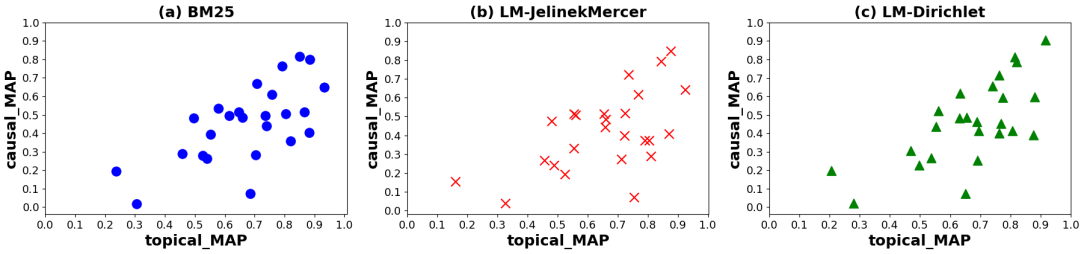


Figure 3.5: Distribution of AP scores per query for classical retrieval models.

3.6.3 Observations

From our results we make a number of observations. *Firstly*, it is clear from Table 3.3 that, irrespective of examined model architecture, the performance of traditional retrieval algorithm drops considerably as it attempts to find causal information, in comparison with topical search. *Secondly*, BM25 improves recall marginally over linearly smoothed language models. However, Dirichlet-smoothed LM appears to be as efficient as BM25 in terms of precision. *Thirdly*, as discussed in Section 3.6.1, topic narrations are expected to lead us to the causal chain of any query event and should deviate the search from topical relevance to causal. In practice, BM25-TN proves to be competent in terms of capturing more cause-related information than topical in the retrieved relevant set (i.e., increased recall), which is our primary intention. *Fourthly*, it is evident that blindly formulating any query that itself mentions the search intention (i.e., BM25-CS), or expanding a query with terms that are closely associated in the vector space of the target collection (i.e., BM25-W2V), is not adequate to harness the search scope; rather it might deviate the search intention from the actual topic to a large extent by adding noise.

To obtain a better understanding of document associations, we plot per-query MAP histograms for both topical and causal relevance for three of the standard

retrieval frameworks (see Figure 3.6.2). Also, we show the topical-causal MAP distributions for each of the 25 queries in Figure 3.6.2. In Section 3.4, we argued that cosine similarity values between topical and causal set of documents are influenced by; (i) the number of causal factors (inversely proportional); (ii) whether the query has any association with familiar entities (holds inverse relation). The results show that the MAP values obtained for sets of topics justify this argument. For example, **topic-6**: Babri Masjid demolition case against *Advani* (Indian Politician), **topic-22**: *Lalu Prasad Yadav* (Minister of Indian Parliament and was accused for multiple scams) convicted etc. achieved lower MAP for causality task as compared to topical. Conversely, for cases, such as **topic-8**: Court blocks facebook in Pakistan (single cause query and no important entity), **topic-21**: Praveen Mahajan accused (non-public figure) etc. traditional models performed well in terms of causality.

3.7 Conclusions

Causal retrieval is important in situations where a user's search is focused on finding the plausible causes of an event mentioned in the search query. For instance, when a user wishes to investigate the chain of preceding occurrences in the context of event-driven news. We have observed that there is a gap in the literature in terms of research on causality search. In an effort to mitigate this gap, we have formally defined the problem of *causal information retrieval*, and explained how it differs from traditional topical search. Furthermore, we have conducted experiments which demonstrate that traditional methods from the information retrieval literature, which are focused on topical relevance, provide limited utility in finding causally-relevant documents. This re-enforces the view that causal information retrieval remains an open challenge which is worthy of further research in the IR community.

Taking this into account, we have proposed a model for a *recursive* causal retrieval model that will help users to perform in-depth exploration in terms of causality pertaining to a news event, and the chain of causes which led to that event. It is important to note that the pilot causal dataset used in this chapter to explore how causal relevance differs from topical relevance does not support the precise identification of the exact span of causally-relevant text, which could be one or more segments, within a given news article. Rather, the pilot dataset contains more coarse-grained information. Specifically, in response to a user's causal query, this dataset offers a list of plausible documents that

might contain the underlying causes of the queried event.

To the best of our knowledge, there is no such dataset available that focuses on extracting the document excerpts indicating causally relevant information. Therefore, in the next chapter, we discuss the unique characteristics of the dataset required for this research and our course of action towards constructing it. Note that the rest of the thesis makes use of the newly annotated fine-grained dataset that focuses on capturing only the first level causal information, rather than extracting causes at lower levels. Since our proposed model in Figure 3.3 is recursive in nature, the retrieval performance at any current stage influences greatly its subsequent course of action. Thus, the more we retrieve cause-specific documents (i.e., document excerpts in our case) in response to the initial effect in the form of a query, the better the recursive queries that we identify further down the chain of causes. In contrast, a poor set of initially-retrieved documents would likely lead to poor results further down the chain. Therefore, accurately identifying first level causes represents a fundamental challenge in causal retrieval that we address in detail later in the thesis.

CHAPTER

FOUR

A NEW DATASET FOR CAUSAL RETRIEVAL

4.1 Introduction

Since the research of causal information retrieval itself is novel, there currently exists no off-the-shelf benchmark dataset for causal retrieval system evaluation. The reason for this is that our objective of capturing causal relevance is entirely different from that of traditional cause-effect textual entailed problems (Riaz & Girju, 2014; Tanaka *et al.*, 2012; Chang & Choi, 2006). In the existing textual entailment research, for a given effect, the relevant cause(s) are expected to be immediately evident for that event, e.g. *seismic plate shift causes earthquake*. In contrast, we are particularly concerned with uncovering a list of plausible causes behind a query event across the whole collection, rather than being limited to a single sentence encapsulating both cause and effect fragments.

In Chapter 3 we discussed experiments based on a small pilot dataset which was annotated at a *document level* – i.e., for any user input query having causal information need, the dataset provides us with a list of relevant documents that are likely to contain triggering causes of the query event. Each of these relevant documents might contain one or multiple causes embedded within it. However, on the basis of our initial findings, it was evident that this form of annotation is insufficiently fine-grained, as it does not allow us to distinguish between multiple distinct causes which might appear separately within the same document. In contrast, our subsequent work within this thesis centers on the development of retrieval techniques to effectively identify concise text fragments in response to user queries. Consequently, to develop and evaluate new methods to solve this type of problem requires the creation of a new causal

task-specific dataset. In this chapter, we detail the construction of a carefully-annotated dataset which we use to conduct our causality-driven IR research later in this thesis. Additionally, we have made this dataset accessible to the IR community for further research in this area[1].

We first outline the dataset characteristics in Section 4.2, providing details around the source of the data, selected queries, and relevance assessments. Furthermore, annotation protocol with a sample annotation is detailed in Section 4.3. Section 4.4 describes the characterization of the annotated dataset and finally, in Section 4.5, we discuss the challenges that were faced during the annotation process.

4.2 Dataset Characteristics

A dataset for the standard IR ad-hoc retrieval task is comprised of three components: a) a document collection, b) a set of queries, and c) a set of relevance assessments for each query.

In the context of the first component (i.e. the document collection), it is evident that the task of causal retrieval is particularly relevant in the context of a corpus of news documents. Such documents often offer diverse perspectives on contemporary events, such as elections, legal cases, and sporting events. Expert views and analysis of such events will often inform likely directions from which the current state-of-affairs might lead. Consequently, it is reasonable to assume that news articles from the past could contain information that describe the potential causes leading to a present event. Furthermore, there are situations where studying a collection of news articles over time proves valuable, enabling the tracking of event evolution, the identification of trends, and a comprehension of how past factors may have contributed to the occurrence of a specific event.

The second component (i.e. queries for the causal retrieval task) should correspond to those queries which specifically describe an event in time (e.g. 'the outbreak of a war between two or more nations', 'a major economic crisis'). Events for which there is a single self-evident cause (e.g. the cause is revealed in the article about the effect itself) are not interesting from the perspective of the causal retrieval task definition. Some concrete examples of such a direct cause-effect relationships are: i) news about heavy rainfall in a region ac-

[1]Dataset available at `https://github.com/suchanadatta/CARD-dataset.git`

Table 4.1: Excerpts of relevant documents (both topical and causal) for a query seeking information on Osama bin Laden's assassination from the dataset introduced in Chapter 3.

	Query - Assassination of Osama-bin-Laden
Topical	Pakistan's President Asif Ali Zardari today said that the whereabouts of Al Qaida leader Osama bin Laden remained a mystery...
RelDoc: 1	was a suspicion that he could be dead... Zardari said US officials had told him that they had no trace of the Al Qaida chief.
	...a leaked foreign intelligence document published....a loud buzz that Osama bin Laden may have died of typhoid in Pakistan last month, but no country would confirm anything...
RelDoc: 2	...citing an uncorroborated report from the Saudi secret services that the leader of al Qaida terror network had died. The chief of al Qaida was a victim of a severe typhoid crisis while in Pakistan on August 23, 2006, the document said...
Causal	An audio tape broadcast... sounds like the voice of Osama bin Laden threatening attacks against US allies,...If it genuinely is bin Laden's voice, makes references to recent events such as last months Bali bombings and the Chechen hostage siege in Moscow...
RelDoc: 1	warned US allies that they would be targets of new attacks...The United States blames bin Laden and his Al Qaida network for the September 11, 2001, hijacked plane attacks on America that killed more than 3,000 people, ...
	Osama bin Ladens al Qaida network may be plotting spectacular attacks inside the US,...Bin Laden and Al Qaida have been blamed by Washington for the hijacked aircraft attacks on September 11, 2001, which killed about 3,000 people...
RelDoc: 2	Al Qaida may favour spectacular attacks that meet several criteria: high symbolic value, mass casualties, severe damage to the US economy and maximum psychological trauma, the FBI said...

companied with the news about flooding in certain localities; ii) news about a mass shooting by a gunman followed by the news on his arrest. In contrast to these direct cause-event relationships, here we are interested in more complex events, where pieces of causal relations are spread across a number of different articles, with multiple opinions on subject matters open to different interpretations (e.g. it is difficult to find a single direct cause for the drop in the pound value prior to Brexit). In such cases, we might have multiple news articles which present different viewpoints, opinions, and expert analyses. When combined, these can provide a comprehensive view of the causes that play around a given event.

The criteria for the third component (relevance assessments) naturally differ in the context of causal retrieval. In this case, a document's relevance is determined by its connection to a potential cause of the specified effect in a given query, as opposed to simply topical relevance. Table 4.1 illustrates the differences between the two types of relevance for a sample query seeking informa-

tion on the assassination of Osama bin Laden. While the concept of *traditional relevance* aligns with the topic itself (the two sample documents relevant to the topic discuss the possibility of bin Laden's death), the sample *causally-relevant* documents offer insights into several events that eventually might contributed to bin Laden's death (such as 'Bali bombings', 'hijacked aircraft attacks which killed more than 3000 people', and 'severe damage to US economy', among others).

4.2.1 Document Collection

The base collection that we chose for this research is the widely-used TREC Washington Post Collection[2]. This data contains five years of news articles, from 2012 to 2017. This corresponds to over 600,000 documents covering all Washington Post content from that time period: articles, columns, and blogs. The documents are stored in 'JSON-lines' format[3], where each document is represented as a single line of JSON. The textual content of each article is broken into content paragraphs, with interspersed media such as images and videos referenced by URLs. Those links point back to the website of The Washington Post and, according to the Post, should persist at those URLs for the foreseeable future.

There are a considerable number of duplicate documents in the collection. This occurs as, at times, the Post will republish an article, and the provenance history is not represented in the data. Prior to performing any experiments, we cleaned the collection to remove documents with identical content (including the document identifier). As with the TREC News track[4], there are many near-duplicate articles in the collection, which we preserve as is.

4.2.2 Queries

As queries (topics), we used our prior knowledge about the news events in the collection to select a set of events, such that for each event it can be reasoned that a number of factors could have led to that event. We exclude those cases where the causality factor is either too obvious (mentioned in the same document also describing the query event) or the number of such factors is too

[2] `https://trec.nist.gov/data/wapost/`
[3] `https://jsonlines.org/`
[4] `http://trec-news.org/`

```
<top>
<num>Number:321</num>
<docid>9171debc316e5e2782e0d2404ca7d09d</docid>
<url>https://www.washingtonpost.com/news/worldviews/
wp/2016/09/01/women-are-half-of-the-world-but-only
-22-percent-of-its-parliaments/</url>
</top>
```

Figure 4.1: Sample query article in XML format.

small in number (≤ 1).

Specifically, we selectively chose topics from the TREC News Track[5] topic set comprising 201 queries in total. We thus ensured that a query is representative of an event that occurred during the period covered by the target collection (between 2012–2017). We curated a total of 45 topics that have causal link for our study. Each topic comprises a *docid* ('id' field in the Washington Post corpus documents), a *url* ('article url' field in the documents). Both indicate the query article. A sample topic is shown in Figure 4.2.2.

4.2.3 Relevance Assessments

As discussed in Section 3.5.2, for standard IR, manual relevance assessment typically involves creating a pool of documents by first combining the top-ranked documents retrieved by a number of systems. In the case of causal retrieval, such a pooling approach is unlikely to work well, since there currently exist no empirically-established models for causal IR. However, relying on the proposed causal relevance model and a number of standard topical relevance IR models (e.g. BM25, LM) alone cannot ensure the inclusion of all truly-relevant documents in the pool. Table 4.1 depicts the two different types of relevance. However, our focus in this thesis is only on causal relevance.

In our work, we focus our attention on the relevance space defined by the TREC News Track for the *background linking* task (Soboroff *et al.*, 2018). Specifically, we manually evaluate the relevance of documents corresponding to the same set of 45 queries used in the background linking task. We opt for this selection because the task closely aligns with our own objective of extracting causal links. More precisely, this task aims to provide evaluation data to support researchers in developing systems that can help users contextualize news

[5] http://trec-news.org/

articles as they are reading them. For instance, news websites often incorporate links to related articles in sidebars, at the end of articles, or embedded within the text.

The primary distinction between background linking and our research in causality lies in our central objective. While background linking typically entails offering a general context (which may lack causal relevance), our aim revolves around presenting users with *causally-connected contexts*. Additionally, we aim to capture the concept of causation in a more fine-grained way. Specifically, our goal is to provide users with a list of document excerpts that highlight the causal triggers of the query, rather than presenting the entire document. This makes our problem even more challenging than the established background linking task.

4.3 Annotation Process

This section describes the end-to-end annotation including the specific details on particularly how we selected annotators, protocols followed by them, annotation tool used followed by a sample annotation example for clear understanding of the readers; and finally a couple of challenges faced during annotation and how do we overcome those challenges.

4.3.1 Annotator Selection

In order to capture fine-grained causal text fragments from the long-form news articles provided by The Washington Post, we appointed human annotators to produce manual judgments for the selected queries, who would be prepared to undertake considerable background studies to understand the context of the news articles and the events which they describe. Therefore, instead of using a number of anonymous annotators who might blindly perform annotations, we employed two local academic annotators from separate disciplines and background to ensure the diversification of opinions. Both annotators worked independently over a two month period.

4.3.2 Annotation Protocol and Tool

As mentioned in Section 4.2.2, to construct the new causality-driven dataset, we select a subset of TREC News Track queries comprising 45 queries in total, with a selection criteria of having causal information need in the query. For annotation, we use a subset of the relevance judgments from the existing background linking task. That task provides graded relevance scores in the set $\{0, 2, 4, 8, 16\}$, where, 0 = the document provides little or no useful background information; 2 = the document provides some useful background or contextual information that would help the user understand the broader story context of the target article; 4 = the document provides significantly useful background; 8 = the document provides essential useful background; and 16 = the document must appear in the sidebar otherwise critical context is missing. Based on the grades above, for our annotation purpose, we consider only those judged documents having relevance scores ≥ 4, which are likely to contain *causally-relevant* background information in relation to their respective query events.

Before commencing the annotation process, we provided annotators with comprehensive guidelines concerning data handling and the specific nature of expected annotations. These instructions included sample queries along with their corresponding lists of causes. To ensure the quality of our annotations, we also curated a series of test queries. For these, we manually identified all the causal text excerpts in a recursive manner and then prompted the annotators to perform the same independently. When their assessments closely aligned with ours, we considered their annotations as reasonable and their subsequent judgments as dependable. Instances of annotator disagreements were resolved through a majority voting policy.

Label Studio[6] was used for annotation. This open-source data labeling platform supports multiple projects, users, and data types within a unified interface. It allows users to perform different types of labeling with many data formats. Also, users can integrate Label Studio with machine learning models to supply predictions for labels (pre-labels) or to perform continuous active learning. More details about this tool can be found here[7].

[6] `https://labelstud.io`

[7] `https://labelstud.io/guide/get_started.html#Quick-start`

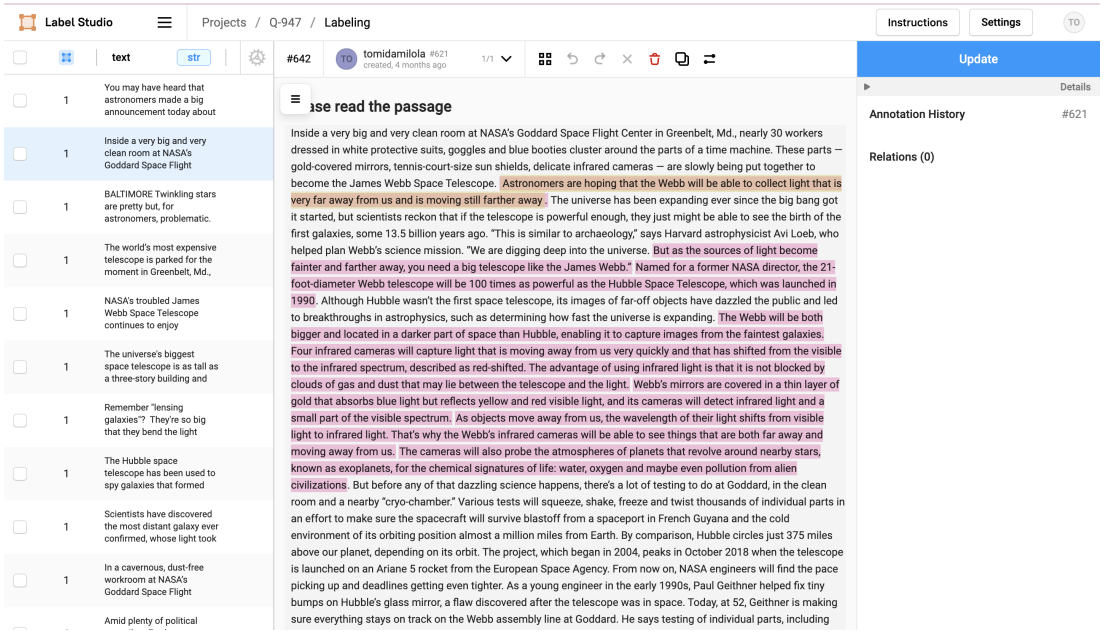


Figure 4.2: Sample annotation snippet from Label Studio interface.

4.3.3 Sample Annotation

As discussed above, annotators were provided with guidelines around how to make judgments on the given data and what they are expected to provide as outputs. We did not expect annotators to have clear prior knowledge on every topic that will be presented for annotation. Therefore, we aimed to provide sufficient background information relating to the target topics. For each query, we provided a list of full Washington Post articles that were judged as being relevant in the TREC background linking task. Annotators then manually highlighted the causally-relevant text fragments within those articles via the Label Studio interface. No restrictions were imposed on the length of the highlighted fragments, so these could range from short text snippets to fragments spanning multiple sentences.

As an example, consider an article from the Washington Post [8] on *'why might the Webb space telescope replace Hubble?'*. A sample annotation for this document using Label Studio is shown in Figure 4.3.3. Label Studio allows users to create multiple labels at the same time. In our case, for each document, we provide annotators with two separate labels, namely 'cause' (red in color) and 'causal-query' (in yellow). These two labels map to the first level causes and

[8] https://www.washingtonpost.com/national/health-science/webb-space-telescope-promises-astronomers-new-scientific-adventures/2014/11/17/4b2533f0-4e64-11e4-babe-e91da079cb8a_story.html

further subsequent causes respectively, as discussed in our general framework depicted in Figure 3.3 in Chapter 3. To illustrate the label 'causal-query' further, consider the text segment 'Astronomers are hoping that the Webb will be able to collect light that is very far away from us and moving still further away' in yellow as in the Figure 4.3.3. As per the model architecture in Figure 3.3 in Chapter 3, this causal indicative text segments at the first level is also supposed to be annotated as the second level potential query event, as user might want to explore more on the underlying 'why' aspect of this highlighted text excerpt, labeled as 'cause'. However, this thesis only aims to retrieve the first level causes, leaving the second level causal annotations as our future scope of research.

The annotation process thus yields a 3-valued tuple, `<QueryID QueryText CausalText>` for each examined text segment as shown in the Figure 4.3.3 and a list of potential next level of causal queries for the respective initial query event. Annotators finally obtained a collection of similar annotations, covering a range of topics and articles.

4.4 Characterization of the Dataset

Based on the relevance grades outlined in Section 4.3.2, for our annotation purposes we consider only those documents having relevance scores ≥ 4 for all 45 causal queries. This relevance threshold reduces the total number of documents to be annotated from $2,183$ to 907. Each of these 907 documents was annotated by two individuals. We then combine their outcomes by taking the union of the two annotation sets. This yields a total of 704 causal text fragments for 45 queries, i.e., on an average nearly 16 text fragments are extracted for each query. The overview of the dataset is presented in Table 4.2.

Note that, in our causal query set, there are 4 queries (IDs: 626, 841, 855, 904) that have only one document to be annotated, when the relevance threshold is set to ≥ 4 for annotation. In some cases, there are documents which despite being judged as highly relevant in terms of containing background information, with relevance score ≥ 4 by TREC News Track, we however do not find any relevant information as per our objective connected to causality. There are 76 such documents in the annotation set which means per query we have nearly 2 documents with no causal annotation.

It is evident that this particular causal information extraction task will involve

Table 4.2: Summary of the data manually annotated for causal research. The columns '$|\bar{Q}|$' and '$\#\bar{Rel}$' denote average number of query terms and average number of relevant documents, respectively.

Collection	#Docs	#Topics	$\|\bar{Q}\|$	#Rel	$\#\bar{Rel}$
Washington Post 2012-2017	603,074	45	11	704	16.76

some subjectivity issues in almost all the cases, leading to disagreements between annotators. The issue of tackling such disagreements are discussed in the next section. However, to estimate the average disagreements, we measure the Cohen's Kappa coefficient[9] (κ) and it is observed that κ score was in the range of $0.6 \geq \kappa \leq 0.8$ for each query, which suggests that there is broadly a high level of agreement between annotators.

4.5 Annotation Challenges

Data annotation has always been a challenging task in terms of many aspects in the retrieval literature (Soboroff *et al.*, 2018; Craswell *et al.*, 2020) retrieval task is no exception. While we had annotators from different disciplines and backgrounds, there were a number of common challenges which arose during the process.

Firstly, given the fact that news events often involve different perspectives and viewpoints, there could be high chance of low agreements among annotators. We provided annotators with a detailed sample annotation to make them aware of this subjectivity aspect of the task. However, since this form of annotation may sometimes be opinion-based rather than entirely fact-based, we aimed not to restrict annotators with a strict manual on what exact relevant information we are seeking in response to a causal query. Rather, we encouraged them to read the background around each query event and then annotate documents as per their own rationale. Following this strategy led us to face various questions from the annotators about the coverage of each topic in general.

As an example, we consider the topic in the dataset *topic 397: 'why do some Takata airbags need to be replaced twice?'*. One annotator observed that there were some causes provided in the data that explained why airbags should be

[9]It is a statistic that is used to measure inter-rater reliability (and also intra-rater reliability) for qualitative (categorical) items

initially replaced, but not necessarily why they should be replaced twice. So a question arose around whether reasons for a single replacement should also be annotated as genuine causes. In such scenario, group discussions were conducted to clarify the scope of annotation and so as to minimize disagreements.

Secondly, there were a few cases where the exact text of the topics that we obtained from the TREC News Track had to be reformulated. For example, none of the documents related to *topic 948: 'why are Australian forces accused of war crimes in Afghanistan?'* contained information about accusation of Australian forces, but rather focused on incidents relating to US forces. We had to resolve these issues by manually reformulating queries, guided by thorough reading of various background sources.

In addition to these primary challenges, there were instances where documents had been labeled as *highly relevant* according to TREC judgments, but no pertinent information regarding potential background causes was discovered. This emphasizes the significant difference between the concepts of causality and topical relevance, further underscoring the complexity of the task.

4.6 Conclusions

This chapter provides an in-depth description of the complete process involved in constructing the new causality-driven adhoc retrieval dataset, hereafter referred to as the CARD. Given the lack of an off-the-shelf benchmark dataset for causal retrieval, we have made the CARD available to the research community to support further work in the area. The remainder of the thesis will make use of this newly created dataset as a key component in our experimental evaluations. In the next chapter, we propose an unsupervised causal retrieval model, which is evaluated on both our initial pilot collection and the newly-annotated dataset.

CHAPTER

FIVE

CAUSAL RETRIEVAL: AN UNSUPERVISED APPROACH

5.1 Introduction

In earlier chapters, we discussed the distinction between conventional retrieval techniques, which aim to fetch potentially relevant documents based on user queries, and the focus of this thesis: *causal search*. In this chapter, we introduce methods to accurately retrieve a set of documents that might have *caused* or led to an *effect* or event specified in a user's query. Causality-based retrieval systems have potential applications in contextualizing events specified in queries for the purpose of analysis and decision-making. In instances where multiple potential causes exist for an event, such as the 'drop in the value of the British pound', these systems can collate diverse opinions on the possible causes. This allows users to assess the merits of each viewpoint (e.g., 'delay in implementing Brexit deal', 'uncertain situation of UK politics' etc.).

Our causality-driven IR harnesses the efficiency of the well-established traditional relevance model (RLM) (Lavrenko & Croft, 2001). We discuss the RLM in detail in Section 5.2. Precisely, after executing an initial retrieval on the collection, a top-scored set of pseudo-relevant documents is selected. There is one linear interpolation parameter which weighs the importance of the original query terms. The conditional probabilities of any term coming from the document belongs to the pseudo-relevant document set and if the term is a part of the initial query itself with respect to the same document are computed using smoothed maximum likelihood estimations. Terms with high probability values are then used as candidate terms for query expansion. This estimated relevance model seeks to select terms which are frequent in the top-retrieved

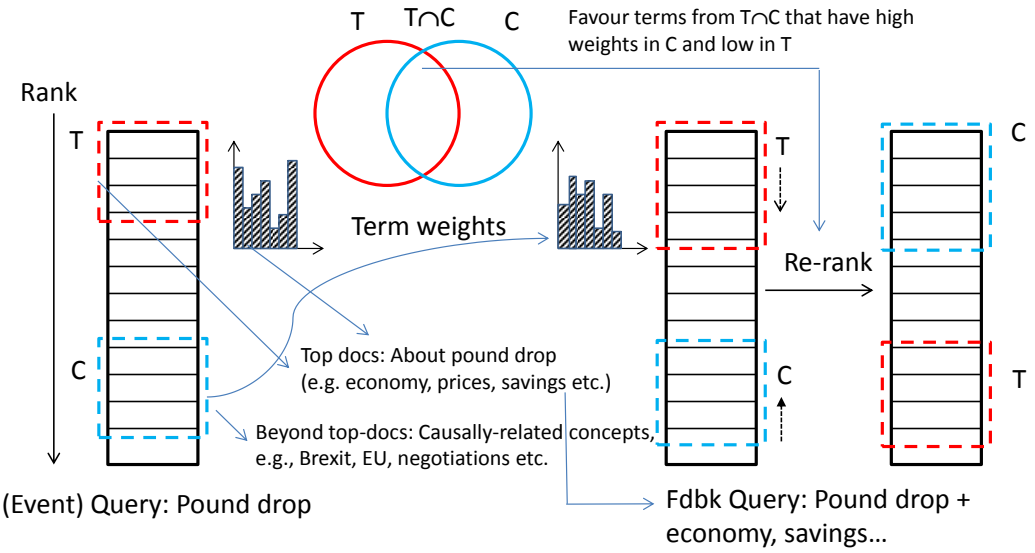


Figure 5.1: Depiction of the schematics of the proposed Factored Causal Relevance Model (FCRLM).

documents as the expansion term. As reported in other work (Lavrenko & Croft, 2001; Ganguly *et al.*, 2012; Salakhutdinov & Mnih, 2008; Roy *et al.*, 2016; Mackie *et al.*, 2023), this technique has been seen to be effective in improving the performance of topical information need. However, to satisfy a causal information need, retrieval systems need to employ expertise apart from topical similarity of the query and the relevant documents. Hence, the standard straightforward RLM approach might not be as useful for causality detection as it is proven to be for addressing topicality based retrieval scenario.

In this chapter, we seek to find out the causally relevant information in an unsupervised way following an IR perspective. In particular, for retrieving causally relevant information in response to a query, we employ a relevance feedback model with the assumption that causally relevant documents would have only a partial term overlap with the topically relevant ones (e.g. although the top-retrieved documents retrieved for the query 'pound drop' are likely not to contain terms, such as 'Brexit' or 'EU', a number of documents beyond the top-ranks would), and also a majority of these causally relevant documents would use a different set of terms to describe a number of possible causes leading to the query event (e.g., some topics related to Brexit are not correlated with the pound drop).

To address this expected behavior of term weight distribution for topical and causal relevance, in this chapter we propose a two-step feedback model, where the purpose of the first step is to estimate a distribution of terms that are topi-

cally relevant to the query, and that of the second step is to prefer those terms that are relatively infrequent in the distribution estimated in the first step, which in effect leads to deviating away from topical relevance towards potential causal relevance. The overall idea is schematically depicted in Figure 5.1. The figure illustrates our hypothesis that topical relevance (T) corresponds to the top-ranked documents, whereas the documents that are causally relevant (C) may exist further down the ranked list. We collect term distribution information from both these sets to use in a second-step retrieval process with a broader, less-focused query. Finally, we favor those documents which comprise terms that have higher weights in C and lower in T.

In the next section, we first illustrate the technical details of the traditional relevance feedback model. Next, we formally explain our proposed unsupervised RLM-based causal retrieval model, namely, Factored Causal Relevance Model (FCRLM), followed by comprehensive experiments and analysis.

5.2 Relevance Feedback and Query Expansion

Vocabulary mismatch (Furnas *et al.*, 1987) is a major challenge in the IR domain, which exists for the retrieval models discussed above. Let D be a relevant document corresponding to a user query Q. It may happen that Q and D use different sets of words to describe the same concept. In such a scenario, it might not be possible for a model to retrieve D simply on the basis of overlapping keywords. For instance, in a search related to nuclear power, we might have one document that uses the term "nuclear power", while another which uses the alternative term "atomic energy". This is where *relevance feedback* comes into play, allowing the retrieval system to refine its search and improve subsequent results. After a user submits a query, they provide feedback on the relevance of the initially retrieved documents. Based on this feedback, the system adjusts its search criteria and performs another iteration of retrieval.

Relevance feedback techniques can be grouped into three primary categories based on the way in which feedback is acquired:

- **Explicit relevance feedback**: Search results are directly specified as being relevant and non-relevant by a user.
- **Implicit relevance feedback**: User activity determines the feedback (e.g. search history, click-through rates, dwell time).

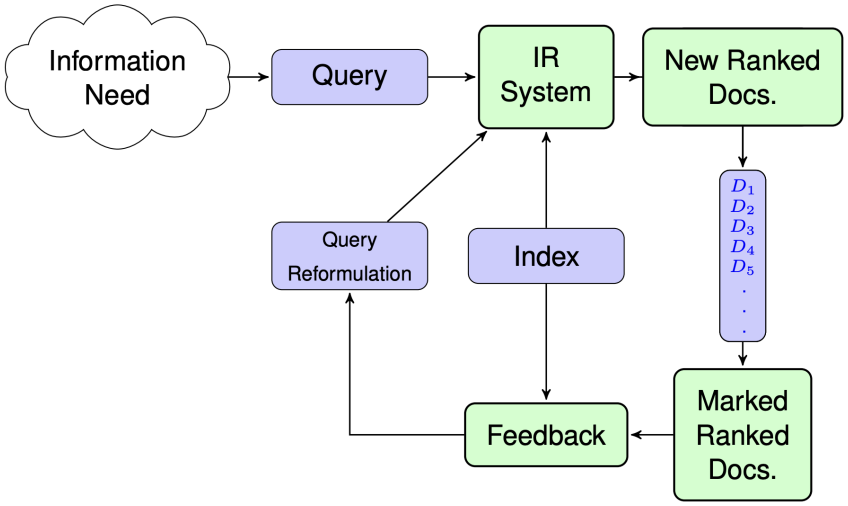


Figure 5.2: A conceptual view of explicit relevance feedback.

- **Pseudo-relevance feedback**: The top n documents in the initial search are considered to be relevant.

An illustrative example of explicit relevance feedback is shown in Figure 5.2. In practice, both explicit and implicit feedback can be difficult to obtain. Instead, systems often have to rely on pseudo-relevance feedback. Thus, we focus now on techniques involving this type of feedback. Relevance based language models (RLMs) (Jaleel *et al.*, 2004; Lavrenko & Croft, 2001) represent a widely-adopted methodology for query expansion that is dependent on pseudo-relevance feedback. RLMs hypothesize that, for a given query $Q = \{q_1, ..., q_k\}$; where $q_1, ..., q_k$ are query terms, a latent probability distribution R exists to generate both Q and the documents relevant to it (see Figure 5.2). Subsequently, an approximation of R is carried out based on Q and the relevant documents. The potential expansion terms are selected from the terms of R which have high probability weights. Thus, the performance of the query expansion technique relies extensively on an accurate estimation of R. The M top-ranked pseudo-relevant documents are usually considered as being relevant documents when training data is unavailable.

The terms of a given query Q serve as the exclusive evidence regarding the relevance model. In other words, the set $\{q_1, ..., q_k\}$ is definitively generated from R. Hence, the probability density function (PDF) of RLM is given by

$$P(w|R) \simeq P(w|Q) = \frac{P(w,Q)}{P(Q)} \sim P(w,Q) = P(w, q_1, ...q_k) \tag{5.1}$$

where $P(w|R)$ is the probability of sampling a term w from R, $P(w|Q)$ is its

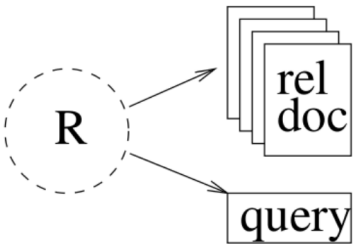


Figure 5.3: The query and its relevant documents are both random samples drawn from the underlying relevance model R.

approximation (i.e., the conditional probability of observing w along with $\{q_1, ..., q_k\}$. As per Equation (5.1), the estimation of the PDF $P(w|R)$ is essentially the same as estimating the joint probability of observing w with $\{q_1, .., q_k\}$. The joint PDF $P(w, Q)$ can subsequently be estimated using either (i) independent and identically distributed (IID) sampling, or (ii) conditional sampling. We now discuss each of these in turn.

IID sampling. Conditional sampling of w together with $q_1, ..., q_k$ is conducted to maintain the same distribution underlying a top-ranked document. Thus, the probability estimation of $P(w|R)$ can be expressed as:

$$P(w|R) \simeq P(w, Q) = \sum_{d \in D} P(w|d) \prod_{q \in Q} P(q|d) \tag{5.2}$$

Conditional sampling. In this approach, conditional sampling of w together with $q_1, ..., q_k$ is carried out from document models which are not dependent. Herein, each document model corresponds to a top ranked retrieved document. Thus, the estimate of $P(w|R)$ is given by:

$$P(w|R) = P(w) \prod_{q \in Q} P(q|w) = P(w) \prod_{q \in Q} \sum_{d \in D} P(d|w) P(q|d) \tag{5.3}$$

In Equations (5.2) and (5.3), the set of top M retrieved documents is denoted by D, while the variant of RLM in Equation 5.2 is referred to as **RM1**. Equation 5.3 is an expression of **RM3**.

As with RLMs, instead of giving importance to the terms in Q, only a term in the initial retrieved document list is considered. Work by Jaleel *et al.* (2004) achieved significant improvements over traditional models by linearly interpolating the relevance model with the query model and explicitly considering

the terms in the query:

$$P'(w|R) = (1 - \phi)P(w|R) + \phi P(w|Q) \tag{5.4}$$

Here MLE is used to compute $P(w|Q)$ and Equation (5.2) (RM1) or Equation (5.3) (RM2) can be used to compute $P(w|R)$. The contribution from both the language models is controlled by the interpolation term $\phi \in [0, 1]$. The final models are referred to as RM3 and RM4 respectively, corresponding to RM1 and RM2. It has been established by prior work (Lv & Zhai, 2009b) that RM3 outperforms all its variants, i.e. RM1, RM2 and RM4, which is why RM3 is considered as potential baseline model in the experiments carried out in the current thesis (see Equation (5.5)).

$$P'(w|R) = (1 - \phi) \left(\sum_{d \in D} P(w|d) \prod_{q \in Q} P(q|d) \right) + \phi P(w|Q) \tag{5.5}$$

Thus, RM3 and RLM are used interchangeably in the present work.

The top N terms with the highest probability distribution weights are chosen from the estimated probability distribution model R. These are subsequently used as the expansion terms. Sum normalization is applied to the weight of the N expansion terms before the task of retrieval is executed with query expansion. It is therefore vital that higher weights are assigned to the relevance model estimate exclusive to the relevant models only and assigning lower weights to the terms which are common. Otherwise, the performance may deviate from the expectation in case the document is subjected to any form of filtering which in the present case would imply excluding the metadata.

The initial retrieval and the retrieval with the expanded query can be performed with any model. Given that RLM is based on language model-based query expansion, in this thesis we conduct experiments using both baseline and expanded retrieval, employing two major smoothing techniques (JM and Dirichlet) and report the best, i.e., JM smoothing in this case. This is a common approach in the literature (Jaleel *et al.*, 2004; Lavrenko & Croft, 2001; Lv & Zhai, 2009b). For more information on language models and various smoothing methods, please refer to Appendix A.2.

Based on the concept of RLM (i.e. RM3), in the next section we formalize our 2-step factored causal retrieval framework as depicted in Figure 5.1.

5.3 Factored Causal Relevance Model

This thesis proposes a general workflow of a user's experience in a hypothetical interactive causal search interface in Chapter 3 (see Figure 3.3) where at each level the main challenge involves capturing the top-ranked causally-relevant documents pertaining to the query event. Consequently, one of the research questions that this thesis investigates is if we can develop an unsupervised system that is capable of generating a list of potential causes either as a whole document or as document excerpts at the sub-document level (refer to the RQ-2 in Chapter 1). To address this question, this chapter proposes a two-step feedback model. In the first step, the objective is to estimate a distribution of terms that are *topically relevant* to the query. Following this, the second step aims to prioritize those terms that are relatively infrequent in the distribution estimated in the first step. This deliberate shift away from topical relevance is intended to highlight potential causal relevance.

We now formalize the ideas behind the FCRLM model presented in Figure 5.1. Given the distinct inherent characteristics of these two term distributions – where topical terms align semantically closer to the query and causal terms are more subtle – we now present the details of the term estimation via a two-step feedback model (see Figure 5.3). As the first step, we intend to find a set of terms that denote a set of concepts related to the main topic of the query. The purpose of this step is to expand the set of initial query terms because a small number of initial query terms is likely not to contain adequate information to help find the causally relevant terms (and eventually the documents themselves). Formally, we estimate a standard *topical* relevance model, θ_T as

$$P(w|\theta_T) = \sum_{i=1}^{M} P(w|D_i) \prod_{q \in Q} P(q|D_i), \tag{5.6}$$

where the weights $P(w|\theta_T)$ capture the co-occurrences between terms in the M top-retrieved documents, and a query term $q \in Q$. In the next step, we estimate a relevance model θ_C as a function of the topical relevance model, θ_T, i.e., assuming the observed distribution in the RLM to be the one estimated from Equation 5.6. Considering terms from the first step as observed terms in an RLM has the effect of making the query more general, which according to our hypothesis is more useful to find causal relevance (as compared to a specific query pointing to an effect event).

The second step aims to balance a trade-off. On the one hand, terms in the

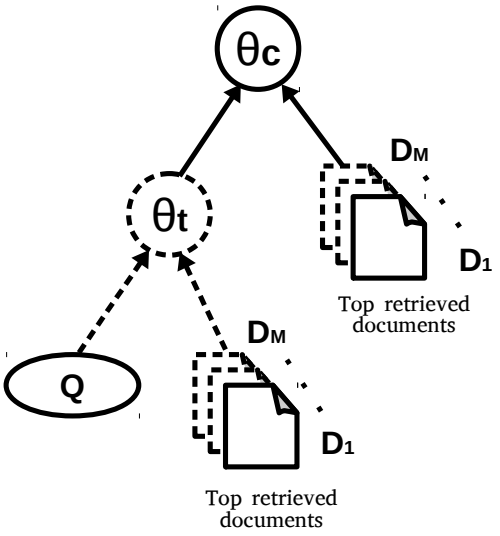


Figure 5.4: Depiction of the estimation details of the proposed Factored Causal Relevance Model (FCRLM).

causal model should correlate well with the general concepts of θ_T. On the other hand, they should not correlate well with the query (effect event). In contrast to standard RLM estimation, we employ an *odds-ratio* between the probability of a term occurring within a document retrieved in the second step (with a more general query), and the probability of the same term occurring within a document retrieved in the first step (a more specific effect event query). Formally we define:

$$P(w|\theta_C, \theta_T) = \sum_{i=1}^{M} P(w|D_i) \prod_{t \in \theta_T} \frac{P(t|D_i)}{P(t|\theta_T)}. \tag{5.7}$$

Inspecting the ratio term in the product sign of Equation 5.7 reveals that this estimation approach favors words that are:

1. Relatively infrequent in θ_T (i.e., terms not too specific to the effect event itself). This is due to a decrease in the denominator.

2. Frequent in the top-ranked documents retrieved with the more general query, (i.e., terms corresponding to a more general aspect of the query effect, which in fact could overlap with a number of potential causes). This is due to an increase in the numerator.

Finally, by substituting Equation 5.6 into Equation 5.7, we get the final estima-

tion model:

$$P(w|\theta_C, \theta_T) = \sum_{i=1}^{M} P(w|D_i) \prod_{t \in \theta_T} \frac{P(t|D_i)}{\sum_{j=1}^{M} P(t|D_j) \prod_{q \in Q} P(q|D_j)}. \tag{5.8}$$

The qualifier 'factored' in the proposed model name suggests that the topical model needs to be estimated first, which then leads to estimating the causal one. This is reflected in Equation 5.8 above.

5.4 Experiments and Results

5.4.1 Dataset

In Chapter 3, we conducted our initial investigation for causality-driven IR on a pilot dataset, PCRD, which is annotated at the document level. That is, for a given input query, any causal retrieval model is evaluated on the basis of number of causally relevant documents being retrieved, where a document corresponds to an entire news article. In Chapter 4 we introduced a new dataset, CARD, which offers more fine-grained information than PCRD, in a sense that we can evaluate models based on the number of causally-connected document *excerpts* being retrieved. In this chapter, we make use of both the datasets (see the overview of the datasets in Table 3.2 and Table 4.2) in order to examine the robustness of our proposed causal retrieval model, FCRLM.

5.4.2 Implementation Details

We used Apache Lucene[1] for indexing the collection of documents. Specifically, for the PCRD-based experiments we create the Lucene index as a collection of whole documents (i.e., news articles), whereas in the case of CARD we indexed the data at a sentence level, so that each sentence is considered to be a separate document. The proposed method and the baselines were also implemented using the Lucene API. Terms from the documents were stemmed using the Porter Stemmer, and we removed stopwords appearing in the SMART stopword list[2].

[1] `https://lucene.apache.org/`
[2] `https://www.lextek.com/manuals/onix/stopwords2.html`

5.4.3 Methods Investigated

Our first baseline is a standard IR model (specifically, LM-JM) without feedback, where λ is the smoothing parameter. Our approach adopts a two-step feedback model, specifically RLM (Lavrenko & Croft, 2001; Jaleel *et al.*, 2004). Therefore, for comparison, we use a single-step conventional RLM as our secondary baseline. To determine the effectiveness of the odds-ratio-based selection mechanism – which prioritizes common terms from both topical and causal relevance sets (θ_T and θ_C) such that terms are more likely in θ_C than in θ_T – we incorporate a relatively straightforward two-step feedback method as a third baseline. Specifically, we use standard RLM (Equation 5.6) in a subsequent step with the expanded query extracted from θ_T (i.e., $Q = \theta_T$) to re-estimate a second-step θ_T. We refer to this baseline as 'RLM-2step'.

As another baseline, we employ a standard RLM to estimate θ_C in a single step. Instead of assuming that the top-retrieved documents are relevant, we assume that the documents beyond the top-retrieved ones could be useful to estimate causal relevance. In practice, we swap the top M documents with an alternative set of M documents, $C_r = \{D_p, \ldots, D_{p+M}\}$, where $p > M$ (i.e., an interval of documents of size M following the top-M). This baseline makes use of the causal relevance assumption only and it disregards information from the top-retrieved ones. We name this baseline 'CRLM' (RLM with causal relevance).

The next methodology that we investigate involves making queries more *specific* for a given causal information need. To illustrate with an example, the query 'drop in pound value' can be made more specific for a causal need by *explicitly* adding causality-related keywords such as 'causes' and 'reasons' to the query – e.g. 'reasons for the drop in pound value' or 'causes for the drop in pound value'. The purpose of reformulating the queries in this way is to investigate whether existing retrieval models can adequately address the causal information need if queries themselves explicitly indicate that information is sought on the causes related to an event and not the event itself.

In our experiments, we automatically constructed a set of causality-related keywords, which we add to make an initial query on an event more specific to seeking the causes for the event. One potential way to create this set is to leverage a resource like WordNet (Miller, 1995) and select keywords that are related to the seed word 'cause'. However, we follow a more general approach that makes use of a pre-trained set of word vectors[3] to identify a set of words

[3]The 300-dimensional vectors trained with skip-gram word2vec on Google News data

Table 5.1: List of causality-related keywords added to an initial query to explicitly seek information on the causes of an event, rather than the event itself.

cause, induce, lead, precipitate, produce, provoke, breed, compel, elicit, evoke, hatch, incite, introduce, kickoff, kindle, motivate, reason

that are semantically related to the two seed words 'cause' and 'reason'. That is, words with corresponding vectors having a high level of similarity to the seed vectors in the embedded space. To ensure that the reformulated queries do not deviate towards different types of information needs, we manually removed any named entities appearing in the candidate set of nearest neighbors. This process yielded 17 distinct terms, as enumerated in Table 5.1.

To determine the effectiveness of standard retrieval models when applied to causally reformulated queries (using additional terms from Table 5.1) in extracting causal information, we apply all the previously described baseline methods to this set of queries. We distinguish these by appending the suffix 'CSR' (indicating causality specific reformulation), such as 'No-QE-CSR' and so on.

5.4.4 Parameter Settings

All feedback methods have a common parameter, denoted as M, which determines the number of top-ranked documents considered for feedback. Each method's parameter is individually fine-tuned using a grid search in the set of $\{10, 20, 30, 40, 50\}$. The purpose of the tuning here is not to claim that the best performing method also generalizes in the best possible way for other topics, as is the case when tuning *learning* parameters in a supervised task. Rather, our aim is to ensure a level playing field when comparing different unsupervised techniques. Another shared parameter is the number of terms, T, having the highest weight values, $P(w|R)$, which are used to calculate the KL divergence for re-ranking in a standard RLM framework (Lavrenko & Croft, 2001). In our proposed model, FCRLM, we introduce an extra parameter, T', which represents the number of top-ranked feedback terms used during the second feedback step[4]. We tune the value of both T and T' from the set $\{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$.

[4]Source code: `https://github.com/suchanadatta/Factored-Causal-RLM.git`

5.4.5 Evaluation

We evaluate our proposed FCRLM with two differently annotated datasets as explained in Section 5.4.1, i) PCRD - at the document level where an entire news article is considered as relevant and ii) CARD - at the sub-document level in which case any document snippet having causal indicative information is counted as relevant. In this section we describe our evaluation strategy with both set of relevance judgments.

Document level evaluation. When comparing retrieval effectiveness across different approaches, we report the mean average precision (MAP) along with normalized discounted cumulative gain (nDCG). More details about these metrics are in Appendix A.4. Given the exploratory nature of searching for causally-related information, where users are likely prepared to look beyond the first page of results, we consider recall as a key metric. Thus, in addition to precision-focused metrics like MAP and nDCG, we evaluate system effectiveness based on the number of relevant documents found within the top 100 results, $recall@100$ or simply, recall. Finally, we also provide scores for $P@5$, which typically captures the ability of a system to find relevant information within the first page of search results.

Sub-document level evaluation. Evaluating a system that retrieves information at the sub-document level, such as document excerpts, is likely to be more complex compared to document-level evaluation. In document-level evaluation, for a given input query, the comparison between the list of relevant documents and the retrieved relevant list is conducted through exact content matching using their document IDs. Consequently, traditional Information Retrieval (IR) evaluation metrics are well-suited for this approach. In contrast, retrieval at sub-document level, i.e., capturing causal indicative piece of text from a whole article might not manifest the concept of exact match between relevance judgments and pseudo-relevant documents. Rather, we are more encouraged to measure the similarity between the retrieved relevant and true relevant document sets. Therefore, it is likely that given a query, a pseudo-relevant document obtained in response to that query might score in a diverse range while comparing with true relevant document set. In other words, for the same pseudo-relevant document, it might score quite high similarity with a true relevant document, however, similarity could be worse for a separate judged document. In such situations, it is hard to decide if the pseudo-relevant document is worth pushing towards the top of the result list. In order to ad-

dress this issue, we consider the maximum of all the similarities obtained for a given pseudo-relevant document and binarize it (i.e., map it to $\{0, 1\}$) to indicate if the same document is relevant or not. We illustrate the scenario with an example as follows.

Example 5.4.1. Consider an input query q retrieves 5 pseudo-relevant documents (note that each document is nothing but a document excerpt), say, $\{d_1, d_2, ..., d_5\}$ and there are 3 judged documents available for q, such as, $\{r_1, r_2, r_3\}$. Now, to decide if any document from $\{d_1, d_2, ..., d_5\}$ to be labeled as relevant, we measure the cosine similarity between each of $\{d_1, d_2, ..., d_5\}$ with $\{r_1, r_2, r_3\}$. Let us consider the similarity values of d_1 with $\{r_1, r_2, r_3\}$ yield as $\{0.9, 0.3, 0.5\}$ respectively. We then take the maximum of all the similarities and map it to a binary value, which is mapped to 1 in this case as the $MAX(0.9, 0.3, 0.5) = 0.9$ and any value $>= 0.5$ is binarized as 1; 0 otherwise. Thus, we can rank the pseudo-relevant documents based on their binary relevance and consequently, standard retrieval evaluation metrics, such as MAP, recall etc. can be leveraged to compare the retrieval effectiveness among multiple retrieval systems.

5.4.6 Results

Table 5.2 shows the comparisons obtained between the proposed method and the baselines on the two different datasets. We now discuss a number of observations stemming from these results.

First, it can be seen that a standard (topical) feedback approach, such as RLM, results in a marginal improvement over the initial retrieval step (No-QE). This indicates that applying off-the-shelf relevance feedback approaches is unlikely to prove effective when seeking causally-relevant information.

Second, simply applying RLM twice in succession yields only slight enhancements in MAP and nDCG, but this comes at the expense of decreased P@10 and recall scores. This indicates that using term expansion to diversify the query for identifying causally relevant terms might not be effective. According to our hypothesis, these terms are less likely to emerge solely from top-ranked documents, as assumed in a standard feedback model. The two step feedback thus contributes to making the query more noisy (less specific to the topical information need) without effectively reformulating it to emphasize its pertinent causal aspects.

Table 5.2: Comparison of retrieval effectiveness between FCRLM and a number of baseline models reported on both pilot (PCRD) and new dataset (CARD). The improvements achieved by FCRLM are found to be statistically significant with respect to all the baselines (t-test with $p < 0.05$).

		PCRD				CARD			
		MAP	nDCG	P@5	Recall	MAP	nDCG	P@5	Recall
Baselines	No-QE	0.3212	0.4909	0.2909	0.4991	0.2201	0.3898	0.1898	0.3744
	RLM	0.3441	0.5906	0.3300	0.4833	0.2430	0.4895	0.2289	0.4471
	CRLM	0.2933	0.5280	0.3050	0.4578	0.1922	0.4269	0.2039	0.3911
	No-QE-CSR	0.1687	0.2157	0.1747	0.2731	0.0676	0.1146	0.0736	0.1563
	RM3-CSR	0.1758	0.2968	0.1894	0.2664	0.0747	0.1957	0.0883	0.1854
	CRLM-CSR	0.1541	0.3280	0.2500	0.3425	0.0530	0.2269	0.1489	0.2398
Proposed	FCRLM	**0.3645**	**0.6197**	**0.4100**	**0.5164**	**0.2534**	**0.4986**	**0.2489**	**0.4658**

Third, it is clear that CRLM, which employs a relatively straightforward heuristic of relevance feedback using documents found further down the ranked list, also fails to improve results. This indicates that terms from mid-ranged documents, which aren't closely related to the primary information need, are more likely to be off-topic than to possess causal relevance.

Fourth, the strategy of enhancing query specificity by directly adding causal terms from Table 5.1 generally leads to reduced retrieval performance across the baselines. This can be attributed to the fact that many causally-relevant documents might not explicitly state the potential reasons for an event using cause-indicative terms.

Next, we observe that our proposed model, FCRLM, outperforms all other baseline approaches on each retrieval effectiveness metric. This confirms the hypothesis that terms which are less common in the topical feedback model, but simultaneously exhibit a higher likelihood of appearing in top-retrieved documents during a second-step feedback, most accurately capture the concept of causal relevance. The factored nature of FCRLM and the use of the odds-ratio between the two factors facilitates retrieving documents that are not directly related to the query, but rather represent the potential set of causally-related precursors.

In addition to providing aggregated results, in Figure 5.4.6 we also present the per-query results, demonstrating that the improvements obtained with FCRLM are generally consistent across a number of PCRD topics, whereas the merit of FCRLM seems to be inconsistent while experimenting with CARD (see the bottom part of Figure 5.4.6). This conforms the fact that capturing the nuance causal relevance at the sub-document level is even more challenging

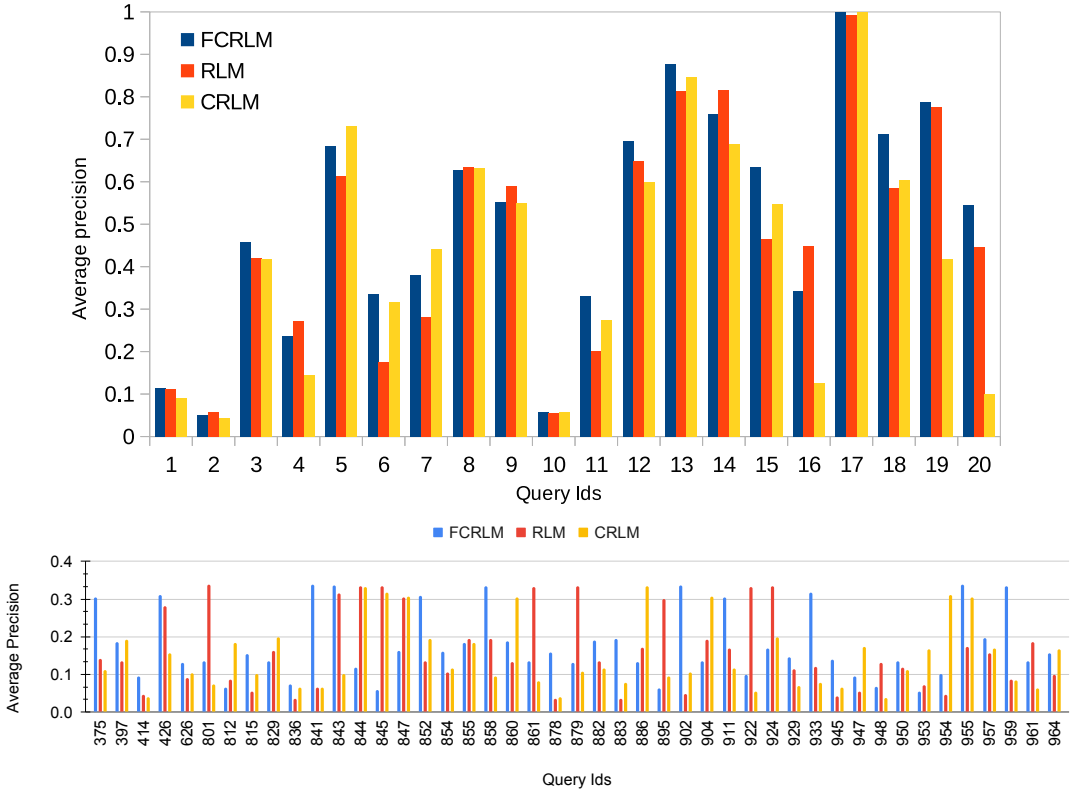


Figure 5.5: Per-query performance analysis of the queries from both PCRD (top) and CARD (bottom) in terms of average precision. The difference in the blue and red bars indicates the improvement of the proposed FCRLM over RLM and CRLM.

compared to the document level retrieval.

5.4.7 Further Discussion

We now present a qualitative comparison between the retrieval effectiveness of RLM-2step (the second best performing in terms of MAP) and FCRLM. We do this by inspecting the top set of expansion terms (ranked by their weights) from each method, as shown in Table 5.3 for a sample topic both from the PCRD and the CARD. The PCRD topic ('Why did Shashi Tharoor resign as member of parliament?') pertains to a semi-political scandal that resulted in the resignation of Shashi Tharoor, a member of the Indian parliament (see the top of Table 5.3). *Lalit Modi*[5], the then *IPL*[6] chairman, *tweeted* that Shashi's friend (with whom allegedly Shashi was having an *affair*), *Sunanda Pushkar*, received *free equity* by team *Kochi*. It is seen from the highlighted words of

[5]Emphasized words in this story indicate causal relevance.
[6]A contending team in the Indian Premier League.

Table 5.3: Top 20 expansion terms selected by RLM-2step, and FCRLM for an example query from PCRD and CARD. Causally relevant terms that are exclusively estimated by FCRLM only are bold-faced.

PCRD query	Why did Shashi Tharoor resign as member of parliament?
RLM-2step	tharoor, shashi, ipl, resign, lalit, controversi, kochi, modi, consortium, embarrass, junior, minist, kerala, sweat, congress, manmohan, bcci, clear, lead, member
FCRLM	tharoor, shashi, ipl, **pushkar**, resign, lalit, controversi, **sunanda**, kochi, modi, **bjp**, parliament, involv, member, improprieti, consortium, **affair**, explain, **tweet**, embarras
CARD query	Why might the Webb space telescope replace Hubble?
RLM-2step	hubble, space, telescope, james, light, pass, astronomy, atmosphere, planet, figure, molecule, image, solar, current, technology, faint, far, diameter, power, galaxy
FCRLM	hubble, space, telescope, james, **exoplanet**, **chemical**, astronomy, **wavelength**, planet, **blue**, **revolve**, diameter, solar, galaxy, dark, faint, **gold**, pass, **absorb**, image

Table 5.3 that FCRLM was successful in estimating high likelihoods for terms that are relevant to the aforementioned chain of events, such as 'Sunanda', 'affair', 'tweet' and so on, which its counterpart, RLM-2step, was unable to find.

We also show the merit of FCRLM on a topic from CARD, e.g. 'Why might the Webb space telescope replace Hubble?'. Similar to the PCRD topic, our proposed FCRLM shows success in capturing terms that are highly relevant to the chain of triggering causes of the query event. The important terms estimated by FCRLM with high likelihood are bold-faced (see the bottom part of the Table 5.3).

5.5 Conclusions

In order to address research question RQ-2 as introduced in Chapter 1, we have hypothesized that the nature of *causal relevance* differs from that of traditional *topical relevance*. While documents exhibiting causal relevance might share some term overlap with those deemed topically relevant to a query, we expect that many of these documents will use a unique set of terms. These distinct terms capture the various causes potentially contributing to the effects outlined in the query. On the basis of this hypothesis, in this chapter we have proposed a novel model, Factored Causal Relevance Model (FCRLM),

that seeks the potential chain of causes of an event using the idea of relevance feedback.

From the experiments reported in Section 5.4, it is evident that pseudo relevance feedback (PRF) based models can enhance average retrieval effectiveness over a sufficiently large number of queries. However, PRF often introduces a drift into the original information need, negatively impacting the performance for certain queries. In tasks like causal retrieval, where cause-effect relationships are nuanced, such drifts can significantly impair the overall efficiency of an IR model. In these circumstances, selectively employing PRF -— applying feedback only when it is likely to be beneficial – could help mitigate this problem and enhance retrieval outcomes. The merit of selective feedback motivates the rest of the thesis to focus on analyzing input queries to estimate their specificity with respect to a collection. We hypothesize that if the query is specific to any given collection, i.e. standard retrieval models are able to separate out relevant documents from the rest of the collection, in this case applying feedback might impose some deviation from the original information need. On the other hand, feedback might be significantly impactful if the input query turns out to be not specific to the collection. With this hypothesis, in the next chapter we aim to estimate the specificity of any given query in a supervised way by proposing two data-driven query performance prediction models. The merit of these proposed models further leads us to develop a selective feedback model with the ability to reduce the penalty of blind feedback.

CHAPTER

SIX

ESTIMATING QUERY SPECIFICITY

6.1 Introduction

This thesis introduces a unique notion of ad-hoc search, where the information need is not explicitly linked to the query's central topic, but instead necessitates identifying information that could have precipitated the event in the query. In Chapter 5 we proposed an unsupervised pseudo-relevance feedback algorithm, FCRLM, which relies on the simple yet effective heuristic that causally relevant terms are often not directly related to the core topic of the query. To leverage such terms, we rely on high term sampling probabilities from documents that lie further down the retrieved list of topically-relevant documents, and high co-occurrence likelihoods with the query terms to filter out potential noise. Experimental results indicate that FCRLM successfully captures causal information at both document and sub-document levels, surpassing the performance of other feedback-based benchmarks. However, our thorough analysis on per query performance both on PCRD and CARD dataset (see Figure 5.4.6) leads us to conclude that the overall retrieval effectiveness could have been enhanced if feedback was applied selectively only in cases where the users' queries required more clarity in the form of feedback.

From an IR point of view, this can be framed as the task of predicting the performance of any input query with respect to a given collection, which is also known as query performance prediction (QPP). The key idea here is if an input query turns out to be specific to a given collection, i.e. the top ranked list in response to that query holds a distinctiveness with respect to the rest of the collection, then that query is likely to be adequate to capture the relevant information on its own; if not then the initial input query is required to be reformulated with the help of feedback. With this hypothesis, this chap-

ter proposes two entirely data-driven supervised QPP approaches that aim to improve the estimation of query specificity with respect to the existing QPP approaches, and thus provides better accountability for selective feedback. It is to be noted that we confirm the merits of our proposed models via rigorous experiments on standard ad-hoc IR benchmark datasets which subsequently leads us to design our selective feedback model for our causality intended task. In the following sections, we detail different existing classical QPP approaches to give the reader a better understanding on this subject, followed by recent researches on QPP and our proposed QPP methodologies.

6.2 Query Performance Prediction

Query performance prediction, which estimates the specificity of an input query relative to a particular collection, has been an active area of research in IR over a number of years (Carmel & Yom-Tov, 2010; Cronen-Townsend *et al.*, 2002, 2006; Hauff *et al.*, 2008; Kurland *et al.*, 2011; Shtok *et al.*, 2010, 2012; Roitman, 2017; Thomas *et al.*, 2017; Zhou & Croft, 2006, 2007). This interest stems from the usefulness of QPP in gauging the satisfaction of queries' underlying information needs without requiring the availability of relevance assessments. This is particularly important because the retrieval effectiveness of IR models can vary substantially for queries with different characteristics (Zendel *et al.*, 2019), whether they range from specific to general (Carterette *et al.*, 2014), or from short to verbose (Gupta & Bendersky, 2015).

Generally speaking, QPP is intended to automatically estimate the retrieval effectiveness of a query without relying on relevance judgments (Yom-Tov *et al.*, 2005; Diaz, 2007). Instead, a QPP method typically relies on two broad sources: *i) pre-retrieval* information, which is available from the collection statistics of an index; and *ii) post-retrieval* information, which becomes available only after a top-set of documents is actually retrieved in response to a given query.

6.2.1 Pre-retrieval Approaches

A pre-retrieval estimator uses aggregated collection-level statistics (e.g., maximum or average of the inverse document frequencies of the query terms) as a measure of the QPP estimate of an input query. This is based on the assumption that queries with higher QPP estimates are likely to lead to a more

topically-coherent set of top-documents (Hauff *et al.*, 2008; He & Ounis, 2004; Zhao *et al.*, 2008), making them likely candidates for effective retrieval.

6.2.2 Post-retrieval Approaches

A post-retrieval estimator uses information from top-retrieved documents to gauge their topical distinction from the rest of the collection, with a greater difference suggesting potentially higher retrieval quality (Cronen-Townsend *et al.*, 2002). Various evidences extracted from the top-retrieved documents have been shown to be useful in the context of different post-retrieval QPP estimation methods. For instance, the KL divergence between the language model of the top-retrieved documents and the collection model as employed in Clarity (Cronen-Townsend *et al.*, 2002), the aggregated values of the information gains of each top-retrieved document with respect to the collection in WIG (Weighted Information Gain) (Zhou & Croft, 2007), the skew of the RSVs measured with variance in NQC (Normalized Query Commitment) (Shtok *et al.*, 2012), and ideas based on the clustering hypothesis for a pairwise document similarity matrix (Diaz, 2007).

Among the different ways of using retrieval status values (RSVs) for post-retrieval QPP estimation, assessing the standard deviation of retrieval scores has consistently been employed as an indicator of query performance (Pérez-Iglesias & Araujo, 2010; Shtok *et al.*, 2012; Tao & Wu, 2014). A higher standard deviation has been linked to a reduced likelihood of query drift (Shtok *et al.*, 2012; Carmel *et al.*, 2006). This has led researchers to improve the estimation of standard deviation by applying a bootstrap sampling approach to the top-retrieved list (Roitman *et al.*, 2017). Other work in this area revisited the estimation of NQC, claiming that NQC computation can be derived as a scaled calibrated-mean estimator (Roitman, 2019).

We now describe the technical details of the aforementioned post-retrieval QPP approaches, which are relevant to work conducted later in this thesis. In general, given a query, Q, a post-retrieval QPP method estimates the probability of successfully retrieving useful information in response to Q, $P(S|Q)$, as a function Φ of the query itself and its top-k retrieved document set M_k. Formally speaking:

$$P(S|Q) \approx \Phi(Q, M_k(Q)),\ M_k = \{D_i\}_{i=1}^{k}. \tag{6.1}$$

Existing post-retrieval QPP methods use different variants of the function $\Phi(Q, M_k(Q))$. We will outline some of these forms next.

Normalized Query Commitment (NQC) (Shtok *et al.*, 2012) is a commonly used post-retrieval QPP method that predicts the retrieval effectiveness of a query using the standard deviation of the document scores. This follows the hypothesis that a query with a well-defined information need is likely to lead to a more non-uniform (heavy-tailed) distribution of the RSVs. To compute the variance of the RSVs in NQC, the function Φ of Equation 6.1 takes the form

$$\Phi_{\text{NQC}}(Q, M_k(Q)) \stackrel{\text{def}}{=\!=} \frac{\sqrt{\frac{1}{k}\sum_{i=1}^{k}(P(D_i|Q) - \bar{P}(D|Q))^2}}{P(Q|C)}, \tag{6.2}$$

where $P(D_i|Q)$ denotes the similarity score of the document D_i to Q, $\bar{P}(D|Q)$ denotes the mean of the RSVs, and $P(Q|C)$ is the similarity of Q to the collection as computed by aggregating collection statistics over the query terms.

Scaled Calibrated NQC (SCNQC) (Roitman, 2019) is a generalization of NQC that involves a number of parameters, both in terms of calibration and scaling. The optimal values of these parameters are found via coordinate ascent or a grid-based exploration. This measure is formally written as

$$\Phi_{\text{SCNQC}}(Q, M_k(Q)) \stackrel{\text{def}}{=\!=} \frac{1}{k}\sum_{i=1}^{k}\left[P(D_i|Q)\left(\frac{1}{P(Q|C)}\right)^{\alpha}\left(\frac{P(D_i|Q) - \bar{P}(D|Q)}{\sqrt{P(D_i|Q)}}\right)^{\beta}\right]^{\gamma}, \tag{6.3}$$

where the expressions $P(D_i|Q)$, $\bar{P}(D|Q)$, and $P(Q|C)$ carry the same meaning as in Equation 6.2. Additionally, α is an idf-weighting factor, β is a weighting factor associated with the deviations in scores, and γ is a calibration parameter.

Weighted Information Gain (WIG) (Zhou & Croft, 2007) uses the aggregated value of the information gain of each top-retrieved document with respect to the collection. The more topically distinct a document is from the collection, the higher its gain will be. This means that the underlying hypothesis of WIG is mostly similar to that of NQC. The average of these information gains represents the topical distinctiveness of the entire set of top documents. Formally,

$$\Phi_{\text{WIG}}(Q, M_k(Q)) \stackrel{\text{def}}{=\!=} \frac{1}{|M_k(Q)|}\sum_{D \in M_k(Q)}\frac{1}{\sqrt{|Q|}}\sum_{q \in Q}\log P(D|Q) - \log P(q|C), \tag{6.4}$$

where $P(D|Q)$ denotes the score of a document D with respect to the query Q, and $P(q|C)$ denotes the collection statistics of a query term $q \in Q$. The original

authors proposed the use of $1/\sqrt{|Q|}$ as a normalization constant so that the WIG scores across queries of different lengths become comparable.

Clarity (Cronen-Townsend *et al.*, 2002) estimates a RLM distribution of term weights from a set of top-ranked documents and then computes its KL divergence with the collection model. The hypothesis is that higher the KL divergence score, the higher the QPP estimate. For estimating the clarity score of a query Q, the generic function Φ of Equation 6.1 takes the form

$$\Phi_{\text{Clarity}}(Q, M_k(Q)) \stackrel{\text{def}}{=\!=} \sum_{w \in V_{M_k(Q)}} P(w|\theta_{M_k(Q)}) \log \frac{P(w|\theta_{M_k(Q)})}{P(w|\theta_C)}, \tag{6.5}$$

where C denotes the collection, $M_k(Q)$ denotes the set of top-k retrieved documents for a query Q, and $V_{M_k(Q)}$ is the vocabulary of $M_k(Q)$. The values $\theta_{M_k(Q)}$ and θ_C correspond to the relevance model estimated from $M_k(Q)$ and the collection's language model, respectively.

UEF (Shtok *et al.*, 2010) differs from the estimators discussed so far in the sense that it involves estimating a confidence score for a set of top documents itself. This is based on the assumption that the value of the estimator itself is potentially more reliable for certain sets of top-retrieved documents than others. As a first step, the UEF method estimates the robustness of a set of top-retrieved documents by checking the relative stability in the rank order before and after relevance feedback (e.g., by RLM). The higher the perturbation of a ranked list following the feedback operation, the greater the likelihood that the retrieval effectiveness of the initial list was poor. This in turn suggests that a smaller confidence should be associated with the QPP estimate of such a query. Formally,

$$\Phi_{\text{UEF}}(Q, M_k(Q), \phi) \stackrel{\text{def}}{=\!=} \sigma(M_k(Q), M_k(\theta_Q))\phi(Q, M_k(Q)), \tag{6.6}$$

where $\phi(Q, M_k(Q))$ is a base QPP estimator (e.g. WIG or NQC), $M_k(\theta_Q)$ denotes the re-ranked set of documents post-RLM feedback, the RLM being estimated on the initially retrieved set of top-k documents $M_k(Q)$, and σ is a rank correlation coefficient of two ordered sets (e.g. Spearman's ρ or Kendall's τ).

6.3 Recent QPP Research

In Section 6.2, we introduced the basic concept of a query performance prediction (QPP) method, which estimates the likelihood of relevance of the top-

retrieved documents by measuring the distinctiveness of the information need of a query with respect to the overall topic distribution of the collection. In other words, a QPP method evaluates the feasibility of separating the top-retrieved documents from the rest of the collection based on topicality (Zhou & Croft, 2007; Hauff, 2010; Shtok *et al.*, 2012; Zamani *et al.*, 2018a; Roy *et al.*, 2019). These techniques can enable an IR system to reflect on its retrieval quality for a specific query, even in the absence of relevance assessments (Diaz, 2007). Consequently, a QPP method can enable an IR system to use this estimate to retrieve more relevant information by applying a number of additional processing steps, executed either independently of user input or through direct user engagement. Instances of user-agnostic processing include the selective application of pseudo-relevance feedback (Roitman & Kurland, 2019; Cao *et al.*, 2008). This strategy involves the automatic augmentation of a user's initial query to retrieve more informative content during a subsequent retrieval step (Lavrenko & Croft, 2001; Roy *et al.*, 2016; Zamani *et al.*, 2016; Montazeralghaem *et al.*, 2020). Methods requiring user engagement include query suggestion (Mitra *et al.*, 2014), or presenting the user with a list of potentially useful query reformulations (Feild & Allan, 2013; Li *et al.*, 2012; Rha *et al.*, 2017; Ahmad *et al.*, 2019). QPP methods are intended to allow a selective application of these user-agnostic or user-aware processing steps to further improve the quality of the retrieved information for those queries for which a QPP method estimates a low likelihood of success in finding relevant information (Roitman & Kurland, 2019).

Kurland *et al.* (2012) showed that the QPP task is equivalent to ranking clusters of similar documents based on their relevance with respect to a query. More recently, Zendel *et al.* (2019) investigated the use of alternative expressions of a user's information needs to improve QPP effectiveness, such as variants of an input query, A study by Diaz (2007) reported that a spatial analysis of vector representations of top-retrieved documents can provide useful cues for improving QPP effectiveness. This concept is also incorporated into our data-driven model, which employs convolutions over interaction matrices to harness these spatial relationships. Other standard deviation-based approaches, somewhat similar to NQC, have also been reported to work well for the QPP task (Cummins *et al.*, 2011; Cummins, 2014). Apart from the weakly supervised neural approach of WS-NeurQPP (Zamani *et al.*, 2018a), a QPP unsupervised approach that uses cluster hypothesis of word vectors in an embedded space was proposed by Roy *et al.* (2019).

Recent research has highlighted a close association between the findings of learning to rank (LTR) and QPP studies. Notably, it has been reported that the set of features that are useful for LTR can also prove beneficial for QPP (Chifu *et al.*, 2018; Déjean *et al.*, 2020). Moreover, the mechanism of two levels of interaction (both between queries and documents, and across queries) has also been reported to be useful for LTR (Mitra *et al.*, 2017). In addition to the DRMM approach (Guo *et al.*, 2016), other work proposing end-to-end LTR approaches have been proposed (Xiong *et al.*, 2017; Zamani *et al.*, 2018b). In particular, the ColBERT model was introduced by Khattab & Zaharia (2020), which is a fine-tuned BERT model (Devlin *et al.*, 2019) using pairwise ranking loss. As a precursor to end-to-end supervised approaches, unsupervised approaches have addressed term semantics by using dense word vectors (Ganguly *et al.*, 2015; Roy *et al.*, 2016; Yilmaz *et al.*, 2019).

Motivated by the recent success of end-to-end deep neural models for ranking tasks (Asadi *et al.*, 2011; Cohen *et al.*, 2018; Khattab & Zaharia, 2020; Dehghani *et al.*, 2017) and recommendation tasks (Ferrari Dacrema *et al.*, 2019; Li *et al.*, 2017; Wu *et al.*, 2019; Smirnova & Vasile, 2017), in this chapter we present two different supervised end-to-end neural approach for QPP, which we refer to as Deep-QPP and qppBERT-PL. Unlike unsupervised approaches that rely on various statistics of document score distributions, our proposed approaches are entirely data-driven. Deep-QPP is a term overlap-based encoding model that leverages the word embedding interactions in the same way as the DRMM model. In contrast, qppBERT-PL is a transformer-based encoding that uses the BERT architecture which takes as input the contextual embeddings of the terms for each pair comprising a query and its top-retrieved document. Detailed descriptions of the architecture of both Deep-QPP and qppBERT-PL are provided in Sections 6.4 and 6.5, respectively.

6.4 CNN-based Predictor: Deep-QPP

As we mentioned earlier, in contrast to unsupervised approaches that rely on various statistics of document score distributions, Deep-QPP is entirely data-driven. Furthermore, unlike weakly supervised approaches (e.g. Zamani *et al.* (2018a)), our approach does not rely on the outputs coming from different QPP estimators. In particular, Deep-QPP leverages information from the semantic interactions between the terms of a query and those in the top-documents which it retrieves. The architecture of the model comprises multiple layers

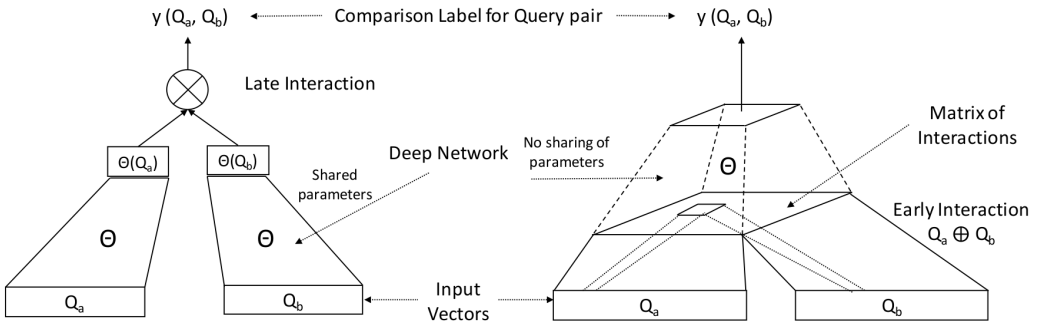


Figure 6.1: While representation-based models rely on *late interaction* involving shared parameters (left), interaction-based models, on the other hand, make use of *early interactions* transforming paired instances into a single input.

of 2D convolution filters, followed by a feed-forward layer of parameters. In the following sections, we explain the end-to-end architecture of Deep-QPP in detail.

6.4.1 Deep-QPP Model Description

The key working principle of Deep-QPP is based on capturing term-semantic interactions at two levels: first, at the *intra-query* level, to model the interaction between the queries themselves and their top-retrieved documents, and then at the *inter-query* level, to model their relative specificity measures.

6.4.1.1 Representation vs. Interaction

The fundamental difference between a representation-based model and an interaction-based model (Guo *et al.*, 2016) is illustrated in Figure 6.4.1. We see that the former constructs a representation of each instance from a pair of inputs, and then optimizes this representation so as to maximize the likelihood of predicting a function involving this pair (see the left diagram in Figure 6.4.1). In contrast, an interaction-based model first *transforms* a paired data instance into a single instance via an interaction operator $\oplus : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}^p$. Here d and p denoted the sizes of the raw and the transformed inputs, respectively.

We now discuss the type of interaction suitable for a supervised deep QPP approach. For QPP, the objective function that should be learned from the reference labels is a comparison between a pair of queries, Q_a and Q_b. More concretely, this comparison is an indicator of the relative difficulty between the queries, i.e., whether Q_a is more difficult than Q_b or vice versa.

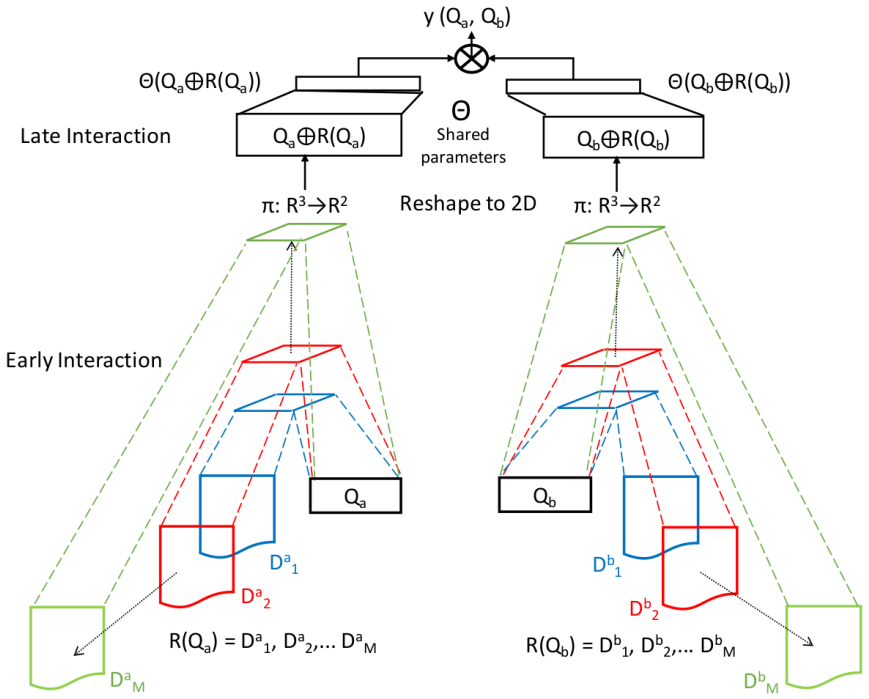


Figure 6.2: Unlike an entirely representation-based or interaction-based model (Figure 6.4.1), our model combines the benefits of both early and late interactions. This addresses: a) the interaction of the terms in the top-retrieved documents of a query with the constituent terms of the query itself; b) the characteristic pattern of these interactions to estimate the comparison function $y(Q_a, Q_b)$ between a pair of queries. Each individual query-document interaction is indicated by a different color.

While pre-retrieval QPP approaches rely solely on information from a query itself, it has been shown that post-retrieval approaches, which make use of additional information from the top-retrieved documents of a query (Zhou & Croft, 2007; Shtok *et al.*, 2012), usually yield better performance. Therefore, we also include information from the top-retrieved documents in the form of *early interactions*. We refer to these as the *intra-query* interactions. The parameters of these interactions are then optimized through a *late interaction* process between the queries, aiming to identify distinctive characteristics of these initial interactions to determine which query within the pair is more straightforward. An overview of our model is shown in Figure 6.4.1.

6.4.1.2 Query-Document Interactions

In unsupervised post-retrieval QPP approaches, the interaction between the terms in a query and those of the top-retrieved set takes the form of statically-defined functions. These measures aim to capture the distinctiveness of the top-retrieved set with respect to the entire collection. For instance, NQC, as

described by Shtok *et al.* (2012), uses the skewness of document retrieval scores to assess this distinction. In contrast, WIG, introduced by Zhou & Croft (2007), evaluates the information gain derived from the top-retrieved set compared to the whole collection. The intra-query interaction shown in Figure 6.4.1 involves computing an interaction between the terms of a query and those in its top-retrieved set of documents. This output then acts as an input when learning an optimal specificity function from the data.

Documents to consider for interaction. A common strategy for post-retrieval QPP approaches that works well as a specificity estimator involves measuring the distinctiveness between the set of documents positioned at the top ranks and the remainder of the retrieved set. The standard deviation of the document similarity scores in NQC (i.e., expected difference from the average score) provides an estimate for the topic distinctiveness of the top set.

We incorporate this insight into our approach as follows. Instead of using only a set of top-k documents, we use information from both the upper and the lower parts of a ranked list. The objective here is to capture the differences in the interaction patterns between a set of highly similar documents (found in the upper part of a ranked list) versus those that are not as similar (located in the lower part). As we shall see, this can provides useful cues for QPP.

Formally, we denote the set of documents considered for interaction with a query Q as $R(Q)$, which is comprised of a total of $\mathcal{M} = t + b$ documents, including the top-t and the bottom-b ranked ones. The index of the bottom-most document considered for interaction computation is specified by a parameter N. This means that the lower part of the ranked list, comprised of b documents are, in fact, those ranked from N to $N - b + 1$. For example, a value of $t = 10$ and $b = 20$ means that $R(Q) = \{D_1, \ldots, D_{10}\} \cup \{D_{81}, \ldots, D_{100}\}$.

In our experiments, we treat t and b as hyper-parameters (see Section 6.4.7), and restrict N to a value of 100 since it is unlikely that any evidence from documents beyond the top-100 would be useful for the QPP task.

Interaction between each query term and a document. We now describe how we compute the query-document interaction matrices for each document $D \in R(Q)$ and a query Q. As a first step, we calculate the cosine similarities between the embedded representations of terms – one from the query Q_a and the other from the document D_i^a. As is the case with DRMM (Guo *et al.*, 2016), the distribution of similarities between the j^{th} query term q_j and constituent terms of D_i^a is then transformed into a vector of fixed length p. This is done by com-

puting a histogram of the similarity values over a partition of p equally-spaced intervals defined over the range of these values (i.e., the interval $[-1, 1)$). The β^{th} component ($\beta = 1, \ldots, p$) of this interaction vector indicates the number of terms yielding similarities that lie within the β^{th} partition of $[-1, 1)$. That is

$$(q_j \oplus D_i^a)_\beta = \sum_{w \in D_i^a} \mathbb{I}\big[\frac{2(\beta-1)}{p} - 1 \leq \frac{\mathbf{q}_j \cdot \mathbf{w}}{|\mathbf{q}_j||\mathbf{w}|} < \frac{2\beta}{p} - 1\big], \tag{6.7}$$

where both $\mathbf{q}_j \in \mathbb{R}^d$ and $\mathbf{w} \in \mathbb{R}^d$, and the interaction vector $q_i \oplus D_i^a \in \mathbb{R}^p$, and $\mathbb{I}[X] \in \{0, 1\}$ is an indicator variable which takes the value of 1, if a property X is true and 0 otherwise.

Example 6.4.1. If p=4, the interval $[-1, 1)$ is partitioned into the set $\{[-1, -0.5), [-0.5, 0), [0, 0.5), [0.5, 1)\}$. For a 3-term document d, if the cosine similarities are 0.2, -0.3 and 0.4 with respect to a query term q, then $q \oplus d = (0, 1, 2, 0)$.

Collection statistics based relative weighting. The specificity of query terms (i.e., collection statistics, such as IDF) contributes to the effective estimate of QPP scores, both for pre-retrieval and post-retrieval approaches. Therefore, we incorporate the idf values of each query term as a factor within the interaction patterns to relatively weigh the contributions from the interaction vectors $q_j \oplus D_i^a$. In our proposed approach, we use a generalized version of Equation 6.7, where we incorporate the idf factor as a part of the interaction vector components, i.e.,

$$(q_j \oplus D_i^a)_\beta = \log(\frac{N_0}{n(q_j)}) \sum_{w \in D_i^a} \mathbb{I}\big[\frac{2(\beta-1)}{p} - 1 \leq \frac{\mathbf{q}_j \cdot \mathbf{w}}{|\mathbf{q}_j||\mathbf{w}|} < \frac{2\beta}{p} - 1\big], \tag{6.8}$$

where $n(q_j)$ denotes the number of documents in the collection containing the j^{th} query term q_j, and N_0 is the total number of documents in the collection.

Overall interaction between a query and a document. Each p-dimensional interaction vector computed for the j^{th} query term forms the j^{th} row of the overall interaction matrix between the query Q_a and the i^{th} document D_i^a. This matrix, $Q_a \oplus D_i^a \in \mathbb{R}^{k \times b}$ is thus given by

$$Q_a \oplus D_i^a = [(q_1 \oplus D_i^a)^T, \ldots, (q_k \oplus D_i^a)^T]^T, \tag{6.9}$$

where k is a predefined upper limit for the number of terms in a query. Zero-padding is used for the row indices exceeding the number of query terms,

i.e., $(q_j \oplus D_i^a) = \{0\}^b, \forall j > |Q_a|$. Referring back to Figure 6.4.1, each $k \times p$ interaction matrix between a query Q_a and a document D_i^a corresponds is indicated by a colored rectangle, which are shown in the planes above the queries and documents.

Interaction between a query and its top-retrieved set. Finally, each individual document-query interaction matrix, when stacked up one above the other in the order of the document ranks, yields an interaction tensor of order $M \times k \times p$. Formally, we define:

$$Q_a \oplus R(Q_a) = \begin{bmatrix} Q_a \oplus D_1^a \\ \cdots \\ \cdots \\ Q_a \oplus D_M^a \end{bmatrix} \tag{6.10}$$

6.4.2 Layered Convolutions for QPP

After constructing the local interactions of a query with its top-retrieved set of documents (i.e., the intra-query interactions), the next step is to extract convolutional features from the 3^{rd} order interaction tensor, $Q_a \oplus R(Q_a) \in \mathbb{R}^{M \times k \times b}$ between a query Q_a and its top-retrieved set $R(Q_a)$. To this end, we first need to slice the 3^{rd} order tensor into separate matrices (2^{nd} order tensors). We can then apply 2D convolution to each of these to extract distinguishing features from the raw data of query-document interactions.

Background on 2D convolution. Before describing how we slice the tensor above into matrices, we summarize the architecture that we employ to extract useful features from the lower-dimensional slices of the interaction tensor. For a detailed discussion of the 2D convolution operation, consult Rodríguez-Sánchez *et al.* (2015). Formally speaking, if $\mathbf{X} \in \mathbb{R}^{M \times P}$ represents an input data matrix, and if $\mathbf{W}^{(l)} \in \mathbb{R}^{k_l \times k_l}$ ($k_l \mod 2 = 1$, i.e., k_l an odd number) denotes the kernel weight matrix of the l^{th} layer, conveniently represented as $(W_{-\lfloor k/2 \rfloor}^{(l)}, \ldots, 0, \ldots, W_{\lfloor k/2 \rfloor}^{(l)})$, then the outputs of layer-wise convolution, generally speaking, are given by

$$\mathbf{h}_{r,c}^{(l)} = f^{(l)}\Big(\sum_{i=-\lfloor k/2 \rfloor}^{\lfloor k/2 \rfloor} \sum_{j=-\lfloor k/2 \rfloor}^{\lfloor k/2 \rfloor} \mathbf{W}_{i,j}^{(l)} \mathbf{h}_{r+i,c+j}^{(l-1)}\Big), \tag{6.11}$$

for each $l = \{1, \ldots, L\}$ (L being the total number of layers), where $\mathbf{h}^{(l-1)} \in \mathbb{R}^{M^{(k-1)} \times P^{(k-1)}}$ is the output obtained from the previous layer of the convolution filter, with $h^{(1)} = X$, $M^{(1)} = M$ and $P^{(1)} = P$. The function $f^{(l)}$ is an aggregation

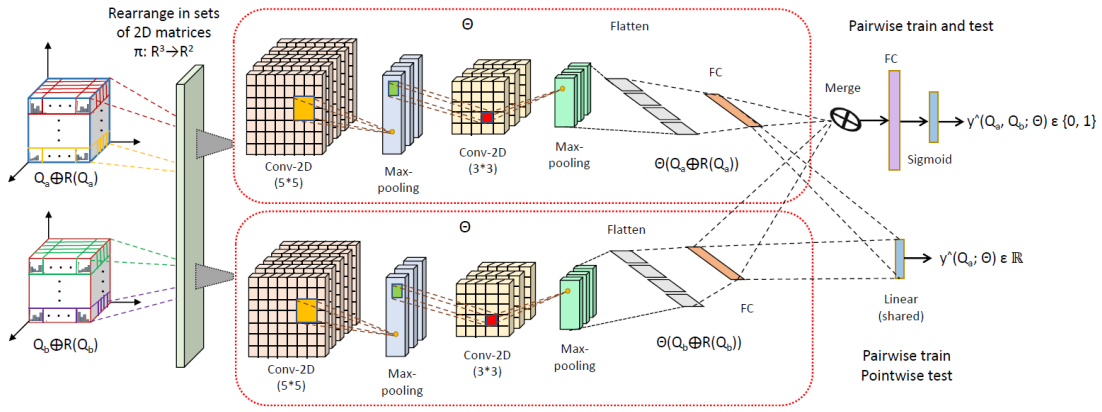


Figure 6.3: Our proposed end-to-end QPP model comprising a Siamese network of shared parameters of layered convolutional feature extraction, followed by either i) merge (concatenation) and a fully connected (FC) layer with a Sigmoid loss for pairwise testing (Equation 6.13) yielding a binary comparison indicator between a pair, or ii) a linear activation layer with pairwise hinge loss for pointwise testing yielding a score for a given query (Equation 6.14). Since the interaction for MDMQ and SDSQ are matrices with a single row only, the two layers of convolution filter sizes for these approaches are 1×5 and 1×3 (see Section 6.4.3).

function that, generally speaking, progressively reduces the size of the convolutional representations, $\mathbf{h}^{(l)}$, across layers. Aggregation methods commonly applied in computer vision include the MaxPooling (Christlein *et al.*, 2019) and AvgPooling (Shen *et al.*, 2014) functions.

Late interactions with convolutional features. A more detailed view of the late interaction across a query pair is shown in Figure 6.4.2. Referring to the notation from Equation 6.11, we employ $L = 2$ convolution layers, and use $k_1 = 5$ and $k_2 = 3$ (i.e., a 5x5 filter for the first layer and a 3x3 for the second one). The aggregate function of each layer l, $f^{(l)}$, is set to the MaxPooling operation.

After extracting the convolutional features for each query vs. top-documents interaction tensor (shown as the two cuboids towards the extreme left of Figure 6.4.2), we employ the standard practice of merging the convolutional filter outputs of each query into a single vector (shown as the 'merge' operation) (Wang *et al.*, 2020; Byerly *et al.*, 2020). Following the merge operation, which now combines abstract features extracted from the local interactions of the two queries into a single vector, we apply a fully connected dense layer. Depending on whether we test the network in a pointwise or pairwise manner, the loss function is set to either the `Sigmoid` function or a function that seeks to maximize the accuracy of the comparison function between pairs. Section 6.4.4

provides more details on the network training process.

6.4.3 Reshaping the Interaction Tensor

There exists a number of different choices for slicing up the interaction tensor of Equation 6.10 into a set of matrices for the purpose of separately applying 2D convolution on each and then combining the features. This is shown as the *reshaping* function $\pi : \mathbb{R}^3 \mapsto \mathbb{R}^2$ in Figure 6.4.2. We now discuss each alternative, examining their respective strengths and weaknesses in the context of QPP.

In our nomenclature, we categorize reshaping functions based on whether the information across i) top-retrieved documents is merged together, or ii) query terms are merged together. A part of the name thus uses the characters `D` to denote the top-retrieved set, and `Q` to denote query terms. To indicate 'merging', we use the letter '`M`' and to denote its counterpart, we use the letter '`S`' (separate). For instance, the name `MDMQ` means that the information from both top-documents and query terms are merged together.

Merged Documents Merged Query-terms (MDMQ). This is the most coarse-grained way to reduce the dimensionality of the interaction tensor of order 3 (Equation 6.10). It involves reducing the $M \times k \times p$ tensor to a flattened vector of dimensionality Mkp, which can still be imagined to be a matrix of dimension $1 \times Mkp$, thus allowing 1D convolutions to be applied. This method extracts abstract features at an aggregate level rather than for individual documents separately. However, this may not be desirable because, in standard QPP methods such as WIG and NQC, an individual contribution from each document score is responsible for the predicted specificity measure.

Separate Documents Merged Query-terms (SDMQ). This corresponds to the most intuitive strategy for grouping an interaction tensor, $Q \oplus R(Q)$, by considering the i^{th} row for each $i = 1, \ldots, M$, $Q \oplus D_i$, as a matrix of dimension $k \times p$. This allows the extraction of abstract features from each document separately in relation to the whole query. Thus, it takes into account the compositionality of the query terms while simultaneously avoiding the conflation of information across documents. This approach is consistent with the implementation of most unsupervised post-retrieval QPP methods in practice.

Merged Documents Separate Query-terms (MDSQ). Contrary to grouping the interaction tensor row-wise, for this method we slice out the constituent

matrices column-wise. Each matrix is thus of dimension $M \times p$, and there are a total k of them, on each of which we apply 2D convolution for feature extraction. This QPP method thus does not take into account the compositionality of the constituent query terms while considering the semantic interactions. Rather it treats the whole set of top-retrieved documents in an aggregated manner, which is also somewhat counter-intuitive because a document at the very top rank should be treated in a different manner from the very bottom one, i.e. the one at M^{th} rank.

Separate Documents Separate Query-terms (SDSQ). This is the most fine-grained approach, which considers every interaction vector between the j^{th} query term and i^{th} document (see Equation 6.8 as a separate candidate for convolutional feature extraction. Each such interaction vector between a query-term and a document is of dimension p and there are a total of Mk such vectors. As with the MDMQ approach, we apply 1D convolution on these vectors.

A point to note is that, although Figure 6.4.2 shows the convolution filters as 5×5 and 3×3, for MDMQ and SDSQ approaches, these filters are of size 1×5 and 1×3 respectively.

6.4.4 Deep-QPP Training

In Chapter 3 we empirically confirm the inadequacy of traditional short queries to capture causal relevance. To overcome this problem, we apply a relevance feedback approach to expand queries with causally related terms, which proved to be effective in Chapter 5. However, Chapter 5 also illustrates that expansion can often cause query-drift (see Figure 5.4.6). As a result, it frequently requires performing a comparison between a pair of queries to determine which one is more specifically relevant to the collection. Such a decision-based framework is likely to be helpful in minimizing query-drifts. The later part of this thesis, in fact, offers one such decision-based pipeline. On the other hand, traditional pre/post-retrieval QPP estimators rank a list of input queries based on their individual specificity score to the collection; the higher the score, the better. Such query ranks are useful when judging the retrievability of IR models.

With this motivation, Deep-QPP network in Figure 6.4.2 is trained based on instances of query pairs with two different objectives – pointwise and pairwise. In the pairwise case, the network directly learns the comparison function, i.e., a binary indicator of the anti-symmetric relation between a query pair. On

the other hand, the pointwise objective aims to predict a QPP score, instead of the relative ordering of specificity between a pair. Before describing these objectives, we first provide details around obtaining the data instances and the reference labels.

Instances and ground-truth labels. Given a training set of queries $\mathcal{Q} = \{Q_1, \ldots, Q_m\}$, we construct the set of all unordered pairs of the form (Q_a, Q_b), such that $\forall a, b \leq m$ and $b > a$. The reference label, $y(Q_i, Q_j)$, of a paired instance is determined by a relative comparison of the retrieval effectiveness obtained by a system with a target metric. The retrieval effectiveness, in turn, is computed with the help of the available relevance assessments. Formally speaking, if $\mathcal{M}$ denotes an IR evaluation measure (e.g., average precision or AP), which is a function of i) the known set of relevant documents - $\mathcal{R}(Q)$ for a query $Q \in \mathcal{Q}$, and ii) the set of documents retrieved with a model $\mathcal{A}$ (e.g., LM-Dir (Zhai & Lafferty, 2001)), then

$$y(Q_a, Q_b) = \text{sgn}(\mathcal{M}(Q_a; \mathcal{R}(Q_a)) - \mathcal{M}(Q_b; \mathcal{R}(Q_b))), \tag{6.12}$$

where $\text{sgn}(x) = 0$ if $x \leq 0$ or 1 otherwise. For all our experiments, we use either AP@100 or nDCG@20 as the target metric $\mathcal{M}$. As the IR model, $\mathcal{A}$, we employ LM-Dir with the smoothing parameter $\mu = 1000$ following QPP literature (Shtok *et al.*, 2012). We emphasize that the results of our experiments are mostly insensitive to the choice of either target metric or IR model.

Pairwise objective. For this objective, the Deep-QPP model is trained to maximize the likelihood of correctly predicting the indicator value of the comparison between a given pair of queries. The purpose here is to learn a data-driven generalization of the comparison function. During the testing phase, the model outputs a predicted value of the comparison between a pair of queries unseen during the training phase. The output layer for the pairwise objective thus constitutes a Sigmoid layer, predicting values of $y(Q_a, Q_b)$ (Equation 6.12) as a function of the network parameters denoted as $\hat{y}(Q_a, Q_b; \Theta)$. During the training phase, the parameter updates seek to minimize the standard squared loss between the ground-truth and the predicted labels:

$$\mathcal{L}(Q_a, Q_b) = (y(Q_a, Q_b) - \hat{y}(Q_a, Q_b; \Theta))^2 \tag{6.13}$$

Pointwise objective. For pointwise testing, the network takes a single query Q as a test input, rather than a pair of queries. Instead of predicting a binary indicator comparison, the network outputs a score $\hat{y}(Q; \Theta)$ that can be used as

Table 6.1: Characteristics of the datasets used for Deep-QPP experiments.

Collection	#Documents	Topic Set	$\|\mathcal{Q}\|$	Avg. Q_len	Avg. #Rel
Disks 4 & 5	528,155	TREC-Rb	249	2.68	71.21
MS MARCO Passage	8,841,823	TREC-DL	79	2.42	52.34

an estimated measure of specificity of Q. To allow for pointwise testing, the output from the shared layer of parameters goes into a linear activation unit predicting a real-valued score $\hat{y}(Q;\Theta)$, which is a function of one query (rather than a pair). This can be seen in the bottom-right part of Figure 6.4.2. Instead of training the network on a merged representation of a query pair, the loss function includes separate contributions from the two parts of the network corresponding to each query. The objective here is to update the parameters for maximizing the comparison agreements between the reference and the predicted scores. Specifically, we minimize the following hinge loss:

$$\mathcal{L}(Q_a, Q_b) = \max(0, 1 - \text{sgn}(y(Q_a, Q_b) \cdot (\hat{y}(Q_a;\Theta) - \hat{y}(Q_b;\Theta)))) \tag{6.14}$$

6.4.5 Experiments

Collections. We evaluate Deep-QPP on two standard ad-hoc IR test collections, namely TREC Robust (comprised of news articles) and the MS MARCO passage collection (Nguyen *et al.*, 2016), which comprises of over 8.8M passages, along with a set of over 500K topics and relevant document pairs. For evaluation, we used the depth-pooled queries of TREC DL tasks from 2019 and 2020 (Craswell *et al.*, 2019, 2020). Table 6.1 provides an overview of the data used in the experiments.

Train and test splits. Since our proposed Deep-QPP method is a supervised one, we first require a training set of queries to learn the model parameters and then a test set for evaluating the effectiveness of the model. Following the standard convention in the literature (e.g. Zamani *et al.* (2018a); Shtok *et al.* (2012); Zendel *et al.* (2019)), we employ repeated partitioning (30 times) of the set of queries into 50:50 splits and report the average values of the correlation metrics (see Section 6.4.7) computed over the 30 splits.

A major difference in our setup, in contrast to existing QPP approaches, is the use of a training set. While the training set for unsupervised approaches serve the purpose of *tuning the hyper-parameters* of a model by grid search, in our case, it involves *updating the learnable parameters* of the neural model using a

method such as stochastic gradient descent.

Hyper-parameter tuning. The most common hyper-parameter for existing unsupervised QPP approaches is the number of top-M documents considered when computing the statistics on the document retrieval scores, as in NQC and WIG, or to estimate a relevance feedback model, as in Clarity and UEF (see Section 6.4.6 for more details). We tune this parameter via grid search on the training partition. Following the setup of Zamani *et al.* (2018a), the values used in grid search were $\{5, 10, 15, 20, 25, 50, 100, 300, 500, 1000\}$.

6.4.6 Methods Investigated

We compare our supervised Deep-QPP approach with a number of standard unsupervised QPP approaches, and also a more recent weak supervision-based neural approach (Zamani *et al.*, 2018a). We do not include QPP methods that leverage external information, such as query variants (Butman *et al.*, 2013). Using query variants has been shown to improve the effectiveness of unsupervised QPP estimators and it is also likely that including them in our supervised end-to-end approach could lead to further improvements. However, since the main objective of our experiments is to investigate whether a deep QPP model can outperform existing methods, we leave the use of external data for future exploration. Furthermore, we have excluded pre-retrieval QPP approaches, like Average IDF or Max IDF, as numerous previous studies have shown that post-retrieval approaches tend to outperform them (Cronen-Townsend *et al.*, 2002; Zhou & Croft, 2007; Shtok *et al.*, 2012; Zamani *et al.*, 2018a).

Unsupervised approaches. We consider a number of baselines that solely make use of term weight heuristics to measure the specificity estimates of queries. These methods mainly differ in the way in which they calculate the similarity of the top-retrieved set of documents from the rest of the collection. The unsupervised baselines (Clarity, NQC, WIG, UEF) to which we compare our proposed approach were previously discussed in Section 6.2.2.

Supervised approaches. Our choice of supervised baselines is guided by two objectives: first, to show that (strong) supervision using the ground-truth of relative query performance is better than the existing approach of weak supervision on QPP estimation functions (Zamani *et al.*, 2018a), and second, to demonstrate that a mixture of both early and late interactions is better than purely content-based ones (see Figures 6.4.1 and 6.4.1). Specifically, we consider the following:

- **Weakly Supervised Neural QPP (WS-NeurQPP)** (Zamani *et al.*, 2018a). The key difference between WS-NeurQPP and Deep-QPP lies in the source of information used and also the objective of the neural end-to-end models. The former uses weak supervision to approximate the scores of individual QPP estimators so as to learn an optimal combination. As inputs, it uses the retrieval scores, along with the word embedded vectors. However, unlike our approach, WS-NeurQPP does not consider the interactions between terms, making it a purely representation-based approach.

- **Siamese Network (SN).** This approach is an ablation of the Deep-QPP model (Figure 6.4.2). Here instead of feeding in the interaction tensors between a query and its top-retrieved documents, we simply input the dense vector representations of queries in pairs. We experiment with two different types of dense vector inputs - one where we used pre-trained RoBERTa vectors (Liu *et al.*, 2019) obtained using the HuggingFace library (web, 2021b), and the other, where we used the sum of the skip-gram (Mikolov *et al.*, 2013) word embedded vectors (trained on the respective target collections) of constituent terms as the dense representation of a query for input. We name these two ablations as **SN-BERT** and **SN-SG**, respectively.

- **No Intra-Query Interaction.** As another ablation of Deep-QPP, we only use the interaction between the terms of the query pairs themselves. The interaction tensor between a pair of queries is a 2^{nd} order tensor, i.e., a $k \times p$ matrix. This is a purely interaction-based method, and in principle, is similar to DRMM (Guo *et al.*, 2016), with the added layer of 2D convolutions. Thus, we denote this baseline as DRMM.

6.4.7 Experimental Settings

Implementation. We used the Java API of Lucene 8.8 (luc, 2021) for indexing and retrieval. This library is also used to implement the existing unsupervised QPP baselines. Both Deep-QPP and the supervised baselines were implemented using Keras (ker, 2021). The code for our proposed method is available for research purposes[1].

Metrics. As discussed in Section 6.4.4, the Deep-QPP model can be trained

[1] `https://github.com/suchanadatta/DeepQPP.git`

using either the pairwise and the pointwise objectives. The pointwise test use-case is the standard practice in existing QPP studies, where given a query, a QPP model predicts a score indicative of the retrieval effectiveness. For this use-case, we evaluate the effectiveness of the QPP methods with standard metrics used in the literature: a) Pearson's-r correlation between the AP values of the queries in the test-set and the predicted QPP scores; b) a ranking correlation measure, specifically Kendall's τ between the ground-truth ordering (increasing AP values) of the test-set queries and the ordering induced by the predicted QPP scores. For additional information about these metrics, please refer to Appendix A.4.2.

In the pairwise case, the network is presented with pairs of queries from the test set, for which it then predicts binary indicators of the relative order of queries within the pairs. As a QPP effectiveness measure, we report the average accuracy of these predictions, i.e., whether a predicted relation as given by the $\mathrm{Sigmoid}$ output from Deep-QPP, $\hat{y}(Q_a, Q_b; \Theta)$, matches the ground-truth that $AP(Q_a) < AP(Q_b)$. Since $\hat{y}(Q_a, Q_b; \Theta) \in [0, 1]$, we binarize this value to $\{0, 1\}$ with the threshold of 0.5, thus indicating a prediction of whether Q_a is a more difficult query than Q_b or vice versa.

Deep-QPP hyper-parameters. For both our proposed method and for the semantic analyzer component of the weakly supervised baseline WS-NeurQPP, we use skip-gram word vectors of dimension 300 trained on the respective document collections with a window size of 10 and 25 negative samples. Another hyper-parameter in Deep-QPP is the number of intervals (bins) p used to compute the interactions in Equation 6.8. Both in Table 6.3 and 6.3, we report results with $p = 30$ (as per the settings of the DRMM paper (Guo *et al.*, 2016)). We later investigate the effect of varying this parameter on the effectiveness of Deep-QPP (see Figure 6.6).

We observed that, after conducting a number of initial experiments, excluding the idf of terms in the interaction tensors consistently led to worse results than when including them. Therefore, in all our experiments with Deep-QPP, we use the idf-weighted interactions as given in Equation 6.8. Another hyper-parameter used in the Deep-QPP model to prevent overfitting is the dropout probability, which we initially set to 0.2 based on our experimental findings.

Table 6.2: A comparison of the QPP effectiveness between Deep-QPP, and a set of unsupervised and supervised baselines (shown in the 1^{st} and the 2^{nd} groups, respectively). The average accuracy and the correlation values (see Section 6.4.7) of Deep-QPP over the best performing baseline - WS-NeurQPP, are statistically significant (t-test with over 97% confidence).

Methods	Metric : AP@100					
	TREC-Robust			TREC DL		
	Pairwise	Pointwise		Pairwise	Pointwise	
	Accuracy	P-r	K-τ	Accuracy	P-r	K-τ
Clarity	0.6251	0.4863	0.3140	0.6120	0.1911	0.0641
NQC	0.6720	0.5269	0.4041	0.7030	0.2654	0.1518
WIG	0.6613	0.5440	0.4279	0.6829	0.2492	0.1920
UEF	0.6941	0.5523	0.4154	0.7217	0.3162	0.1959
SN-BERT	0.6613	0.5208	0.4169	0.6902	0.2317	0.1441
SN-SG	0.6349	0.5112	0.3987	0.6273	0.2110	0.1154
DRMM	0.5871	0.4730	0.3710	0.6023	0.2014	0.1141
WS-NeurQPP	0.8123	0.7215	0.5090	0.7727	0.5192	0.2828
Deep-QPP (MDMQ)	0.7857	0.6988	0.4981	0.7414	0.4636	0.2495
Deep-QPP (SDSQ)	0.7210	0.6303	0.4018	0.6844	0.4208	0.2401
Deep-QPP (MDSQ)	0.8006	0.7203	0.4989	0.7426	0.4840	0.2575
Deep-QPP (SDMQ)	**0.8420**	**0.7404**	**0.5434**	**0.8045**	**0.5532**	**0.3130**

6.4.8 Results and Analysis

Table 6.2 and 6.3 present the QPP results for all the methods considered in our experiments. Firstly, we observe that the existing supervised approach for QPP, WS-NeurQPP, outperforms the unsupervised approaches (NQC, WIG and UEF), which is consistent with the observations reported by Zamani *et al.* (2018a).

Secondly, we see that the ablation baselines of Deep-QPP involving a purely representation-based approach (SN-BERT and SN-SG), or a purely interaction-based one (DRMM), perform worse than Deep-QPP. This is primarily because these baselines lack the additional source of information—interactions of queries with the top-retrieved set of documents, which Deep-QPP is able to leverage. This observation also reflects the fact that post-retrieval QPP approaches, incorporating additional information from top-documents, typically outperform their pre-retrieval counterparts (Shtok *et al.*, 2012).

Third and most importantly, it is apparent that Deep-QPP outperforms WS-NeurQPP, which confirms the hypothesis that explicitly learning the relative specificity of query pairs with an end-to-end (strongly) supervised model is better able to generalize than a weakly supervised approach which learns an optimal combination of statistical predictors.

Another observation is that the SDMQ version of the reshaping function π :

Table 6.3: Similar to Table 6.2, we compare the QPP effectiveness of Deep-QPP with a set of unsupervised and supervised baselines. The only difference here is that the QPP effectiveness is evaluated based on nDCG@20.

Methods	Metric : nDCG@20					
	TREC-Robust			TREC DL		
	Pairwise	Pointwise		Pairwise	Pointwise	
	Accuracy	P-r	K-τ	Accuracy	P-r	K-τ
Clarity	0.6118	0.3529	0.2462	0.6101	0.0923	0.0714
NQC	0.6689	0.4261	0.3017	0.6916	0.3105	0.1987
WIG	0.6629	0.3915	0.2407	0.6710	0.2780	0.1823
UEF	0.6792	0.5029	0.3510	0.6925	0.3320	0.1854
SN-BERT	0.6529	0.5023	0.3624	0.6724	0.2241	0.1334
SN-SG	0.6147	0.4736	0.3561	0.6231	0.2049	0.1283
DRMM	0.5629	0.4038	0.3119	0.6004	0.1927	0.1201
WS-NeurQPP	0.7973	0.5913	0.4126	0.7614	0.3928	0.2337
Deep-QPP (MDMQ)	0.7632	0.5649	0.3619	0.7189	0.3509	0.2185
Deep-QPP (SDSQ)	0.7284	0.5112	0.3065	0.6753	0.3124	0.2014
Deep-QPP (MDSQ)	0.7824	0.5601	0.3245	0.7037	0.3518	0.2100
Deep-QPP (SDMQ)	**0.8371**	**0.6315**	**0.4614**	**0.7903**	**0.4431**	**0.2554**

$\mathbb{R}^3 \mapsto \mathbb{R}^2$ (see Section 6.4.3 and Figure 6.4.2) turns out to be the most effective, as we might expect. This also conforms to the way in which unsupervised QPP approaches generally work, i.e., by first making use of the information from each top-retrieved document (e.g. its score in NQC and WIG) and then computing an aggregate function over them (e.g. their variance in NQC, and relative gains in WIG).

To further compare Deep-QPP to WS-NeurQPP, we report the training-time efficiency of both approaches in Figure 6.4. Due to a much larger number of trainable parameters and larger input dimensionality (dense word vectors instead of interactions between the dense vectors), WS-NeurQPP takes significantly longer to execute compared to Deep-QPP. The total number of trainable parameters for WS-NeurQPP is 4.7M, approximately 2.5X the number of parameters in Deep-QPP (1.9M).

Hyper-parameter sensitivity of Deep-QPP. Figure 6.5 demonstrates that using the top-10 and bottom-10 documents for interaction computation (as explained in Section 6.4.1.2) yields the best results. This suggests that selecting an appropriate number of documents is important for learning the QPP comparison function, avoiding both overly small and overly large document sets.

Figure 6.6 shows the effects of different bin sizes p (as used in Equation 6.8), when computing the interactions between queries and the documents retrieved at top and bottom ranks. A value of 30 turned out to be optimal, which is similar to the previously reported optimal value for interaction computation

Figure 6.4: Deep-QPP, in addition to being more effective than WS-NeurQPP, also outperforms WS-NeurQPP in terms of training time due to the much smaller number of parameters (1.9M vs. 4.7M).

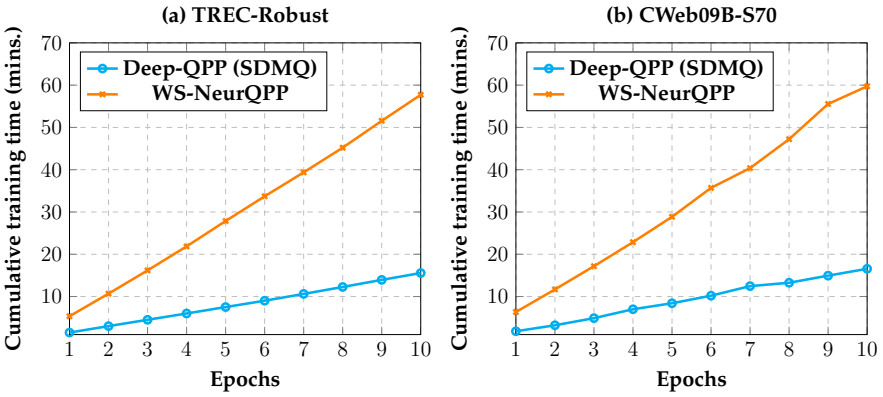
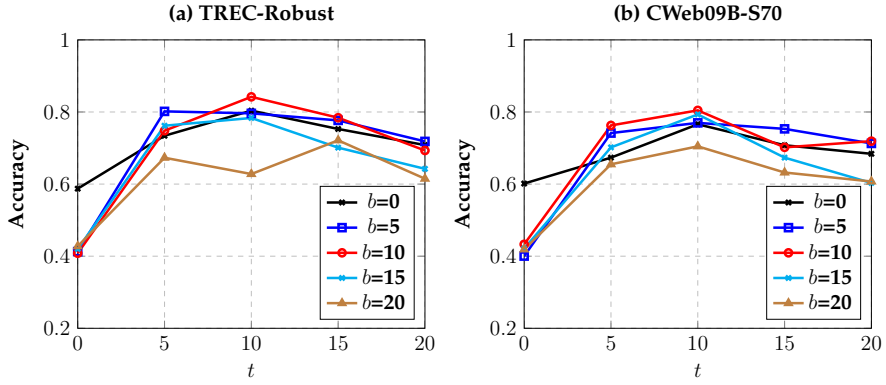


Figure 6.5: Sensitivity of Deep-QPP to the number of top (t) and bottom (b) documents included for interaction computation on QPP effectiveness (see Section 6.4.1.2). The limiting case of $(t, b) = (0, 0)$ corresponds to the situation when we simply use the interaction between query terms themselves (i.e., the DRMM baseline).

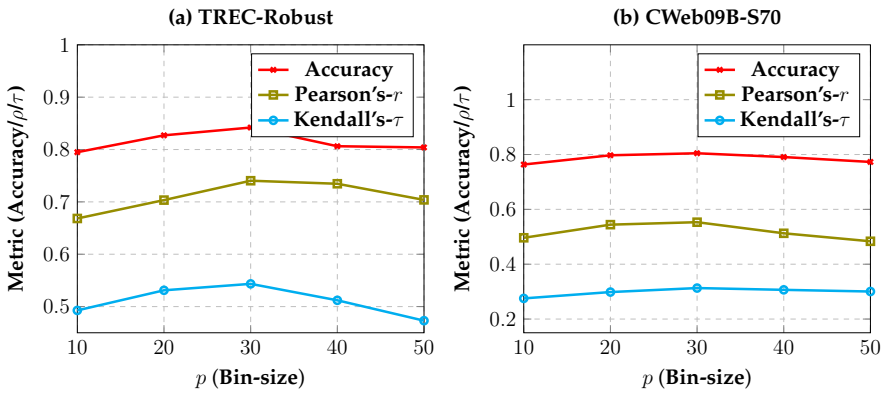


in the LTR task (Guo *et al.*, 2016).

6.4.9 Discussion

Our proposed method Deep-QPP is trained on pairs of queries to capture their relative retrieval performance. The main disadvantage of a pairwise strategy is that the number of pairs is quadratic with respect to the training set size, thus causing a significant increase in training time. As a solution, Arabzadeh *et al.* (2021) showed that a pointwise approach, which makes use of a cross-encoding based interaction of the BERT vectors of constituent query and document terms (an architecture that has been shown to be effective for passage and document relevance ranking (Nogueira & Cho, 2019; MacAvaney *et al.*,

Figure 6.6: Sensitivity of Deep-QPP with respect to the bin-size, p.



2019)), can perform well at QPP in practice. Instead of predicting a relative measure of query difficulty across a pair, the training objective in this case seeks to directly predict a retrieval effectiveness measure (e.g. MRR@10).

Motivated by the success of BERT-based QPP models, such as BERT-QPP (Arabzadeh *et al.*, 2021), in the next section we propose a groupwise query estimation framework that combines both cross-query and cross-document information across groups to learn the query performance predictor. Similar to BERT-QPP, this is also a regression-based model that predicts individual score for each query-document pair. However, the final QPP score per query is obtained by aggregating predictions in each group. Both of our proposed CNN and BERT-based QPP approaches played the role of building blocks of our proposed selective feedback model in the next chapter (i.e. Chapter 7).

6.5 Transformer-based Predictor: qppBERT-PL

6.5.1 Model Description

We now provide the detailed architecture of our proposed transformer-based query performance predictor. The objective of the model is to predict the retrieval effectiveness or QPP score for a query Q, as a function of the query itself and the ordered set of top-retrieved documents $M_k = \{D_1, D_2, \ldots, D_k\}$ in the form $\psi(Q, M_k) \mapsto \mathbb{R}$.

Network architecture. To model the input M_k as an ordered set of documents, we make use of a recurrent neural network to encode the documents in sequence. Specifically, we use LSTM units (Hochreiter & Schmidhuber, 1997;

Chen *et al.*, 2017) for modeling the sequence (see Figure 6.5.1). As discussed previously, an important decision choice with QPP estimators relates to the size of the top-retrieved document set. In the case of many popular QPP methods, such as NQC and WIG, these have been reported to work well when using information from the top-100 documents. Encoding such long sequences of documents, which are themselves sequences of words, is likely to introduce noise into the process. Consequently, we segment the ordered set M_k into equal sized partitions (chunks) of smaller ordered sets, namely

$$M_k = \bigcup_{i=1}^{\lfloor k/p \rfloor} \{M_k^{(i)}\} = \{D_1, \ldots, D_p\} \cup \ldots \{D_{k-p+1}, \ldots, D_k\}, \tag{6.15}$$

where $p(< k)$ is the size of each partition. Each partitioned list of documents $M_k^{(i)} = \{D_i, D_{i+1} \ldots, D_{i+p}\}$, along with the query Q, constitutes an input instance. We employ a BERT-based cross-encoder architecture to model the interactions between the query and the document terms, followed by an LSTM-encoded representation of this interaction sequence (see Figure 6.5.1). Formally,

$$\begin{aligned}
\Theta_{Q,D_i} &= \text{BERT}([\text{CLS}]q_o, q_1, \ldots, q_{|Q|}[\text{SEP}]d_1, d_2, \ldots, d_{|D_i|}) \\
\Theta_{Q,M_k^{(i)}} &= \text{LSTM}(\theta_{Q,D_i}, \ldots, \theta_{Q,D_{i+p}}; \theta_{LSTM}) \\
\hat{y}(Q, M_k^{(i)}) &= \text{SOFTMAX}(\phi^T.\Theta_{Q,M_k^{(i)}}),
\end{aligned} \tag{6.16}$$

where θ_{LSTM} and ϕ denote the parameters corresponding to the LSTM and a fully connected layer, respectively, Θ_{Q,D_i} denotes the BERT (Devlin *et al.*, 2019) encoding of the query-document pair (Q, D_i), and $\hat{y}(Q, M_k^{(i)})$ denotes the predicted output through a `SOFTMAX` layer.

We name our proposed method as **qppBERT-PL**, based on the following naming convention. Since the model makes use of a sequence of chunked documents, it can be categorized as a listwise-document approach, which is why we include the suffix 'L' (denoting Listwise). On the other hand, since we incorporate the relative position (rank) information of the documents (so as to distinguish one input chunk from another), we include the suffix 'P' in the name to denote the Position or absolute ranks of documents.

Incorporating rank embeddings. Since we provide the information from the top-k retrieved list as separate chunks $M_k^{(i)}$, each of size p, we require a way to establish a link between these input chunks. A convenient way of doing this is by incorporating positional information into the embedded documents

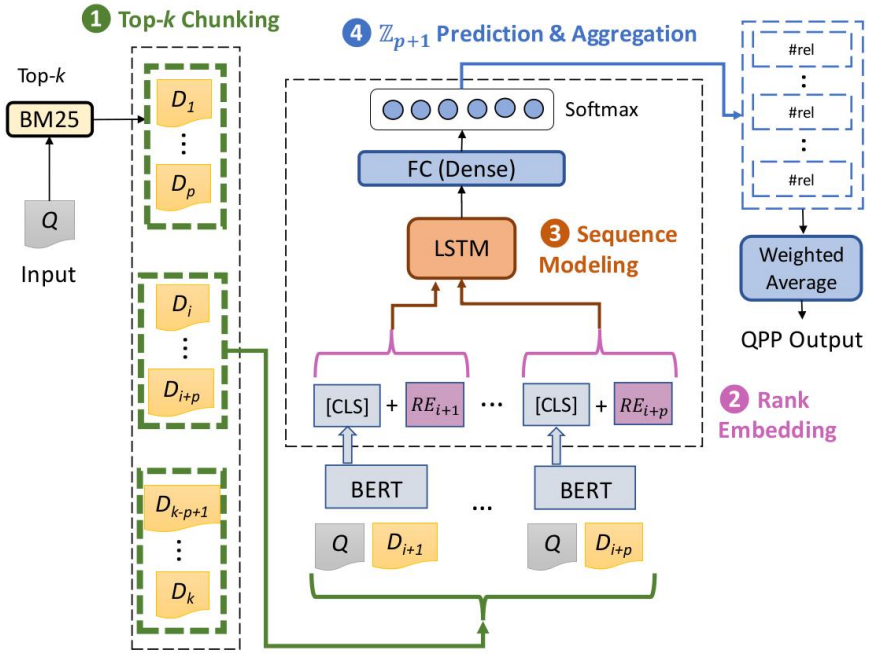


Figure 6.7: Schematics of our proposed neural model 'qppBERT-PL' for a given query Q and a list of top-ranked k documents, $\{D_1, D_2, \ldots, D_k\}$ partitioned into $\lfloor k/p \rfloor$ chunks, each of size p. The query-document cross-encoded representations ($\Theta(Q, D_i)$) for each chunk along with the rank of a document in the form of BERT positional embeddings (Devlin *et al.*, 2019), are then encoded via LSTMs (to represent each chunk as a sequence rather than a set of documents). A fully connected layer (FC) then follows terminating at a $p+1$ dimensional Softmax representing the probability of finding r relevant documents within this p-sized chunk ($r \in \{0, 1, \ldots, p\}$). Through our experiments, we show that (1) top-k chunking, (2) rank embedding, (3) sequence modeling, and (4) count prediction/aggregation are all important components of our approach.

(Θ_{Q,D_i}). We borrow an idea from the BERT model itself, which uses positional embeddings to indicate the relative positions of each token in the input text. Similarly, for a chunk $M_k^{(i)}$ comprised of documents $\{D_i, D_{i+1}, \ldots, D_{i+p}\}$, we add an embedding tied to i (i.e., the document's rank) to Θ_{Q,D_i} representation. It is important to note that our objective here is to model the sequence of *documents*, and not the sequence of the words themselves, as is the case in many NLP tasks.

Training objective. The ground-truth values that the network seeks to predict correspond to the number of relevant documents in each partition $M_k^{(i)}$, which is an integer between 0 (none of the documents are relevant) and p (all documents are relevant). To account for the likelihood of these $p+1$ possible integer (categorical) values, we model the output layer as a $p+1$ dimensional Softmax.

Inducing query ranks from the estimator. As a final step, we compute a

Table 6.4: Average number of relevant documents for each set of queries used for the evaluation of qppBERT-PL.

	MS MARCO Dev	TREC-DL'19	TREC-DL'20
#Queries	6980	43	54
Avg #Rel	1.1	58.2	38.7

weighted average from the outputs of the network, $\hat{y}(Q, M_k^{(i)})$, predicted for each p-sized partition of the top-k documents to obtain an aggregated score. The rationale for using a weighted average is to favour the predicted relevance contributions from the chunks towards the top of the ranked list, in comparison to the ones that are at the bottom. Formally, we compute:

$$\psi(Q, M_k) = \sum_{i=1}^{\lfloor k/p \rfloor} \frac{\hat{y}(Q, M_k^{(i)})}{i}. \tag{6.17}$$

The resulting $\psi(Q, M_k)$ scores are subsequently used to sort the set of input queries in descending order.

6.5.2 Experimental Setup

Datasets. We conduct experiments on the well-known MS MARCO passage collection (Nguyen *et al.*, 2016), which comprises of over 8.8M passages, along with a set of over 500K topics and relevant document pairs. To evaluate the results of our QPP experiments, we follow the approach of Arabzadeh *et al.* (2021), using the validation set of relevance-assessed queries, commonly known as "Dev". As in Arabzadeh *et al.* (2021), we report our experiments on the queries used in the TREC DL tasks from 2019 and 2020 (Craswell *et al.*, 2019, 2020). Table 6.4 provides an overview of the three datasets used to evaluate qppBERT-PL.

In contrast to the MS MARCO queries, those appearing in the TREC DL tasks use depth pooling for relevance assessment, and hence are associated with a higher number of relevant documents, on an average, per query. TREC DL uses a graded form of relevance. For our experiments, as per the official metric used in the track, we treat only the relevance level of 2 when computing the AP values. In line with prior work, we estimate the performance of BM25 results and we perform indexing and BM25 retrieval using PISA (Mallia *et al.*, 2019). For QPP evaluation, we employ two widely-used correlation measures:

Pearson's-r and Kendall's-τ (denoted as P-r and K-τ, respectively). While the former is a standard statistical correlation measure between two sets of values, the latter is a measure of the relative number of agreements in ranking order between two ordered sets of values.

Unsupervised baselines. As reported in Table 6.2 and 6.3, we include a number of traditional unsupervised QPP approaches as baselines: Clarity (Cronen-Townsend *et al.*, 2002), Weighted Information Gain (WIG) (Zhou & Croft, 2007), Normalized Query Commitment (NQC) (Shtok *et al.*, 2012) and UEF (Shtok *et al.*, 2010), a method which applies a base QPP estimator to aggregate estimates from a number of subsets sampled from the top-retrieved set. The contribution from each subset depends on the relative stability in the rank order before and after relevance feedback with that set. As the base estimator for UEF, we used NQC, since it yields the most effective results compared to other post-retrieval estimators. In addition, we employ a generalized model of NQC which claims that NQC computation can be derived as a scaled calibrated-mean estimator, namely SCNQC (Roitman, 2019). All of these QPP baselines are discussed in detail in Section 6.2.2.

In line with the work by Arabzadeh *et al.* (2021), we use a small subset to tune the hyper-parameters of the unsupervised baseline QPP approaches. This comprised of 100 queries randomly sampled from the MS MARCO Dev topic set. The two main tuned hyper-parameters were the number of top-retrieved documents considered for the post-retrieval QPP methods (such as NQC, WIG, and SCNQC), and the number of documents for relevance feedback in UEF. The baseline method SCNQC involves a number of hyper-parameters related to scaling and calibrating the NQC estimation, which we tune by grid search similar to Roitman (2019).

Supervised baselines. As our first supervised point of comparison, we consider WS-NeurQPP (Zamani *et al.*, 2018a). Unlike BERT-QPP and our proposed approach, this method is not an end-to-end supervised model since it requires inputs in the form of the outputs from several QPP estimators. It then employs weak supervision to learn an optimal combination of the estimators. As the next supervised baseline, we include the cross-encoder version of BERT-QPP (Arabzadeh *et al.*, 2021) (as it outperforms the bi-encoder version). We refer to this baseline as BERT-QPP.

Ablations derived from the baseline BERT-QPP. Arabzadeh *et al.* (2021) only used a single document to train a regression model for predicting the target

Table 6.5: A summary of extensions of the originally proposed BERT-QPP method (Arabzadeh *et al.*, 2021), which act as ablations in our study. The original BERT-QPP method is included as one of our baselines. Prediction type$[0, 1]$ indicates a regression model, whereas $\mathbb{Z}_n$ denotes an n-class classification.

	Model	Type	Pred. Type	Seq.	Chunked	RE
g	BERT-QPP	Baseline	$[0, 1]$	✗	✗	✗
h	+ Seq.	Ablation	$[0, 1]$	✓	✗	✗
i	+ Seq. + RankEmb	Ablation	$[0, 1]$	✓	✗	✓
j	qppBERT-PL	Proposed	$\mathbb{Z}_{p+1}$	✓	✓	✓
k	– Seq.	Ablation	$\mathbb{Z}_{k+1}$	✗	✗	✓
l	– Chunked	Ablation	$\mathbb{Z}_{k+1}$	✓	✗	✓
m	– RankEmb	Ablation	$\mathbb{Z}_{p+1}$	✓	✓	✗
n	– Chunked – RankEmb	Ablation	$\mathbb{Z}_{k+1}$	✓	✗	✗

IR evaluation metric value. Since our model makes use of more than one document, we extend the original BERT-QPP method by additionally encoding the information from top-k retrieved as a sequence (similar to our proposed approach). This is performed so as to ensure a fair comparison between the methods.

We consider two extended versions of this approach. In the first version (see row h of Table 6.5), we only include the information from the top-k as a flat sequence with no chunking. In the second version, we include the rank embedding information similar to our model (see row i of Table 6.5). One of the main differences of our proposed model qppBERT-PL with the ones shown in the extensions to BERT-QPP (rows h and i of Table 6.5) is that the latter ones are regression models (see the 'Pred.' column of the table). These extensions to the BERT-QPP approach act as ablations with respect to our complete model setup, thus allowing us to conduct more fair, comprehensive comparisons.

Ablations of our proposed model. In relation to our proposed model qppBERT-PL, we study several ablations by selectively removing one or more sources of information. First, instead of encoding the information from top-k as a sequence, we simply use the top-retrieved documents, the only difference with BERT-QPP now being we include the rank embedding (see row k of Table 6.5). As our second ablation, instead of presenting a partitioned input of the top-k documents to the qppBERT-PL, we learn to predict the number of relevant documents on the entire top-k set by applying a $(k + 1)$ dimensional Softmax (see row l of Table 6.5). Similarly, we derive our third ablation by removing the rank embedding information from qppBERT-PL (see row m). Finally, we remove both the chunk-based workflow and the rank embedding information to derive another ablation (row n of Table 6.5).

Table 6.6: Comparison between the QPP effectiveness achieved by the proposed qppBERT-PL method and other baseline methods. The differences between the best results, obtained by qppBERT-PL (bold-faced), and the next best performing method, BERT-QPP, are significant (t-test with 95% confidence).

Type		Models	MS MARCO Dev				TREC-DL'19				TREC-DL'20			
			MRR@10		AP@100		MRR@10		AP@100		MRR@10		AP@100	
			P-r	K-τ	P-r	K-τ	P-r	K-τ	P-r	K-τ	P-r	K-τ	P-r	K-τ
Baselines	a	NQC	0.331	0.298	0.285	0.227	0.239	0.185	0.183	0.107	0.259	0.243	0.179	0.124
	b	Clarity	0.173	0.248	0.172	0.207	0.156	0.147	0.096	0.113	0.239	0.215	0.107	0.129
	c	WIG	0.193	0.215	0.215	0.203	0.192	0.133	0.133	0.089	0.260	0.241	0.143	0.096
	d	UEF(NQC)	0.347	0.313	0.294	0.227	0.254	0.235	0.189	0.112	0.275	0.291	0.200	0.126
	e	SCNQC	0.334	0.310	0.304	0.228	0.261	0.251	0.204	0.123	0.284	0.298	0.215	0.141
	f	WS-NeurQPP	0.215	0.197	0.173	0.193	0.156	0.126	0.129	0.133	0.271	0.253	0.133	0.112
	g	BERT-QPP	0.520	0.411	0.326	0.301	0.350	0.363	0.268	0.202	0.343	0.341	0.233	0.195
	h	+ Seq.	0.463	0.360	0.301	0.312	0.345	0.333	0.265	0.193	0.277	0.218	0.258	0.190
	i	+ Seq. + RankEmb	0.473	0.370	0.328	0.285	0.323	0.332	0.253	0.167	0.303	0.236	0.252	0.172
Proposed	j	qppBERT-PL	**0.562**	**0.448**	**0.354**	**0.327**	**0.413**	**0.403**	**0.301**	**0.247**	**0.422**	**0.392**	**0.303**	**0.251**
	k	– Seq.	0.512	0.386	0.303	0.283	0.357	0.349	0.274	0.193	0.345	0.320	0.271	0.200
	l	– Chunked	0.520	0.413	0.331	0.274	0.373	0.326	0.290	0.225	0.370	0.333	0.297	0.231
	m	– RankEmb	0.519	0.392	0.320	0.267	0.361	0.328	0.285	0.232	0.352	0.331	0.293	0.215
	n	– Chunked – RankEmb	0.405	0.329	0.293	0.285	0.309	0.299	0.260	0.159	0.217	0.198	0.199	0.184

Implementation-specific details. All supervised methods were trained on the MS MARCO training split of the data. The dimension of the hidden layer for the LSTM cells was set to 768 and that of the dense layer was set to 100, i.e., $\theta_{LSTM} \in \mathbb{R}^{768}$ and $\phi \in \mathbb{R}^{100}$ (see Equation 6.16). For the supervised models, we executed one epoch through the training set with a batch size of 16 as prescribed by the BERT-QPP paper (Arabzadeh *et al.*, 2021). For the classification methods, we used a cross-entropy loss. Parameter updates were performed using the Adam optimizer with a learning rate of 0.01[2].

For the regression-based methods (rows g to i of Table 6.5), we used the AP@100 values as the ground-truth for regression. In contrast, the ground-truth for our proposed model and its ablation variants (methods in the bottom group of Table 6.5) is the number of relevant documents of each chunk, or the total number of relevant documents in top-k, if chunking is not applied.

6.5.3 Results and Analysis

Table 6.6 presents a summary of the results of our experiments. We first observe that our proposed approach, qppBERT-PL (row j), outperforms all baseline approaches for all three datasets and across all measures. This includes BERT-QPP (g), the prior state-of-the-art approach for this task. The relative improvement ranges from 8.1% (MRR@10 P-r on MS MARCO Dev) to 30.0%

[2]Source code available at `https://github.com/suchanadatta/qppBERT-PL.git`

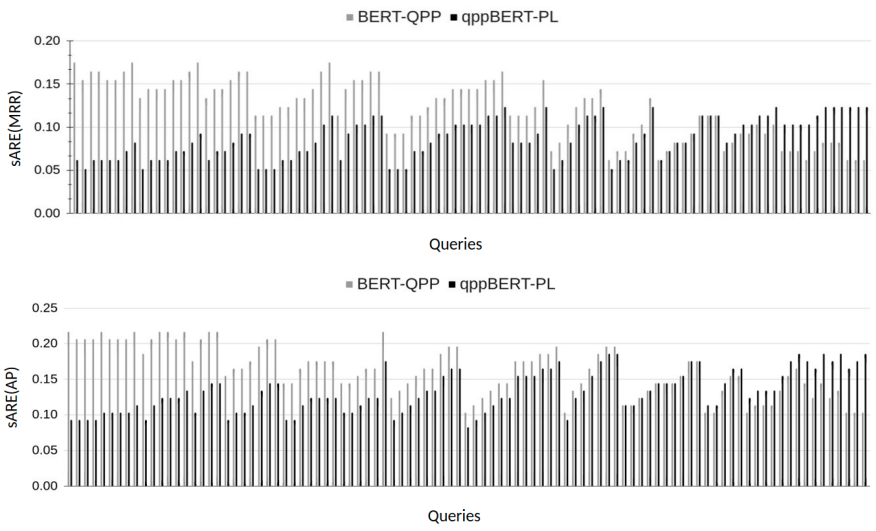


Figure 6.8: Per-query comparisons of QPP effectiveness between qppBERT-PL and BERT-QPP in terms of scaled Absolute Rank Error (sARE) (Faggioli *et al.*, 2021) computed with MRR (left) and AP (right). Comparisons are made on the TREC-DL dataset, comprising 97 queries. It can be seen that our method (qppBERT-PL) exhibits lower (hence more effective) sARE values on an average (bars with smaller heights).

(AP@100 P-r on TREC-DL'20), clearly showing a marked increase in the ability to predict query performance. The relative improvement on datasets with deeply-annotated labels (TREC-DL'19 and 20) were consistently higher than on the sparsely-annotated MS MARCO Dev set (+11.0–30.0% compared to +8.1–9.0%). All other baselines we explored (rows a–f) yielded even poorer results. This clearly shows that our proposed qppBERT-PL is more effective at predicting query performance than existing methods.

To test whether our proposed sequential modeling approaches can also improve the previous state-of-the-art model, we conduct ablations on BERT-QPP. In the results, rows h and i correspond to two versions of BERT-QPP that are generalised to closer match qppBERT-PL by, rather than consuming only the top retrieved item, taking the top-k (and optionally including rank embeddings). We find that this approach tends to lead to a decrease in QPP performance. In two cases, the approaches can lead to a slight increase in performance (AP@100 K-τ on MS MARCO Dev and AP@100 P-r on TREC-DL'19). Meanwhile, note that the sequence modeling appears to be critical for the success of qppBERT-PL; when sequence modeling is removed from the model (row k), QPP performance drops considerably. These results suggest that attempting to predict ranking effectiveness scores directly from sequences is challenging for models to learn using existing techniques.

We now explore the effect of the proposed rank embedding and chunking components of qppBERT-PL. We observe that when we remove chunking (i.e., make a binary decision about each individual document, row l) or the rank embedding (i.e., do not provide the model with information about the absolute rank of the documents, row m), QPP performance drops to the level of around or below BERT-QPP. When *both* of these components are removed (row n), the performance drops even further. These results suggest that not only information about surrounding documents is necessary to estimate QPP well, but also the absolute rank of the documents within the ranked list.[3] The former observation is aligned with findings in neural ranking (Nogueira *et al.*, 2019a). However, to the best of our knowledge, the latter has not been observed in other contexts. Hence, we find that both chunking and Rank Embeddings are critical components in our proposed method.

Analysis. We now focus on the per-query QPP effectiveness of the TREC-DL topic set. To do this, we employ the metric scaled Absolute Rank Error (sARE) proposed in Faggioli *et al.* (2021). More concretely, the sARE metric computes the absolute difference between the position (rank) of a query when ordered by a ground-truth retrieval effectiveness metric (e.g. AP) and when ordered by the estimated QPP scores.

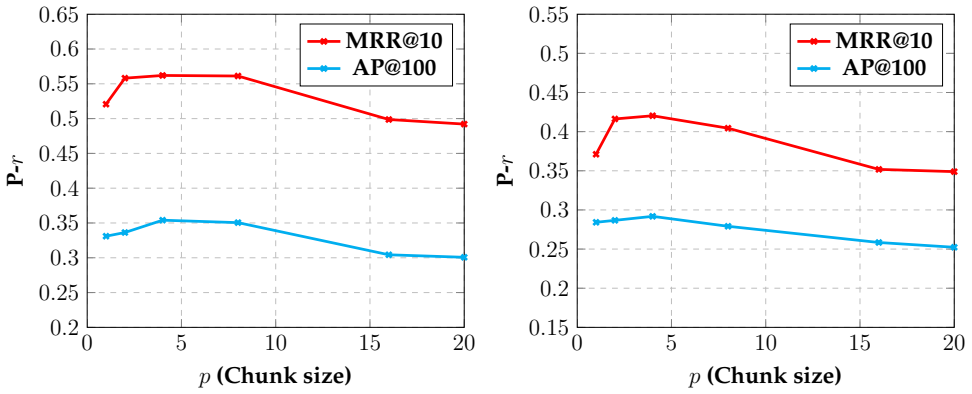
Figure 6.5.3 plots the sARE metric values for each query for our method and the best performing baseline, namely BERT-QPP (as observed from Table 6.6). By comparing the plots in Figure 6.5.3, both with MRR (sARE(MRR)) and AP (sARE(AP)), we observe that qppBERT-PL leads to lower rank errors than BERT-QPP on the TREC-DL dataset (on an average the dark-shaded bars are shorter than the lightly shaded ones).

Furthermore, we investigate the sensitivity of our proposed model, qppBERT-PL, to the chunk size p. We conduct a grid search for the optimal chunk size over the set $\{1, 2, 4, 8, 16, 20\}$. Figure 6.9 shows that the best results are obtained on both MS MARCO Dev and TREC-DL with a chunk size of 4. We observe that our method is somewhat insensitive to the chunk size parameter for $p \in [2, 8]$. For values of p lying outside of this range, the effectiveness of qppBERT-PL can decrease considerably.

The sensitivity plot of Figure 6.9 thus illustrates that prediction usually performs well when the model is able to leverage information from a set of documents, rather than a single one. This is reflected by the relatively low value

[3]It is important to remember that the query performance itself is induced from individual chunk estimations in a final step, where rank information is provided.

Figure 6.9: Sensitivity of qppBERT-PL on the MS MARCO Dev set (left) and the TREC-DL query set (right) with respect to the chunk size parameter (p).



of QPP effectiveness obtained with $p = 1$. However, using information from too many documents is likely to confuse the model, as can be seen from the decrease in QPP effectiveness for $p > 8$.

6.6 Conclusions

Our experiments in Chapter 5 demonstrated that augmenting queries blindly via relevance feedback may lead to poor retrieval effectiveness, because it often deviates some queries from their original information need. Therefore, we hypothesized that augmenting queries selectively is likely to reduce query drifts, improving overall retrieval effectiveness. With this idea in mind, in this chapter, we revisited the QPP task with the aim of estimating the difficulty of queries, which could potentially lead to improved retrievability.

We have proposed two new entirely data-driven supervised query performance predictors – one based on 2D convolutional networks, the other using a transformer-based estimator. Both are effective in estimating the specificity of an input query with respect to a given collection. In other words, the extent to which a retrieval model can distinguish the relevant documents from the rest of the collection for the input query. This allows us to automatically estimate the retrieval quality of a search system for a query without the presence of relevance assessments. The success of both of our proposed models in comprehensive experimental evaluations motivates us to develop a selective feedback model as a downstream task of our QPP approaches. In the next chapter, we will explain the end-to-end architecture of this selective feedback model and how it improves causal retrieval effectiveness.

CHAPTER

SEVEN

SELECTIVE RELEVANCE FEEDBACK FOR CAUSAL IR

7.1 Introduction

The keywords entered by a user as a query to a search engine are often inadequate for expressing the user's information need, creating a *lexical gap* between the query text and the relevant documents (Belkin *et al.*, 1982). Standard pseudo relevance feedback (PRF) methods, such as the relevance model (Lavrenko & Croft, 2001) and its variants (Ganguly *et al.*, 2012; Salakhutdinov & Mnih, 2008; Roy *et al.*, 2016; Mackie *et al.*, 2023), can overcome this problem and ultimately yield improvements in retrieval effectiveness. Generally speaking, PRF methods are designed to enrich a user's initial query with distinctive terms from the top-ranked documents (Rocchio, 1971; Mitra *et al.*, 1998; Xu & Croft, 2000).

Despite the demonstrated success of PRF in improving retrieval effectiveness, a number of studies have identified certain limitations of this strategy (Billerbeck & Zobel, 2004; Lv & Zhai, 2009a; Cronen-Townsend *et al.*, 2004; Deveaud *et al.*, 2018). For the most part, these limitations share a common theme: there is no consistent PRF setting that works well across a wide range of queries. In simpler terms, *one size does not fit all*. To illustrate this idea, Figure 7.1 depicts a scenario where nearly 38.9% of the queries from TREC DL'20 topic set are penalized as a result of PRF. Not only do standard datasets confirm this, but our experimental findings with a causal dataset in Chapter 5 also support this observation (see Figure 5.4.6). Keeping this in mind, this chapter proposes an adaptive relevance feedback framework that includes a data-driven supervised neural approach to optimize retrieval effectiveness by applying feedback

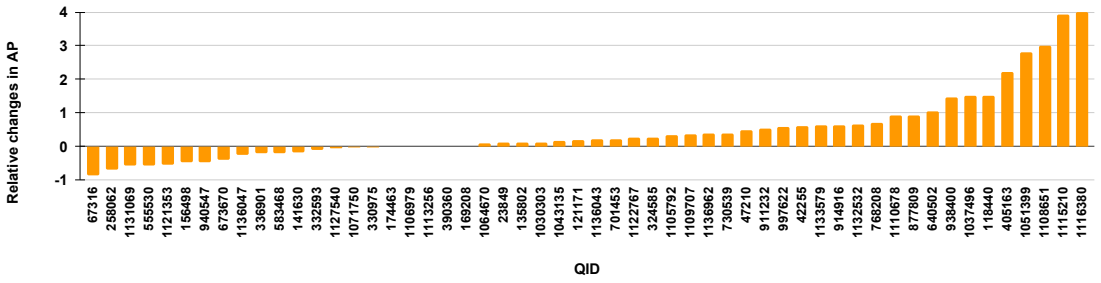


Figure 7.1: Relative changes in AP, i.e., (AP(post-fdbk) - AP(pre-fdbk))/AP(pre-fdbk), for TREC DL'20 queries. We observe that many queries are negatively impacted by PRF (bars below the x-axis).

on queries in a selective fashion Our experiments on both standard ad-hoc IR dataset, e.g. MS MARCO (Nguyen *et al.*, 2016) and newly created causality-driven dataset, CARD in Chapter 4 confirms the effectiveness of selective feedback for better retrieval.

Previous work has shown that not all documents contribute equally well to PRF, as certain documents may impair retrieval effectiveness when used to expand a query (Lee *et al.*, 2008; Bashir & Rauber, 2009). This can even be true when relevant documents are used to enrich a query's representation (Terra & Warren, 2005). It has also been observed that some queries are amenable to more aggressive query expansion, while others work better with more conservative settings (Ogilvie *et al.*, 2009). Moreover, not all terms might contribute equally well in terms of enriching the representation of a query (Cao *et al.*, 2008; He & Ounis, 2009), which suggests that a selective approach to PRF can potentially improve the overall system effectiveness.

Rather than following the previous approaches on adapting the number of feedback terms (Ogilvie *et al.*, 2009) or attempting to choose a robust subset of documents for PRF (Lee *et al.*, 2008; Bashir & Rauber, 2009), we rather focus on solving the more fundamental decision question of "*whether or not to apply PRF for a given query*" (Cronen-Townsend *et al.*, 2004; Lv & Zhai, 2009a) through the use of a supervised data-driven approach. We hypothesize that selectively applying feedback to queries well-suited for PRF can improve the overall success of information retrieval. This approach aims to prevent query drift in situations where feedback might otherwise be counterproductive.

The main novelty of our proposed selective pseudo relevance feedback (SRF) approach is that, in contrast to existing work on selective PRF, we propose a data-driven supervised neural model for predicting which queries are con-

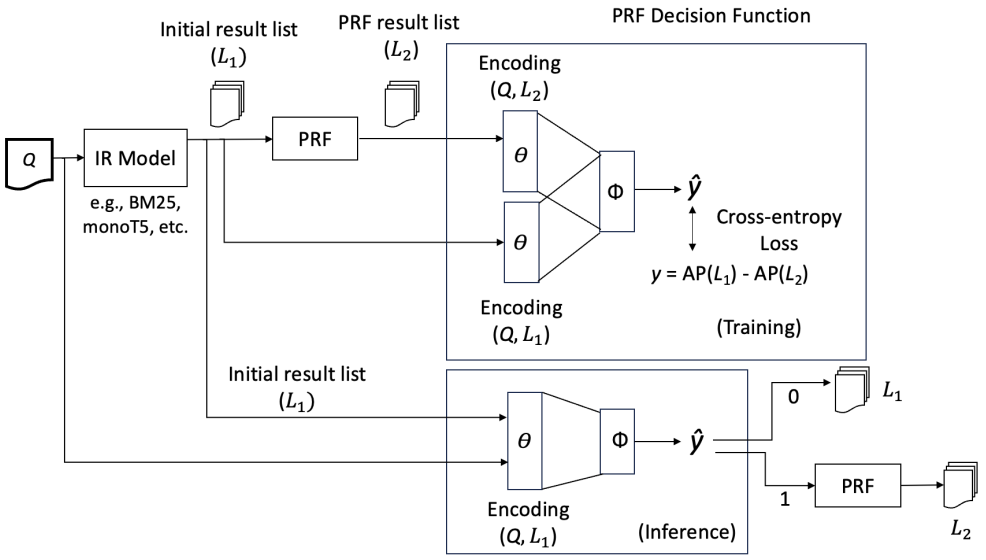


Figure 7.2: A schematic diagram of selective feedback. The main contribution of this model is a supervised data-driven approach towards realizing the decision function.

ducive to PRF. Specifically, during the training phase we make use of the relevance assessments to learn a decision function that can predict whether it is useful to apply PRF. The process considers both the initial query and the top-ranked documents with and without feedback. During the inference phase, we make use of only a part of the shared parameter network which predicts whether PRF is to be applied. This strategy reduces computational costs for queries where PRF should eventually be ignored. The overall process is depicted in Figure 7.1.

A key advantage of our SRF approach is that it can be applied to the output ranked list obtained by *any retrieval model*, ranging from sparse models (e.g., BM25, LM-Dir) to dense ones such as MonoBERT (Nogueira *et al.*, 2019b). Moreover, in the SRF workflow it also is possible to use *any PRF model* to enrich a query's representation, ranging from sparse models (e.g., RLM) to dense ones (e.g., ColBERT-PRF from Wang *et al.* (2023)); from generative ones (e.g., GRF from Mackie *et al.* (2023)) to the new FCRLM approach that we proposed in Chapter 5.

7.2 Related Research

The evolution of relevance feedback in IR spans from traditional query expansion models (Ogilvie *et al.*, 2009; Cao *et al.*, 2008) to cluster-based feedback

document selection (Lee *et al.*, 2008; He & Ounis, 2009). Prior research has considered both unsupervised selective feedback (Cronen-Townsend *et al.*, 2004) and feature-driven methods (Lv & Zhai, 2009a). Several existing methods, both supervised and unsupervised, hinge on *decision-based relevance feedback*. A common unsupervised approach involves using Query Performance Prediction (QPP) scores (Shtok *et al.*, 2012; Zhou & Croft, 2007; Shtok *et al.*, 2010; He & Ounis, 2007), which we include as a baseline. The higher the QPP score, the greater the chance of identifying relevant documents at the top rank positions with the initial query. However, high variances in retrieval status values, as seen in neural re-rankers like MonoBERT, can make QPP scores deceptive. To avoid such heuristics, our method focuses solely on query terms and the documents retrieved by that query in order to learn the selection function.

PRF on and for dense retrieval. Recently, the community has seen a significant interest in feedback for dense retrieval to boost performance. Precursors to dense feedback models made use of word embeddings for PRF. For instance, Roy *et al.* (2016) proposed a generalized RLM built upon word embeddings, while Zamani *et al.* (2016) leveraged non-negative matrix factorization to bridge the semantic gap between the terms in a query and the corresponding top-retrieved documents.

Work by Yu *et al.* (2021) explored relevance feedback principles within dense retrieval models. In a separate study, Li *et al.* (2022a) examined the quality of feedback signals, contrasting conventional models such as those developed by Rocchio (1971) with dense retrieval models like those based on ANCE (Xiong *et al.*, 2020), concluding that the dense retrievers demonstrated greater robustness. Representation models, such as ColBERT (Khattab & Zaharia, 2020), can allow us to append additional embedding layers to the query representation, as demonstrated by Wang *et al.* (2021). This method employed contextualized PRF to cluster and rank feedback document embeddings in order to select suitable expansion embeddings, thus improving document ranking. In other work, Zhuang *et al.* (2022) leveraged implicit feedback from historical clicks for relevance feedback in dense retrieval. The authors introduced counterfactual-based learning-to-rank, showing that historic clicks can be highly informative in terms of relevance feedback. Finally, Li *et al.* (2022b) proposed the idea of combining feedback signals from both sparse and dense retrievers in the context of PRF.

More recently, PRF on dense IR models has garnered significant interest (Li *et al.*, 2018; Naseri *et al.*, 2021; Zheng *et al.*, 2020; Wang *et al.*, 2023). The concept

of 'dense for PRF' was first motivated by Montazeralghaem *et al.* (2020), who proposed a reinforcement-based learning algorithm designed to explore and exploit various retrieval metrics, aiming to learn an optimized PRF function. Following the recent success of LLMs, Mackie *et al.* (2023) proposed a generative feedback method (GRF) that makes use of LLM generated long-form texts to build a probabilistic feedback model. In contrast, our work aims to develop a generic PRF strategy that does not apply feedback blindly, but rather learns a selection function in a supervised manner to analyze the suitability of relevance feedback for each query irrespective of sparse or generative PRF.

Selective PRF. Prior work in this area has considered either fully unsupervised strategies (Cronen-Townsend *et al.*, 2004) or feature-based supervised approaches (Lv & Zhai, 2009a) for selective relevance feedback (SRF). The former makes use of QPP-based measures to predict if a query should be expanded, where the decision depends on whether the QPP score exceeds a given threshold. On the other hand, existing supervised approaches first represent each query as a bag of characteristic features derived from its top-retrieved set of documents. A classifier is subsequently trained on these features to predict whether or not a query should be expanded (Lv & Zhai, 2009a).

7.3 Selective Feedback Model Description

7.3.1 A Generic Decision Framework for PRF

In this section, we formally describe the generic framework for selective PRF. Given a set of queries $\mathcal{Q} = \{Q_1, \ldots, Q_n\}$, a standard relevance feedback model M uses the information from the top-retrieved documents of each query to enrich its representation, i.e., $M : Q \mapsto \phi_M(Q)$. Consequently, each query $Q \in \mathcal{Q}$ is transformed to an enriched representation $\phi_M(Q)$, which is then used either for re-ranking the initial list, or to execute a second-step retrieval.

Unlike the standard PRF setting, a decision-based selective PRF framework first applies a *decision function*, $\theta : Q \mapsto \{0, 1\}$, which outputs a Boolean to indicate whether the retrieval results for Q are likely to be improved by applying PRF. As per our proposal, the overall PRF process on the set of queries $\mathcal{Q}$ does not blindly use the expanded query $\phi_M(Q)$ for each $Q \in \mathcal{Q}$. Rather, it makes use of the function $\theta(Q)$ for each query Q to decide independently whether to output the initial ranked list or to use an enriched query representation $\phi_M(Q)$,

as obtained by a PRF model M. This leads to either re-ranking the initial list or re-retrieving a new list via a second stage retrieval. Thus, the top-k ranked list of documents, $L_k(Q) = \{D_1^Q, \ldots, D_k^Q\}$, retrieved for a query Q, in addition to being a function of the query Q itself, is thus also a function of i) the feedback model M, ii) the enriched query representation $\phi_M(Q)$, and iii) the decision function θ. Formally,

$$L_k(Q) = \begin{cases} \sigma(Q), & \text{if } \theta(Q) = 0 \\ \sigma(\phi_M(Q)), & \text{if } \theta(Q) = 1, \end{cases} \tag{7.1}$$

where $\sigma(Q)$ denotes a retrieval model (e.g., BM25) that outputs an ordered set of k documents sorted by the similarity scores. Previous approaches have explored the use of both unsupervised and supervised approaches for addressing this decision problem. We now briefly explain both strategies in our own context.

Unsupervised decision function. An unsupervised approach, such as that proposed by Cronen-Townsend *et al.* (2004), applies a threshold parameter on a QPP estimator function $\theta_{\text{QPP}} : Q \mapsto [0, 1]$[1]. More concretely, if the predicted QPP score is lower than the threshold parameter, it is likely to indicate that the retrieval performance for the query has scopes for further improvement and subsequently PRF is applied for this query. Formally speaking, the decision function of an unsupervised approach takes the form

$$\theta(Q) \stackrel{\text{def}}{=\!=} \mathbb{I}(\theta_{\text{QPP}} < \tau), \tag{7.2}$$

where $\tau \in [0, 1]$ is the threshold parameter.

Supervised decision function. An unsupervised function $\theta(Q)$ as per Equation 7.2 depends only on the information of a query and its top-retrieved documents. In a supervised approach, this decision additionally depends on the enriched query representation and its top-retrieved documents. More precisely, a supervised PRF decision is a parameterized function of features of: i) the query Q, ii) its top-retrieved documents $L_k(Q)$, iii) the enriched query $\phi_M(Q)$, and iv) its top-retrieved set $L_k(\phi_M(Q))$ (Lv & Zhai, 2009a). The corresponding training process makes use of a set of queries, denoted $\mathcal{Q}_{\text{train}}$, for which ground-truth indicator labels are available. These labels are calculated by comparing the retrieval performance of the original query against that of the enhanced query,

[1]While a QPP estimate is not generally required to lie within $[0, 1]$, in practice the estimated value can be normalized within the unit interval.

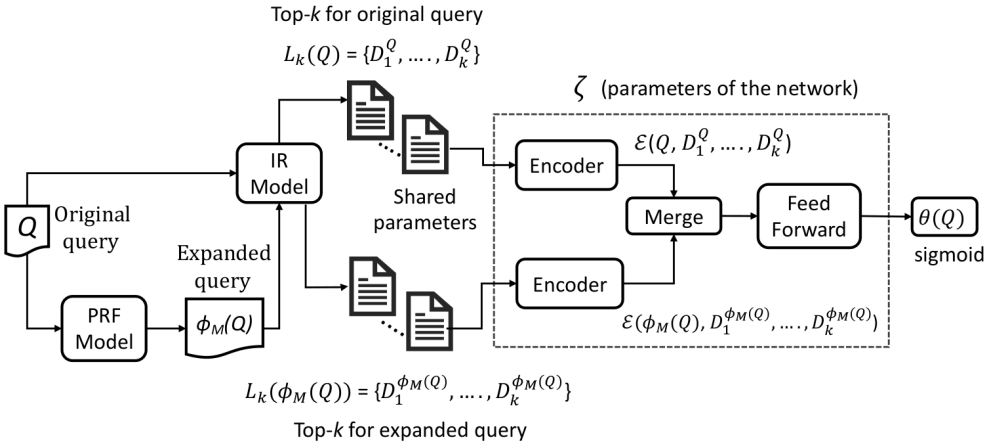


Figure 7.3: A schematic overview of the data-driven modeling of the decision function for selective relevance feedback.

using the available relevance assessments to measure effectiveness. Formally,

$$y(Q) = \mathbb{I}(\mathrm{AP}(\phi_M(Q)) > \mathrm{AP}(Q)), \tag{7.3}$$

where $\mathrm{AP}(Q)$ denotes the average precision of a query $Q \in \mathcal{Q}_{\text{train}}$.[2]

The ground-truth indicator values of Equation 7.3 are used to learn the parameters of a classifier function to yield a supervised version of the decision function θ, given by:

$$\begin{aligned} \theta(Q) &\stackrel{\text{def}}{=\!=} \zeta \cdot \mathbf{z}_{Q,\phi_M(Q)}, \text{ where} \\ \theta(Q) &\approx \underset{\zeta}{\operatorname{argmin}} \sum_{Q' \in \mathcal{Q}'_{\text{train}}} (y(Q') - \zeta \cdot \mathbf{z}_{Q',\phi_M(Q')})^2. \end{aligned} \tag{7.4}$$

In the above, ζ represents a set of learnable parameters, with $\mathbf{z}_{Q',\phi_M(Q')}$ denoting a set of features extracted from both the original query Q' and the enriched query $\phi_M(Q')$ along with the features from their top-retrieved set of documents $L_k(Q')$ and $L_k(\phi_M(Q'))$. The variable $y(Q')$ (as defined in Equation 7.3) denotes the ground-truth indicating whether PRF should be applied for Q'.

The optimal parameter vector ζ, as learned from a training set of queries $\mathcal{Q}_{\text{train}}$, is then used to predict the decision for any new query Q. The exact features we use are described later in Section 7.4. In the next section we describe a data-driven approach that makes use of the terms in a query and those in the top-retrieved documents towards a data-driven learning of the decision function with deep neural networks.

[2]Here, we explore only average precision. We note that other measures of query effectiveness could be used as well, depending on the needs of the application.

7.3.2 Deep Learning of PRF Decision

Motivated by the merits of the QPP methods we proposed previously in Chapter 6, we first provide a generic description of our proposed neural model for selective feedback, and then describe two concrete architectural realisations of the generic neural framework — one with convolutional operations, and the other with transformers. We will see that the latter yields higher effectiveness at the cost of increased run-times for training and inference.

We adopt an approach similar to that summarized in Equation 7.4, which involves training a supervised approach to learn if PRF should be applied for a query. However, unlike Equation 7.4, instead of making use of a specific set of extracted features, the learning objective makes use of the terms present in the documents and the queries. As with Equation 7.4, we make use of both the content of the original query Q and its enriched form $\phi_M(Q)$, along with their top-retrieved sets. Formally,

$$\theta(Q) \stackrel{\text{def}}{=\!=} \zeta \cdot (\mathcal{E}(Q, D_1^Q, \ldots, D_k^Q) \oplus \mathcal{E}(\phi_M(Q), D_1^{\phi_M(Q)}, \ldots, D_k^{\phi_M(Q)})), \tag{7.5}$$

where $\theta(Q)$ is learned by computing:

$$\underset{\zeta}{\text{argmin}} \sum_{Q' \in \mathcal{Q}'_{\text{train}}} (y(Q') - \zeta \cdot (\mathcal{E}(Q', L_k(Q')) \oplus \mathcal{E}(\phi_M(Q'), L_k(\phi_M(Q'))))^2. \tag{7.6}$$

In Equation 7.6, $\mathcal{E}$ is a parameterized function for encoding the interaction between a query Q and its top-retrieved documents, $L_k(Q)$. This encoding function maps a query (a sequence of query terms) and a sequence of documents (which are themselves sequences of their constituent terms) to a fixed length vector, i.e., $\mathcal{E} : Q, L_k \mapsto \mathbb{R}^p$ (p an integer, e.g., for BERT embeddings $p = 768$). Here $\oplus$ indicates an interaction operation (e.g., a merge layer in a neural network) between the query-document encodings corresponding to the original query and the enriched one. A schematic overview of the proposed data-driven neural model is shown in Figure 7.3.1.

We will now detail two specific realizations of the encoding function $\mathcal{E}$. One version represents documents and queries as collections of terms, ignoring their order, while the other considers them as ordered sequences.

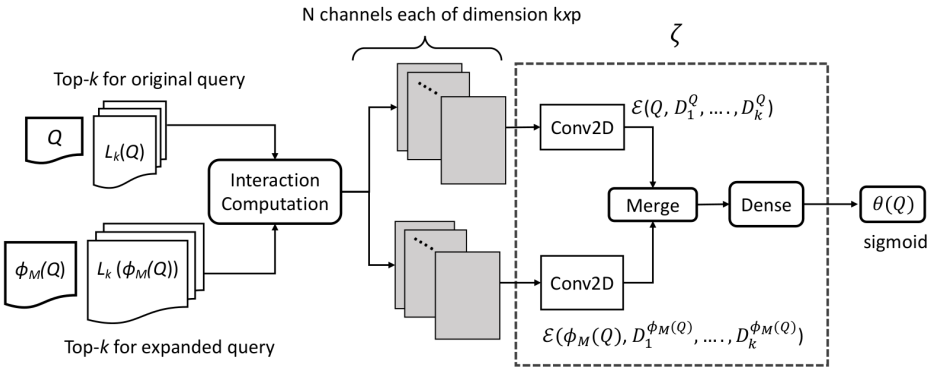


Figure 7.4: A concrete realization of the encoding of the query-document pairs via the use of DRMM-based early interaction. Interaction vectors between each query term and a top-retrieved document are stacked to form a matrix. The corresponding matrices for each top-retrieved document are considered as separate channels of input data, which are passed through 2D convolutional filters.

7.3.3 Term Overlap-based Encoding

Our first implementation follows from the word embedding-based interactions in the deep relevance matching model (Guo *et al.*, 2016). Rather than applying separate encoders for documents and queries, this method first computes the interaction between a query Q and a top-retrieved document $D_i^Q \in L_k(Q)$ as a fixed length vector by quantizing the cosine similarity values between every term pair – one from the query Q and the other from the document D_i^Q. The quantization step involves the use of a hyper-parameter p, which is the number of intervals in which the range of the cosine similarity values ($[-1, 1]$) is partitioned.

Specifically, the value of the β^{th} component ($\beta = 1, \ldots, p$) of the interaction vector between a query and a top-retrieved document is obtained by counting the number of terms that yield similarities which lie within the β^{th} partition, i.e.,

$$(q_j \oplus D_i^Q)_\beta = \sum_{w \in D_i^Q} \mathbb{I}\big[\frac{2(\beta-1)}{p} - 1 \leq \frac{\mathbf{q}_j \cdot \mathbf{w}}{|\mathbf{q}_j||\mathbf{w}|} < \frac{2\beta}{p} - 1\big], \tag{7.7}$$

where both $\mathbf{q}_j \in \mathbb{R}^d$ and $\mathbf{w} \in \mathbb{R}^d$ denote the embedded vectors corresponding to the j^{th} query term $q_j \in Q$ and a term w of the i^{th} document D_i^Q. Here $\mathbb{I}[X] \in \{0, 1\}$ is an indicator variable which takes the value of 1, if a property X is true and 0 otherwise. Following the findings of Guo *et al.* (2016), we use the inverse document frequencies (IDFs) of query terms to weigh (via scalar multiplication) the interaction tensors. Note that the IDF factors are not shown in Equation 7.7 for the sake of brevity.

By construction, the interaction vector itself between a query term and a top-retrieved document is a p-dimensional vector, i.e., $q_i \oplus D_i^Q \in \mathbb{R}^p$. We then construct a matrix for the overall interaction between the query Q with a top-retrieved document D_i^Q by stacking the interaction vectors for each query term. Finally, we stack together the matrices for each of the k documents in $D_i^Q \in L_k(Q)$ to yield a tensor of order 3.

In a similar manner, we also obtain the interaction tensor with the enriched query $\phi_M(Q)$. Note that, to ensure that the interaction tensors for the original and the expanded queries are of the same dimensions, the length of a query needs to be set to a maximum value, say N. As a result, the dimensions of the interaction tensors become $k \times N \times p$.

Ideally, the hyper-parameter N is the sum of maximum query length (n) in the dataset and the number of feedback terms (t) selected to construct the expanded query, $\phi_M(Q)$. Therefore, the parameter n is simply computed from the dataset, whereas t is treated as a hyper-parameter and its optimal value is obtained from the set $\{5, 10, 15, 20, 25, 30, 35, 40\}$. In fact, our reported results in Table 7.2 and 7.3 uses the optimal values of t found via grid search.

In the next step, following the idea of our proposed QPP model in Chapter 6 (refer to Section 6.4), we then employ a 2D convolutional neural network that takes as input a $k \times N \times p$ dimensional interaction tensor – one for the original query and the other for the expanded one (see Figure 7.4). This slices the tensor into N channels and transforms each to yield a fixed-length vector after the standard flattening step, following the application of the convolutional filters. Formally, the query-document encoding function, as per the notation in Equation 7.6, is thus defined as

$$\mathcal{E}(\bar{Q}, L_k(\bar{Q})) = \text{Conv2D} \begin{bmatrix} [q_1 \oplus D_1^{\bar{Q}}] & \dots & [q_1 \oplus D_k^{\bar{Q}}] \\ \dots & \dots & \dots \\ [q_N \oplus D_1^{\bar{Q}}] & \dots & [q_N \oplus D_k^{\bar{Q}}] \end{bmatrix}. \tag{7.8}$$

where $\bar{Q}$ refers to either an original query Q or its expanded form $\phi_M(Q)$.

The $\oplus$ interaction in Equation 7.6 between the 2D-CNN encoded vectors is then given by a merge operation, followed by a dense layer of parameters, finally leading to a sigmoid for the binary prediction of the PRF decision function. The 2D-CNN coupled with the merge and the dense layers thus define the full set of parameters ζ.

For our experiments, we use skip-gram word vectors of dimension 300 trained

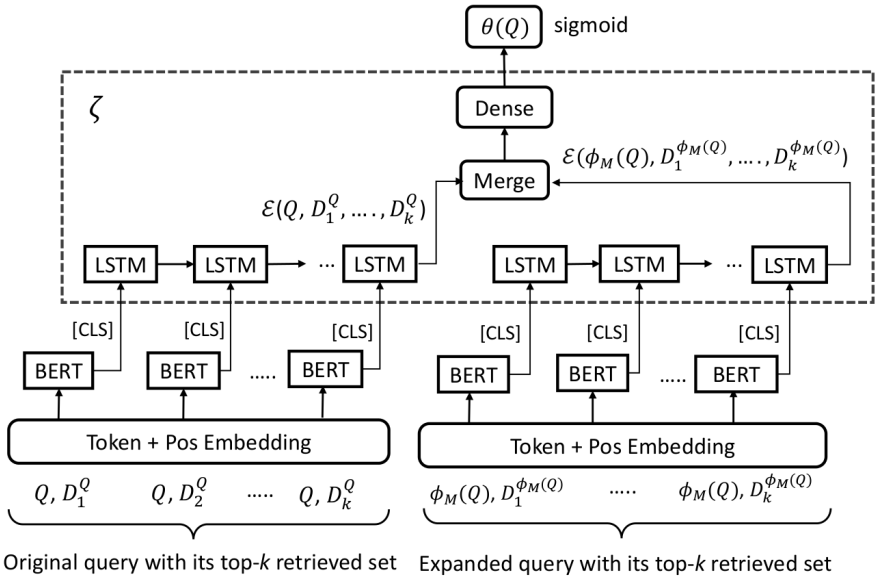


Figure 7.5: A concrete realization of the query-document encoding function via the use of transformers. This model, unlike the one shown in Figure 7.4, involves late interactions and is able to model the sequence of terms within a document and also the order of the documents within the top-k set.

on the respective document collections, with a window size of 10 and 25 negative samples. We use two layers of stacked 2D convolution with kernel $k_1 = 5$ and $k_2 = 3$ (i.e. a 5x5 filter for the first layer and a 3x3 for the second one). We refer to this particular realization of the generic data-driven approach as **Deep-SRF-CNN** (Deep Selective Relevance Feedback with the use of 2D convolutions).

7.3.4 Transformer-based Encoding

The 2D-CNN-based encoding makes use of individual word vectors to obtain interaction tensors, which are then supplied as inputs to a neural network. Unlike the idea of early interactions, the transformer-based encoding uses the BERT architecture which takes as input the contextual embeddings of the terms for each pair comprising a query Q and its top-retrieved document $D_i^Q \in L_k(Q)$. The 768 dimensional '[CLS]' representations of each *query-document* pair is then encoded with LSTMs as a realisation of the encoded representation of a query and its top-retrieved set, i.e., to define $\mathcal{E}(Q, L_k(Q))$ as per the notation of Equation 7.6.

Similar to the architecture described in Section 7.3.3, we also obtain a BERT-based encoding of the expanded query $\phi_M(Q)$ and its top-retrieved set and

then merge the two representations before passing them through a feed-forward network. More formally,

$$\mathcal{E}(\bar{Q}, L_k(\bar{Q})) = \text{LSTM}(\text{BERT}(\bar{Q}, D_1^{\bar{Q}})_{\text{[CLS]}}, \ldots, \text{BERT}(\bar{Q}, D_k^{\bar{Q}})_{\text{[CLS]}}). \tag{7.9}$$

As in Equation 7.7, the variable $\bar{Q} \in \{Q, \phi_M(Q)\}$, i.e., in one branch of the network it corresponds to the original query, whereas in the other it corresponds to the expanded one. Figure 7.3.4 shows the transformer-specific implementation of the encoding function. In this case, the set of learnable parameters ζ comprises of the LSTM and the fully connected (dense) layer parameters, as shown in Figure 7.3.4. We name this particular model **Deep-SRF-BERT** (Deep Selective Relevance Feedback with the use of BERT transformers).

During the inference stage of the model, only the component that relates to the original query and its top-ranked documents is employed to predict the output variable (a sigmoid). If this output exceeds 0.5, it indicates that PRF should be applied.

7.3.5 Model Confidence-based PRF Calibration

Prior work has applied confidences of prediction models to improve retrieval effectiveness (Cohen *et al.*, 2021). In our work, we use the uncertainties in the prediction of the decision function to further improve search results. Rather than only reporting either results with or without relevance feedback, we make use of the confidence of the decision function $\theta(Q)$ to combine the results from the two lists – one without feedback and the other with feedback. Specifically, if the supervised model outlined in Section 7.3.1 is decisive in its choice between $L_k(Q)$ (the list retrieved for the original query) and $L_k(\phi_M(Q))$ (the list retrieved for the expanded query), then one of the rankings is expected to dominate over the other. However, when the model $\theta(Q)$ itself is not confident about the prediction, we can potentially achieve better results if we "meet somewhere in the middle".

Formally, we propose a rank-fusion based method, where the fusion weights are obtained from the predictions of the PRF decision model $\theta(Q)$. The predicted value $\theta(Q)$ (a sigmoid) represents the probability of classifying the decision into one of the two outcomes – the closer $\theta(Q)$ is to 0, the higher is the model's confidence in not applying feedback, and similarly the closer $\theta(Q)$ is to 1, the higher is the model's confidence in applying PRF. The predicted val-

ues of $\theta(Q) \in [0, 1]$ can thus be used as weights to fuse the two different ranked lists, i.e., the fusion score $\sigma_F(Q, D)$ of a document D for a query Q is given by

$$\sigma_F(Q, D) = \frac{1 - \theta(Q)}{\text{Rank}(D, L_k(Q))} + \frac{\theta(Q)}{\text{Rank}(D, L_k(\phi_M(Q)))}, \tag{7.10}$$

where the notation $\text{Rank}(D, L)$ denotes the rank of a document D in a list L. If $D \notin L$, then the rank is set to a large value $\aleph(> k)$. For our evaluations, we use the value 1000, which was higher than all values of k considered in the experiments.

For values of $\theta(Q)$ close to 0.5 (i.e., the highest uncertainty in prediction), the fusion-based approach leads to a more uniform contribution from both the lists. In contrast, a value of $\theta(Q)$ close to 0 ensures that the majority of the score contribution comes from the original query (since $1-\theta(Q) >> \theta(Q)$), and a similar argument applies for $\theta(Q) \rightarrow 1$, in which case the major contribution comes from the second term on the right-hand side of Equation 7.10.

7.4 Evaluation

7.4.1 Methods Investigated

We now evaluate the methods proposed earlier in this chapter. In addition to conducting experiments with our proposed model Deep-SRF-BERT (Figure 7.3.4), we also incorporate the confidence-based calibration with rank fusion (Equation 7.10 and 7.11), which we denote by adding the suffix R2F2 [3]. For comparison purposes, we consider a range of unsupervised and supervised methods. Some baselines correspond to existing methods, while others represent extensions of alternative approaches. The latter allow us to provide a fair comparison, such as by using a more recent QPP method instead of the originally-proposed clarity score (Cronen-Townsend *et al.*, 2004).

PRF is a standard non-selective relevance feedback model, namely RLM (Lavrenko & Croft, 2001). We use the RM3 version of the model as reported by Jaleel *et al.* (2004), which is a linear combination of the weights of the original query model and new expansion terms. In fact, we use RLM as one of the base PRF model M which means that the standard RLM degenerates to a specific case of the generic selective PRF framework of Equation 7.1 with

[3]Source code : `https://github.com/suchanadatta/AdaptiveRLM.git`

$\theta(Q) = 1 \, \forall Q \in \mathcal{Q}$, i.e., when for each query we use its enriched form $\phi_M(Q)$.

R2F2 refers to an adaptation of the Reciprocal Rank-based Fusion (RRF) Cormack *et al.* (2009), a simple yet effective approach for combining the document rankings from multiple IR systems. For our task, instead of combining ranked lists from two different retrieval models, we merge the ranked lists of the original and the expanded queries, i.e., $L_k(Q)$ and $L_k(\phi_M(Q))$ as per our notations. We name the adapted method Reciprocal Rank Fusion-based Feedback (R2F2).

Formally, the score for document D after fusion is given by

$$\sigma_F(Q, D) = \frac{1 - \alpha}{\text{Rank}(D, L_k(Q))} + \frac{\alpha}{\text{Rank}(D, L_k(\phi_M(Q)))}, \tag{7.11}$$

where, similar to Equation 7.10 $\text{Rank}(D, L)$ denotes the rank of a document in a list L (this being a large number $\aleph$ if $D \notin L$), and $\alpha \in [0, 1]$ is a linear combination hyper-parameter that we adjust with grid search on each training fold. A lower value of α puts more emphasis on the initial retrieval list, whereas a higher value ensures that the feedback rank of a document contributes more. Equation 7.11 is a special case of Equation 7.10 with a constant value of $\theta(Q) = \alpha$ for each query Q.

QPP-SRF is an adaptation of the method proposed by Cronen-Townsend *et al.* (2004), where the QPP score of a query is used as estimate to decide if PRF should be applied for that query (see $\theta(Q)$ in Section 7.3.1). The idea here is that a high QPP score is already indicative of an effective retrieval performance, in which case, the method avoids any further risk of potentially degrading the retrieval quality with query expansion. We refer to this method as QPP-based selective relevance feedback (QPP-SRF). The method requires a base QPP estimator for obtaining the θ_{QPP} scores.

To choose the QPP estimator, we conducted a set of initial experiments using several standard unsupervised QPP approaches. Our proposed supervised QPP method qppBERT-PL in Chapter 6 demonstrated the best downstream retrieval effectiveness. Therefore, we report results of QPP-SRF combined with qppBERT-PL, where training is conducted using the settings as mentioned in Section 6.5.2. A key parameter for QPP-SRF is the threshold value τ ($\tau \in [0, 1]$) which controls the decision around whether PRF is applied or not. In our experiments we tune τ on the train folds. To ensure that the threshold can be applied for any QPP estimate, we normalize the QPP estimates in the range $[0, 1]$.

TD2F is an unsupervised selective feedback approach that is conceptually similar to QPP-SRF (Cronen-Townsend *et al.*, 2004). Rather than using a QPP method, it computes the difference of the term weight distributions across the sets of documents retrieved with the original and the expanded queries, i.e., the sets $L_k(Q)$ and $L_k(\phi_M(Q))$ as per our notations introduced in Section 7.3.1. Formally,

$$\theta(Q) = \frac{1}{|V|} \sum_{t \in V} \log P(t|L_k(Q)) - \log P(t|L_k(\phi_M(Q))), \tag{7.12}$$

where the set V denotes the vocabulary of the two lists, i.e., $V = V_{L_k} \cup V_{L_k(\phi_M(Q))}$. As per Cronen-Townsend *et al.* (2004), we set the feedback decision threshold τ to a value such that over 95% of the queries satisfy the criterion that $\theta(Q) \leq \tau$. We name this method as Term Distribution Divergence based Feedback, or TD2F for short.

LR-SRF is the only existing supervised method that uses the query features, along with their top-retrieved documents, to predict the PRF decision (Lv & Zhai, 2009a). The ground-truth labels for learning the decision function is obtained for a training set of queries with existing relevance assessments (i.e. $y(Q) = \mathbb{I}(\text{AP}(\phi_M(Q)) > \text{AP}(Q))$). The method then uses Equation 7.4 to train a feature-based logistic regression classifier. In particular, the experiments reported by Lv & Zhai (2009a) used the following features for training the logistic regression model: i) the clarity of top-retrieved documents (Cronen-Townsend *et al.*, 2002), ii) the absolute divergence between the query model Q and the relevance model (Lavrenko & Croft, 2001), iii) the Jensen-Shannon divergence between the language model of the feedback documents (Lin, 2006), and iv) the clarity of the query language model. We refer to this method as Regression-based Selective Relevance Feedback (LR-SRF).

7.5 Experimental Setup

7.5.1 Dataset and Train-Test Splits

Our retrieval experiments are conducted both with a standard ad-hoc IR dataset, the MS MARCO passage collection (Nguyen *et al.*, 2016) and our newly proposed causal ad-hoc dataset, CARD, introduced in Chapter 4. The relevance of the passages in the MS MARCO collection are more of personal-

Table 7.1: Summary of the data used in our SRF-based experiments. The columns '$|\bar{Q}|$' and '$\#\bar{Rel}$' denote average number of query terms and average number of relevant documents.

Collection	#Docs	Topic Set	#Topics	$\lvert\bar{Q}\rvert$	$\#\bar{Rel}$
		MS MARCO Train	502,939	5.97	1.06
MS MARCO Passage	8,841,823	TREC DL'19	43	5.40	58.16
		TREC DL'20	54	6.04	30.85
Washington Post	600,000	CARD	45	11	16.76

ized in nature which we detailed in Section 6.5.2. A common practice is to use the TREC DL topic sets, which contains depth-pooled relevance assessments on the passages of the MS MARCO collection. For TREC DL, we conduct experiments on a total of 97 queries from the years 2019 and 2020 (Craswell *et al.*, 2020, 2019). Table 7.1 provides an overview of the dataset used for our selective feedback experiments.

Since MS MARCO has a dedicated training set, we use a random sample of 5% of queries (constituting a total of approximately 40K queries) to train the supervised models in our experiments, whereas evaluation is conducted on the TREC DL (both '19 and '20) query sets. On the other hand, the experiments on CARD dataset is conducted following the standard k-fold cross validation where at each step any $(k-1)$ folds are used for training the model and the remaining fold is made use for testing. We repeat this for k times and report the average outcome. It is worth mentioning here that for the MS MARCO experiments, we use a small sample from the training set since the training process requires executing a feedback model (e.g., RLM) for all queries. Therefore, the model needs to learn a task-specific encoding for each query-document pair, both for the original and the expanded queries.

To investigate the generalization ability of our selective feedback model on MS MARCO dataset, we employ RLM as the feedback approach to train the decision function (Figure 7.3.1). During inference, we employ three different PRF approaches, namely RLM, ColBERT-PRF (Wang *et al.*, 2023) and GRF (Mackie *et al.*, 2023) to test the effectiveness of selective feedback. Whereas, to ensure the robustness of the proposed Deep-SRF in terms of causal retrieval, we make use of FCRLM (see Chapter 5 for more details) for feedback at the time of training and both FCRLM and RLM were used at the time of inference for fair comparisons.

7.5.2 Parameter Settings

A common parameter for all the methods is the number of top-retrieved documents k used for the feedback process and also for training the supervised PRF decision models. For each method we tune the $k \in [5, 40]$ via grid search on the training folds, and use the optimal value on the test fold. We use the same approach to tune the parameter α in Equation 7.10, which controls the importance of the feedback process for the rank-based fusion methods. For the R2F2-based methods, we conduct a grid search for α in the set $\{0, 0.1, \ldots, 1\}$. The number of terms used for relevance feedback was tuned for the collection and we use the optimal value across all the methods considered.

To obtain the initial retrieval list, we use both sparse and dense models. As a sparse model, we employ BM25 (Robertson *et al.*, 1995) to retrieve the top-1000 results from both MS MARCO and CARD collections. It is worth mentioning here that while for MS MARCO, the top list is comprised of small passages, in case of CARD dataset each sentence is considered as a document. A supervised neural model, namely, MonoT5 (Nogueira *et al.*, 2019b) is employed which operates by re-ranking the top-1000 of BM25. Note that MonoT5 model was trained only on the MS MARCO training queries.

Both RLM and FCRLM have a shared parameter, that is the number of terms, T, having the highest weight values, $P(w|R)$, which are used to calculate the KL divergence for re-ranking in a standard RLM framework (Lavrenko & Croft, 2001). FCRLM introduces an extra parameter, T', which represents the number of top-ranked feedback terms used during the second feedback step. We choose the value of T and T' via a grid search from the set $\{5, 6, 7, ..., 20\}$.

7.6 Results

Main observations. The key results of our experiments are reported in Table 7.2 and 7.3 for MS MARCO and CARD dataset , respectively. We see that the accuracy level of the decisions is quite satisfactory, even for the unsupervised threshold-based approaches. The scores also indicate that more accurate PRF decisions usually lead to an increase in retrieval effectiveness.

From Chapter 5 it is observed that our proposed FCRLM outperforms RLM significantly (see Table 5.2). Our intuition was that queries those were penalized due to applying feedback blindly (refer to Figure 5.4.6) would likely to

Table 7.2: Comparison of different SRF approaches on the TREC DL (2019 and 2020) topic sets with BM25 and MonoT5 set as the initial retrieval models. MAP values are computed for top-1000 documents. Paired t-test ($p < 0.05$) shows a significant improvement of Deep-SRF over the best performing baselines (comparing bold-faced results with the underlined ones).

		BM25 (ϕ: RLM)			BM25 (ϕ: GRF)			BM25 (ϕ: ColBERT-PRF)		
	Methods	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10
Baselines	No PRF	N/A	0.3766	0.5022	N/A	0.3766	0.5022	N/A	0.3766	0.5022
	PRF	N/A	0.4321	0.5134	N/A	0.4883	0.6226	N/A	0.4514	0.6067
	R2F2	N/A	0.4381	0.5140	N/A	0.5094	0.6332	N/A	0.4968	0.6184
	QPP-SRF	0.7835	0.4400	0.5152	0.7844	0.5321	0.6667	0.7742	0.5238	0.6400
	TD2F	0.7611	0.4392	0.5135	0.7580	0.4579	0.5900	0.7642	0.4910	0.6038
	LR-SRF	0.7842	0.4411	0.5154	0.7784	0.5107	0.6512	0.7854	0.5254	0.6414
Ours	Deep-SRF-CNN	0.7890	0.4522	0.5189	0.7944	0.5466	0.6692	0.8012	0.5403	0.6578
	Deep-SRF-CNN-R2F2		0.4619	0.5246		0.5521	0.6710		0.5495	0.6624
	Deep-SRF-BERT	**0.8081**	0.4705	0.5374	**0.8093**	0.5654	0.6821	**0.8165**	0.5631	0.6765
	Deep-SRF-BERT-R2F2		**0.4961**	**0.5486**		**0.5730**	**0.6839**		**0.5785**	**0.6873**
	Oracle	1.0000	0.5038	0.5528	1.0000	0.5876	0.6941	1.0000	0.5820	0.6936

		MonoT5 (ϕ: RLM)			MonoT5 (ϕ: GRF)			MonoT5 (ϕ: ColBERT-PRF)		
	Methods	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10
Baselines	No PRF	N/A	0.5062	0.6451	N/A	0.5062	0.6451	N/A	0.5062	0.6451
	PRF	N/A	0.5081	0.6463	N/A	0.5200	0.6487	N/A	0.5297	0.6491
	R2F2	N/A	0.5112	0.6484	N/A	0.5241	0.6494	N/A	0.5324	0.6502
	QPP-SRF	0.7963	0.5189	0.6559	0.7871	0.5313	0.6604	0.7900	0.5419	0.6673
	TD2F	0.7789	0.5071	0.6453	0.7670	0.4991	0.6403	0.7612	0.5179	0.5986
	LR-SRF	0.7958	0.5180	0.6543	0.7980	0.5422	0.6628	0.7928	0.5500	0.6654
Ours	Deep-SRF-CNN	0.8012	0.5233	0.6597	0.8011	0.5478	0.6683	0.7967	0.5565	0.6693
	Deep-SRF-CNN-R2F2		0.5287	0.6609		0.5518	0.6696		0.5579	0.6710
	Deep-SRF-BERT	**0.8152**	0.5306	0.6640	**0.8160**	0.5529	0.6694	**0.8067**	0.5624	0.6733
	Deep-SRF-BERT-R2F2		**0.5317**	**0.6659**		**0.5607**	**0.6719**		**0.5711**	**0.6746**
	Oracle	1.0000	0.5416	0.6786	1.0000	0.5722	0.6803	1.0000	0.5801	0.6821

be minimized if feedback were applied selectively, improving retrieval effectiveness further. Comparing 'RLM' results from Table 5.2 with 'PRF' results in Table 7.3 confirms the correctness of our initial intuition and the results further improve by applying Deep-SRF-BERT. The same can be concluded from the results obtained for TREC DL topic set in Table 7.2.

Next, we observe that supervised selective PRF approaches yield improved results over their unsupervised counterparts both for ad-hoc and causal datasets. Of particular interest is the fact that a data-driven approach outperforms the feature-based approach, as per our original hypothesis in Section 7.1. Given the success of Deep-SRF-BERT over Deep-SRF-CNN observed on TREC DL queries in Table 7.2, we report results only with the data-driven approach, Deep-SRF-BERT for our causal selective feedback (see Table 7.3). We see that the results are further improved through a soft combination of the initial and feedback lists via a confidence-based calibration (Deep-SRF-BERT-R2F2).

An interesting finding is that the SRF decision function trained on RLM on a set of queries generalizes well not only for a different set of queries (the test set),

Table 7.3: Comparison of different SRF approaches on the CARD topic sets with BM25 as the initial retrieval model. Similar to Table 7.2 MAP values are computed for top-1000 documents. A significant improvement of Deep-SRF over the best performing baselines (comparing bold-faced results with the underlined ones) are shown via paired t-test ($p < 0.05$). It is also noticeable that leveraging the top ranked documents obtained by FCRLM over RLM improves the retrieval effectiveness significantly for CARD topic sets (shown via paired t-test ($p < 0.05$)) by comparing the numbers in vertical columns in each group.

		BM25 (ϕ: RLM)			BM25 (ϕ: FCRLM)		
	Methods	Accuracy	MAP	nDCG@10	Accuracy	MAP	nDCG@10
Baselines	No PRF	N/A	0.2201	0.2844	N/A	0.2201	0.2844
	PRF	N/A	0.2487	0.3011	N/A	0.2571	0.3078
	R2F2	N/A	0.2503	0.3048	N/A	0.2611	0.3123
	QPP-SRF	0.6645	0.2531	0.3072	0.6743	0.2662	0.3170
	TD2F	0.6345	0.2489	0.3011	0.6402	0.2586	0.3100
	LR-SRF	0.6667	0.2556	0.3081	0.6781	0.2691	0.3202
Ours	Deep-SRF-BERT	**0.6932**	0.2645	0.3142	**0.7063**	0.2711	0.3193
	Deep-SRF-BERT-R2F2		**0.2671**	**0.3151**		**0.2742**	**0.3200**
	Oracle	1.0000	0.2856	0.3342	1.0000	0.2930	0.3320

but also across different feedback models as observed both in Table 7.2 and 7.3. This suggests that the queries which improve with RLM also improve with other feedback models, such as GRF or ColBERT-PRF. This can be seen from the GRF and the ColBERT-PRF group of results for both BM25 and MonoT5 in Table 7.2. This entails that the SRF based decision function does not need to be trained for specific PRF approaches, which makes it more suitable to use in a practical setup. The results reported in Table 7.3 also confirm the generalized nature of our proposed Deep-SRF-BERT in a sense that the decision function that is trained on FCRLM is capable of improving performance of RLM as well.

We observe that the best results obtained by our method are close to those achieved by an *oracle* for MS MARCO; whereas the performance achieved by CARD dataset shows difference from that of its oracle. This observation again emphasizes the fact that capturing subtle causal relevance is way more challenging compared to its topical counterpart. In the ideal oracle scenario, PRF is applied *only if* the AP of a query is actually improved (i.e., the oracle uses the relevance assessments for the test queries). The fact that the results from Deep-SRF-BERT are close to the oracle suggests that further attempts to increase the accuracy of PRF decisions may have little impact on retrieval effectiveness, likely due to saturation effects.

Per-query analysis. To provide further context, Table 7.4 shows examples of queries both from TREC DL and CARD dataset. Firstly, we see that the average differences in the AP values before and after feedback are mostly higher for the

Table 7.4: Contingency tables of the Deep-SRF-BERT model with sample queries both from TREC DL (top) and CARD (bottom). Here, $|Q|$ is the count of queries for each of the 4 possible cases of prediction (true/false positives and true/false negatives), and $\overline{\Delta\text{AP}}$ denotes the average ΔAP values of each cell, where $\Delta\text{AP}(Q) = \frac{\text{AP}(\phi_M(Q)) - \text{AP}(Q)}{\text{AP}(Q)}$.

		Actual			
		$\Delta\text{AP} > 0$		$\Delta\text{AP} \leq 0$	
Predicted	$\Delta\text{AP} > 0$	What is active margin? Exon definition Biology	$\lvert Q\rvert = 59$ $\overline{\Delta\text{AP}} = 0.1302$	Why is Pete Rose banned from hall of fame? What are best foods to lower cholesterol?	$\lvert Q\rvert = 8$ $\overline{\Delta\text{AP}} = 0.0525$
	$\Delta\text{AP} \leq 0$	Define BMT medical Who is Robert Gray?	$\lvert Q\rvert = 11$ $\overline{\Delta\text{AP}} = 0.0246$	Do Google docs auto save? How many sons Robert Kraft has?	$\lvert Q\rvert = 19$ $\overline{\Delta\text{AP}} = 0.0737$

		Actual			
		$\Delta\text{AP} > 0$		$\Delta\text{AP} \leq 0$	
Predicted	$\Delta\text{AP} > 0$	Why do some Takata airbags need to be replaced twice? Why are more police dogs dying in the line of duty?	$\lvert Q\rvert = 22$ $\overline{\Delta\text{AP}} = 0.1442$	Why does Alfonso Fanjul open to investing in Cuba? Why is amphetamine use affecting our waterways?	$\lvert Q\rvert = 9$ $\overline{\Delta\text{AP}} = 0.0164$
	$\Delta\text{AP} \leq 0$	Why is Haiti seeing a surge in cholera? Why did Lego release audio braille instructions?	$\lvert Q\rvert = 6$ $\overline{\Delta\text{AP}} = 0.0121$	Why might future medical breakthroughs come from IT industry? Why is China projecting military power into the South China Sea?	$\lvert Q\rvert = 8$ $\overline{\Delta\text{AP}} = 0.0823$

green cells, which indicates that the penalty incurred due to queries for which the model (Deep-SRF-BERT) predicts incorrectly is not too high. This also conforms to the fact that at close to 80% accuracy, Deep-SRF-BERT achieves results close to the oracle. Secondly, a manual inspection of the examples reveals that the queries for which the Deep-SRF-BERT model correctly decides to apply PRF appear to be those with under-specified information needs. In other words, these are queries that would likely benefit from enrichment. An example of such a query is 'what is active margin' or 'Why do some Takata airbags need to be replaced twice?' in Table 7.4.

7.7 Conclusions

Pseudo-relevance feedback (PRF) has the potential to improve average retrieval effectiveness across a sufficiently large number of queries. However, PRF can also lead to a deviation from the original information need, which may reduce the retrieval effectiveness for certain queries. Additionally, in the context of searching for causal information, the nuanced nature of cause-and-effect relationships means that such query drifts result in more significant

penalties. While a selective application of PRF can potentially alleviate this issue, previous approaches have largely relied on unsupervised or feature-based learning to determine whether a query should be expanded. We revisited the problem of selective PRF from a deep learning perspective, presenting a model that is entirely data-driven and trained in an end-to-end manner. We introduced two different architectures – one that involves early interaction between queries and their top-retrieved documents, and another that involves a late interaction between the query-document encodings obtained via transformers. We also made use of the confidence estimates of our models to effectively combine the information from the original queries and their expanded versions to further improve retrieval effectiveness. In our experiments, we applied this selective feedback on a number of different combinations of ranking and feedback models, demonstrating that our proposed approach consistently improves retrieval effectiveness for both sparse and dense ranking models, with the feedback models being either sparse, dense or generative both in ad-hoc and causal search paradigm.

CHAPTER

EIGHT

FINAL CONCLUSIONS

In this chapter we revisit the necessity of introducing a new search paradigm that addresses causal information needs and the corresponding concerns about a lack of suitable retrieval systems. We re-examine the research challenges that were identified following a comprehensive literature review and we explore how the contributions presented in the technical chapters of this thesis make important strides towards causality-driven search paradigm. We conclude with a discussion of the limitations of our work, along with an exploration of several promising avenues for future research in building a more accountable and effective causal IR landscape.

8.1 Main Contributions

Traditional information retrieval systems are primarily concerned with locating materials that are topically relevant and descriptive of a certain query term. In settings like news article collections, users typically search for documents that not only depict a news event but also delve into the sequence of events that potentially led to its occurrence. These connections can be complicated, involving multiple causative elements. We define the problem of *causal information retrieval* as a result of this information need.

In Chapter 2, we presented a full review of numerous significant research gaps in this field, all of which were subsequently addressed in the technical chapters of this thesis. We briefly remind the reader of the research gaps that were originally highlighted in Chapter 1:

- Is a typical search system sufficient for detecting causally significant information, or does a new research paradigm, namely *causal information*

retrieval, need to be introduced? (Discussed in detail in Section 1.2.1 and addressed throughout Chapter 3).

- Can we create a system that generates a list of plausible causes, represented as short text segments, in response to any given causal query without any supervision? (Discussed in detail in Section 1.2.2 and addressed throughout Chapter 5).
- Can we develop a supervised decision-making pipeline that can determine when query reformulation is required to capture causal relevance? (Discussed in detail in Section 1.2.3 and addressed throughout Chapter 6 and 7).

In Chapter 2 we emphasize that while there is a long history of diverse work in the general area of causality, the existing techniques we identified have only considered limited forms of causal relations at the sentence or document level via a comprehensive literature review. In some cases, such as patterns and contingent discourses, these methods require prior knowledge about causal events, whereas in others, they require some predefined lexical, syntactic, or morphological relations. These techniques, however, do not address the nuanced causes and effects found in larger document collections, such as those we target to capture using retrieval models.

The first two technical chapters of this thesis (Chapter 3 and 4) illustrated how the proposed causal retrieval task differs from standard retrieval problems, showcasing several notable contributions:

- **Creating an initial pilot dataset.** We created an initial pilot dataset for the novel causal document retrieval task that enumerates a list of cause indicative documents in response to an user's query.
- **Standard retrieval models are not adequate for causality.** A series of rigorous experiments were conducted on the pilot dataset to demonstrate that standard retrieval models do not suffice for causality due to the subtle nature of causally-relevant documents in relation to query events.
- **Recursive causal retrieval framework.** We proposed a new recursive causal retrieval framework design that allows for in-depth exploration of a query to find a chain of likely causes.
- **A fine-grained novel causal dataset.** We developed a newly annotated, fine-grained dataset that was created specifically to meet the needs of the

retrieval framework, namely, identifying precise pieces of information within causally relevant documents. This has been made available for other researchers working in the area.

Subsequently, in Chapter 5 we hypothesized that, while there may be some term overlap between causally relevant documents and those that are topically relevant for a query, a significant portion of these documents will use a distinct set of terms to describe various potential causes that could result in specific effects. In tandem with methodological advances, this chapter also made contributions in the evaluation of causal IR, which are summarized as follows:

- **Unsupervised causal feedback model.** We proposed an unsupervised feedback model for estimating a distribution of terms that are relatively rare but have high weights in the topically relevant distribution, indicating potential causal relevance.
- **Significant improvements in causal information search.** We demonstrated that this feedback model is significantly more effective than traditional IR models and several other causality heuristic baselines in detailed experiments on both ad-hoc IR datasets and our newly created causal dataset.

As a further contribution of this thesis, Chapter 7 presented a new supervised approach for improving retrieval effectiveness in the context of causality. The fundamental concept is to analyze input queries and predict their performance relative to the collection (as discussed in Chapter 6). This allows for determining whether or not to use feedback to capture a larger number of relevant documents, while minimizing the risk of query drift. The main contributions of these two chapters can be summarized as:

- **CNN-based query performance predictor.** We proposed a data-driven end-to-end convolutional neural framework for predicting query specificity in ad-hoc retrieval.
- **Cross-encoder-based query performance predictor.** A novel end-to-end neural cross-encoder-based approach for estimating the specificity of an input query was introduced and validated across a number of benchmark IR datasets.

- **Selective feedback approach.** We proposed a new deep-learning framework for the decision-making pipeline based on the data-driven convolutional and cross-encoder-based query estimators that we proposed in Chapter 6.
- **Retrieval effectiveness of selective approach.** We showed the model's effectiveness in a large number of experiments on standard benchmark datasets and our newly proposed causal dataset.

Collectively, the three research question outlined in Section 1.2 have been directly addressed in the technical chapters of this thesis. Consequently, our work offers significant contributions toward causality-driven ad-hoc information retrieval. However, it is important to acknowledge several limitations and potential avenues for future research, which warrant further discussion.

8.2 Promising Avenues for Future Work

Recursive causal information retrieval. In Chapter 3, we proposed an architecture for a *recursive* causal retrieval model that can help users to perform in-depth exploration in terms of causality pertaining to a news event, and the chain of causes which led to that event. Since our proposed model is recursive in nature, the retrieval performance at any current stage influences greatly its subsequent course of action. Thus, the more we retrieve cause-specific documents (i.e., document excerpts in our case) in response to the initial effect in the form of a query, the better the recursive queries that we identify further down the chain of causes. In contrast, a poor set of initially-retrieved documents would likely lead to poor results further down the chain. Thus, accurately pinpointing the first-level causes represents a key challenge in causal retrieval, a topic we have thoroughly addressed in this thesis.

As future work, we intend to explore ways to construct deeper causal chains of events in a recursive manner. That is, instead of outputting a ranked list of documents as potential causes to a given query event, we intend to extract events from the retrieved articles, treat them as queries in turn, and retrieve a list of further causes (see Figure 3.3). We would like to create a more fine-grained dataset for building such a recursive causal pipeline. The dataset we introduced in Chapter 4 could naturally be extended for this purpose. We also aim to carry out user studies to explore effective methods for integrating causally-relevant content within a standard search interface.

Smarter selective feedback. In Chapter 7, we proposed an adaptive relevance feedback framework that includes a data-driven supervised neural approach to optimize retrieval effectiveness by applying feedback on queries in a selective fashion. By testing this approach using multiple neural architectures and over different standard test collections, we observed that it performs favorably compared to alternative strategies. Furthermore, it approached the performance of an oracle system, which always perfectly decides whether or not to apply PRF. We found that different neural architectures exhibited different trade-offs in terms of computational efficiency and performance.

This work opens the door to interesting future research directions. While we find our approach to be effective, it does require the execution of PRF to identify the quality of the results. It may be useful to investigate methods to identify whether this PRF step is worth executing, thus potentially reducing the computational cost for queries where PRF is eventually deemed unnecessary. Further work could also examine strategies for predicting the parameters of PRF itself, such as the number of relevant documents to include.

Generalized QPP architecture. Chapter 6 proposed a new 'Pointwise-Query, Listwise-Document' approach, qppBERT-PL for query performance prediction. We found that the model yields significant relative improvements in QPP compared to the existing literature. To the best of our knowledge, this is the first contribution in QPP that transforms the pointwise QPP objective into a listwise classification task.

As a BERT-based model, qppBERT-PL currently faces limitations due to the maximum length of a BERT sequence (512 tokens). In the future, we aim to adapt the model to handle longer documents. This could be achieved by dividing a lengthy document into smaller segments and then aggregating the information from these segments. Additionally, we are keen to investigate alternative neural architectures and training objectives to reduce the computation time involved in this listwise-document approach.

8.3 Closing Comments

The field of Causal Information Retrieval has experienced a significant and dynamic evolution, with researchers striving to decipher complex patterns within textual content. Existing research on causality has considered relations either at the sentence level or within a single document. In some cases, these

methods require prior knowledge about causal events, while in other cases they necessitate some predefined lexical, syntactic, or morphological relations. However, these techniques do not cover the nuanced causes and effects in larger document collections, such as news collections. Motivated by the potential challenges of causal IR and the growing deployment of neural architecture in information retrieval, this thesis has demonstrated the promise of relevance feedback-based models in relation to capturing causal relevance from the collection. The two primary conclusions of this PhD thesis are: first, the factored relevance method demonstrates greater efficiency for causality-based IR compared to standard relevance feedback; and second, selectively applying feedback to factored relevance further enhances retrieval effectiveness.

APPENDIX

ONE

IR BACKGROUND AND RELATED CONCEPTS

A.1 Introduction to Information Retrieval

Information Retrieval (IR) revolves around the science of searching. Classic examples might include searching a telephone directory for a specific number or finding the closest hospitals. For much of the 20th century, IR remained relatively unexplored, limited to niche applications like library article searches or referencing old legal cases. However, with the emergence of the World Wide Web, the significance of IR has surged, making it essential across numerous fields and a cornerstone of daily information consumption worldwide. This section provides a brief introduction to IR, together with an architectural view of the relevant processes. We will explore essential concepts in IR and the evaluation methodologies that are essential for understanding the research presented in this thesis.

Despite the technological advancements, textual data remains the predominant medium for storing online information. This thesis focus on *textual information retrieval*, which can be defined as:

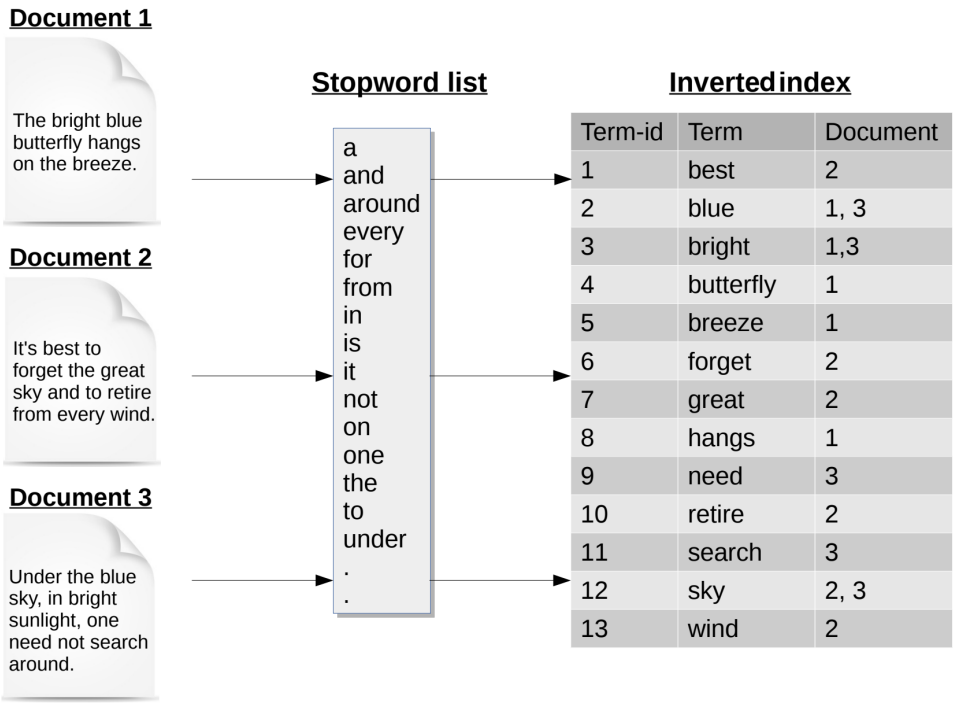
> Given an information need, and a collection of documents stored in unstructured textual form, Information Retrieval is the process of finding documents from that collection which satisfy the information need.

Typically the information need will be encoded as a user-specified search *query*, while the response of the IR system will take the form of a ranked list of documents, where the most relevant documents appear at the top of the list.

In a broad sense, the standard process of IR can be split into two distinct steps: i) indexing, and ii) retrieval. The initial process of indexing typically involves processing and storing the collection of documents (often referred to as *corpus*) in a manner that facilitates the subsequent recall of information. Prior to indexing, non-informative terms, sometimes referred to as *stopwords*, are usually removed and *stemming* is performed to reduce terms to their root form. An *inverted index* is generated from the collection as part of the indexing process. This index is comprised of two primary components: the inverted list (often called a posting list) and the dictionary. The inverted list associates each term from the vocabulary with a list of documents where that term appears. On the other hand, the dictionary stores all of the unique terms present in the vocabulary.

A key consideration for IR revolves around the ability to access information in a timely manner. For traditional indexing techniques, this is achieved by storing the dictionary in the primary memory, with pointers to each inverted list which in turn is stored in the secondary memory. A weight is usually assigned to each document containing a term that is stored in the inverted list. These weights are taken into account when ranked retrieval is carried out. During the process of indexing, the dictionary stores per-term collection

Figure A.1: A conceptual model of indexing for IR.



statistics which ensures that all essential information needed to compute the retrieval score for a document is extracted in one single lookup operation. An example illustrating the process of indexing is presented in Figure A.1 for a corpus of three documents.

A.2 Baseline Retrieval Models

Search engines rely on specific algorithms, often called *retrieval models* or *retrieval functions*, to find relevant content in response to an input query. Several of these models have proven to be highly efficient. In this thesis, we implement language model-based retrieval methods that are dependent on smoothing techniques, notably on smoothing techniques, such as Jelinek-Mercer and Dirichlet (Zhai, 2008; Zhai & Lafferty, 2001). The relevance of these strategies lie in the fact that they are frequently used as baseline retrieval methods (e.g. Zamani & Croft, 2016, 2017; Zheng & Callan, 2015; Guo *et al.*, 2016; Paik, 2015). Considerable research in this area has also involved the use of probabilistic and information theoretic methods, including BM25 (Robertson & Walker, 1994; Robertson & Zaragoza, 2009) and measuring divergence from randomness (Amati & Van Rijsbergen, 2002). The next subsection describes key techniques, such as relevance feedback and relevance-based language models – a leading state-of-the-art query expansion (QE) technique using relevance feedback. These serve as baselines for several experiments conducted later in this thesis.

A.2.1 Language Modeling

Here, we introduce the underlying concept of the language model-based retrieval pipeline. Consider a query Q, a document d, and a language model $\mathcal{D}$ estimated from d. The posterior probability $P(\mathcal{D}|Q)$ gives the document score with respect to Q in decreasing order. The estimate of $P(\mathcal{D}|Q)$ is obtained for d when the collection is being indexed with the aid of prior probability $P(Q|d)$ as per Bayes rule (Ponte & Croft, 1998; Zhai & Lafferty, 2001):

$$\begin{aligned} p(d|Q) &= \frac{p(Q|\mathcal{D}.p(\mathcal{D}))}{\sum_{d' \in C} p(Q|\mathcal{D}')} \propto p(Q|\mathcal{D}).p(\mathcal{D}) = p(\mathcal{D}).\prod_{q \in Q} p(q|\mathcal{D}) \\ &\propto \prod_{q \in Q} p(q|\mathcal{D}) \end{aligned} \tag{A.1}$$

Let V denote the corpus vocabulary size and $p(w_i|d)$ denote the probability of randomly picking a word w_i from d. Then a unigram language model, $\mathcal{D} = \{p(w_i|d)\}_{i\in[1,|V|]}$, represents an approximation of the language model $\mathcal{D}$ associated with d. Following this, the retrieval score of d for query Q can be defined as:

$$\begin{aligned} \text{Score}(Q, d) &= p(Q|\mathcal{D}) \\ &= \prod_{q\in Q} p(q|d) \end{aligned} \tag{A.2}$$

Jelinek-Mercer smoothing. From Equation (A.2), it is evident that a missing query term from d would result in a score of 0. This necessitates addressing the zero probability issue in the language mode $\mathcal{D}$. A common solution involves computing the Maximum Likelihood Estimate (MLE) of $p(w_i|d)$ for a background language model for the entire collection, C and interpolating the same. Formally, this is given by:

$$\begin{aligned} p(Q|\mathcal{D}) &= \prod_{q\in Q} [\lambda p(q|d) + (1-\lambda)p(q|C)] \\ &= \prod_{q\in Q} \lambda \frac{tf(q,d)}{|d|} + (1-\lambda)\frac{cf(q)}{|C|} \end{aligned} \tag{A.3}$$

where $tf(q, d)$ and $cf(q)$ denote the number of times q occur in d and in C respectively, $|d|$ and $|C|$ indicate the size of document d and collection C respectively, and $\lambda = [0, 1]$ is the interpolation parameter. The method described by Equation (A.3) is often termed as the language model with Jelinek-Mercer smoothing or linear smoothing, abbreviated as LM-JM throughout this thesis.

Dirichlet smoothing. Another widely-used method is Dirichlet smoothing, which relies on Bayesian estimation, in contrast to MLE in LM-JM. The model is similar to the one presented in Equation (A.3), with the key difference being in the interpolation parameter with a dynamic coefficient dependent on the length of the document. This is given by

$$P(Q|\mathcal{D}) = \prod_{q\in Q} \frac{tf(q,d) + \mu p(q|C|)}{|d| + \mu} \tag{A.4}$$

where $tf(q, d)$ is the term frequency of q in d, and $p(Q|C)$ is the probability of occurrence of q in C. The interpolation parameter μ can be interpreted as the pseudo count of words obtained through prior probabilities. Typically, the value of μ is set within the range $[100, 5000]$. The approach presented in Equation (A.4) is commonly referred to as language model with Dirichlet smooth-

ing, abbreviated as LM-Dir henceforth.

A.2.2 BM25

BM-25 is a probabilistic-based function used in traditional information retrieval systems to rank documents based on their relevance to a given query, which offers better length normalization factors (Robertson & Walker, 1994; Robertson & Zaragoza, 2009). Here, the retrieval score of a document d with respect to its query q is computed as

$$\text{Score}(Q,d) = \sum_{q \in Q} log \frac{D - df(q) + 0.5}{df(q) + 0.5} \frac{tf(q,d)(k_1+1)}{tf(q,d) + k_1(1 - b + b\frac{|d|}{avgdl})} \tag{A.5}$$

where D denotes the number of documents in the collection, $df(q)$ denotes the number of documents in the collection containing the term q, $tf(q,d)$ denotes the number of occurrences of q in d, $avgdl$ is the average document length of collection, and b is a length normalization parameter. The tuning parameter k_1 is designed to calibrate document term frequency scaling.

A.3 Word Embeddings

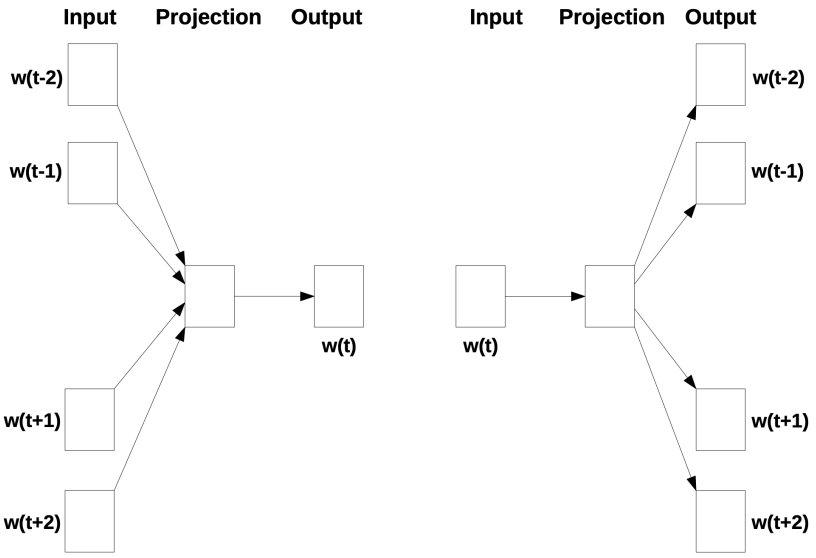
The idea that terms can be represented as vectors in a multi-dimensional space has been in existence for quite sometime (Collobert *et al.*, 2011; Tredici & Bel, 2015; Gharbieh *et al.*, 2016; Joshi *et al.*, 2016; Salehi *et al.*, 2015). However, when (Mikolov *et al.*, 2013) first introduced the *word2vec* algorithm, it popularized thee use of word embedding in natural language processing. This technique and its counterparts, such as *GloVe* (Pennington *et al.*, 2014), rely on using vectors to represent words in an abstract dimensional space through which the semantic similarity between two constituent terms of a document is studied.

Word embeddings can capture linguistic patterns such conceptual combination and laws of analogy through basic algebraic operations on their vectors. A number of useful characteristics of word embeddings are listed below:

1. An approximation of the *semantic similarity* between two words can be provided by the cosine similarity between their respective vectors.

2. A simple addition of the embedded vectors can be useful in capturing the effect of conceptual composition. For instance, *vec('Bann')* might

Figure A.2: A graphical illustration of of the *continuous bag-of-words* (left) and the *skip-gram* (right) models of `word2vec`.



be a close approximation of the resultant vector obtained by adding *vec('Ireland')* and *vec('river')*.

3. Laws of analogy are obeyed by the embedded vectors. For example, the operation *vec('Paris')* - *vec('France')* + *vec('Italy')* yields an approximate representation of *vec('Rome')*.

4. Embedded vectors can serve as features for various supervised text processing tasks such as document classification, named entity recognition, and sentiment analysis, with their inherent semantic information making them useful when performing these tasks.

Semantic distances between terms, used to deduce the similarity function, play a key role when ranking documents, making the first characteristic above crucial in our own work.

The Skip-gram and Continuous Bag-of-Words (CBOW) are the two word2vec models employed to build word embeddings. A single hidden layer neural network is used by both of these models. The key difference is that CBOW predicts a word from its context, whereas Skip-gram predicts the context from a given word. Figure A.2 provides a visual representation of these models.

Negative sampling is the key algorithm which is used to train the neural network in word2vec. For every word, a positive evidence set $\mathcal{D}$ is established consisting of different words in a particular context. In a similar manner, a

negative evidence set $\mathcal{D}'$ outside of the word is also defined. The probability that a word, w_c is present in the context of a set of words w_t for a given ordered pair (w_t, w_c) can be written as:

$$arg\ max_\theta \prod_{(w_t,w_c)\in\lceil} P(D_{w_t,w_c} = 1|w_t, w_c) \prod_{(w_t,w_c)\in\mathcal{D}'} P(D_{w_t,w_c} = 0|w_t, w_c) \quad \text{(A.6)}$$

Maximizing the probability of sampling w_t from its context $\mathcal{D}$ and at the same time minimizing the same from outside the context i.e. $\mathcal{D}'$ is the primary purpose of the objective function for training the RNN. When vector representation of words are used, the optimization function may be represented by

$$arg\ max_\theta \sum_{(w_t,w_c)\in\mathcal{D}} \log\sigma(v_{w_c}.v_{w_t}) - \sum_{(w_t,w_c)\in\mathcal{D}'} \log\sigma(v_{w_c}.v_{w_t}) \quad \text{(A.7)}$$

where $v(w)$ denotes the embedded representation of the word w and the sigmoid function is calculated as $\sigma(x) = \frac{1}{1+e^{-}x}$. The inner product of vectors for w_t with respect to w_c is maximized by using Equation (A.7), ensuring a higher level of similarity for these vectors, while simultaneously reducing the similarity with respect to the vectors outside of the context.

A.4 Evaluation Methodology

A.4.1 Retrieval Evaluation Metrics

We now discuss the various metrics used to compare the performance of different retrieval methods in the experiments in this thesis. For a required set of information, let us assume a set of queries $\mathbb{Q}$. Let Rel_Q denote the total number of documents relevant to $\{Q : Q \in \mathbb{Q}\}$ present in the collection. A ranked list $L = \{d_1, d_2, ..., d_n\}$ contains a set of n documents retrieved by applying a function $\mathcal{F}$. These documents are ordered based on their retrieval scores, such that $\{score(d_i, Q) \geq score(d_j, Q)\ \forall i < j\}$.

Mean Average Precision (MAP). Mean Average Precision (MAP) evaluates the retrieval quality of a model across recall levels by yielding a single-figure measure. Amongst other retrieval evaluation metrics, a satisfactory discriminative property with significant stability is exhibited by MAP (Manning *et al.*, 2008). in the top K retrieved documents For each relevant document retrieved in the top k documents, the *average precision* is indicative of the mean of the precision

values for a given information need. Formally, each document is ranked depending on the position at which it is retrieved, i.e. $Rank(d_i) = i, i \in \mathbb{N}$. The average precision (AP) for query Q is then defined as

$$AP(Q) = \frac{1}{|Rel_Q|} \sum_{d_m \in RelRet_Q} \frac{m}{Rank(d_m)} \tag{A.8}$$

where $RelRet_Q = \{d_1, d_2, ..., d_m\}$ is the set of m relevant documents retrieved among the documents of $Ret_Q (Ret_Q \subseteq RelRet_Q)$, such that $n \geq m$. Thus, the overall MAP score of $\mathbb{Q}$ is given by:

$$MAP(\mathbb{Q}) = \frac{1}{|\mathbb{Q}|} \sum_{Q \in \mathbb{Q}} AP(Q) \tag{A.9}$$

This metric is usually calculated on the top n documents, where n is generally set to 1000.

Precision at Rank k (P@k). In the ranked list of documents L, let $Relevant(L_k)$ be the total number of relevant documents up to the rank k. The precision at depth k is therefore given by:

$$P@k = \frac{Relevant(L_k)}{k} \tag{A.10}$$

For a given topic set $\mathbb{Q}$, averaging $P@k$ for all the queries gives the value of $P@k$ in a way similar to MAP. To evaluate the rank at depth 5, k is set to 5 in this thesis.

Recall at Rank k (Recall@k). We define the recall at depth k for a query Q as:

$$Recall@k = \frac{Relevant(L_k)}{Rel_Q} \tag{A.11}$$

The recall at rank k is computed by averaging the individual recall values over all the queries for the entire topic set $\mathbb{Q}$. The value of k is set to 1000 for the experiments presented in this thesis.

Mean Reciprocal Rank (MRR). This metric is used to assess systems that retrieve a ranked list of responses to input queries. Reciprocal rank is the multiplicative inverse of the rank of the first correct answer, i.e., 1 for first, $1/2$ for second, $1/3$ for third, and so on. If there is no document retrieved, the reciprocal rank is 0. The average reciprocal rank is the sum of the reciprocal ranks of

results for a query set $\mathbb{Q}$, which can be expressed as:

$$MRR(\mathbb{Q}) = \frac{1}{|\mathbb{Q}|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \tag{A.12}$$

Normalized Discounted Cumulative Gain at Rank k (nDCG@k). The discounted cumulative gain (DCG) for k retrieved relevant documents sums the relevance of these documents in relation to the current query (cumulative), while also including a penalty for relevant documents placed later in the ranked list (discounted). Therefore, DCG can be calculated as

$$DCG = \sum_{i=1}^{ranks} \frac{Gain_i}{\log_2(i+1)} \tag{A.13}$$

where $Gain_i$ is the relevance score of the i^{th} retrieved document.

Normalized discounted cumulative gain (nDCG) normalizes the DCG score with respect to the ideal discounted cumulative gain (IDCG), which represents the DCG of the ideal ranking. Thus, nDCG can be expressed as

$$nDCG@k = \frac{DCG@k}{IDCG@k} = \frac{\sum_{i=1}^{k(actual\ order)} \frac{Gain_i}{\log_2(i+1)}}{\sum_{i=1}^{k(ideal\ order)} \frac{Gain_i}{\log_2(i+1)}} \tag{A.14}$$

which yields values $\in [0, 1]$, with higher values indicating better performance.

A.4.2 QPP Evaluation Metrics

Pearson's Correlation Coefficient ($P - r$) is used to assess the strength of a linear relationship between two variables x and y. A value of $r = 1$ indicates a perfect positive correlation, while $r = -1$ indicates a perfect negative correlation. We compute r as

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x}).(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}.\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{A.15}$$

where $\bar{x}$ and $\bar{y}$ denote the sample mean of x and y, respectively.

Kendall's Correlation Coefficient ($K - \tau$) quantifies the similarity of two ranked transformed data orderings x and y. Its output can be interpreted as the probability such that, as x increases, y increases, re-scaled from -1 to 1.

Formally we define

$$\tau_{xy} = \frac{2}{n(n-1)} \sum_{i<j} sgn(x_i - x_j).sgn(y_i - y_j) \tag{A.16}$$

where n denotes the number of pairs and $sgn()$ is the standard $sign$ function. Note that the equation above is only applicable in cases where there are no ties in the sample data.

BIBLIOGRAPHY

(2021). Apache Lucene. https://lucene.apache.org/, accessed: 2021-05-25.

(2021a). Causality-driven Adhoc Information Retrieval. https://cair-miners.github.io/CAIR-2021-website/, accessed: 2021-08-04.

(2021b). Huggingface Transformers. https://huggingface.co/transformers/, accessed: 2021-05-25.

(2021). Keras. https://keras.io/, accessed: 2021-05-25.

Ahmad, W.U., Chang, K. & Wang, H. (2019). Context attentive document ranking and query suggestion. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, 385–394, Association for Computing Machinery, New York, NY, USA.

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

Arabzadeh, N., Khodabakhsh, M. & Bagheri, E. (2021). BERT-QPP: Contextualized pre-trained transformers for query performance prediction. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, 2857–2861, Association for Computing Machinery, New York, NY, USA.

Asadi, N., Metzler, D., Elsayed, T. & Lin, J. (2011). Pseudo test collections for learning web search ranking functions. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, 1073–1082, Association for Computing Machinery, New York, NY, USA.

Asghar, N. (2016). Automatic extraction of causal relations from natural language texts:A comprehensive survey. *CoRR*, **abs/1605.07895**.

Austin, P. (2011). An introduction to propensity score methods for reducing the effects of confounding in observational studies. *Multivariate Behavioral Research*, **46**, 399–424.

Bashir, S. & Rauber, A. (2009). Improving retrievability of patents with cluster-based pseudo-relevance feedback documents selection. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, 1863–1866, Association for Computing Machinery, New York, NY, USA.

BBC Middle East editor (2020). Five reasons why Israel's peace deals with the UAE and Bahrain matter. `https://www.bbc.com/news/world-middle-east-54151712`, online; accessed 20 August 2021.

Beamer, B. & Girju, R. (2009). Using a bigram event model to predict causal potential. In *Proceedings of the 10th International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing '09, 430–441, Springer-Verlag, Berlin, Heidelberg.

Belkin, N.J., Oddy, R.N. & Brooks, H.M. (1982). Ask for information retrieval: Part i. background and theory. *Journal of documentation*.

Billerbeck, B. & Zobel, J. (2004). Questioning query expansion: An examination of behaviour and parameters. In *Proceedings of the 15th Australasian Database Conference - Volume 27*, ADC '04, 69–76, Australian Computer Society, Inc., AUS.

Blanco, E., Castell, N. & Moldovan, D. (2008). Causal relation extraction. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC'08)*, European Language Resources Association (ELRA), Marrakech, Morocco.

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J. & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani & K.Q. Weinberger, eds., *Advances in Neural Information Processing Systems 26*, 2787–2795, Curran Associates, Inc.

Butman, O., Shtok, A., Kurland, O. & Carmel, D. (2013). Query-performance prediction using minimal relevance feedback. In *Proceedings of the 2013 Conference on the Theory of Information Retrieval*, ICTIR '13, 14–21, Association for Computing Machinery, New York, NY, USA.

Byerly, A., Kalganova, T. & Dear, I. (2020). A branching and merging convolutional network with homogeneous filter capsules. *CoRR*, **abs/2001.09136**.

Cao, G., Nie, J.Y., Gao, J. & Robertson, S. (2008). Selecting good expansion terms for pseudo-relevance feedback. In *SIGIR '08*, 243–250, Association for Computing Machinery, New York, NY, USA.

Carmel, D. & Yom-Tov, E. (2010). Estimating the query difficulty for information retrieval. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, 911, Association for Computing Machinery, New York, NY, USA.

Carmel, D., Yom-Tov, E., Darlow, A. & Pelleg, D. (2006). What makes a query difficult? In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, 390–397, Association for Computing Machinery, New York, NY, USA.

Carterette, B., Kanoulas, E., Hall, M.M. & Clough, P.D. (2014). Overview of the TREC 2014 session track. In *Proceedings of TREC 2014*.

Chang, D.S. & Choi, K.S. (2005). Causal relation extraction using cue phrase and lexical pair probabilities. In K.Y. Su, J. Tsujii, J.H. Lee & O.Y. Kwong, eds., *Natural Language Processing – IJCNLP 2004*, 61–70, Springer Berlin Heidelberg.

Chang, D.S. & Choi, K.S. (2006). Incremental cue phrase learning and bootstrapping method for causality extraction using cue phrase and word pair probabilities. *Information Processing & Management*, **42**, 662–678.

Chen, Q., Zhu, X., Ling, Z., Wei, S., Jiang, H. & Inkpen, D. (2017). Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics ACL 2017, Volume 1: Long Papers*, 1657–1668.

Chifu, A.G., Laporte, L., Mothe, J. & Ullah, M.Z. (2018). Query performance prediction focused on summarized letor features. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, 1177–1180, Association for Computing Machinery, New York, NY, USA.

Christlein, V., Spranger, L., Seuret, M., Nicolaou, A., Kral, P. & Maier, A. (2019). Deep generalized max pooling. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 1090–1096, IEEE Computer Society, Los Alamitos, CA, USA.

Cohen, D., Foley, J., Zamani, H., Allan, J. & Croft, W.B. (2018). Universal approximation functions for fast learning to rank: Replacing expensive regression forests with simple feed-forward networks. In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '18, 1017–1020.

Cohen, D., Mitra, B., Lesota, O., Rekabsaz, N. & Eickhoff, C. (2021). Not all relevance scores are equal: Efficient uncertainty and calibration modeling for deep retrieval models. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, 654–664, Association for Computing Machinery, New York, NY, USA.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K. & Kuksa, P. (2011). Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, **12**, 2493–2537.

Cormack, G.V., Clarke, C.L.A. & Buettcher, S. (2009). Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, 758–759, Association for Computing Machinery, New York, NY, USA.

Craswell, N., Mitra, B., Yilmaz, E., Campos, D. & Voorhees, E.M. (2019). Overview of the TREC 2019 deep learning track. In *Proceedings of the Twenty-eighth Text REtrieval Conference, TREC 2019*, vol. 1265 of *NIST Special Publication*.

Craswell, N., Mitra, B., Yilmaz, E. & Campos, D. (2020). Overview of the TREC 2020 deep learning track. In *Proceedings of the Twenty-Ninth Text REtrieval Conference, TREC 2020*, vol. 1266 of *NIST Special Publication*.

Cronen-Townsend, S., Zhou, Y. & Croft, W.B. (2002). Predicting query performance. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '02, 299–306, Association for Computing Machinery, New York, NY, USA.

Cronen-Townsend, S., Zhou, Y. & Croft, W.B. (2004). A framework for selective query expansion. In *Proceedings of the 13th ACM Conference on Information and Knowledge Management*, CIKM '04, 236–237, ACM, New York, NY, USA.

Cronen-Townsend, S., Zhou, Y. & Croft, W.B. (2006). Precision prediction based on ranked list coherence. *Information Retrieval*, **9**, 723–755.

Cummins, R. (2014). Document score distribution models for query performance inference and prediction. *ACM Transactions on Information Systems*, **32**.

Cummins, R., Jose, J. & O'Riordan, C. (2011). Improved query performance prediction using standard deviation. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, 1089–1090.

Datta, S., Ganguly, D., Roy, D., Bonin, F., Jochim, C. & Mitra, M. (2020). Retrieving potential causes from a query event. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, 1689–1692, New York, NY, USA.

Dawid, A.P. (2010). Beware of the dag! In *Proceedings of Workshop on Causality: Objectives and Assessment at NIPS 2008*, vol. 6 of *Proceedings of Machine Learning Research*, 59–86, PMLR, Whistler, Canada.

Dehghani, M., Zamani, H., Severyn, A., Kamps, J. & Croft, W.B. (2017). Neural ranking models with weak supervision. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, 65–74, Association for Computing Machinery, New York, NY, USA.

Déjean, S., Ionescu, R.T., Mothe, J. & Ullah, M.Z. (2020). Forward and backward feature selection for query performance prediction. In *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, SAC '20, 690–697, Association for Computing Machinery, New York, NY, USA.

Deveaud, R., Mothe, J., Ullah, M.Z. & Nie, J.Y. (2018). Learning to adaptively rank document retrieval system configurations. *ACM Transactions on Information Systems*, **37**.

Devlin, J., Chang, M., Lee, K. & Toutanova, K. (2019). BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, 4171–4186, Association for Computational Linguistics.

Diaz, F. (2007). Performance prediction using spatial autocorrelation. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, 583–590, Association for Computing Machinery, New York, NY, USA.

Do, Q.X., Chan, Y.S. & Roth, D. (2011). Minimally supervised event causality identification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, 294–303, Association for Computational Linguistics, USA.

Faggioli, G., Zendel, O., Culpepper, J.S., Ferro, N. & Scholer, F. (2021). An enhanced evaluation framework for query performance prediction. In *Advances in Information Retrieval*, 115–129, Springer International Publishing, Cham.

Feild, H. & Allan, J. (2013). Task-aware query recommendation. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, 83–92, Association for Computing Machinery, New York, NY, USA.

Ferrari Dacrema, M., Cremonesi, P. & Jannach, D. (2019). Are we really making much progress? a worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems*, RecSys '19, 101–109, Association for Computing Machinery, New York, NY, USA.

Frank Gardner (2020). With UAE deal, Israel opens tentative new chapter with Gulf Arabs. `https://www.bbc.com/news/world-middle-east-53805828`, online; accessed 20 August 2021.

Furnas, G.W., Landauer, T.K., Gomez, L.M. & Dumais, S.T. (1987). The vocabulary problem in human-system communication. *Commun. ACM*, **30**, 964–971.

Ganguly, D., Leveling, J. & Jones, G. (2012). Cross-lingual topical relevance models. In M. Kay & C. Boitet, eds., *Proceedings of COLING 2012*, 927–942, The COLING 2012 Organizing Committee, Mumbai, India.

Ganguly, D., Roy, D., Mitra, M. & Jones, G.J.F. (2015). Word embedding based generalized language model for information retrieval. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*, 795–798, ACM.

Gharbieh, W., Bhavsar, V.C. & Cook, P. (2016). A word embedding approach to identifying verb-noun idiomatic combinations. In *Proceedings of the 12th Workshop on Multiword Expressions, MWE@ACL 2016, Berlin, Germany, August 11, 2016*, Association for Computer Linguistics.

Girju, R. (2003). Automatic detection of causal relations for question answering. In *Proceedings of the ACL 2003 Workshop on Multilingual Summarization and Question Answering - Volume 12*, MultiSumQA '03, 76–83, Association for Computational Linguistics, USA.

Gordon, A.S., Bejan, C.A. & Sagae, K. (2011). Commonsense causal reasoning using millions of personal stories. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*, AAAI'11, 1180–1185, AAAI Press.

Gordon, A.S., Kozareva, Z. & Roemmele, M. (2012). Semeval-2012 task 7: Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, SemEval '12, 394–398, Association for Computational Linguistics, USA.

Granger, C.W.J. (2001). *Investigating Causal Relations by Econometric Models and Cross-Spectral Methods*, 31–47.

Guo, B., Zhang, X., Wang, Z., Jiang, M., Nie, J., Ding, Y., Yue, J. & Wu, Y. (2023). How close is ChatGPT to human experts? Comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*.

Guo, J., Fan, Y., Ai, Q. & Croft, W.B. (2016). A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, 55–64, Association for Computing Machinery, New York, NY, USA.

Gupta, M. & Bendersky, M. (2015). Information retrieval with verbose queries. *Foundations and Trends in Information Retrieval*, **9**, 91–208.

Harradon, M. *et al.* (2018). Causal learning and explanation of deep neural networks via autoencoded activations. *ArXiv*, **abs/1802.00541**.

Hashimoto, C., Torisawa, K., Kloetzer, J. & Oh, J.H. (2015). Generating event causality hypotheses through semantic relations. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, AAAI'15, 2396–2403, AAAI Press.

Hashimoto, C. *et al.* (2012). Excitatory or inhibitory: A new semantic orientation extracts contradiction and causality from the web. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 619–630.

Hashimoto, C. *et al.* (2014). Toward future scenario generation: Extracting event causality exploiting semantic relation, context, and association features. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 987–997.

Hauff, C. (2010). Predicting the effectiveness of queries and retrieval systems. *SIGIR Forum*, **44**, 88.

Hauff, C., Hiemstra, D. & de Jong, F. (2008). A survey of pre-retrieval query performance predictors. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, 1419–1420, Association for Computing Machinery.

He, B. & Ounis, I. (2004). Inferring query performance using pre-retrieval predictors. In *String Processing and Information Retrieval*, 43–54.

He, B. & Ounis, I. (2007). Combining fields for query expansion and adaptive query expansion. *Inf. Process. Manage.*, **43**, 1294–1307.

He, B. & Ounis, I. (2009). Finding good feedback documents. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, 2011–2014, Association for Computing Machinery, New York, NY, USA.

Hiemstra, D. (2001). *Using language models for information retrieval*. Citeseer.

Hochreiter, S. & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, **9**, 1735–1780.

Inui, T. & Okumura, M. (2005). Investigating the characteristics of causal relations in Japanese text. In *Proceedings of Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, 37–44.

Jaleel, N.A., Allan, J., Croft, W.B., Diaz, F., Larkey, L.S., Li, X., Smucker, M.D. & Wade, C. (2004). Umass at TREC 2004: Novelty and HARD. In *Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004*, vol. 500-261 of *NIST Special Publication*, National Institute of Standards and Technology (NIST).

Joshi, A., Tripathi, V., Patel, K., Bhattacharyya, P. & Carman, M.J. (2016). Are word embedding-based features useful for sarcasm detection? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, 1006–1011, Association for Computational Linguistics.

Kaplan, R.M. & Berry-Rogghe, G. (1991). Knowledge-based acquisition of causal relationships in text. *Knowledge Acquisition*, **3**, 317–337.

Khattab, O. & Zaharia, M. (2020). *ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT*, 39–48. Association for Computing Machinery, New York, NY, USA.

Kiciman, E. (2018). Causal inference over longitudinal data to support expectation exploration. In *Proceedings of the 41st International ACM SIGIR Conference on Research Development in Information Retrieval (SIGIR'18)*, 1345–1345.

Kiciman, E. & Thelin, J. (2018). Answering what if, should i, and other expectation exploration queries using causal inference over longitudinal data. In *Proceedings of the Conference on Design of Experimental Search and Information Retrieval Systems (DESIRES)*, 9–15.

Koriat, A., Ma'ayan, H. & Nussinson, R. (2006). The intricate relationships between monitoring and control in metacognition: Lessons for the cause-and-effect relation between subjective experience and behavior. In *Journal of Experimental Psychology: General*, vol. 135(1), 36–69.

Kozareva, Z. (2012). Cause-effect relation learning. In *Workshop Proceedings of TextGraphs-7: Graph-based Methods for Natural Language Processing*, 39–43.

Kuo, M., Barnes, M. & Jordan, C. (2019). Do experiences with nature promote learning? converging evidence of a cause-and-effect relationship. *Frontiers in Psychology*, **10**.

Kurland, O., Shtok, A., Carmel, D. & Hummel, S. (2011). A unified framework for post-retrieval query-performance prediction. In *Proceedings of the Third International Conference on Advances in Information Retrieval Theory*, ICTIR'11, 15–26.

Kurland, O., Raiber, F. & Shtok, A. (2012). Query-performance prediction and cluster ranking: Two sides of the same coin. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, 2459–2462, New York, NY, USA.

Lattimore, F. & Ong, C.S. (2018). A primer on causal analysis. *ArXiv*, **abs/1806.01488**.

Lavrenko, V. & Croft, W.B. (2001). Relevance based language models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, 120–127, Association for Computing Machinery, New York, NY, USA.

Lee, K.S., Croft, W.B. & Allan, J. (2008). A cluster-based resampling method for pseudo-relevance feedback. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, 235–242, Association for Computing Machinery, New York, NY, USA.

Li, C., Sun, Y., He, B., Wang, L., Hui, K., Yates, A., Sun, L. & Xu, J. (2018). NPRF: A neural pseudo relevance feedback framework for ad-hoc information retrieval. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 4482–4491, Association for Computational Linguistics, Brussels, Belgium.

Li, H., Mourad, A., Koopman, B. & Zuccon, G. (2022a). How does feedback signal quality impact effectiveness of pseudo relevance feedback for passage retrieval. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, 2154–2158, Association for Computing Machinery, New York, NY, USA.

Li, H., Wang, S., Zhuang, S., Mourad, A., Ma, X., Lin, J. & Zuccon, G. (2022b). To interpolate or not to interpolate: Prf, dense and sparse retrievers. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, 2495–2500, Association for Computing Machinery, New York, NY, USA.

Li, J., Ren, P., Chen, Z., Ren, Z., Lian, T. & Ma, J. (2017). Neural attentive session-based recommendation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, 1419–1428, Association for Computing Machinery, New York, NY, USA.

Li, P. & Mao, K. (2019). Knowledge-oriented convolutional neural network for causal relation extraction from natural language texts. *Expert Systems with Applications*, **115**, 512–523.

Li, R., Kao, B., Bi, B., Cheng, R. & Lo, E. (2012). Dqr: A probabilistic approach to diversified query recommendation. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, 16–25, Association for Computing Machinery, New York, NY, USA.

Lin, J. (2006). Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, **37**, 145–151.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. & Stoyanov, V. (2019). Roberta: A robustly optimized BERT pretraining approach. *CoRR*, **abs/1907.11692**.

Lv, Y. & Zhai, C. (2009a). Adaptive relevance feedback in information retrieval. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, 255–264, Association for Computing Machinery, New York, NY, USA.

Lv, Y. & Zhai, C. (2009b). A comparative study of methods for estimating query language models with pseudo feedback. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM 2009, Hong Kong, China, November 2-6, 2009*, 1895–1898, ACM.

MacAvaney, S., Yates, A., Cohan, A. & Goharian, N. (2019). Cedr: Contextualized embeddings for document ranking. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Mackie, I., Chatterjee, S. & Dalton, J. (2023). Generative relevance feedback with large language models. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'23, 2026–2031, Association for Computing Machinery.

Mallia, A., Siedlaczek, M., Mackenzie, J. & Suel, T. (2019). PISA: performant indexes and search for academia. In *Proceedings of the Open-Source IR Replicability Challenge co-located with 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, OSIRRC@SIGIR 2019*, 50–56.

Manning, C.D., Raghavan, P. & Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, 3111–3119, Curran Associates Inc., Red Hook, NY, USA.

Miller, G.A. (1995). WordNet: A lexical database for english. *Communications of the ACM*, **38**, 39–41.

Mitra, B., Shokouhi, M., Radlinski, F. & Hofmann, K. (2014). On user interactions with query auto-completion. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, 1055–1058, Association for Computing Machinery, New York, NY, USA.

Mitra, B., Diaz, F. & Craswell, N. (2017). Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, 1291–1299, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE.

Mitra, M., Singhal, A. & Buckley, C. (1998). Improving automatic query expansion. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'98, 206–214, Association for Computing Machinery, New York, NY, USA.

Montazeralghaem, A., Zamani, H. & Allan, J. (2020). A reinforcement learning framework for relevance feedback. In *Proceedings of the 43rd International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '20, 59–68, Association for Computing Machinery, New York, NY, USA.

Narendra, T. *et al.* (2018). Explaining deep learning models using causal inference. *ArXiv*, **abs/1811.04376**.

Naseri, S., Dalton, J., Yates, A. & Allan, J. (2021). CEQE: contextualized embeddings for query expansion. In *Advances in Information Retrieval - 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 - April 1, 2021, Proceedings, Part I*, vol. 12656 of *Lecture Notes in Computer Science*, 467–482, Springer.

Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R. & Deng, L. (2016). MS MARCO: A human generated machine reading comprehension dataset. In *CoCo@NIPS*, vol. 1773 of *CEUR Workshop Proceedings*.

Nogueira, R., Yang, W., Cho, K. & Lin, J.J. (2019a). Multi-Stage Document Ranking with BERT. *ArXiv*, **abs/1910.14424**.

Nogueira, R.F. & Cho, K. (2019). Passage re-ranking with BERT. *CoRR*, **abs/1901.04085**.

Nogueira, R.F., Yang, W., Cho, K. & Lin, J. (2019b). Multi-stage document ranking with BERT. *CoRR*, **abs/1910.14424**.

Ogilvie, P., Voorhees, E. & Callan, J. (2009). On the number of terms used in automatic query expansion. *Information Retrieval*, **12**, 666–679.

Oh, J.H., Torisawa, K., Hashimoto, C., Sano, M., De Saeger, S. & Ohtake, K. (2013). Why-question answering using intra- and inter-sentential causal relations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Paik, J.H. (2015). A probabilistic model for information retrieval based on maximum value distribution. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, 585–594, Association for Computing Machinery, New York, NY, USA.

Palchowdhury., S. *et al.* (2011). Overview of FIRE 2011. In *Multilingual Information Access in South Asian Languages - Second International Workshop, FIRE 2010*, 1–12.

Paul, M.J. (2017). Feature selection as causal inference: Experiments with text classification. In R. Levy & L. Specia, eds., *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), Vancouver, Canada, August 3-4, 2017*, 163–172, Association for Computational Linguistics.

Pearl, J. (1995). Causal diagrams for empirical research. *Biometrika*, **82**, 669–688.

Pearl, J. & Mackenzie, D. (2018). *The Book of Why: The New Science of Cause and Effect*. Basic Books, Inc., 1st edn.

Pearl, J. & Paz, A. (2022). *GRAPHOIDS: Graph-Based Logic for Reasoning about Relevance Relations OrWhen Would x Tell You More about y If You Already Know z?*, 189–200. Association for Computing Machinery, New York, NY, USA, 1st edn.

Pennington, J., Socher, R. & Manning, C.D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, 1532–1543, ACL.

Pérez-Iglesias, J. & Araujo, L. (2010). Standard deviation as a query hardness estimator. In *String Processing and Information Retrieval*, 207–212, Springer Berlin Heidelberg, Berlin, Heidelberg.

Ponte, J.M. & Croft, W.B. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, 275–281, Association for Computing Machinery, New York, NY, USA.

Radinsky, K. & Horvitz, E. (2013). Mining the web to predict future events. In *Proceedings of the 6th ACM International Conference on Web Search and Data Mining*, WSDM '13, 255–264, Association for Computing Machinery, New York, NY, USA.

Radinsky, K., Davidovich, S. & Markovitch, S. (2012). Learning causality for news events prediction. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, 909–918, Association for Computing Machinery, New York, NY, USA.

Raina, R., Madhavan, A. & Ng, A.Y. (2009). Large-scale deep unsupervised learning using graphics processors. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, 873–880, Association for Computing Machinery, New York, NY, USA.

Rha, E.Y., Shi, W. & Belkin, N.J. (2017). An exploration of reasons for query reformulations. In *Diversity of Engagement: Connecting People and Information in the Physical and Virtual Worlds - Proceedings of the 80th ASIS&T Annual Meeting, ASIST 2017*, vol. 54, 337–346, Wiley.

Riaz, M. & Girju, R. (2010). Another look at causality: Discovering scenario-specific contingency relationships with no supervision. In *Proceedings of the 4th IEEE International Conference on Semantic Computing (ICSC 2010), September 22-24, 2010, Carnegie Mellon University, Pittsburgh, PA, USA*, 361–368, IEEE Computer Society.

Riaz, M. & Girju, R. (2014). In-depth exploitation of noun and verb semantics to identify causation in verb-noun pairs. In *Proceedings of the SIGDIAL 2014 Conference, The 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 161–170, Association for Computer Linguistics.

Richard, S. & Peter, S. (2008). *Causal Structure Search: Philosophical Foundations and Future Problems*.

Richardson, T. & Spirtes, P. (2000). Ancestral graph markov models. *Annals of Statistics*, **30**.

Rink, B., Bejan, C.A. & Harabagiu, S.M. (2010). Learning textual graph patterns to detect causal event relations. In *Proceedings of the Twenty-Third International Florida Artificial Intelligence Research Society Conference, May 19-21, 2010, Daytona Beach, Florida, USA*, AAAI Press.

Robertson, S. & Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, **3**, 333–389.

Robertson, S.E. & Walker, S. (1994). Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In B.W. Croft & C.J. van Rijsbergen, eds., *SIGIR '94*, 232–241, Springer London, London.

Robertson, S.E., Walker, S., Hancock-Beaulieu, M., Gatford, M. & Payne, A. (1995). Okapi at TREC-4. In *Proceedings of The Fourth Text REtrieval Conference, TREC 1995, Gaithersburg, Maryland, USA, November 1-3, 1995*, vol. 500-236 of *NIST Special Publication*, National Institute of Standards and Technology (NIST).

Rocchio, J.J. (1971). *Relevance Feedback in Information Retrieval*. Prentice Hall, Englewood, Cliffs, New Jersey.

Rodríguez-Sánchez, A.J., Fallah, M. & Leonardis, A. (2015). Editorial: Hierarchical object representations in the visual cortex and computer vision. *Frontiers in Computational Neuroscience*, **9**, 142.

Roitman, H. (2017). An enhanced approach to query performance prediction using reference lists. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, 869–872, Association for Computing Machinery, New York, NY, USA.

Roitman, H. (2019). Normalized query commitment revisited. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, 1085–1088, Association for Computing Machinery, New York, NY, USA.

Roitman, H. & Kurland, O. (2019). Query performance prediction for pseudo-feedback-based retrieval. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, 1261–1264, Association for Computing Machinery, New York, NY, USA.

Roitman, H., Erera, S. & Weiner, B. (2017). Robust standard deviation estimation for query performance prediction. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '17, 245–248, Association for Computing Machinery, New York, NY, USA.

Roy, D., Ganguly, D., Mitra, M. & Jones, G.J. (2016). Word vector compositionality based relevance feedback using kernel density estimation. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, 1281–1290, Association for Computing Machinery, New York, NY, USA.

Roy, D., Ganguly, D., Mitra, M. & Jones, G.J. (2019). Estimating gaussian mixture models in the local neighbourhood of embedded word vectors for query performance prediction. *Information Processing and Management*, **56**, 1026 – 1045.

Salakhutdinov, R. & Mnih, A. (2008). Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, 880–887, Association for Computing Machinery, New York, NY, USA.

Salehi, B., Cook, P. & Baldwin, T. (2015). A word embedding approach to predicting the compositionality of multiword expressions. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 977–983, Association for Computational Linguistics.

Selvaraju, R.R. *et al.* (2016). Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, **abs/1610.02391**.

Shen, Y., He, X., Gao, J., Deng, L. & Mesnil, G. (2014). A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, CIKM '14, 101–110.

Shtok, A., Kurland, O. & Carmel, D. (2010). Using statistical decision theory and relevance models for query-performance prediction. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, 259–266.

Shtok, A., Kurland, O., Carmel, D., Raiber, F. & Markovits, G. (2012). Predicting query performance by query-drift estimation. *ACM Transactions on Information Systems*, **30**.

Smirnova, E. & Vasile, F. (2017). Contextual sequence modeling for recommendation with recurrent neural networks. In *Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems*, DLRS 2017, 2–9.

Soboroff, I., Huang, S. & Harman, D. (2018). TREC 2018 news track overview. In E.M. Voorhees & A. Ellis, eds., *Proceedings of the Twenty-Seventh Text REtrieval Conference, TREC 2018, Gaithersburg, Maryland, USA, November 14-16, 2018*, vol. 500-331 of *NIST Special Publication*, National Institute of Standards and Technology (NIST).

Sun, Y., Xie, K., Liu, N., Yan, S., Zhang, B. & Chen, Z. (2007). Causal relation of queries from temporal logs. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, 1141–1142, Association for Computing Machinery, New York, NY, USA.

Suppes, P. (1970). *A Probabilistic Theory of Causality*. Amsterdam: North-Holland Pub. Co.

Tanaka, S., Okazaki, N. & Ishizuka, M. (2012). Acquiring and generalizing causal inference rules from deverbal noun constructions. In *COLING 2012, 24th International Conference on Computational Linguistics, Proceedings of the*

Conference: Posters, 8-15 December 2012, Mumbai, India, 1209–1218, Indian Institute of Technology Bombay.

Tao, Y. & Wu, S. (2014). Query performance prediction by considering score magnitude and variance together. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, CIKM '14, 1891–1894, Association for Computing Machinery, New York, NY, USA.

Terra, E.L. & Warren, R. (2005). Poison pills: harmful relevant documents in feedback. In *Proceedings of the 2005 ACM CIKM International Conference on Information and Knowledge Management, Bremen, Germany, October 31 - November 5, 2005*, 319–320, ACM.

Thomas, P., Scholer, F., Bailey, P. & Moffat, A. (2017). Tasks, queries, and rankers in pre-retrieval performance prediction. In *Proceedings of the 22nd Australasian Document Computing Symposium*, ADCS 2017, Association for Computing Machinery.

Tredici, M.D. & Bel, N. (2015). A word-embedding-based sense index for regular polysemy representation. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing, VS@NAACL-HLT 2015*, 70–78, Association for Computational Linguistics.

Voorhees, E.M. & Harman, D. (1999). Overview of the eighth text retrieval conference (TREC-8). In *Proceedings of The Eighth Text REtrieval Conference, TREC 1999, Gaithersburg, Maryland, USA, November 17-19, 1999*, vol. 500-246 of *NIST Special Publication*, National Institute of Standards and Technology (NIST).

Wang, K., Zhang, P. & Su, J. (2020). A text classification method based on the merge-LSTM-CNN model. *Journal of Physics: Conference Series*, **1646**, 012110.

Wang, X., Macdonald, C., Tonellotto, N. & Ounis, I. (2021). Pseudo-relevance feedback for multiple representation dense retrieval. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '21, 297–306, Association for Computing Machinery, New York, NY, USA.

Wang, X., MacDonald, C., Tonellotto, N. & Ounis, I. (2023). Colbert-prf: Semantic pseudo-relevance feedback for dense passage and document retrieval. *ACM Transactions on the Web*, **17**.

Wood-Doughty, Z., Shpitser, I. & Dredze, M. (2018). Challenges of using text classifiers for causal inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, 4586–4598, Association for Computational Linguistics.

Wu, L., Sun, P., Fu, Y., Hong, R., Wang, X. & Wang, M. (2019). A neural influence diffusion model for social recommendation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, 235–244.

Xiong, C., Dai, Z., Callan, J., Liu, Z. & Power, R. (2017). End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, 55–64, Association for Computing Machinery.

Xiong, L., Xiong, C., Li, Y., Tang, K., Liu, J., Bennett, P.N., Ahmed, J. & Overwijk, A. (2020). Approximate nearest neighbor negative contrastive learning for dense text retrieval. *CoRR*, **abs/2007.00808**.

Xu, J. & Croft, W.B. (2000). Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems*, **18**, 79–112.

Yilmaz, Z.A., Wang, S., Yang, W., Zhang, H. & Lin, J. (2019). Applying BERT to document retrieval with birch. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 19–24.

Yom-Tov, E., Fine, S., Carmel, D. & Darlow, A. (2005). Learning to estimate query difficulty: Including applications to missing content detection and distributed information retrieval. In *Proceedings of SIGIR '05*, 512–519, Association for Computing Machinery.

Yu, H., Xiong, C. & Callan, J. (2021). *Improving Query Representations for Dense Retrieval with Pseudo Relevance Feedback*, 3592–3596. Association for Computing Machinery, New York, NY, USA.

Zamani, H. & Croft, W.B. (2016). Estimating embedding vectors for queries. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*, ICTIR '16, 123–132, Association for Computing Machinery, New York, NY, USA.

Zamani, H. & Croft, W.B. (2017). Relevance-based word embedding. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, 505–514, Association for Computing Machinery, New York, NY, USA.

Zamani, H., Dadashkarimi, J., Shakery, A. & Croft, W.B. (2016). Pseudo-relevance feedback based on matrix factorization. In *Proceindgs of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, 1483–1492.

Zamani, H., Croft, W.B. & Culpepper, J.S. (2018a). Neural query performance prediction using weak supervision from multiple signals. In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '18, 105–114, Association for Computing Machinery.

Zamani, H., Mitra, B., Song, X., Craswell, N. & Tiwary, S. (2018b). Neural ranking models with multiple document fields. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*, WSDM'18, 700–708.

Zendel, O., Shtok, A., Raiber, F., Kurland, O. & Culpepper, J.S. (2019). Information needs, queries, and query performance prediction. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, 395–404, New York, NY, USA.

Zhai, C. (2008). Statistical language models for information retrieval a critical review. *Foundations and Trends in Information Retrieval*, **2**, 137–213.

Zhai, C. & Lafferty, J. (2001). A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, 334–342, Association for Computing Machinery, New York, NY, USA.

Zhang, J. (2008). Causal reasoning with ancestral graphs. *Journal of Machine Learning Research*, **9**, 1437–1474.

Zhao, S., Wang, Q., Massung, S., Qin, B., Liu, T., Wang, B. & Zhai, C. (2017). Constructing and embedding abstract event causality networks from text snippets. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, WSDM '17, 335–344, ACM.

Zhao, W.X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., Liu, P., Nie, J.Y. & Wen, J.R. (2023). A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Zhao, Y., Scholer, F. & Tsegay, Y. (2008). Effective pre-retrieval query performance prediction using similarity and variability evidence. In C. Macdonald, I. Ounis, V. Plachouras, I. Ruthven & R.W. White, eds., *Advances in Information Retrieval*, 52–64, Springer Berlin Heidelberg, Berlin, Heidelberg.

Zheng, G. & Callan, J. (2015). Learning to reweight terms with distributed representations. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, 575–584, Association for Computing Machinery, New York, NY, USA.

Zheng, Z., Hui, K., He, B., Han, X., Sun, L. & Yates, A. (2020). BERT-QE: Contextualized Query Expansion for Document Re-ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 4718–4728, Association for Computational Linguistics.

Zhou, B. *et al.* (2015). Learning deep features for discriminative localization. *CoRR*, **abs/1512.04150**.

Zhou, Y. & Croft, W.B. (2006). Ranking robustness: A novel framework to predict query performance. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, CIKM '06, 567–574, Association for Computing Machinery.

Zhou, Y. & Croft, W.B. (2007). Query performance prediction in web search environments. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, 543–550.

Zhuang, S., Li, H. & Zuccon, G. (2022). Implicit feedback for dense passage retrieval: A counterfactual approach. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, 18–28, ACM, New York, NY, USA.