

IMPROVING USER SPARSE QUERY INTERPRETATION  
THROUGH PSEUDO-RELEVANCE RETRIEVAL METHODS

QUANLI PEI

A THESIS SUBMITTED TO THE FACULTY OF GRADUATE STUDIES  
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR THE DEGREE OF

MASTERS OF ARTS

GRADUATE PROGRAM IN INFORMATION SYSTEMS AND TECHNOLOGY  
YORK UNIVERSITY  
TORONTO, ONTARIO

December 2024

© Quanli Pei, 2024

# Abstract

Despite the rapid development of information retrieval technology, understanding sparse user query remains a significant challenge. Users often input short, ambiguous, or context-lacking queries when searching, making it difficult for retrieval systems to capture user intent.

This thesis focuses on this critical issue and proposes three innovative models based on Pseudo-Relevance Feedback: CNRoc, CLRoc, and LLM-PRF, with the aim of enhancing the performance of retrieval systems.

The CNRoc model enriches query expansions by incorporating external conceptual knowledge, enabling it to capture the subtle meanings of query terms and generate more semantically relevant expansion terms. The CLRoc model combines weak and strong relevance signals, utilizing Contrastive Learning to optimize document selection and enhance the alignment between user intent and result documents. The LLM-PRF model integrates Large Language Model to improve the query representation capability of dense retrieval systems, further enhancing the understanding of user intent. Experimental results demonstrate that these models significantly outperform traditional methods in multiple evaluation metrics, providing effective solutions for handling sparse query. Ultimately, this thesis lays the groundwork for future advancements in Information Retrieval, ensuring that users can more effectively retrieve the information they want and make informed decisions.

# Acknowledgements

As I approach the end of my master's journey, I am filled with immense gratitude for all those who have supported and assisted me along the way.

First, I would like to express my gratitude to my supervisor, Professor Jimmy Huang. His unwavering guidance in my academic pursuits and encouragement during challenging times have been invaluable. He not only provided me with full funding for my studies, allowing me to focus entirely on my research but also instilled in me a passion for knowledge and inquiry. His rigorous academic attitude and profound expertise have deeply inspired me, allowing me to grow throughout my research journey.

Secondly, I also want to express my sincere gratitude to Professor Min Pan. His guidance and encouragement during my undergraduate studies laid a solid foundation for my subsequent graduate education. I am grateful for the insights and support he provided on my academic path.

In addition, I want to express my gratitude to all the members of the Information Retrieval & Knowledge Management Research Lab. Thank you for their companionship, discussions and support, which made this journey less lonely.

Lastly, I would like to express my gratitude to my family for their understanding, support and encouragement. They are the driving force behind my pursuit of dreams, and I appreciate their silent support.

This thesis is not only the result of my personal efforts, but also a collective achievement of all those who care for and support me. Thank you all once again!

# Contents

Abstract	ii
Acknowledgments	iii
Table of Contents	vii
List of Tables	ix
List of Figures	x
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Contributions . . . . .	3
1.3 Thesis Organization . . . . .	5
<b>2 Literature Review and Background</b>	<b>7</b>
2.1 Overview of Information Retrieval . . . . .	7
2.1.1 Information Retrieval . . . . .	7
2.1.2 Relevance Feedback . . . . .	11
2.1.3 Pseudo-Relevance Feedback . . . . .	14
2.2 Overview of Research on Pseudo-Relevance Feedback . . . . .	16
2.2.1 Based on Vector Space Model . . . . .	16
2.2.2 Based on Language Model . . . . .	18

2.2.3	Based on Positional Information . . . . .	20
2.2.4	Based on Deep Learning Model . . . . .	22
2.2.5	Based on Generative LLM and Others . . . . .	24
2.3	Chapter Summary . . . . .	27
<b>3</b>	<b>Query Expansion via Semantic Network</b>	<b>29</b>
3.1	Chapter Introduction . . . . .	29
3.2	Semantic Network . . . . .	30
3.3	CNRoc: Query Expansion via Semantic Network . . . . .	33
3.3.1	Semantic Related Term Extraction . . . . .	33
3.3.2	Query Expansion Term Generation . . . . .	35
3.3.3	The Adaptive PRF Model . . . . .	36
3.4	Experimental Setup . . . . .	37
3.4.1	Experimental Datasets and Analytical Metrics . . . . .	37
3.4.2	Baseline Model . . . . .	40
3.4.3	Hyperparameter Settings . . . . .	41
3.5	Experimental Results and Analysis . . . . .	42
3.5.1	Validation Against Baseline and Strong Baseline Models . . . . .	42
3.5.2	Validation Against the SOTA PRF Models . . . . .	44
3.5.3	Validation Against Neural IR models . . . . .	47
3.5.4	Hyperparameter Impact Analysis . . . . .	50
3.5.5	Case Study . . . . .	52
3.6	Chapter Summary . . . . .	53
<b>4</b>	<b>Document Matching Optimization via Contrastive Learning</b>	<b>55</b>
4.1	Chapter Introduction . . . . .	55
4.2	Contrastive Learning . . . . .	56
4.3	CLRoc: Document Matching Optimization via Contrastive Learning . . . . .	58

4.3.1	Selection of Query Expansion Terms . . . . .	58
4.3.2	Adapting the Rocchio Model . . . . .	62
4.4	Experimental Results and Analysis . . . . .	63
4.4.1	Experimental Datasets and Analytical Metrics . . . . .	63
4.4.2	Baseline Model . . . . .	64
4.4.3	Hyperparameter Settings . . . . .	65
4.5	Experimental Setup . . . . .	66
4.5.1	Validation Against Baseline and Advanced Models . . . . .	66
4.5.2	Validation Against the SOTA PRF Models . . . . .	69
4.5.3	Hyperparameter Impact Analysis . . . . .	72
4.5.4	Case study . . . . .	74
4.6	Chapter Summary . . . . .	75
<b>5</b>	<b>Optimizing Dense Retrieval via Large Language Model</b>	<b>77</b>
5.1	Chapter Introduction . . . . .	77
5.2	Dense Retrieval . . . . .	78
5.3	Large Language Model . . . . .	81
5.4	LLM-PRF: Optimizing Dense Retrieval via Large Language Model . . . . .	83
5.4.1	Generative External Knowledge Acquisition . . . . .	83
5.4.2	Pseudo-Relevance Feedback Injection . . . . .	84
5.4.3	Dense Retrieval Integration . . . . .	85
5.5	Experimental Setup . . . . .	86
5.5.1	Experimental Datasets and Analytical Metrics . . . . .	86
5.5.2	Baseline Model . . . . .	87
5.5.3	Hyperparameter . . . . .	87
5.6	Experimental Results and Analysis . . . . .	87
5.6.1	Validation Against with Baseline Models . . . . .	87
5.6.2	Validation Against Strong Models . . . . .	89

5.7 Chapter Summary . . . . .	91
<b>6 Conclusion and Future Work</b>	<b>92</b>
6.1 Summary of Contributions . . . . .	92
6.2 Future Work . . . . .	93
<b>A Published Papers</b>	<b>116</b>

# List of Tables

3.1	Validation Datasets for the CNRoc Model . . . . .	37
3.2	Evaluation of MAP Metrics: CNRoc Model vs. Baseline Models . . . . .	45
3.3	Evaluation of P@10 Metrics: CNRoc Model vs. Baseline Models . . . . .	46
3.4	Evaluation of NDCG and MRR Metrics: CNRoc Model vs. Baseline Models	47
3.5	Evaluation of MAP Metrics: CNRoc Model vs. SOTA Models . . . . .	48
3.6	Evaluation of MAP Metrics: CNRoc Model vs. Neural IR Models . . . . .	49
3.7	Evaluation of P@10 Metrics: CNRoc Model vs. Neural IR Models . . . . .	49
3.8	Evaluation of NDCG Metrics: CNRoc Model vs. Neural IR Models . . . . .	49
3.9	Evaluation of MRR Metrics: CNRoc Model vs. Neural IR Models . . . . .	50
3.10	Case Study of CNRoc . . . . .	52
4.1	Validation Datasets for the CLRoc Model . . . . .	64
4.2	Evaluation of MAP Metrics: CLRoc Model vs. Baseline and Strong Baseline Models . . . . .	67
4.3	Evaluation of P@10 Metrics: CLRoc Model vs. Baseline and Strong Baseline Models . . . . .	68
4.4	Evaluation of NDCG and MRR Metrics: CLRoc Model vs. Baseline and Strong Baseline Models . . . . .	69
4.5	Evaluation of MAP Metrics: CLRoc Model vs. SOTA PRF Models . . . . .	70
4.6	Case Study of CLRoc . . . . .	73

5.1	Validation Datasets for the LLM-PRF . . . . .	86
5.2	Evaluation of MAP, NGCD@10 and R@1000 Metrics: LLM-PRF Model vs. Baseline Models . . . . .	88
5.3	Evaluation of MAP, NGCD@10 and R@1000 Metrics: LLM-PRF Model vs. Strong Dense Retrieval Models . . . . .	90

# List of Figures

2.1	Flowchart of Information Retrieval . . . . .	9
2.2	Flowchart of Relevance Feedback . . . . .	12
2.3	Flowchart of Pseudo-Relevance Feedback . . . . .	15
3.1	ConceptNet . . . . .	32
3.2	Flowchart of CNRoc . . . . .	34
3.3	CNRoc’s experimental results of different pseudo-relevance documents . . . . .	42
3.4	Sensitivity of CNRoc . . . . .	51
4.1	SimCLR Data Augmentation . . . . .	57
4.2	SimCSE Framework . . . . .	60
4.3	Framework for obtaining the strong relevance signal weight based on Contrast Learning . . . . .	61
4.4	Sensitivity of CLRoc . . . . .	74
5.1	Dense Retrieval Dual-Encoder Architecture . . . . .	79
5.2	Some LLM Applications . . . . .	81
5.3	Flowchart of Generative Knowledge Acquisition . . . . .	84

# Chapter 1

## Introduction

### 1.1 Motivation

In the digital age of today, the Internet has emerged as an extensive repository of information that fundamentally reshapes how we gather and use knowledge in various sectors, particularly in news and e-Commerce. The rapid growth of data has transformed the Internet into an essential resource, profoundly influencing our learning habits and decision-making processes. Beyond serving as a medium for communication and entertainment, the Internet has become a vital source of information that shapes our daily lives.

In the realm of news, individuals can swiftly access a wealth of information on current events, emerging trends, and in-depth analyses from a myriad of sources. Major news organizations, independent journalists, and citizen reporters contribute to a diverse media landscape that caters to different perspectives and interests. This instant access not only enhances public awareness but also empowers users to form educated opinions on pressing issues ranging from politics to social justice. Real-time updates provided by news platforms enable citizens to engage with ongoing stories as they unfold, fostering a more informed electorate. Additionally, social media platforms have played a significant role in democratizing news dissemination, allowing users to share stories and perspectives that may not be

covered by traditional media outlets. However, this abundance of information also presents challenges, such as the spread of misinformation and the difficulty in discerning credible sources, leading to concerns about information overload and echo chambers.

Similarly, in the e-Commerce sector, consumers can easily find detailed product information, compare prices, read user reviews, and access ratings, all of which facilitate more informed purchasing decisions. Online shopping platforms have revolutionized the retail experience, allowing consumers to browse a wide range of products from the comfort of their homes. Features such as personalized recommendations, AI-driven chatbots for customer service, and streamlined payment processes enhance the shopping journey, making it more convenient and customized to individual preferences. For example, platforms such as Amazon and Alibaba employ sophisticated algorithms to analyze user behaviour, offering tailored product suggestions that enhance user satisfaction. This shift toward digital retail not only simplifies the purchasing process, but also sets new consumer expectations regarding accessibility, service quality, and delivery speed.

However, despite these advancements, users often face challenges when searching for specific information, particularly when queries are sparse or vague. For example, a user searching for “Best laptops” without additional context, such as intended use (gaming, business, or education), may receive overwhelming results that do not align with their specific needs. This lack of precision can lead to frustration and wasted time, underscoring the need for improved information retrieval methods that can better understand and interpret user intentions.

To address these challenges, Pseudo-Relevance Feedback (PRF) [1] can provide valuable assistance. PRF improves the initial retrieval results by utilizing relevant documents from the first search round to refine the query. This process incorporates additional contextual information, enabling more precise retrieval of relevant data. By employing PRF, we can help users navigate the vast amount of information available online, ultimately improving their search experience. For instance, When a user enters a sparse query, PRF analyzes

the top retrieved documents to identify relevant common themes or terms. This allows the system to enhance the user’s query in accordance with this contextual understanding.

This thesis explores methods to enhance understanding of user sparse queries. We will study advanced retrieval techniques aimed at improving information retrieval systems in both news and e-Commerce sectors. Our objective is to enrich the search experience, facilitating users in finding the information they seek, making informed choices, and engaging more effectively with the digital world.

## 1.2 Contributions

This thesis makes significant contributions by proposing three novel methods to improve information retrieval system efficiency. These techniques pay special attention to Pseudo-Relevance Feedback [2], Dense Retrieval [3], and Query Expansion [4]. Contributions specifically include:

1. CNRoc: Novel Query Expansion Framework via ConceptNet

- We introduce CNRoc, a semantic-based Pseudo-Relevance Feedback model that leverages ConceptNet [5] to enhance query expansion. By incorporating semantic information into the PRF process, Our model greatly enhances the choice of expansion terms, resulting in more accurate retrieval outcomes.
- CNRoc addresses the limitations of traditional Query Expansion methods, capturing the nuanced meanings of query terms through a semantic network. This results in expansion candidate terms that better align with user intent.
- Through rigorous evaluation of standard TREC datasets, CNRoc demonstrates substantial improvements over baseline models and SOTA methods across multiple metrics, including MAP, MRR and NDCG, showcasing its effectiveness in Information Retrieval.

## 2. CLRoc: Document Matching Optimization via Contrastive Learning [6]

- We introduce CLRoc, an innovative probabilistic framework that integrates weak and strong relevance signals to enhance document selection for query expansion. This dual-signal approach enhances the alignment between user query intent and retrieved documents, improving the relevance of retrieval results.
- An innovative linear fusion method is introduced to balance the weights of weak and strong signals, thereby enhancing the quality of query representation and retrieval performance.
- Extensive experiments across six TREC datasets validate that CLRoc significantly outperforms various baseline models in retrieval performance, highlighting its potential application in information retrieval systems.

## 3. LLM-PRF: Optimizing Dense Retrieval via Large Language Model

- We propose a method that integrates the Large Language Model (LLM) [7] with Pseudo-Relevance Feedback to enhance dense retrieval systems. By utilizing external knowledge generated by LLM during the query encoding phase, we enrich query representation with greater semantic depth.
- Our framework employs a Chain of Thought (CoT) [8] reasoning process to acquire contextually relevant insights, thereby improving the retrieval system’s understanding of user intent.
- Comprehensive evaluations on TREC datasets show that the LLM-PRF method outperforms traditional models and consistently exceeds state-of-the-art Dense Retrieval models across multiple metrics, demonstrating its practical advantages.

To conclude, this thesis makes contributions that go beyond introducing novel theoretical frameworks and methodologies; they are substantiated by empirical research validating the effectiveness of these methods. These findings provide new directions for the advancement of

information retrieval technologies, particularly to enhance semantic understanding and user interaction. By integrating various advanced methods, we demonstrate how to significantly improve user search experiences, consequently, information retrieval is being advanced.

## 1.3 Thesis Organization

The structure of this thesis is as follows.

**Chapter 2: Literature Review Background** This chapter thoroughly reviews the current literature on Information Retrieval, emphasizing Pseudo-Relevance Feedback. It identifies key gaps in current research and establishes the foundation for the approaches proposed in this thesis.

**Chapter 3: Query Expansion via Semantic Network** This chapter outlines our proposed CNRoc methodology in the thesis, focusing on the integration of ConceptNet for semantic query expansion. It outlines the datasets, experimental setup, algorithm, and evaluation metrics employed to assess CNRoc’s performance. Lastly, we assess and analyze the model’s effectiveness and discuss its contributions.

**Chapter 4: Document Matching Optimization via Contrastive Learning** The chapter presents the CLRoc framework, a probabilistic approach for document matching optimization based on Contrastive Learning. It details the algorithms, experimental setup, datasets, and evaluation metrics used to assess CLRoc’s performance. Finally, we evaluate and analyze the model’s effectiveness, discussing and summarizing the contributions of this research.

**Chapter 5: Optimizing Dense Retrieval via Large Language Model** This chapter outlines our method LLM-PRF for integrating Large Language Models into the dense retrieval framework. It outlines the experimental setup, the algorithms utilized, and the datasets and metrics applied to evaluate the performance of LLM-PRF. Finally, we evaluate and analyze the model’s effectiveness, discussing and summarizing the contributions of this

research to the field.

Chapter 6: **Conclusion and Future Work** This concluding chapter wraps up the master's thesis, placing the research within a wider context. It further highlights possible opportunities and avenues for future study.

# Chapter 2

## Literature Review and Background

Driven by growing user demands and rapid advancements in technology, Information Retrieval (IR) [2] has seen remarkable progress over the past half-century. Early IR models primarily relied on statistical methods for basic retrieval tasks. However, with continuous technological improvements, more sophisticated techniques have emerged such as Relevance Feedback (RF) [9] and Pseudo-Relevance Feedback (PRF). These methods have demonstrated considerable success in practical applications.

An introduction of IR, RF, and the evolution of PRF will be given at the beginning of this chapter. It will categorize and summarize the current research status of PRF techniques. Finally, the chapter will discuss the existing challenges in PRF research and explore potential work in this area.

### 2.1 Overview of Information Retrieval

#### 2.1.1 Information Retrieval

As information technologies continue to advance, more and more of our social interactions are taking place online. This has led to a huge increase in the amount of data being generated through blogs, microblogs, and other online channels. While this data is incredibly valuable,

it also presents significant challenges for information processing systems. Research shows that a large portion of this data is stored in text format, such as news articles, microblog comments, e-books, and other written resources. As a result, users often struggle to find the information they need within such vast amounts of text, leading to what's known as the "information overload" problem.

In response to this challenge, IR technologies have emerged as an effective solution. Information Retrieval encompasses the organized methods and processes used to represent, store, and retrieve pertinent information based on user queries. Meeting the user's information needs by locating pertinent documents in a large collection is the primary objective of information retrieval, or IR. In order to achieve this, the user often inputs a query into a retrieval system, which then uses sophisticated algorithms to retrieve documents considered most pertinent to the user's inquiry based on assessed relevance scores.

From a technical perspective, IR is characterized by the process of matching a query to documents stored in a database, determining their relevance through algorithmic means, and ranking the documents accordingly. Higher relevance scores indicate a stronger alignment with the user's search intent because they are displayed closer to the top of the retrieval results. The primary areas of focus within IR research include the development of enhanced techniques for the representation, storage, and efficient retrieval of textual information, which continue to be of central importance in addressing the ever-growing complexity of large-scale information set.

The earliest research related to IR originated in the field of library science. In 1945, Bush and colleagues, in their influential paper "As We May Think" [10], introduced the idea of using computers for rapid full-text retrieval, envisioning a future where people could quickly access vast collections of books. In 1957, Luhn et al. [11] proposed representing documents using indexes, with index units made up of terms. They proposed using the overlap between query and document terms to determine the relevance between a user's query and a document.

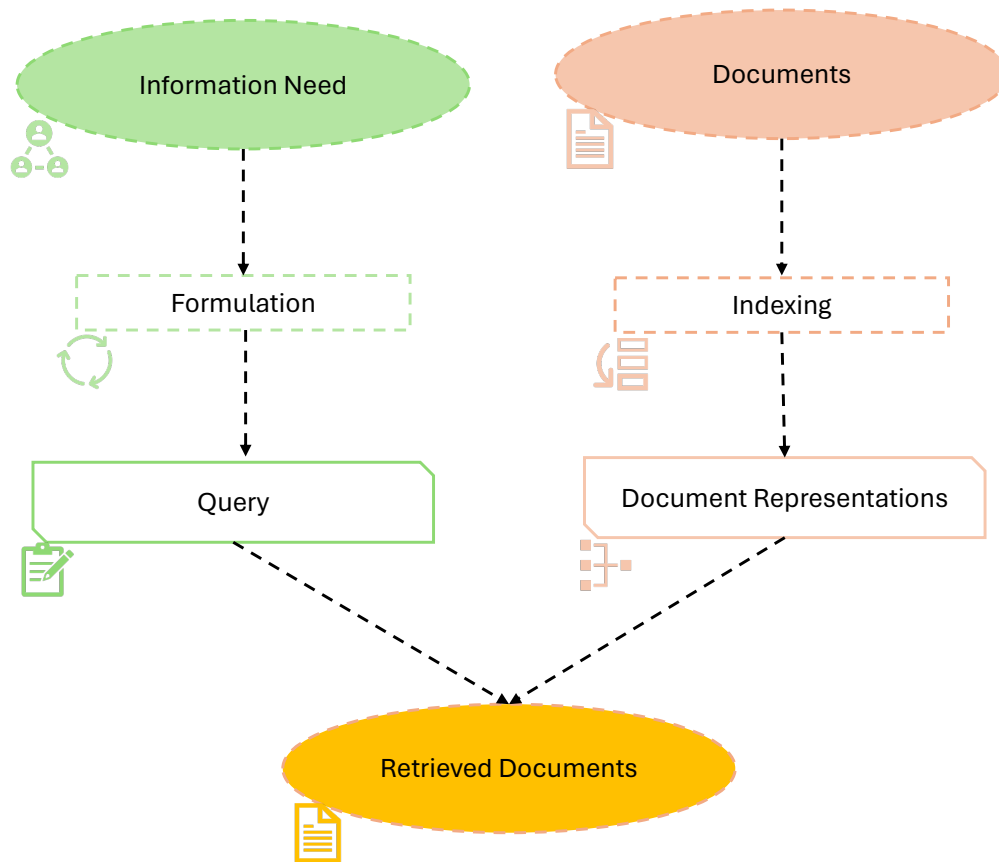


Figure 2.1: Flowchart of Information Retrieval

At its core, Information Retrieval is about efficiently identifying documents from large datasets that align with a user’s information needs. The basic process of Information Retrieval is illustrated in Figure 2.1.

In Figure 2.1, it is important to highlight that the research centers on illustrating the relationship between user information needs and document collections, which directly influences the choice and advancement of information retrieval models.

Over the past half-century, Numerous efficient information retrieval models have been suggested and progressively implemented in search systems, including the Boolean Model

[2], Vector Space Model [12], Probabilistic Model [13], Language Model [14], and Ranking Model [15].

In 1960, Maron and Kuhns proposed the Probabilistic Indexing Model [13], which treats document relevance as a probabilistic event, calculating the proportion of query terms that show up in both relevant as well as irrelevant documents.

In 1972, Sparck Jones introduced the concept of Inverse Document Frequency (IDF) [16], asserting that the importance of a query term decreases as its frequency in the corpus increases.

In 1973, Lancaster and others [2] proposed the Boolean Model, based on set theory, which supports Boolean queries where users can construct search expressions using logical operators such as AND, OR, and NOT.

In 1975, Salton, a founding figure of modern information retrieval, applied the Vector Space Model in the SMART retrieval system [12], where he presented the use of term frequency-inverse document frequency (TF-IDF) weighting to score documents.

In 1976, Robertson and Sparck Jones developed the RSJ framework for information retrieval [17], based on Probabilistic Models, establishing the foundation of the Binary Independence Model (BIM).

In 1994, Robertson and Walker enhanced the BIM model by introducing term weighting in both queries and documents, leading to the creation of the BM25 model [18]. The Okapi BM25 model is currently commonly utilised for web ranking in commercial search engines [19–25].

In 1998, the probabilistic model evolved further when Croft and others proposed the Language Model [14]. This model generates a unique language model for every document, evaluating how likely it is that a document can generate a user’s query and ranking the results according to these probabilities.

In 2003, Learning-to-Rank (L2R) methods began to emerge, with Joachims and Thorsten introducing the RankSVM model [15], which is a pair-wise method for solving the Learning-

to-Rank problem using the Support Vector Machine (SVM) model [26]. In RankSVM, the model uses the predicted values of the current samples as the basis for ranking.

That same year, Freund and colleagues proposed RankBoost [27], which cleverly transforms the ranking problem into a classification problem by constructing a target classifier that establishes relative ordering between pairs of objects.

From 2004 onwards, Information Retrieval theory has seen rapid development, and many representative models have been proposed. These include retrieval models based on Markov Chains [28], Latent Topic Models [25, 29], and Cross-Term Models [30, 31]. Additionally, there are many other retrieval models proposed by prominent scholars [32–39] that are not listed here individually.

These classic models have laid a solid foundation for the advancement of modern information retrieval techniques and provide profound guidance for future research in the field. However, in practice, users often face challenges due to unfamiliarity with retrieval environments or uncertainty regarding the exact information they seek. As a result, the queries they submit are often too short, potentially missing critical terms, which negatively impacts the efficiency of information retrieval systems [40]. To tackle this challenge, information retrieval researchers have suggested several approaches to rebuild the original query, striving to align more closely with users’ actual needs. Among these methods, Relevance Feedback has proven to be particularly effective [41].

### **2.1.2 Relevance Feedback**

In both research and practical use, information retrieval systems often have shortcomings, especially in terms of accuracy and recall. Even the most sophisticated retrieval systems can identify only a portion of the documents relevant to a user’s query, with many significant documents remaining unseen. A primary factor contributing to this low retrieval accuracy is the restricted number of keywords that convey user needs and the various forms these keywords may take. For instance, when a user enters the query term “microphone,” if the

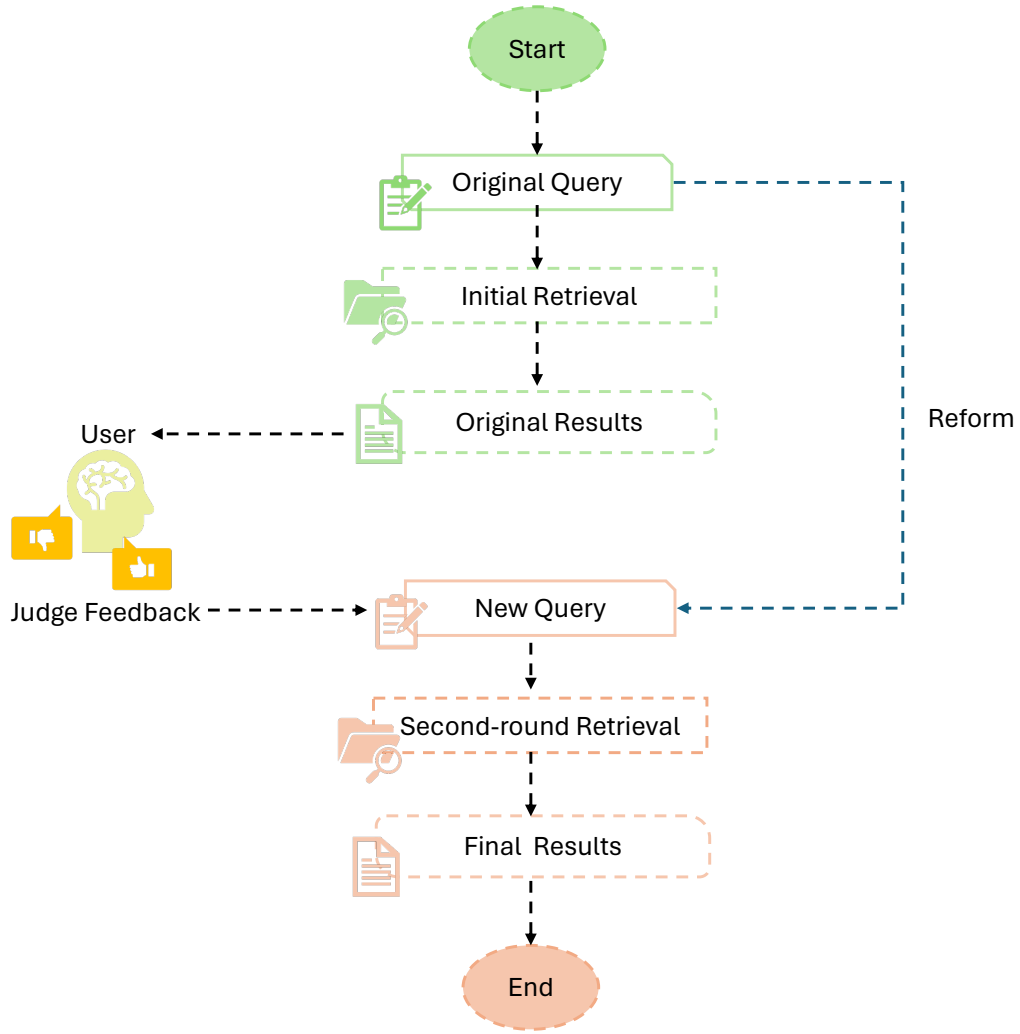


Figure 2.2: Flowchart of Relevance Feedback

retrieval system does not perform query optimization beforehand, documents containing synonyms or related terms, such as “mic” may be overlooked due to the traditional reliance on exact keyword matching. To address this issue, optimizing query terms to improve recall and reduce the omission of relevant documents is an effective solution [21–25].

Current methods for optimizing queries can generally be divided into two categories:

**Global Methods:** These techniques enhance and reorganize the query independently of the original query or the documents obtained in the initial round. The expanded query terms are aligned with semantically related ones terms [23]. Synonym-based query expansion

and spelling check methods fall under this category.

**Local Methods:** By analysing the top-N documents obtained in the initial round, these methods expand the initial query terms. RF is a commonly used QE method based on local optimization [39, 42–44], and numerous studies have demonstrated its effectiveness in IR [9].

Since then, based on classic retrieval models, numerous RF methods for QE have been proposed [43], and these have been widely applied in practice. RF can effectively mitigate the limitations of initial queries in expressing users’ information needs by providing feedback on the relevance of the initial retrieval results, thereby improving the overall retrieval performance.

A widely used query expansion method employed in domains like Information Science and Information Retrieval is called RF. Sometimes referred to as query reformulation, its primary concept is to optimize the initial query through user interaction during the retrieval process [45].

The following is the specific process of RF: First, a query containing terms or representations that reflect the user’s search intent is submitted by the users. The retrieval system performs an initial search using various models and returns a result list of documents. Next, the user evaluates the initial set of documents, marking which documents are relevant (matching their search intent) and which are irrelevant [39]. If the user explicitly marks the results, this is referred to as explicit feedback, whereas if the system analyzes the results automatically, it is called implicit feedback. The system then uses a strategy (such as keyword weighting or query expansion) to modify the initial query in response to user feedback, making it a better representation of the user’s information needs. The system performs a second retrieval with the newly revised query, which yields updated results. This feedback loop can be repeated several times until the system retrieves documents that align closely with the user’s search intent. Clearly, the success of RF hinges on the system’s ability to reformulate the query effectively based on user input. Consequently, the “strategy selection” for query reformulation becomes a crucial area of research aimed at enhancing the

performance of retrieval systems. The specific process is illustrated in Figure 2.2.

RF has been proven to perform effectively to enhance retrieval results. It requires user interaction during the retrieval process to gather feedback, which may involve evaluating and marking the retrieved results, analyzing user browsing behavior and preferences, or statistically analyzing terms and frequencies in the clicked documents. However, most users prefer an automatic retrieval process without human involvement, and they are often reluctant to have their private information collected by the system. Additionally, when user interaction is unavailable, feedback information cannot be obtained. In such cases, Pseudo-Relevance Feedback has emerged as a solution.

### **2.1.3 Pseudo-Relevance Feedback**

PRF offers an automated approach for local analysis and ranks among the most popular query expansion methods in RF. PRF automates the user interaction portion of relevance feedback, allowing retrieval performance improvements without requiring additional user input [43].

Typically, the process of PRF is as follows: Initially, a search is conducted with the original query, yielding the most relevant documents that create an initial set; this is predicated on the premise that the top-N rated documents are related to the query. The query is then expanded and a new query is generated using a weighting or measurement approach. Finally, the system performs a second retrieval on the initial set using a specific retrieval strategy, returning the most relevant documents or document set. Figure 2.3 provides a thorough illustration of the procedure.

While the initial search results in a PRF process may not fully meet user needs, most documents generally reflect the user's query intent. Consequently, they help improve the quality of the query expansion terms. Research on PRF can be traced back to the 1960s and 1970s. In 1971, Rocchio proposed a document-based query expansion method in Salton's SMART retrieval system. This method involved adding high-frequency terms from the top-N

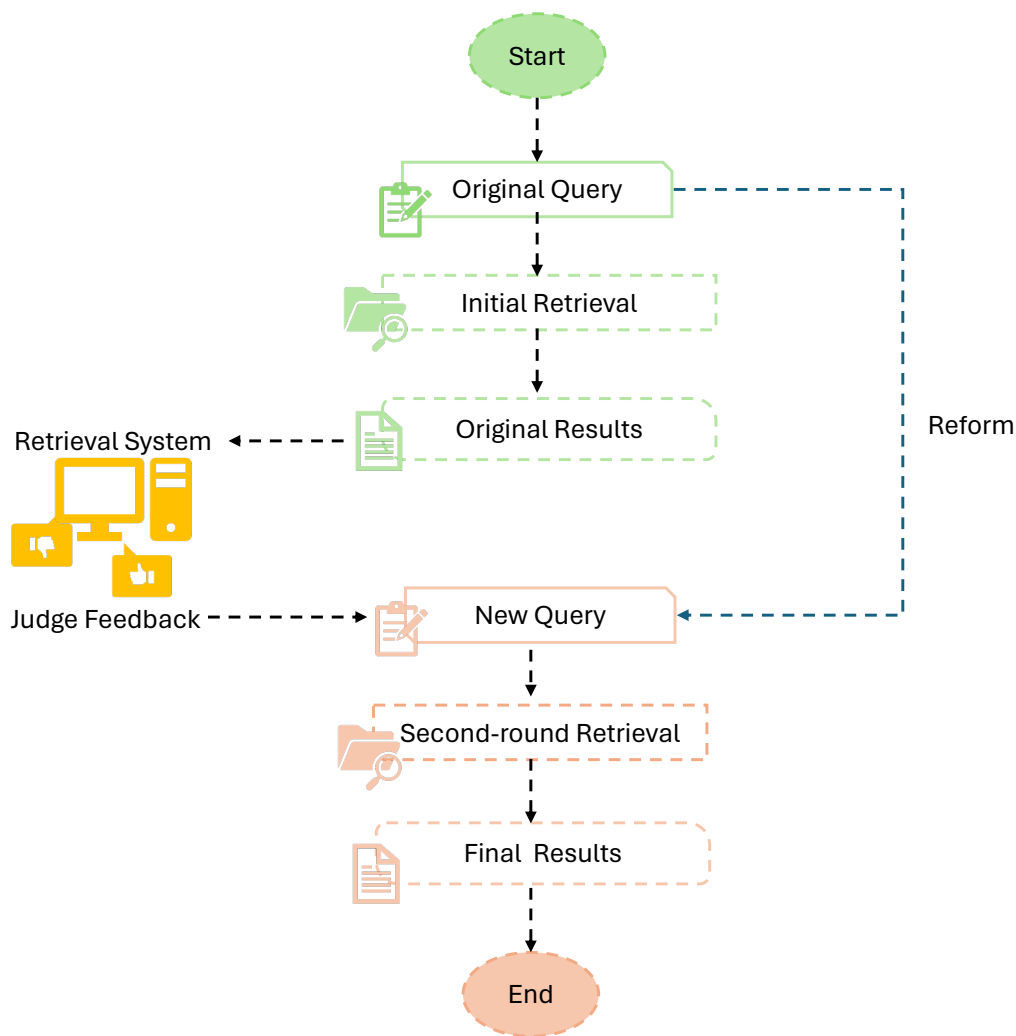


Figure 2.3: Flowchart of Pseudo-Relevance Feedback

retrieved documents to the initial query and redefining query term weights within the Vector Space Model to optimize the query [46], laying the foundation for research on PRF.

Evidence from practical applications demonstrates that PRF enhances the performance of multiple retrieval models. [46–50]. Nonetheless, the method is extremely dependent on the top-N documents’ quality. The final retrieval results may also suffer if the original feedback documents are of poor quality. Many studies have been conducted in the last few decades with the goal of improving retrieval system performance. The function of relevant and irrelevant documents or reducing the weight of irrelevant terms has been the subject of one

line of research. [51–54]. For example, Raman and colleagues [51] extracted better expansion terms from the top-N documents and demonstrated that distinguishing between useful and non-useful expansion terms can significantly improve retrieval performance. Similarly, other PRF methods have focused on improving the QE process [48, 49].

## **2.2 Overview of Research on Pseudo-Relevance Feedback**

PRF serves as a feedback mechanism that eliminates the need for direct user interaction. The top-N ranked documents from the initial retrieval are automatically considered by the system as the most relevant to the user’s needs because of to this method. A second round of retrieval is then performed using these documents after they have been optimised and more expansion terms have been added to the query. Given the characteristics of the retrieval process, the majority of documents in the initial feedback set are typically relevant to the user’s query, making the selection of “query candidate terms” for the second retrieval crucial in improving the quality of the search criteria [9]. As a result, many scholars, both domestically and internationally, have devoted significant research efforts to the study of PRF [1, 39, 51, 55–62], with Text-Based PRF being one of the primary research directions. The following sections will provide a detailed introduction to existing PRF techniques from various perspectives. These include PRF methods based on the Vector Space Model, Language Model, Positional Information, Neural Networks, and other approaches.

### **2.2.1 Based on Vector Space Model**

Research on Natural Language Processing makes extensive use of the VSM [12], especially for tasks like information retrieval and text classification. VSM simplifies text content into vectors composed of feature terms, and the semantic similarity between texts is represented by calculating the cosine distance (Cosine value) between two vectors. Each query document

in the collection and the user’s query are treated as vectors composed of terms by PRF based on VSM during retrieval. By determining the cosine similarity between the query and document vectors, it ranks feedback documents and assesses the relevance of them. Finally, it refines the query by reweighting the feature terms of documents deemed relevant, optimizing the retrieval process.

In the VSM, Relevance Feedback theory assumes that the weight vectors of feature terms from relevant documents are similar, whereas the weight vectors of irrelevant documents differ from those of relevant ones. The fundamental idea is to reconstruct the query so that the results move closer to the feature term weight vector space of the relevant documents [26].

In 1960s, Salton and colleagues first introduced the VSM in the SMART retrieval system [12]. This model represented queries and documents as term vectors, and their similarity was ranked based on the cosine similarity between these vectors [18]. In order to determine document scores, the model also added the term frequency-inverse document frequency (TF-IDF) weighting. These phrases were treated as a collection of orthogonal vectors in Salton’s vector space model in the SMART system. A separate corrective procedure was often needed to account for the relationships between terms in this model.

In 1971, Rocchio introduced a Text-based QE method in Salton’s SMART retrieval system. The method added high-frequency terms from the top-N retrieved documents to the initial query and redefined the query term weights, thereby optimizing the query [46].

The Rocchio algorithm relies on the principle that an ideal query maximizes similarity with relevant documents and reduces similarity with irrelevant ones. As shown in Equation 2.1. Given a specific query, along with some relevant documents  $D_r$  and irrelevant documents  $D_n$ , we can derive a fully expanded query  $Q'$ :

$$Q' = \alpha \times Q_0 + \beta \times \frac{1}{|D_r|} \sum_{d \in D_r} d - \gamma \times \frac{1}{|D_n|} \sum_{d \in D_n} d \quad (2.1)$$

where  $Q_0$  is the initial query,  $d_r$  are the relevant documents,  $d_n$  are the irrelevant documents, and  $\alpha$  and  $\beta$  are the weights for relevant and irrelevant documents, respectively.

From Equation 2.1, we can deduce Rocchio’s approach to modifying the original query: it involves adding high-frequency terms from relevant documents to the query while removing terms that appear in irrelevant documents. In practice, users tend to focus more on documents relevant to the query topic in the initial search results and pay little attention to irrelevant documents (e.g., no explicit set of irrelevant documents is provided). As a result, many Pseudo-Relevance Feedback models do not consider the set of irrelevant documents, leading to various modified versions of the model.

The Rocchio algorithm is a fundamental RF method that integrates relevance feedback within the vector space model, establishing a basis for future studies on PRF.

In 1985, Wong and colleagues proposed a systematic approach, known as the Generalized Vector Space Model (GVSM) [63], to directly compute term correlations from automatic indexing schemes. Addressing the issue that Salton’s model in the SMART system did not account for term correlations [12], they introduced a corrective procedure to consider these relationships. Their experiments demonstrated how this correlation could be incorporated into existing vector-based information retrieval systems with minimal modifications. According to the experimental findings, this approach showed promising improvements in retrieval performance.

### 2.2.2 Based on Language Model

LMs differ from Vector Space Models in their approach to calculating the probability of generating each document for a given query. The theory is that if a query is likely to be generated by the language model that corresponds to a document, then the terms in the query should be used frequently in the content. As a result, the document should closely match the query. Based on the LM approach, ranks documents based on the posterior probability  $p(d|q)$ , as shown in Equation 2.2.

$$p(d|q) = \frac{p(q|d) \cdot p(d)}{\sum_{\bar{d} \in D} p(q|\bar{d}) \cdot p(\bar{d})} \propto p(q|d) \cdot p(d) \quad (2.2)$$

Assuming  $p(d)$  is uniform, the above process simplifies to Equation 2.3.

$$P(d|q) = \prod_{t_q \in q} \left( \lambda \frac{tf(t_q|d)}{|d|} + (1 - \lambda) \frac{\sum_{d \in D} tf(t_q, \bar{d})}{\sum_{\bar{d} \in D} |\bar{d}|} \right) \quad (2.3)$$

Here,  $p(q|d)$  represents the probability that a term randomly drawn from document  $d$  generates query  $q$ . To achieve smoothing, the term frequencies are derived both from the current document  $d$  and from the entire document collection  $D$ . These two events are considered independent, and their probabilities are weighted by  $\lambda$  and  $(1 - \lambda)$ . Zhai and Lafferty in 2001 proposed using the Dirichlet Language Model to rank documents according to their relevance to the query. The goal is to better calculate the weighted values of query terms by using probabilities in the form of logarithmic values, as shown in Equation 2.4.

$$w(q_i, d) = \log P(q_i|d) = \log \left( \frac{dl}{dl + \mu} p_{ml}(q_i|d) + \frac{\mu}{dl + \mu} p_{ml}(q_i|c) \right) \quad (2.4)$$

Where  $c$  represents the document collection,  $d$  is the current document,  $dl$  is the document length, and  $\mu$  is the Dirichlet smoothing parameter.  $p_m(q_i|d)$  and  $p_m(q_i|c)$  represent the probabilities under different models.

The rapid development of LM has facilitated the growth of probabilistic models.

In 2001, Zhai and Lafferty [64] proposed two types of models based on LM. One was based on the likelihood of generating feedback documents, while the other minimized the Kullback-Leibler divergence (KL divergence) between the queries. According to experimental findings, both methods effectively enhanced the quality of query expansion terms.

In 2009, Lv and Zhai [65] compared three advanced methods based on LM in five key benchmark domains. They investigated two different corresponding models (RM3 and RM4) [64], two mixed models (SMM and RMM) [66], and DMM [65]. Their comparison revealed that the SMM model outperformed RM3, and RM3's feedback weighting mechanism improved retrieval stability. RM3 became a reliable method for various retrieval tasks, with its stable performance providing a strong baseline for future experimentation.

While smoothing methods for LM have been shown to be necessary, Hazimeh et al. [57] conducted a detailed analysis in 2015, illustrating the drawbacks of using LM smoothing techniques. They showed that smoothing caused frequently occurring terms to dominate, preventing the effective selection of low-frequency terms during PRF process. To address this, they recommended using additive smoothing with alternative smoothing methods during feedback to better important terms. The experimental results confirmed that, based on LM, additive smoothing was more effective than fixed baseline smoothing methods.

In 2016, with an emphasis on term weighting Dehghani et al. [67] proposed a model based on LM that identified key terms in documents.. They showed that assigning more weight to terms from relevant documents improved query expansion effectiveness. Additionally, their model was resilient to noise in the feedback documents. Finally, their experiments confirmed that term weighting models could capture more accurate query expansion terms by filtering general terms through weighting.

In 1998, Ponte and Croft [14] proposed an LM approach for IR. Subsequently, many variations of LM were introduced, and document ranking based on the first retrieval was redefined according to extended Language Models.

However, methods based solely on Language Models rely heavily on estimating the relevance of documents based on query term frequencies within the documents themselves. This neglects important semantic relationships between terms that are crucial for optimal query expansion. There is still significant room for improvement in fully exploring the latent semantic relationships between queries and documents, making it a subject worthy of further research.

### **2.2.3 Based on Positional Information**

Rocchio’s model is a well-established method for query expansion, with numerous studies demonstrating its ability to enhance information retrieval performance. However, when selecting terms for expansion, this method ignores the relationships—such as proximity—between

candidate terms and query terms. Given that terms that are nearer to the query terms are probably more relevant to the query topic, it is reasonable to assume that the similarity between candidate terms and query terms should influence the query expansion process.

In 2010, Lv et al. addressed the issue that candidate expansion terms extracted by traditional methods might contain irrelevant information to the query topic. They introduced a Positional Relevance Model (PRM) [46], which builds on the Random Forest (RF) approach. PRM takes into account the positions of terms and their proximity, giving greater importance to terms nearer to the query terms, under the assumption that these are more likely to relate to the query topic. To estimate positional relevance scores, they used two methods utilizing distinct sampling processes. Experiments on two large retrieval datasets revealed that the PRM was both effective and robust for pseudo-relevance feedback (PRF), significantly surpassing other relevance models in terms of both document-based feedback and passage-based feedback.

In 2011, Zhao et al. introduced the Cross Term Retrieval (CRTER) [30] model. To simulate term proximity and enhance retrieval efficiency, they included a pseudo-term known as a “cross term” in this model. They assumed that the occurrence of query terms would influence the surrounding text, with the influence gradually weakening as the distance from the occurrence position increases. They described this influence using a shape function. A cross term is produced when two query terms appear in close proximity and their influence shape functions overlap.

In 2012, three PRoc models [42] based on the Rocchio model proposed by Miao et al., considering the proximity relationship between candidate terms and their corresponding queries in feedback documents: (1) PRoc1, which uses a simple N-gram frequency count method within a fixed window; (2) PRoc2, which uses a Kernel-based frequency count method; and (3) PRoc3, which is based on a term frequency model. In comparison to the traditional Rocchio model, all three versions of the PRoc model demonstrated improved robustness and effectively integrated proximity information. Notably, the PRoc3 model displayed superior

adaptability relative to PRoc1 and PRoc2.

In 2017, Montazer-alghaem et al. proposed three additional constraints based on the proximity of query terms and feedback terms in feedback documents: proximity, convexity, and inverse document frequency of queries (IDF) [61]. They revised the more advanced PRF model with the logistic model to satisfy these constraints, and validated the effectiveness of the constraints proposed by experiments.

## 2.2.4 Based on Deep Learning Model

Neural information retrieval involves using superficial or deep neural networks for tasks related to retrieving information [68]. While document lengths might vary greatly, from a few words to hundreds of phrases or more, search searches typically just comprise a few terms. Vector representations of text are used in neural models for information retrieval, which usually include several adjustable parameters. Large parameter sets in machine learning models typically require a substantial amount of training data [69]. Unlike traditional L2R methods [70], which train machine learning models using a group of manually annotated features, the raw text of queries and documents is frequently used as input for neural network models for information retrieval. Large-scale datasets are also necessary for training to learn appropriate text representations. [71]. Therefore, neural methods frequently need an extensive amount of data, in contrast to traditional IR models, and their performance improves with more training data.

Instead of directly comparing queries and documents in vector space, an alternative method uses vector representations of words to identify appropriate expansion candidates from a global corpus. This expansion query is then used to retrieve documents. Numerous researchers have proposed different functions [72, 73] to estimate the relevance between candidate terms and query terms. Each of these functions utilizes vector representations, assessing each candidate term's vector against every query term's vector, and subsequently consolidating the scores. For example, in [72, 73], the relevance of a candidate term  $t_c$  is es-

estimated as shown in Equation 2.5. QE based on word vectors alone tends to perform worse than PRF. However, when combined with PRF, the performance improves significantly.

$$Score(t_c, q) = \frac{1}{|q|} \sum_{t_q \in q} \cos(v_{t_c}, v_{t_q}) \quad (2.5)$$

where  $t_q$  represents a term in the query  $q$ , and  $v_{t_c}$  and  $v_{t_q}$  represent the vector representations of candidate term  $t_c$  and query term  $t_q$ , respectively.

In 2002, a method for expanding semantic queries based on contextual information kept in a semantic encyclopedia was put forth by Akrivas et al. [74]. By using a fuzzy thesaurus from the encyclopedia, they expanded the terms relevant to the query. The query context, which was described as a collection of fuzzy semantic entities, had to be taken into account during the query expansion process. They also utilized user profiles to expand the query topics.

In 2015, Yao et al. [75] refined initial queries through semantic reasoning operations based on context derived from background knowledge. However, they relied on external resources. It is well-known that external resources cannot provide contextual information for all queries, particularly for unpopular or recent queries.

In 2018, Imani et al. addressed the issue that terms selected by Word Vector-based Query Expansion [76] methods were not always helpful for the retrieval process. They developed a neural network classifier to assess the usefulness of query expansion terms and identify effective candidates for expansion. The classifier utilized word vectors as input. Four TREC collections were used for the experiments, and the results showed that using terms the classifier selected for expansion significantly improved retrieval performance when compared to competing baseline models.

In 2018, Chen et al. proposed a method that adjusted context fragments based on PRF into Context-Aware Topic (CAT) [32] model to enhance query representation. Specifically, instead of choosing a series of independent terms, they fully utilized the context of query, focusing on fragments of length  $l$  in pseudo-relevance documents. Additionally, they intro-

duced a model to capture the topic distribution of context fragments relevant to the query. Unlike traditional Topic Models, which infer topics from the entire corpus, they established a bridge between fragments and the corresponding pseudo-relevance documents, enabling more accurate and efficient topic modeling.

In 2022, Pan et al. [77] presented a probabilistic framework grounded in the classic Rocchio model, which incorporates sentence-level semantics through BERT [3] in PRF. They use BERT to encode the query and each term in the feedback document, and they rank the terms according to their relative relevance. This makes it possible to determine each sentence’s semantic similarity score in relation to the query. A term score at the sentence level is generated by combining these semantic scores. In the end, specific factors are modified to balance the weights of terms and sentences, merging the top-K scoring terms to formulate a new query for the following processing stage.

### **2.2.5 Based on Generative LLM and Others**

Information Retrieval is a key technology that helps users find information of Interest. The fact that a user’s query typically only contains a few terms makes it difficult for the retrieval system to completely understand the user’s genuine purpose, which is one of the difficulties in information retrieval. PRF based on the Topic Model [78] can effectively address this issue. Existing research shows that the Topic Model can help uncover thematic relationships between terms.

In 2006, Wei et al. [29] were pioneers in applying Latent Dirichlet Allocation (LDA) [79] to IR for targeted retrieval tasks. Within the language model retrieval framework, topics derived from LDA served as document smoothing models. Experiments demonstrated that this smoothing technique markedly enhanced retrieval performance. Wei and his team assessed the retrieval effectiveness of the Topic Model and examined the search outcomes from three Topic Models in IR. The findings revealed that the Topic Models employed for language model document smoothing boosted retrieval performance, with the LDA model achieving

the highest results. Other studies have also found that applying the Topic Models to pseudo-feedback documents can improve retrieval results [80].

In 2011, Ye et al. [50] suggested a query expansion technique using Topic Modeling to identify topics associated with the query. Through experiments, they demonstrated that, within the framework of several representative QE methods, selecting topics related to the query intent for query expansion significantly outperformed QE methods based on Language Models.

In 2016, Zhang et al. [81] proposed a new PRF strategy aimed at improving the retrieval of relevant biomedical literature by enhancing the quality of pseudo-feedback documents and expansion terms. The principle of this strategy is as follows: first, they applied ontology-based QE to retrieve more relevant feedback documents. Subsequently, they extracted valuable expansion terms from the pseudo-feedback documents utilizing a ranking method that relied on a semantic graph. In order to improve document relevancy, these query expansion terms were eventually added to the user's query. They assessed the effectiveness of their suggested approach using 10-fold cross-validation on the test set. The experimental findings revealed that this approach boosted retrieval performance by 33.8% in comparison to queries based on free-text.

In 2018, Muhammad et al. [82] proposed a new Chi-Square Test [83] method to select expansion terms in PRF. They also discussed how pre-processing affects retrieval performance in the Biomedical domain. In this field, the presence of noisy data can severely hamper the performance of IR systems utilizing PRF. The results indicated that the proposed algorithm surpassed traditional methods within the Biomedical sector.

In 2018, Zhang et al. [84] proposed a PRF method based on LDA. They applied the LDA model to analyze the document collection, revealing hidden topics and characterizing each document as a multinomial distribution across these topics. They evaluated the connection between the documents and the query by determining the likelihood of a document producing the query. Then ranked the documents according to their level of significance.. The highest-

ranked documents were utilized to derive informative terms that would enhance the original query. Experiments showed that this approach surpassed other sophisticated PRF methods.

Since 2023, ChatGPT has emerged as a key technology in IR due to the rapid advancement of artificial intelligence. Huang et al. [85] delve into the impact of ChatGPT on IR tasks, offering insights into its possible future directions. Retrieval-Augmented Generation (RAG) [86] combines retrieval techniques with advancements in deep learning to overcome the static limitations of Large Language Models (LLMs), allowing for the dynamic incorporation of current external information. They also consolidate existing research on RAG, clarify its technological underpinnings, and highlight its potential to broaden the adaptability and applications of LLM [7].

The results indicate that Pseudo-Relevance Feedback have always been a focal point for researchers in IR, both domestically and internationally. With the advent of new technologies, PRF have promising development prospects, although they also present many challenges.

In fact, after reviewing the existing mainstream PRF methods, it is evident that Query Expansion is flourishing as a significant branch of IR. The selection of query expansion terms and the development of relevance matching algorithms form the foundation of these methods. Researchers have modeled, assessed, and improved multiple features, including the significance of term frequency for candidate query expansion terms, the positional context of terms, the distribution of document topics, and the strength of association between terms and query terms. These efforts aim to improve the quality of query expansion terms in PRF and ultimately enhance the precision, diversity, and recall of retrieval results. However, many relevance calculation methods and techniques are still primarily based on probabilistics, with relatively little research focusing on the intrinsic semantic features of characters, words, and sentences. This shortcoming results in existing models being unable to fully understand or represent users' real intent, leading to average precision and overall retrieval effectiveness remaining at relatively low levels, with significant room for improvement.

Building on existing research, this thesis will conduct an in-depth study of PRF from different perspectives. By integrating new technologies, we aim to combine the intrinsic semantic features of characters, words, and sentences in queries and documents with current techniques. This will gradually refine the Generation Model for candidate expansion terms, producing more relevant and higher-quality query expansion terms to improve information retrieval performance. The research conducted here is expected to provide new insights for further research in this field.

## 2.3 Chapter Summary

This chapter first provides a brief overview of the development of Information Retrieval, including the evolution of IR technology, Relevance Feedback, and Pseudo-Relevance Feedback. It then focuses on detailed research on Query Expansion methods based on PRF and surveys various aspects of classic Query Expansion methods. The results indicate that PRF has consistently been a focal point for researchers in the field of IR, both domestically and internationally. With the emergence of new technologies, PRF methods have promising prospects but also face significant challenges.

In fact, after reviewing the existing mainstream PRF, it is evident that as an important branch of information retrieval, PRF techniques are flourishing. The core work in this area revolves around the selection methods for query expansion terms and research into relevance matching algorithms. By examining features such term frequency importance, positional information of words, contextual information, the distribution of topics in the document, and the degree of association with the query terms, researchers model, assess, and enhance the quality of prospective query expansion terms. The ultimate objective is to improve the retrieval results' recall, diversity, and precision. However, most of the current methods for calculating relevance are primarily based on probabilistic and statistical models. There is relatively little research on the intrinsic semantic features of characters, words, and sentences.

This leads to models that do not adequately understand and represent the true intent of user queries, keeping the average precision and overall effectiveness of information retrieval at a relatively low level, leaving much room for improvement.

Building on the existing research, this paper aims to conduct an in-depth study of PRF from different perspectives. By integrating new methods that combine the intrinsic semantic features of characters, words, and sentences in queries and documents with existing work, this research seeks to gradually improve candidate expansion term selection models. The goal is to select or generate more relevant and higher-quality query expansion terms, thus improving the performance of retrieval and providing new ideas for further research in the field.

# Chapter 3

## Query Expansion via Semantic Network

### 3.1 Chapter Introduction

In the PRF process, Query Expansion plays a role because the core success of retrieval performance depends largely on the quality of the candidate terms selected. Traditional PRF methods usually complete term ranking and selection by evaluating the importance of candidate terms in pseudo-relevant documents. These pseudo-relevant documents refer to the top N documents in the final search results that are considered most relevant to the user query. In terms of PRF, the importance of a term is calculated based on its frequency of occurrence in the document or inverse document frequency.

For example, take the term “Apple”. Depending on its meaning, “Apple” may refer to a fruit, or it may refer to a brand of smartphone, such as Hackintosh. In the case of the Bag-of-Words model, this is considered equivalent [87]. In practical applications, relying solely on term frequency and inverse document frequency to select potential expansion terms often results in candidate terms that inadequately capture the semantic relationships associated with the query terms. This uncertainty underscores the necessity for advanced semantic

comprehension in QE.

To address these issues, we suggest a new semantic-based PRF model (CNRoc). Our model leverages ConceptNet to enhance semantic relationships among terms, providing a deeper understanding of their connections. This approach enhances the PRF framework by not only evaluating a query’s relevance to the document collection but also integrating semantic analysis to more effectively identify terms suitable for query expansion. This approach ensures that users receive more relevant feedback documents. Experimental findings validate the CNRoc model’s effectiveness, demonstrating robust performance across multiple metrics. It also surpasses baseline models, SOTA models, and different neural network models. A case study comparison reveals that the expansion terms generated by our model align better with the query’s intended semantic meaning.

## 3.2 Semantic Network

Semantic Network [5], such as ConceptNet, are gaining recognition for their ability to overcome certain limitations of conventional PRF techniques. ConceptNet is a large-scale, multilingual knowledge graph that encodes general human knowledge in a machine-readable format, capturing a wide range of relationships between concepts. In the context of IR, A Semantic Network can improve query expansion by offering semantically related terms relevant to the user’s context.

Early research has shown that incorporating semantic information from external knowledge sources into the information retrieval process can significantly improve retrieval results. This approach can provide the retrieval system with richer contextual information, thereby more accurately understanding the meaning of the query and the content of the document, and re-improving the relevance and quality of search results. For example, Li et al. [88] explored the use of WordNet for QE, showing that incorporating synonyms and related terms can significantly improve retrieval performance. Similarly, Xu and Croft [89] examined the

application of ConceptNet in IR, in this process, the network is used to generate semantically richer expanded terms to better match the user’s query and retrieved documents. These expanded terms can capture the potential meaning of the query and fill in the semantic gaps that may exist in traditional retrieval methods.

However, while the integration of Semantic Network offers a promising approach to query expansion, there are challenges associated with effectively leveraging these resources. One key issue is determining the appropriate weight to assign to the expansion terms generated from the semantic network, as overly emphasizing certain terms can lead to skewed retrieval results. Additionally, the coverage and accuracy of the semantic network itself can impact the quality of the expanded query, highlighting the need for careful selection and tuning of these external resources.

ConceptNet<sup>1</sup> is a knowledge graph originating from the OMCS crowdsourcing project. As depicted in Figure 3.1 . ConceptNet is a semantic network with structured data that contains a large amount of entity knowledge and relationship knowledge, making information easier to calculate, interpret and evaluate [89].

Commonly known as a Knowledge Base (KB), ConceptNet acts as a repository for both structured and unstructured data used in information systems. While KBs focus on storing and retrieving structured information, knowledge mapping transcends this by constructing associative knowledge, including triadic semantic networks. Allan M. Collins, a cognitive scientist and an early advocate for semantic networks, proposed using knowledge representation to explore semantic memory in the human brain [90]. WordNet stands as a quintessential example of early semantic networks. Unlike semantic networks, which prioritize depicting concepts and their interrelationships, knowledge graphs focus on linking data and entities.

ConceptNet is a multilingual knowledge graph that connects words and phrases through labeled edges, representing general knowledge to aid in natural language understanding. It supports over 100 languages, each with more than 10,000 terms, facilitating cross-linguistic

---

<sup>1</sup><https://conceptnet.io/>

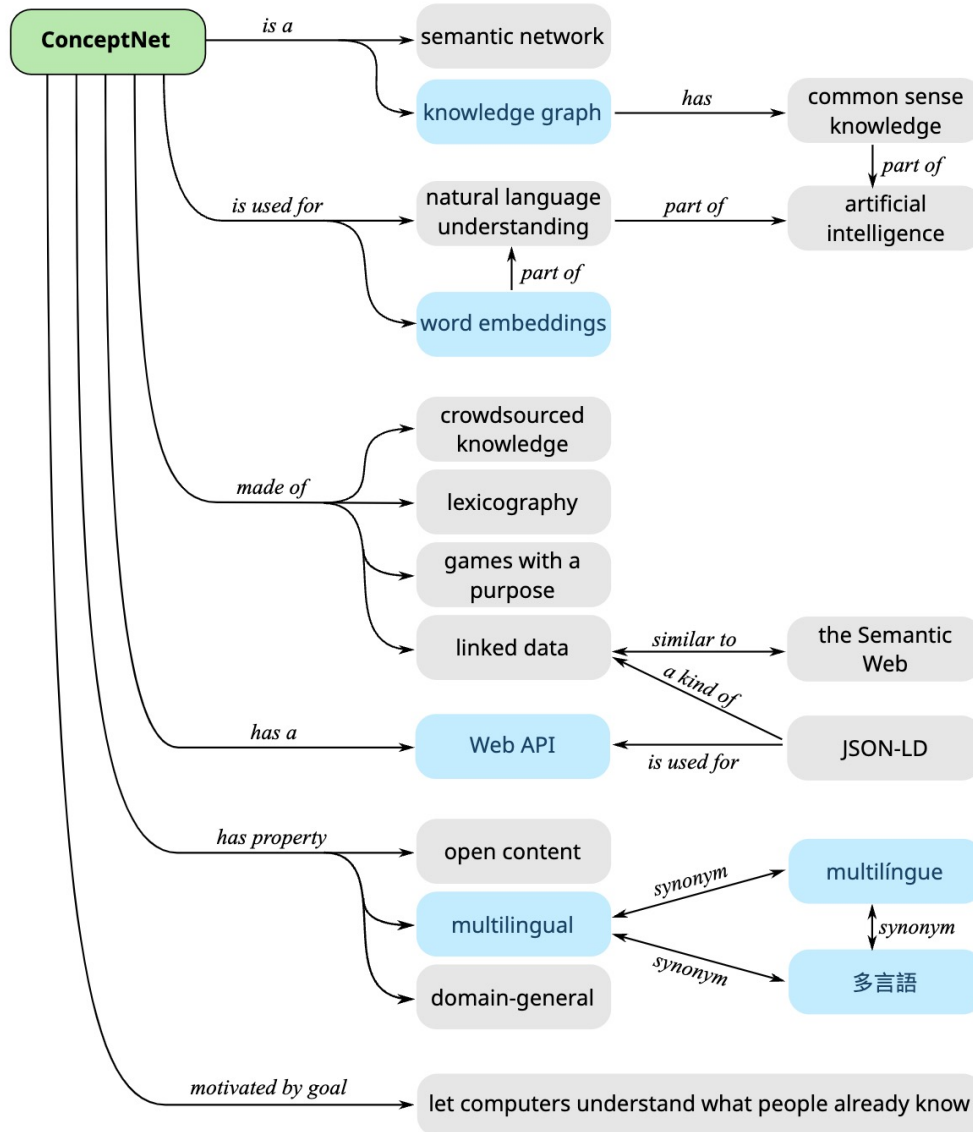


Figure 3.1: ConceptNet

applications. Unlike traditional knowledge graphs, ConceptNet emphasizes natural language descriptions and focuses on word-to-word relationships, making it particularly useful for tasks that require understanding common-sense knowledge and semantic connections between terms

### 3.3 CNRoc: Query Expansion via Semantic Network

Query Expansion (QE) is a crucial technique in modern information retrieval systems, aiming to address the issues of semantic under-specification and vocabulary mismatch in user queries. Traditional QE methods, such as Rocchio-based pseudo-relevance feedback (PRF), heavily depend on the quality of the initial retrieval results and lack external semantic knowledge integration. To tackle these limitations, we propose a novel framework named **CNRoc**, which leverages the semantic network *ConceptNet* and dynamically integrates semantic expansion with PRF.

The framework comprises three main stages, as shown in Figure 3.2: (1) Semantic Related Term Extraction, (2) Query Expansion Term Generation, and (3) Query Refinement and Second-Round Retrieval. Each stage is elaborated below.

#### 3.3.1 Semantic Related Term Extraction

Semantic related term extraction plays a critical role in enriching the query representation. *ConceptNet* is a structured semantic knowledge graph consisting of nodes (concepts) and edges (semantic relationships). Unlike traditional statistical-based QE methods, *ConceptNet* captures diverse semantic relations, such as:

- **Synonymy**: Concepts with identical meanings, e.g., “car” and “automobile.”
- **Hypernymy**: Hierarchical relationships, e.g., “dog isA animal.”
- **Part-Whole**: Compositional relationships, e.g., “wheel partOf car.”
- **Causality**: Cause-and-effect relationships, e.g., “rain Causes wet ground.”

By introducing these relations, the query’s semantic coverage is significantly enhanced, enabling the retrieval system to better interpret the user’s intent.

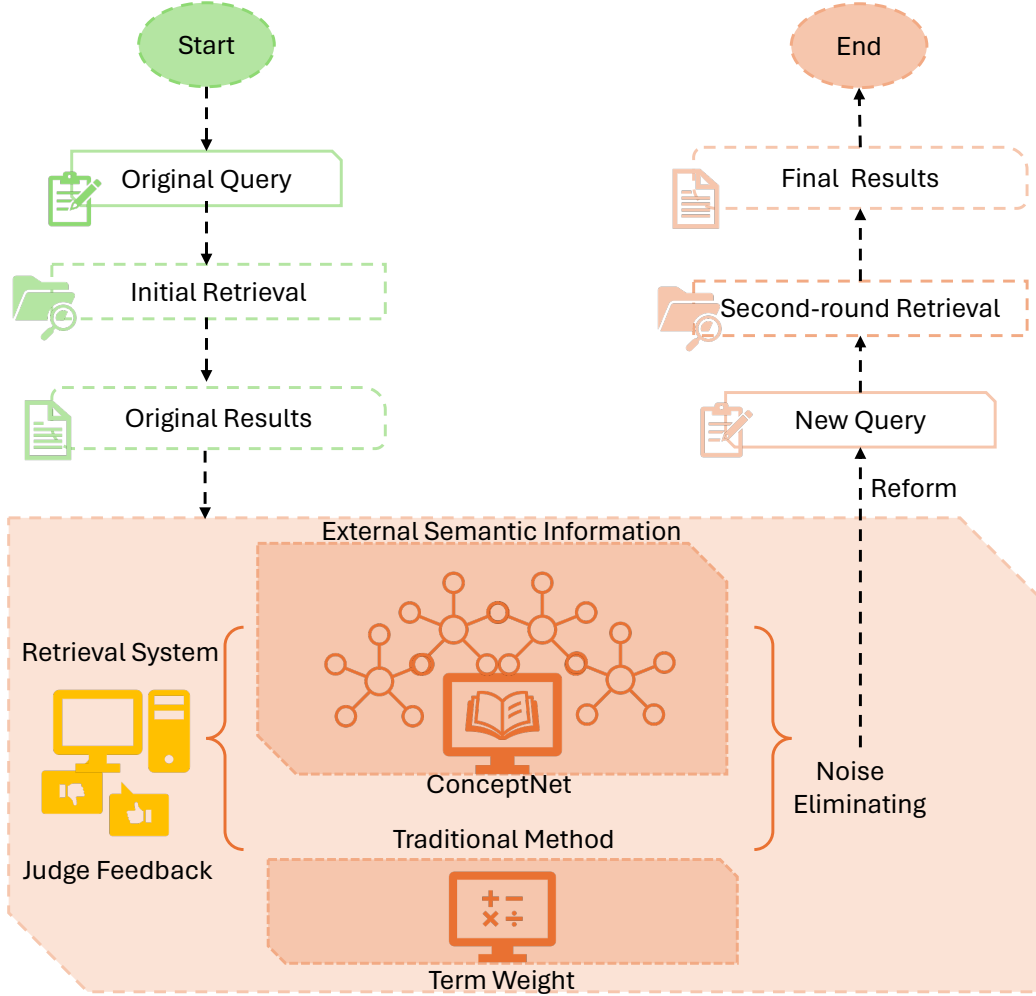


Figure 3.2: Flowchart of CNRoc

1. **Query Term Mapping and Node Retrieval:** For a given query  $Q_0 = \{q_1, q_2, \dots, q_n\}$ , each term  $q_i$  is used as a seed to retrieve directly connected nodes  $C(q_i)$  in ConceptNet, forming the candidate related term set:

$$W_{\text{cand}} = \bigcup_{q_i \in Q_0} C(q_i). \quad (3.1)$$

2. **Depth Control for Indirect Relations:** To capture indirect semantic relations, the retrieval depth  $d$  is set (typically  $d = 50$ ), allowing the inclusion of secondary

connections.

3. **Semantic Relationship Weighting:** Each edge in ConceptNet is associated with a weight representing the strength of the semantic relationship. For a candidate term  $w \in W_{\text{cand}}$ , its semantic similarity to the query is computed as:

$$\text{sim}(w, Q_0) = \max_{q \in Q_0} \frac{\text{weight}(r_{w,q}) \cdot (\vec{w} \cdot \vec{q})}{\|\vec{w}\| \|\vec{q}\|}, \quad (3.2)$$

where  $r_{w,q}$  represents the relationship between  $w$  and  $q$ .

4. **Initial Filtering:** Terms with semantic similarity below a threshold  $\theta$  are discarded. The filtered set  $W_{\text{filtered}}$  serves as input for the next stage.

### 3.3.2 Query Expansion Term Generation

Semantic-related terms derived from ConceptNet may not fully capture the user’s specific information needs. Thus, they must be refined using pseudo-relevance feedback, which provides context-dependent weights by analyzing the high-frequency terms in the initial retrieval results.

1. **Pseudo-Relevance Document Selection:** From the initial retrieval results, the top- $k$  documents are selected as the relevant set  $D_{\text{rel}}$ .
2. **Weight Calculation for Expansion Terms:** For each candidate term  $w \in W_{\text{filtered}}$ , its weight is computed as a unified combination of semantic similarity and statistical relevance:

$$\text{weight}(w) = \delta \cdot (\text{sim}(w, Q_0) + (1 - \delta)\text{TF-IDF}(w, D_{\text{rel}})) \quad (3.3)$$

where  $\delta$  is a scaling parameter controlling the contribution of the combined factors.

3. **Term Selection:** The top- $m$  terms with the highest weights are selected to form the final expansion term set  $W_{\text{exp}}$ .

### 3.3.3 The Adaptive PRF Model

#### Enhanced Rocchio Model

The classic Rocchio algorithm updates the query vector by combining the original query with pseudo-relevance feedback. We extend the formula to incorporate the expansion terms while excluding the non-relevant document component:

$$Q_{\text{new}}^{\text{cn}} = \alpha Q_0 + \beta \sum_{w \in W_{\text{exp}}} \text{weight}(w) \cdot \vec{w} \quad (3.4)$$

where:

- $\alpha, \beta$ : Parameters controlling the contributions of the original query and expansion terms, respectively.
- $Q_0$ : The original query vector.
- $W_{\text{exp}}$ : The set of expansion terms.

#### Second-Round Retrieval

The updated query  $Q_{\text{new}}^{\text{cn}}$  is used for the second-round retrieval, which incorporates both the user’s original intent and enhanced semantic and statistical knowledge. This step improves the relevance and coverage of the retrieved results.

The proposed **CNRoc** framework effectively integrates semantic knowledge from ConceptNet with PRF techniques, addressing the limitations of traditional QE methods. By dynamically weighting semantic expansion terms and incorporating them into the Rocchio model, CNRoc achieves superior performance in improving retrieval relevance and robustness.

## 3.4 Experimental Setup

### 3.4.1 Experimental Datasets and Analytical Metrics

**Experimental Datasets** TREC is a prominent evaluation initiative organized by the National Institute of Standards and Technology (NIST) in collaboration with DARPA (Defense Advanced Research Projects Agency). Its primary aim is to advance the field of information retrieval by providing standardized datasets and evaluation frameworks.

TREC provides large-scale and well-structured datasets for reproducible experiments. Popular datasets as shown in the Table 3.1. The diversity in size and type among these datasets ensures a comprehensive evaluation of our model. By including a mix of news articles, web documents, and governmental content, we can test the robustness, scalability, and generalizability of our information retrieval system. Each dataset poses unique challenges and opportunities, allowing us to fine-tune our model for optimal performance across various types of data.

Table 3.1: Validation Datasets for the CNRoc Model

Collection	Size	Queries	# of Queries	# of Docs
AP90	0.23 Gb	51–100	50	78,321
AP88-89	0.50 Gb	51–100	50	164,597
DISK4&5	1.86 Gb	301–450	150	528,155
WT2G	2.14 Gb	401–450	50	247,491
WT10G	10 Gb	451–550	100	1,692,096
GOV2	426 Gb	701–850	150	25,178,548

By leveraging these varied datasets, we aim to validate the effectiveness and robustness of our proposed information retrieval model across a broad spectrum of real-world scenarios.

- **AP90** The AP90 dataset contains news articles published by the Associated Press (AP) in 1990. It provides a focused collection of reports on current events and topics

from that year, making it a valuable resource for testing information retrieval models on contemporary news data.

- **AP88-89** This dataset includes news content from the AP published in 1988 and 1989. Compared to AP90, it spans a broader temporal range, allowing for the evaluation of retrieval models across multiple years of news data and assessing their consistency over time.
- **DISK4&5** The DISK4&5 collection comprises newsletter-style articles from a range of sources. While the dataset is known for its high-quality content, the diversity of origins introduces variability, making it well-suited for evaluating the robustness of retrieval systems under diverse data conditions.
- **WT2G** The WT2G dataset, initially used in the TREC-8 Web Track, consists of 2 GB of web documents. Its content spans various domains and topics, providing a challenging benchmark for assessing retrieval performance on heterogeneous and unstructured web data.
- **WT10G** As an extension of WT2G, WT10G contains 10 GB of web documents and was employed in the TREC-9 and TREC-10 Web Tracks. This dataset is particularly useful for testing the scalability and efficiency of retrieval systems on mid-sized web collections.
- **GOV2** The GOV2 dataset is a comprehensive crawl of U.S. government websites, featuring over 25 million documents with an uncompressed size of 426 GB. Used in the TREC Terabyte Tracks (2004–2006), it provides a domain-specific benchmark for evaluating retrieval performance on large-scale governmental data.

To evaluate the effectiveness of the proposed retrieval framework, we adopt four widely recognized evaluation metrics: Mean Average Precision (MAP), Precision at 10 (P@10), Normalized Discounted Cumulative Gain (NDCG), and Mean Reciprocal Rank (MRR). Each

metric provides distinct perspectives on the system’s retrieval quality and ranking effectiveness.

- **Mean Average Precision (MAP)** Mean Average Precision (MAP) is a metric used to assess a system’s precision across multiple queries. For each query, it calculates the average precision based on the ranks of relevant documents, and these scores are averaged across all queries. This metric emphasizes both the retrieval of relevant documents and their placement higher in the ranking. The formula for MAP is:

$$MAP = \frac{1}{|N|} \sum_{i=1}^{|N|} \left( \frac{1}{|L_i|} \sum_{j=1}^{|L_i|} Prec_j \right) \quad (3.5)$$

where  $|N|$  is the total number of queries,  $|L_i|$  is the number of relevant documents for query  $i$ , and  $Prec_j$  is the precision at rank  $j$ . A higher MAP score indicates better performance across diverse queries.

- **Precision at 10 (P@10)** Precision at 10 (P@10) calculates the proportion of relevant documents among the top 10 results retrieved by the system. This metric is crucial because users often focus on the first page of results. The formula is:

$$P@10 = \frac{\text{Count of relevant documents in top 10 results}}{10} \quad (3.6)$$

A higher P@10 score reflects the system’s ability to prioritize relevant documents in the top results, improving user satisfaction.

- **Normalized Discounted Cumulative Gain (NDCG)** Normalized Discounted Cumulative Gain (NDCG) evaluates how well a system ranks documents based on their relevance and position. It employs a graded relevance scale, assigning higher importance to relevant documents appearing earlier in the ranked list. The Discounted

Cumulative Gain (DCG) at rank  $p$  is calculated as:

$$DCG_p = \sum_{k=1}^p \frac{score_k}{\log_2(k+1)} \quad (3.7)$$

where  $score_k$  represents the relevance score of the document at rank  $k$ . The NDCG is then computed by normalizing DCG with the Ideal DCG ( $iDCG$ ), representing the best possible ranking:

$$NDCG_p = \frac{DCG_p}{iDCG_p} \quad (3.8)$$

A higher NDCG score signifies better placement of relevant documents at the top, enhancing the user experience.

- **Mean Reciprocal Rank (MRR)** The Mean Reciprocal Rank (MRR) measures how quickly the first relevant document appears for each query. It calculates the average reciprocal rank of the first relevant document across all queries. The formula for MRR is:

$$MRR = \frac{1}{|N|} \sum_{i=1}^{|N|} \frac{1}{pos_i} \quad (3.9)$$

where  $pos_i$  denotes the rank position of the first relevant document for query  $i$ . A higher MRR indicates that the system is efficient at presenting relevant content at the top of the ranked list.

These metrics collectively provide a robust framework for evaluating the performance of information retrieval systems, each addressing different aspects of retrieval effectiveness and user satisfaction. By using these metrics, researchers and practitioners can gain a comprehensive understanding of how well their systems meet users' information needs.

### 3.4.2 Baseline Model

BM25 is a widely adopted probabilistic retrieval model based on the term frequency-inverse document frequency (TF-IDF) principle. It calculates relevance scores by considering term

frequency saturation and document length normalization. BM25 serves as a strong traditional baseline for ad hoc retrieval tasks due to its simplicity and efficiency. The Rocchio algorithm is a classic pseudo-relevance feedback (PRF) model that updates the query representation by leveraging the top-k pseudo-relevant documents. The updated query vector is calculated using a weighted linear combination of the original query and pseudo-relevant document vectors. This baseline allows us to compare the proposed integration of semantic knowledge with traditional feedback methods. Additionally, we assess its performance against enhanced models based on Rocchio, such as PRoc2 [42] and KRoc [44], to further confirm the validity of our method.

### 3.4.3 Hyperparameter Settings

To optimize the hyperparameters and evaluate their impact on the CNRoc framework’s performance, we conducted a range of controlled experiments. Initially, the number of pseudo-relevant documents ( $N$ ) was tested with values of 3, 5, 10, 20, 30, 50, 100, and 200 across six TREC datasets. As shown in Figure 3.3, the results indicated that using 50 pseudo-relevant documents achieved an effective balance between computational efficiency and retrieval accuracy. Based on these findings, this value was adopted for the subsequent stages of the experimental workflow.

In addition to tuning  $N$ , the number of candidate terms for semantic expansion ( $|T_f|$ ) was varied, with options set to 10, 20, 30, and 50. The parameters  $\alpha$ ,  $\beta$ , and  $\delta$  were fine-tuned within the interval  $[0, 1]$  to assess their influence on retrieval effectiveness. Performance was measured using MAP and NDCG metrics computed for the top 1000 retrieved documents, as well as P@10 for the highest-ranking results. To ensure consistent evaluation, queries lacking relevance judgments were excluded from both the indexing and retrieval processes.

We use Porter’s English stemmer [91] to process terms across all datasets and remove 418 stop words based on the standard InQuery list [92]. All models with trained parameters are assessed using the chosen TREC datasets and topics. By applying consistent evaluation

metrics and parameter settings, we ensure an accurate and reliable comparison.

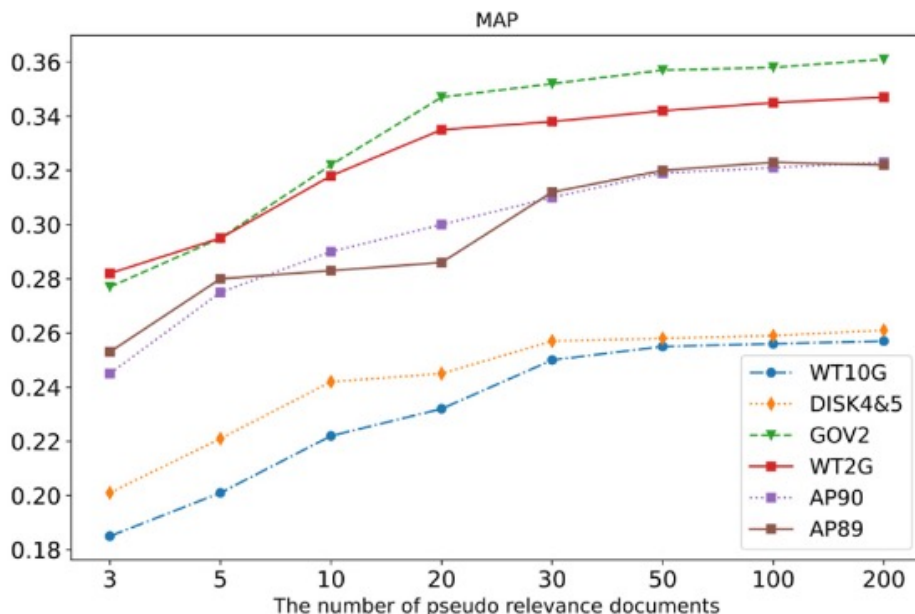


Figure 3.3: CNRoc’s experimental results of different pseudo-relevance documents

## 3.5 Experimental Results and Analysis

### 3.5.1 Validation Against Baseline and Strong Baseline Models

The proposed **CNRoc** framework incorporates conceptual knowledge and contextual insights into the PRF paradigm. Specifically, external semantic information associated with the input query is extracted using external semantic network. Then, QE is achieved by leveraging computations derived from pseudo-relevance documents, and this enhanced information is integrated into the existing PRF method.

To demonstrate the effectiveness of CNRoc, it was compared against the traditional Rocchio baseline. Additionally, further evaluations were conducted using advanced Rocchio-based models, such as **PRoc2** and **KRoc**, to comprehensively validate its performance. These experiments were executed on six TREC datasets, measuring key metrics.

The experimental results, summarized in Tables 3.2 and 3.3, present the averaged MAP

and P@10 scores across varying numbers of query expansion terms. Furthermore, statistical tests were employed to determine the significance of these results, offering insights into the comparative strengths of the models. In addition to these metrics, **NDCG** and **MRR** scores were used to further evaluate the ranking quality of CNRoc, with results detailed in Table 3.4.

As shown in Table 3.2, the MAP values for both CNRoc and Rocchio increase consistently with the number of feedback terms ( $|T_f|$ ). The experimental findings demonstrate that CNRoc consistently surpasses Rocchio-based models in average MAP across a range of  $|T_f|$  values, exhibiting significant improvements across various datasets. Specifically, the maximum MAP gains achieved by CNRoc are recorded as 16.29%, 11.73%, 12.83%, 4.58%, 17.20%, and 7.41% for the six collections (AP90 through GOV2), respectively. Additionally, CNRoc outperforms advanced baselines, such as KRoc and PRoc2, in MAP, delivering the most competitive results across all datasets.

Given that retrieval systems are often evaluated based on their ability to prioritize top-ranked results, P@10 serves as a critical metric. Table 3.3 highlights a comparison of P@10 scores among CNRoc, the Rocchio baseline, and other advanced baselines. The results clearly indicate that CNRoc achieves superior P@10 scores, consistently ranking the highest across all tested datasets.

To further assess the robustness of the model, comparisons were conducted using NDCG and MRR metrics, as presented in Table 3.4. These results reveal that CNRoc achieves substantial improvements in both metrics, underscoring its ability to enhance ranking quality and retrieval effectiveness. By integrating ConceptNet with the PRF framework, CNRoc effectively identifies and utilizes high-quality QE terms, thereby improving search outcomes.

The success of CNRoc can be attributed to its dual approach: (1) focusing on the selection of critical terms for query expansion and (2) leveraging ConceptNet to extract meaningful semantic and conceptual knowledge from pseudo-relevant documents. As a result, CNRoc demonstrates strong adaptability, even in scenarios rich in semantic and conceptual data,

ultimately enhancing the overall search experience.

### 3.5.2 Validation Against the SOTA PRF Models

To further evaluate the efficiency of the proposed model, we conducted a comparative analysis of MAP scores across six standard datasets, benchmarking against several advanced query expansion (QE) models. The **MRF model** identifies QE terms based on their term dependencies. The **IF&FB model** incorporates relational patterns derived from information flow within the language modeling (LM) framework to select expansion terms. Meanwhile, the **RM3 model**, which relies on a probabilistic language modeling approach, selects expansion terms by evaluating the likelihood of generating candidate document queries, leveraging word frequency statistics. The **HAL model**, on the other hand, computes multidimensional distances to identify expansion terms, emphasizing higher semantic relevance for terms closer to the original query. Notably, RM3 has shown strong performance across various retrieval tasks.

Among the probabilistic models evaluated, **PRoc3** was chosen due to its lower sensitivity to variations in parameter settings. Similarly, **TF-PRF** applies co-occurrence frequency-based term selection and normalization techniques, which align closely with the functionality of PRoc2. On the other hand, **SRoc** and **SPRoc2** integrate advanced sentence-level semantics into the PRF process by leveraging BERT embeddings to enhance the representation of queries and documents.

All methods were tested under uniform experimental conditions, ensuring consistent comparisons with standard baselines like BM25. Table 3.5 presents the outcomes, where the bolded values indicate the best-performing results across datasets. These results emphasize the superior MAP scores achieved by the proposed CNRoc framework, which consistently outperformed other evaluated models.

The observed improvements are attributed to how prior methods calculate term weights, often relying on co-occurrence statistics, probabilistic term generation, or contextual embed-

Table 3.2: Evaluation of MAP Metrics: CNRoc Model vs. Baseline Models

Model	$ T_f $	AP90	AP88-89	DISK4&5	WT2G	WT10G	GOV2
Rocchio	10	0.2858	0.2908	0.2307	0.3249	0.2064	0.3232
	20	0.2884	0.2938	0.2328	0.3253	0.2076	0.3263
	30	0.2899	0.2950	0.2337	0.3255	0.2077	0.3276
	50	0.2926	0.2965	0.2349	0.3258	0.2088	0.3296
	Avg	0.2892	0.2940	0.2330	0.3254	0.2076	0.3267
PRoc2	10	0.3031	0.3128	0.2506	0.3328	0.2212	0.3264
	20	0.3132	0.3176	0.2572	0.3333	0.2258	0.3314
	30	0.3204	0.3200	0.2588	0.3406	0.2248	0.3364
	50	0.3243	0.3212	0.2618	0.3344	0.2226	0.3401
	Avg	0.3153	0.3179	0.2571	0.3353	0.2236	0.3336
KRoc	10	0.3287	0.3131	0.2563	0.3338	0.2174	0.3303
	20	0.3367	0.3172	0.2621	0.3357	0.2166	0.3403
	30	0.3357	0.3214	0.2657	0.3348	0.2162	0.3416
	50	0.3383	0.3222	0.2654	0.3451	0.2125	0.3419
	Avg	0.3349	0.3185	0.2624	0.3374	0.2157	0.3385
CNRoc	10	0.3351 <sup>*</sup> (+17.25%)	0.3249 <sup>*</sup> (+11.73%)	0.2561 <sup>*</sup> (+11.01%)	0.3349 <sup>*</sup> (+3.08%)	0.2394 <sup>*</sup> (+15.99%)	0.3461 <sup>*</sup> (+7.09%)
	20	0.3359 <sup>*</sup> (+16.47%)	0.3287 <sup>*</sup> (+11.88%)	0.2595 <sup>*</sup> (+11.47%)	0.3387 <sup>*</sup> (+4.12%)	0.2418 <sup>*</sup> (+16.47%)	0.3487 <sup>*</sup> (+8.86%)
	30	0.3363 <sup>*</sup> (+16.01%)	0.3298 <sup>*</sup> (+11.80%)	0.2677 <sup>*</sup> (+14.55%)	0.3405 <sup>*</sup> (+4.61%)	0.2455 <sup>*</sup> (+18.20%)	0.3530 <sup>*</sup> (+7.75%)
	50	0.3377 <sup>*</sup> (+15.41%)	0.3309 <sup>*</sup> (+11.60%)	0.2684 <sup>*</sup> (+14.26%)	0.3453 <sup>*</sup> (+5.99%)	0.2463 <sup>*</sup> (+17.96%)	0.3557 <sup>*</sup> (+7.92%)
	Avg	<b>0.3363<sup>*</sup></b> (+16.29%)	<b>0.3285<sup>*</sup></b> (+11.73%)	<b>0.2629<sup>*</sup></b> (+12.83%)	<b>0.3403<sup>*</sup></b> (+4.58%)	<b>0.2433<sup>*</sup></b> (+17.20%)	<b>0.3509<sup>*</sup></b> (+7.41%)

**Note:** Bold values indicate optimal results for each dataset.

Table 3.3: Evaluation of P@10 Metrics: CNRoc Model vs. Baseline Models

Model	$ T_f $	AP90	AP88-89	DISK4&5	WT2G	WT10G	GOV2
Rocchio	10	0.4468	0.4571	0.4247	0.4920	0.3092	0.5878
	20	0.4426	0.4571	0.4240	0.4940	0.3071	0.6020
	30	0.4426	0.4551	0.4233	0.4940	0.3082	0.6061
	50	0.4447	0.4571	0.4220	0.4900	0.3082	0.5995
	Avg	0.4442	0.4566	0.4235	0.4925	0.3082	0.5989
PRoc2	10	0.4553	0.4633	0.4335	0.5080	0.3226	0.5822
	20	0.4553	0.4755	0.4393	0.5242	0.3245	0.5914
	30	0.4617	0.4735	0.4380	0.5201	0.3275	0.5837
	50	0.4617	0.4663	0.4407	0.5223	0.3325	0.5852
	Avg	0.4590	0.4694	0.4383	0.5187	0.3267	0.5856
KRoc	10	0.4660	0.4653	0.4313	0.5120	0.3133	0.5878
	20	0.4681	0.4755	0.4387	0.5141	0.3113	0.5839
	30	0.4681	0.4673	0.4380	0.5141	0.3133	0.5895
	50	0.4638	0.4740	0.4393	0.5221	0.3112	0.5951
	Avg	0.4665	0.4699	0.4368	0.5156	0.3128	0.5924
CNRoc	10	0.4898*	0.4759*	0.4493*	0.5261*	0.3617*	0.6075*
		(+9.62%)	(+4.11%)	(+5.79%)	(+6.93%)	(+16.98%)	(+3.39%)
	20	0.4943*	0.4760*	0.4537*	0.5246*	0.3640*	0.6081*
		(+11.68%)	(+4.13%)	(+7.00%)	(+6.19%)	(+18.53%)	(+1.31%)
	30	0.4955*	0.4786*	0.4525*	0.5232*	0.3635*	0.6058*
		(+11.95%)	(+4.56%)	(+6.96%)	(+6.59%)	(+17.94%)	(+0.63%)
50	0.4967*	0.4797*	0.4549*	0.5253*	0.3631*	0.6083*	
	(+11.69%)	(+4.94%)	(+7.80%)	(+7.20%)	(+17.81%)	(+0.63%)	
Avg	<b>0.4940*</b>	<b>0.4775*</b>	<b>0.4526*</b>	<b>0.5248*</b>	<b>0.3630*</b>	<b>0.6075*</b>	
	(+11.21%)	(+4.58%)	(+6.87%)	(+6.56%)	(+17.78%)	(+1.33%)	

**Note:** Bold values indicate optimal results for each dataset.

Table 3.4: Evaluation of NDCG and MRR Metrics: CNRoc Model vs. Baseline Models

Collection	Metric	Rocchio	PRoc2	KRoc	CNRoc
AP90	NDCG	0.6682	0.6689	0.6810	<b>0.6837*</b>
	MRR	0.6106	0.5777	0.6259	<b>0.6421*</b>
AP88–89	NDCG	0.6745	0.6716	0.6812	<b>0.6859*</b>
	MRR	0.5337	0.5415	0.5534	<b>0.5642*</b>
DISK4&5	NDCG	0.6564	0.6580	0.6701	<b>0.6693*</b>
	MRR	0.5903	0.5989	0.6037	<b>0.6078*</b>
WT2G	NDCG	0.7085	0.7093	0.7138	<b>0.7170*</b>
	MRR	0.6830	0.6907	0.6978	<b>0.7136*</b>
WT10G	NDCG	0.5778	0.5838	0.5877	<b>0.6171*</b>
	MRR	0.5207	0.5194	0.5286	<b>0.5374*</b>
GOV2	NDCG	0.7476	0.7534	0.7541	<b>0.7534*</b>
	MRR	0.6544	0.6569	0.7014	<b>0.7468*</b>

**Note:** Bold values indicate optimal results for each dataset.

dings. Unlike these approaches, the proposed framework integrates semantic and conceptual knowledge from ConceptNet into the Rocchio algorithm. By prioritizing semantically relevant terms and refining the initial retrieval process, CNRoc assigns greater weight to meaningful documents, thereby improving overall retrieval performance.

### 3.5.3 Validation Against Neural IR models

To thoroughly assess the performance of CNRoc, we compared it against six widely recognized neural-based information retrieval models: CDSSM [93], DRMM [94], DSSM [95], MatchPyramid (simply as MP in this thesis) [96] and TKL [97]. These experiments utilized the six datasets introduced in Section 3.4.1 and were implemented using the MatchZoo framework. Evaluation metrics including MAP, P@10, NDCG, and MRR were computed to provide a comprehensive comparison.

For training the aforementioned models, each query was paired with one relevant document (positive) and 50 irrelevant documents (negative). To mitigate limitations caused by insufficient training data, the TKL model was pre-trained on the MS Marco dataset and subsequently fine-tuned on the six datasets specified in Section 3.4.1. The hyperparame-

Table 3.5: Evaluation of MAP Metrics: CNRoc Model vs. SOTA Models

Model	AP90	AP88-89	DISK4&5	WT2G	WT10G	GOV2
BM25	0.2738	0.2882	0.2258	0.3192	0.2050	0.3035
MRF	0.2920	0.3088	0.2579	0.3380	0.2214	0.3357
	(+6.65%)	(+7.15%)	(+14.22%)	(+5.89%)	(+8.00%)	(+10.61%)
HAL	0.2810	0.2916	0.2363	0.3285	0.2158	0.3228
	(+2.63%)	(+1.18%)	(+4.65%)	(+2.91%)	(+5.27%)	(+6.36%)
IF&FB	0.2886	0.2971	0.2565	0.3301	0.2180	0.3326
	(+5.41%)	(+3.09%)	(+13.60%)	(+3.41%)	(+6.34%)	(+9.59%)
RM3	0.3082	0.3171	0.2600	0.3326	0.2253	0.3305
	(+12.56%)	(+10.03%)	(+15.15%)	(+4.20%)	(+9.05%)	(+8.90%)
PRoc3	0.3181	0.3179	0.2575	<b>0.3534</b>	0.2256	0.3393
	(+16.18%)	(+10.31%)	(+14.04%)	(+10.71%)	(+10.05%)	(+11.80%)
TF-PRF	0.3074	0.3190	<b>0.2699</b>	0.3448	0.2350	0.3371
	(+12.27%)	(+10.69%)	(+19.53%)	(+8.02%)	(+14.63%)	(+11.07%)
SRoc	0.3271	0.3197	0.2598	0.3444	0.2374	0.3494
	(+19.48%)	(+10.92%)	(+15.04%)	(+7.89%)	(+15.78%)	(+15.11%)
SPRoc2	0.3282	0.3283	0.2642	0.3456	0.2428	0.3412
	(+19.88%)	(+13.91%)	(+16.19%)	(+8.28%)	(+18.45%)	(+12.65%)
CNRoc	<b>0.3363</b>	<b>0.3285</b>	0.2629	0.3403	<b>0.2433</b>	<b>0.3509</b>
	(+22.83%)	(+13.98%)	(+16.43%)	(+6.61%)	(+18.68%)	(+15.62%)

**Note:** **Bold** values indicate optimal results for each dataset.

ter configurations were consistent with the guidelines established in [97], ensuring fair and reliable comparisons across all models.

Table 3.6: Evaluation of MAP Metrics: CNRoc Model vs. Neural IR Models

Dataset	BM25	DSSM	CDSSM	DRMM	MP	TKL	CNRoc
AP90	0.2738	0.1364	0.1069	0.2777	0.2858	0.2998	<b>0.3363</b>
AP88-89	0.2882	0.1425	0.1137	0.2994	0.3022	0.2988	<b>0.3285</b>
DISK4&5	0.2258	0.1247	0.0970	0.2558	0.2532	0.2532	<b>0.2629</b>
WT2G	0.3192	0.1649	0.1399	0.3237	0.3293	0.3347	<b>0.3403</b>
WT10G	0.2050	0.1136	0.1039	0.2155	0.2256	0.2309	<b>0.2433</b>
GOV2	0.3035	0.1658	0.1382	0.3283	0.3155	0.3369	<b>0.3509</b>

**Note:** **Bold** values indicate optimal results for each dataset.

Table 3.7: Evaluation of P@10 Metrics: CNRoc Model vs. Neural IR Models

Dataset	BM25	DSSM	CDSSM	DRMM	MP	TKL	CNRoc
AP90	0.4468	0.2334	0.2135	0.4595	0.4679	0.4721	<b>0.4940</b>
AP88-89	0.4531	0.2417	0.2243	0.4642	0.4686	0.4768	<b>0.4750</b>
DISK4&5	0.4313	0.2251	0.2058	0.4459	0.4402	0.4525	<b>0.4526</b>
WT2G	0.4760	0.2399	0.2520	0.4901	0.4956	0.5097	<b>0.5248</b>
WT10G	0.3061	0.1579	0.1343	0.3283	0.3464	0.3585	<b>0.3630</b>
GOV2	0.5620	0.2764	0.2486	0.5837	0.5994	0.5982	<b>0.6075</b>

**Note:** **Bold** values indicate optimal results for each dataset.

Table 3.8: Evaluation of NDCG Metrics: CNRoc Model vs. Neural IR Models

Dataset	BM25	DSSM	CDSSM	DRMM	MP	TKL	CNRoc
AP90	0.6579	0.3342	0.2896	0.6682	0.6732	0.6779	<b>0.6837</b>
AP88-89	0.6709	0.3411	0.2637	0.6693	0.6734	0.6828	<b>0.6859</b>
DISK4&5	0.6678	0.3479	0.3196	0.6584	0.6685	0.6612	<b>0.6691</b>
WT2G	0.7058	0.3624	0.3523	0.6943	0.7053	0.6972	<b>0.7170</b>
WT10G	0.5819	0.2966	0.2784	0.6022	0.6038	0.6037	<b>0.6171</b>
GOV2	0.7064	0.3295	0.3258	0.7337	0.7363	0.7454	<b>0.7534</b>

**Note:** **Bold** values indicate optimal results for each dataset.

Table 3.9: Evaluation of MRR Metrics: CNRoc Model vs. Neural IR Models

Dataset	BM25	DSSM	CDSSM	DRMM	MP	TKL	CNRoc
AP90	0.5521	0.3744	0.3525	0.5824	0.5758	0.5954	<b>0.6421</b>
AP88-89	0.5378	0.3598	0.3392	0.5436	0.5583	0.5604	<b>0.5642</b>
DISK4&5	0.5608	0.3352	0.2884	0.5786	0.5885	0.5835	<b>0.6078</b>
WT2G	0.6623	0.3738	0.3393	0.6783	0.6847	0.6994	<b>0.7163</b>
WT10G	0.5191	0.2946	0.2658	0.5239	0.5184	0.5283	<b>0.5374</b>
GOV2	0.7073	0.3553	0.3736	0.7392	0.7424	0.7307	<b>0.7468</b>

**Note:** **Bold** values indicate optimal results for each dataset.

As presented in Tables 3.6-3.9, the proposed CNRoc model consistently surpasses other neural information retrieval models across a wide range of datasets. By utilizing ConceptNet, the query expansion method integrates both conceptual knowledge and semantic attributes of individual terms, effectively evaluating their significance beyond contextual dependencies. Unlike the CNRoc framework, models such as **DSSM**, **CDSSM**, **DRMM**, and **Match-Pyramid** rely on recurrent or convolutional neural networks for text representation during training, which makes their performance heavily reliant on contextual patterns.

Additionally, our approach demonstrates notable enhancements in **MAP**, **P@10**, **NDCG**, and **MRR** metrics when compared to the transformer-based **TKL** model. These performance improvements are primarily attributed to the ConceptNet-driven query expansion mechanism, which optimizes search results by incorporating richer semantic layers and refining pseudo-relevant documents. In contrast, the **TKL** model emphasizes limited interactions between queries and document segments, without leveraging conceptual or semantic augmentation to improve query expansion.

### 3.5.4 Hyperparameter Impact Analysis

In Equation 3.3, the hyperparameter  $\delta$  plays a pivotal role in CNRoc by controlling the influence of query expansion terms derived from ConceptNet. It adjusts the balance between

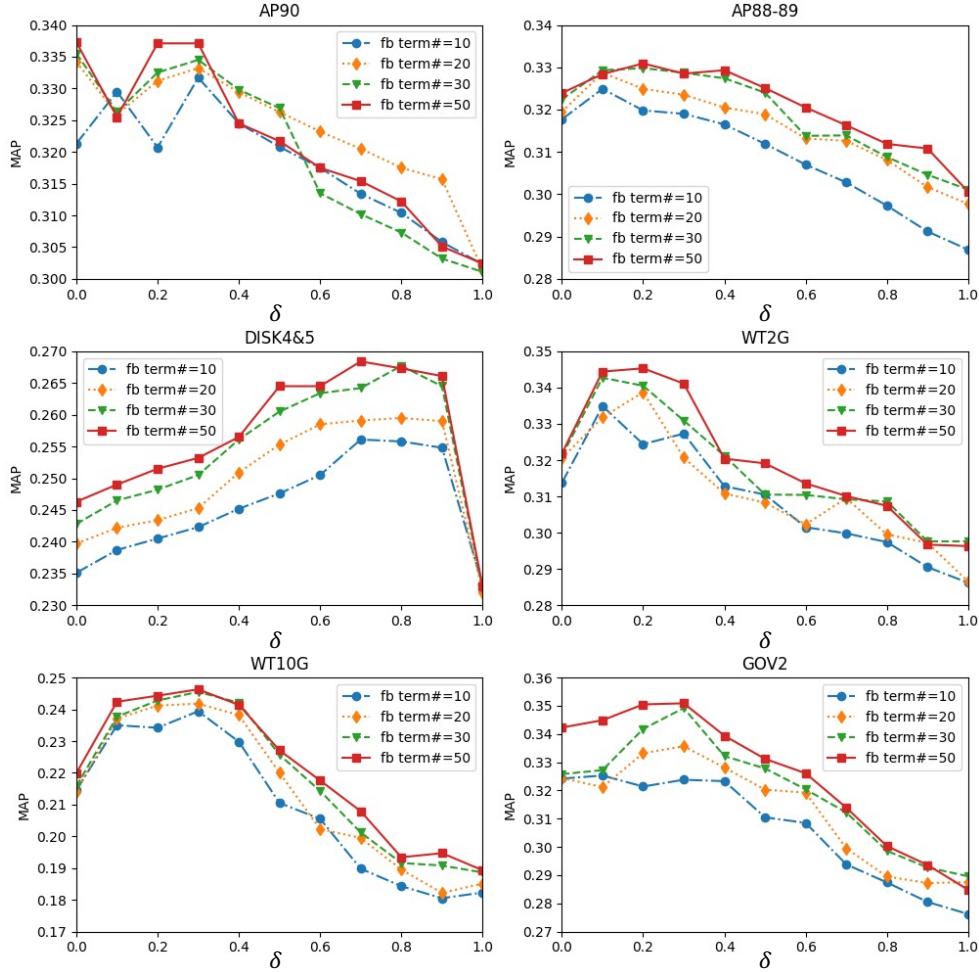


Figure 3.4: Sensitivity of CNRoc

term relevance and semantic enrichment. Given its direct impact on model performance, a comprehensive sensitivity analysis is conducted. As  $\delta$  increases, it incorporates more extensive semantic information and conceptual knowledge into the retrieval process. To evaluate this parameter systematically,  $\delta$  is incremented from 0 to 1 in steps of 0.05 during the experiments. For varying  $\delta$  values, MAP scores are computed using half of the queries (when the total number of queries is odd), with results visualized in Figure 3.4. A fixed  $\delta$  value of 0.8 is also used for certain comparisons, and MAP performance across six TREC datasets is reported in the figure. Here,  $|T_f|$  represents the count of query expansion terms.

As shown in Figure 3.4, the AP90, AP88-89, and WT2G datasets achieve their highest MAP scores when  $\delta$  is within the range of 0.1 to 0.3. For DISK4&5, optimal MAP values

are observed when  $\delta$  ranges between 0.6 and 0.8. Similarly, the WT10G and GOV2 datasets exhibit peak performance for  $\delta$  values between 0.2 and 0.4. Based on these patterns, setting  $\delta$  between 0.1 and 0.4 is recommended for most datasets.

Furthermore, the influence of  $|T_f|$ , which denotes the number of query expansion terms, is analyzed. Figure 3.4 highlights that increasing  $|T_f|$  generally improves MAP scores across all datasets. The most consistent performance gains are achieved at  $|T_f| = 50$ . Thus, a value of  $|T_f| = 50$  is suggested for optimal query expansion in diverse scenarios.

Table 3.10: Case Study of CNRoc

Model	Term	&	Weight					
Rocchio	airbus	northwest	loan	body	subsidies	hill	ec	unfair
	1.0022	0.4873	0.4407	0.3467	0.2948	0.2696	0.2673	0.2356
CNRoc	airbus	subsidies	plane	<b>aircraft</b>	jet	<b>consortium</b>	<b>airline</b>	<b>european</b>
	1.6500	1.0000	0.3339	<b>0.3132</b>	0.2563	<b>0.2355</b>	<b>0.2354</b>	<b>0.2067</b>

**Note:** Bold indicates new terms obtained by our proposed method.

### 3.5.5 Case Study

Table 3.10 presents a comparison of the first eight expansion terms and their corresponding weights derived from both the CNRoc and Rocchio models. For this evaluation, we used the query “Airbus Subsidies”. The terms highlighted in **bold** represent the new expansion terms identified by CNRoc that were not selected by Rocchio.

As shown in the table, both CNRoc and Rocchio successfully identify some relevant terms such as “airbus” and “subsidies”, which are directly related to the query topic “Airbus Subsidies”. Both models also include terms like “jet” and “northwest”, which are useful, but less central to the specific context of “Airbus Subsidies”.

However, there are important differences in the term selection between the two models. Rocchio, for example, selects terms like “loan”, “body”, and “hill”, which are not as strongly related to the query topic and do not contribute much to the understanding of “Airbus

Subsidies”. In contrast, CNRoc effectively identifies additional terms that are much more relevant to the query. These include “aircraft”, “consortium”, “airline”, and “european”, which are directly related to the aerospace industry and more specific to the context of “Airbus Subsidies”.

Furthermore, CNRoc captures terms such as “plane” and “jet”, which are important in the context of the aerospace industry but were not selected by Rocchio. The higher weight given to “aircraft” and “consortium” in CNRoc suggests that it is able to identify terms that are conceptually and semantically aligned with the query topic, outperforming Rocchio in selecting more relevant terms.

This analysis demonstrates that CNRoc has an advantage over Rocchio by utilizing both semantic and conceptual relationships in the query expansion process. By selecting more contextually relevant and conceptually aligned terms, CNRoc provides a better and more accurate query expansion, improving the effectiveness of the retrieval process.

## 3.6 Chapter Summary

This chapter proposed the CNRoc framework, which leverages ConceptNet to enrich query expansion by integrating semantic information into the Pseudo-Relevance Feedback process. The model demonstrated superior performance compared to baseline and state-of-the-art methods across multiple datasets and metrics.

However, the CNRoc framework has certain limitations. Firstly, its reliance on ConceptNet means that the quality of query expansion terms heavily depends on the completeness and accuracy of the external semantic network, which may not cover domain-specific terms or emerging concepts. Secondly, the noise introduced by irrelevant terms in pseudo-relevance documents, even after denoising, can still affect retrieval performance.

To address these limitations in future work, improvements could focus on:

- Enhancing the denoising mechanism by incorporating dynamic weighting schemes or

domain-specific knowledge.

- Expanding the framework to adaptively leverage multiple semantic networks or knowledge bases for broader coverage.
- Introducing a reinforcement learning-based feedback loop [98] to iteratively refine the selection of query expansion terms.
- Reducing computational costs by optimizing the integration of semantic information within the PRF process.

These improvements aim to further enhance the robustness and scalability of the framework, making it applicable to a wider range of information retrieval scenarios.

# Chapter 4

## Document Matching Optimization via Contrastive Learning

### 4.1 Chapter Introduction

Pseudo-Relevance Feedback is a technique for refining queries based on the assumption that the highest-ranked documents in the initial search results are relevant. The system extracts useful expansion terms from these presumed relevant documents and incorporates them into the original query to better fulfill the user’s information requirements. This approach does not rely on direct feedback from the user, which is why it’s called “Pseudo-Relevance Feedback.”

The success of PRF significantly relies on the quality of the chosen pseudo-relevance documents. If these documents lack accuracy or fail to match the user’s query intent, the extracted query expansion terms may not reflect the user’s actual needs, resulting in poorer quality search results. For instance, consider a user searching for “Recent Advancements in Solar Cell Technology.” If the system incorrectly selects pseudo-relevance documents focused on “Traditional Battery Materials,” the expansion terms extracted, such as “Lead-Acid Battery” or “Nickel-Cadmium Battery” would relate to “Batteries” but not to “Solar Cells”.

This mismatch would cause many irrelevant results to appear, significantly lowering precision.

Moreover, poorly selected pseudo-relevance documents can negatively impact recall. The system may overlook highly relevant documents related to “Solar Cells” and focus instead on irrelevant areas, causing important information to be missed. Thus, the accurate selection of pseudo-relevance documents is critical in Pseudo-Relevance Feedback. High-quality query expansion terms can only enhance the effectiveness of the retrieval system if the initial documents closely match the user’s actual needs, thereby boosting both precision and recall.

This chapter introduces a probabilistic framework inspired by the classical Rocchio model. This framework integrates the weights of both weak and strong relevance signals, assisting in the selection of documents pertinent to the query topic and enhancing retrieval performance. First, the proposed pipeline extracts the weak signal weight of terms using the BM25 method, followed by obtaining the strong signal weight of terms using the SimCSE model [99]. Finally, the weak and strong signal weights are combined, with their balance adjusted by tuning specific factors. The resulting refined query is then used for second-round retrieval.

Through a series of experiments on six official TREC datasets, the results across various evaluation metrics demonstrate significant improvements of our proposed model over the corresponding baseline models. This approach provides scholars with a novel way to introduce Contrastive Learning mechanisms into the study of PRF.

## 4.2 Contrastive Learning

**SimCLR(Simple Framework for Contrastive Learning of Visual Representations)** [100] is a self-supervised learning framework developed by Google, designed to learn effective visual representations without the need for labeled data. SimCLR employs Contrastive Learning, aiming to maximize the similarity between augmented versions of the same image (positive pairs) while minimizing the similarity between different images (negative pairs). Different image construction methods are shown in the Figure 4.1.

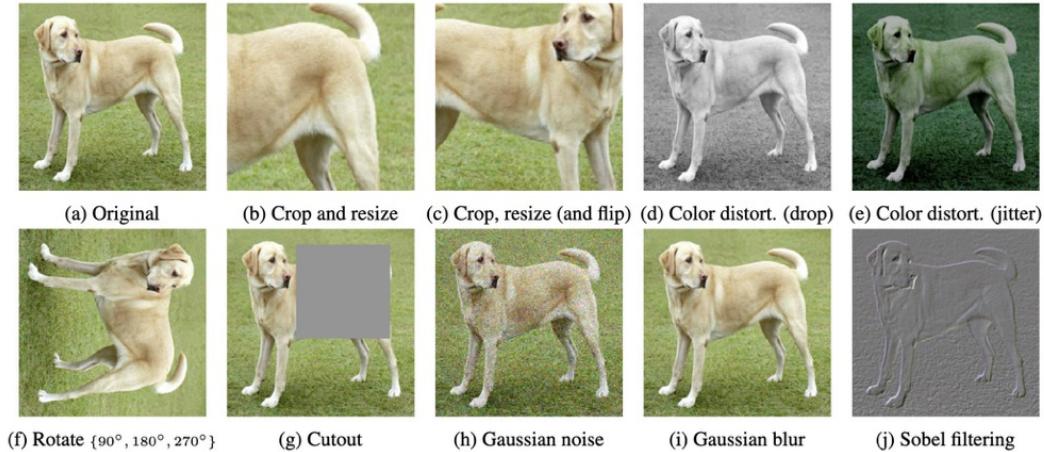


Figure 4.1: SimCLR Data Augmentation

The framework consists of three main components:

- **Data Augmentation:** Each input image is randomly augmented twice, creating two different views of the same image, forming the positive pair. Other images in the batch are considered negative pairs.
- **Encoder:** The augmented images are passed through a shared neural network (typically a ResNet) to extract feature representations.
- **Projection Head:** A small multi-layer perceptron (MLP) is used to map the feature representations into a lower-dimensional space more suited for computing the contrastive loss.

SimCLR’s simplicity and its ability to learn robust visual representations have made it highly effective for tasks like image classification and object detection, even without labeled datasets. Its success also inspired the use of Contrastive Learning in other domains, including NLP.

**SimCSE (Simple Contrastive Sentence Embeddings)** [99] extends the principles of SimCLR to the NLP domain. As shown in Figure 4.2, SimCSE is designed to generate high-quality sentence embeddings by leveraging Contrastive Learning in a similar manner.

It creates positive pairs by applying dropout as a form of data augmentation to the same sentence and considers other sentences within the batch as negative pairs.

SimCSE has been particularly successful in improving tasks like semantic similarity detection, sentence clustering, and text classification. By utilizing unsupervised and supervised variants, SimCSE ensures that sentence-level embeddings capture meaningful semantic information, improving the performance of various downstream NLP tasks.

The application of Contrastive Learning to PRF and IR is still an emerging area of research. Some recent studies have begun exploring this direction and applied Contrastive Learning to re-rank documents in a PRF framework, demonstrating improvements in retrieval performance. However, challenges remain in effectively integrating Contrastive Learning with traditional IR techniques, particularly in terms of balancing the influence of learned representations with established retrieval models like BM25.

Therefore, we propose a probabilistic framework, CLRoc, based on the classical Rocchio model. This framework combines the weights of weak and strong relevance signals to better select documents relevant to the query topic, thereby improving retrieval performance. Our approach introduces a novel method for effectively integrating Contrastive Learning with traditional IR techniques.

## **4.3 CLRoc: Document Matching Optimization via Contrastive Learning**

### **4.3.1 Selection of Query Expansion Terms**

#### **Weak Signal Weight via BM25**

The combination of BM25 and the Rocchio model has been widely recognized as a common baseline in the field of information retrieval [101, 102]. Despite the Rocchio model being proposed in 1971, it remains one of the most effective and popular approaches for PRF. BM25,

a Bag-of-Words scoring function for retrieval, computes term frequency scores in the pseudo-relevant documents, assigning the frequency weight of term  $t$  as follows:  $W_{weak}$ . BM25 has demonstrated robust and excellent performance across various tasks [103]. However, it has limitations in effectively capturing query intent, sentence semantic understanding, and retrieving related documents. Therefore, we consider term frequency as a weak signal, and the score derived from BM25 serves as the weight for the weak signal. The weak signal is measured for a term as follows, BM25+Rocchio uses this function to select query expansion terms:

$$W_{weak} = \log \frac{N_{doc} - N_{doc}^t + 0.5}{N_{doc}^t + 0.5} \times \sum_{i=1}^{N_{doc}^t} \frac{(k_1 + 1) \times tf(t, d_i)}{K + tf(t, d_i)} \times \frac{(k_3 + 1) \times qtf}{k_3 \times qtf} \quad (4.1)$$

The variables in Equation 4.1 are defined as follows:  $N_{doc}$  represents the total number of indexed documents in the datasets, while  $N'_{doc}$  denotes the number of feedback documents obtained from the initial retrieval stage.  $k_1, k_3$ : the tuning constants in this function;  $K$ : it equals  $k_1 \times ((1 - b) + b \times \frac{dl}{avdl})$ , where  $dl$  represents the length of the document and  $avdl$  is the average length of the document;  $N_{doc}^t$ : the document number in which term  $t$  exists;  $d_i$ : the  $i_{th}$  document;  $f(t, d_i)$ : denotes the term frequency within document  $d_i$ ;  $qtf$ : signifies the term frequency found in the query.

### Strong Signal Weight via Contrastive Learning

As shown in Figure 4.2 Contrastive Learning [104–107] is a type of unsupervised learning, where a pile of data, without labels, is given to learning a feature representation on its own. For any data point  $x$ , the Contrastive Learning method aims to learn an encoder  $f$ :

$$score(f(x), f(x^+)) \gg score(f(x), f(x^-)) \quad (4.2)$$

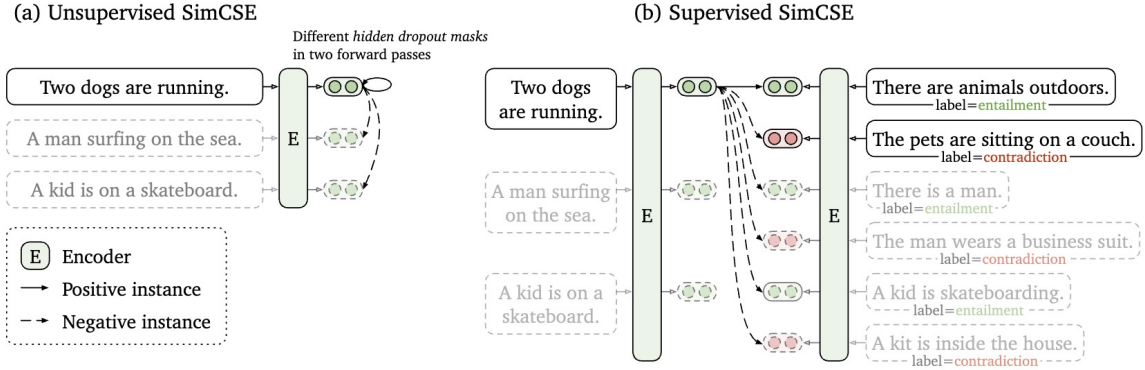


Figure 4.2: SimCSE Framework

Where  $x^+$  is the data points that are similar to  $x$  and are called positive samples;  $x^-$  is the data points that are not similar to  $x$  and are called negative samples; and  $score$  is a function that measures the similarity between two features [99]. To optimize this property, SimCSE is a method to learn sentence embeddings, and it constructs a SoftMax classifier that correctly classifies both positive and negative samples. This will encourage the score function to assign larger values to the positive samples and smaller values to the negative samples:

$$L_N = -E_X \left[ \log \frac{\exp(f(x)^T f(x^+))}{\exp(f(x)^T f(x^+)) + \sum_{i=1}^{N-1} \exp(f(x)^T f(x_i))} \right] \quad (4.3)$$

There are three stages in obtaining the strong signal weight as shown in Figure 4.3.

First, we perform the first round in PRF by BM25 to obtain the top-K pseudo-relevance documents. These top-K pseudo documents could be as relevance feedback signals. Second, we use query  $Q$  and sentence  $S$  in the form of vectors as input to the BERT model, to calculate the degree of relevance of both. Meanwhile, we splice the query with the sentences in the pseudo-relevance documents to form a positive instance and splice the query with the sentences in the non-pseudo documents to form a negative instance and use them for contrast learning by SimCSE, where the core idea of Contrastive Learning is to get closer and closer to the similar instance and further and further from the dissimilar ones.

This process makes it possible for us to subsequently compare the query with a random

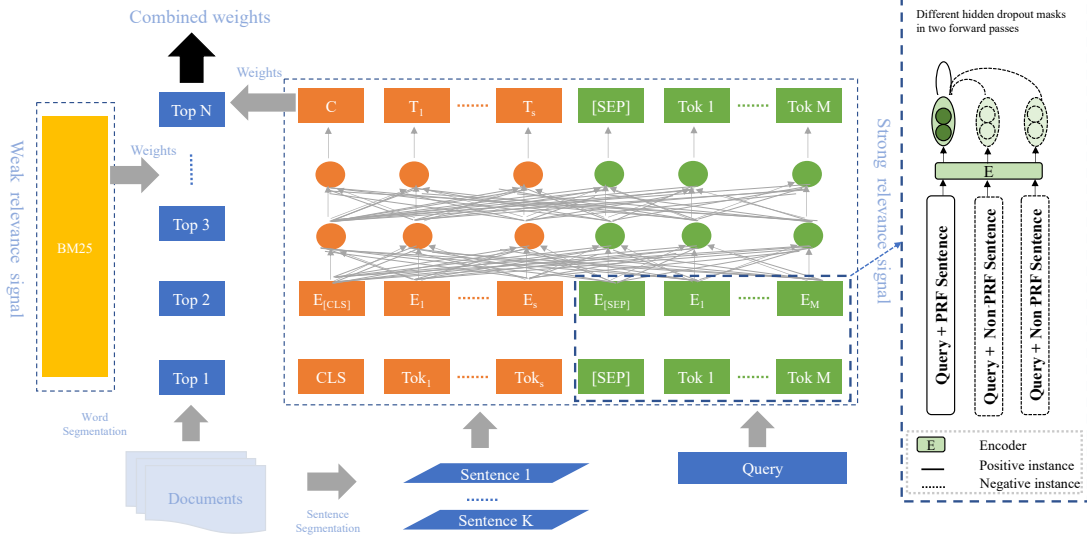


Figure 4.3: Framework for obtaining the strong relevance signal weight based on Contrast Learning

sentence. When performing the calculation, if the sentence has higher relevance to the query, then it will get a higher weight. This mechanism is that the relevance feedback signal is amplified by Contrastive Learning, allowing the model to identify similar utterances more deeply and accurately. Third, since we need to process query expansion, we assign the relevance weight obtained by sentence to each term in this sentence, and the specific score assignment formula is as follows, compared with the weak signal obtained by BM25, this is a strong signal weight calculated by effective cross-calculating.

According to the above description, given a query  $Q$  and document  $d_i$  originating from  $N'_{doc}$ , it is possible to obtain a strong signal correlation between the sentences  $S_{i,j}$  (denoted as  $S_{i,1}, S_{i,2}, \dots, S_{i,n}$ ) from  $d_i$ , and each sentence  $S_{i,n}$ . The query  $Q$  is computed and represented as  $CL(S_{i,j}, Q)$ . Next, the strong signal weight for the term is derived through Equation 4.4:

$$W_{strong} = \sum_{i=1}^{N'_{doc}} \sum_{j \in n(d_i, t)} CL(S_{i,j}, Q) \quad (4.4)$$

Where  $n(d_i, t)$  is the set of sentences in document  $d_i$  where the term  $t$  exists, and  $N'_{doc}$

represents the number of documents returned from the first round of retrieval.

### Combination of Weak and Strong Signal Weights

Our method selects query expansion terms by considering both term frequency and the interaction encoding information between the query and PRF signal in the sentences of the pseudo-relevance document. This method connects query intent with document representation. To compute the weak relevance signal weights, we use the BM25 algorithm. The strong relevance signal weights are obtained through cross-calculation using the SimCSE framework, which combines Contrastive Learning and the Transformer-based model BERT. These two signal weights are linearly fused to derive a new weight, which is assigned to the candidate query expansion terms using Equation 4.5:

$$W_{\text{new}} = x \cdot W_{\text{weak}} + (1 - x) \cdot W_{\text{strong}} \quad (4.5)$$

Here,  $x$  is a constant that varies between 0 and 1.0, serving as a tuning parameter that adjusts the contributions of weak and strong signal weights. We sort by  $W_{\text{new}}$  in descending order, select top-K as query expansion terms,  $Q_{\text{exp}}^{\text{cl}}$  is the query expansion representation in the CLRoc framework.

### 4.3.2 Adapting the Rocchio Model

#### Enhanced Query Representation

We combine the original query with query expansion terms selected based on strong and weak signals to obtain a new query representation. A new query  $Q_{\text{new}}^{\text{cl}}$  is obtained from the following Equation 4.6:

$$Q_{\text{new}}^{\text{cl}} = (1 - y) \cdot Q_{\text{org}}^{\text{cl}} + y \cdot Q_{\text{exp}}^{\text{cl}} \quad (4.6)$$

Here,  $Q_{\text{org}}^{\text{cl}}$  denotes the original query, while  $Q_{\text{exp}}^{\text{cl}}$  represents the expanded query terms.

The variable  $y$  acts as a constant tuning parameter, adjusting the influence of both the original query and the PRF signal. It is essential for us to evaluate the score for each document based on the query vector representation  $Q_{\text{new}}^{\text{cl}}$ .

## Second Retrieval with BM25

In the field of Information Retrieval, there is no doubt that BM25 is the most robust and effective method. Following our implementation of the Rocchio model, we utilize BM25 for the second-stage retrieval process. Therefore, the updated equation can be expressed as:

$$w(Q_{\text{new}}^{\text{cl}}, D_r) = \sum_{i=1}^{N'} \frac{f(Q_i, D_r) \cdot (k_1 + 1) \cdot f_i}{f_i + k_1 \cdot (1 - b + b \cdot \frac{dl}{\text{avg}D_r})} \quad (4.7)$$

## 4.4 Experimental Results and Analysis

### 4.4.1 Experimental Datasets and Analytical Metrics

**Experimental Datasets** To comprehensively evaluate our proposed model, we select the same datasets as in Section 3.4.1 of Chapter 3. Now we provide a brief review, as shown in the Table 4.1, we utilize the datasets provided by TREC, which adhere to international standards. The chosen datasets include AP90, AP88-89, DISK4&5, GOV2WT2G and WT10G, which vary significantly in size and type, allowing for a thorough assessment of our model’s effectiveness.

The diversity in size and type among these datasets ensures a comprehensive evaluation of our model. By including a mix of news articles, web documents, and governmental content, we can test the robustness, scalability, and generalizability of our information retrieval system. Each dataset poses unique challenges and opportunities, allowing us to fine-tune our model for optimal performance across various types of data.

By leveraging these varied datasets, we aim to validate the effectiveness and robustness

Table 4.1: Validation Datasets for the CLRoc Model

Collection	Size	Queries	# of Queries	# of Docs
AP90	0.23 Gb	51–100	50	78,321
AP88-89	0.50 Gb	51–100	50	164,597
DISK4&5	1.86 Gb	301–450	150	528,155
WT2G	2.14 Gb	401–450	50	247,491
WT10G	10 Gb	451–550	100	1,692,096
GOV2	426 Gb	701–850	150	25,178,548

of our proposed information retrieval model across a broad spectrum of real-world scenarios.

**Analytical Metrics** We evaluate our IR system’s performance using four established metrics: MAP, P@10, NDCG, and MRR. Each metric offers distinct insights into various aspects of retrieval effectiveness.

These metrics together create a strong foundation for assessing the performance of IR systems, targeting various elements of retrieval efficacy and user contentment. Utilizing these metrics, researchers and practitioners can develop a thorough insight into how effectively their systems fulfill users’ requirements.

#### 4.4.2 Baseline Model

The baseline model is essential for assessing the performance of our proposed method. Initially, our CLRoc approach introduces a probabilistic framework derived from the classical Rocchio model, which integrates the weights of both weak and strong relevance signals. This integration aids in selecting documents pertinent to the query topic and enhances retrieval performance. To validate our method, a comparison with the original Rocchio model is necessary. Furthermore, it’s important to also compare our model with improved versions based on Rocchio, such as PRoc2 and KRoc, to thoroughly confirm the methods’ effectiveness.

### 4.4.3 Hyperparameter Settings

In our method, the number of feedback documents was set to 50 to achieve an optimal balance between efficiency and accuracy. This configuration was consistently applied in all subsequent experiments. Additionally, the number of query expansion candidates for semantic enhancement was varied as  $|T_f| \in \{10, 20, 30, 50\}$ . To evaluate performance, we measured **MAP** and **NDCG** scores using the top 1000 retrieved documents, alongside **P@10** scores from these results.

The experiments were conducted on **TREC datasets** and their associated topics, ensuring consistent metrics and parameter settings for fair comparison. Queries that were not processed during the indexing or querying stages were excluded. Furthermore, Porter’s English Stemmer [48] was used to process terms in the datasets, removing 418 stop words based on the standard InQuery List [92].

We utilize Contrastive Learning to optimize document matching; the initial retrieval for weak signals was performed using BM25 with the same settings as that of CNRoc in the Section 3.4.3 of Chapter 3, and the range of the adjustable parameters  $x, y$  to  $[0, 1]$ . The BERT model was fine-tuned with a learning rate of  $2e-5$ , a batch size of 32, and trained for 16 epochs. SimCSE was used to generate embeddings, with the margin set to 0.5 in the contrastive loss function. Positive samples were selected from top-ranked pseudo-relevance documents, while negative samples were chosen from irrelevant documents. The selection was based on a similarity threshold, ensuring that positive samples had high semantic similarity and negative samples were sufficiently distinct. The final retrieval used a linear combination of weak and strong signal weights, with the weighting factor empirically set at 0.6 for strong signals, balancing the contribution of semantic content and term frequency.

## 4.5 Experimental Setup

### 4.5.1 Validation Against Baseline and Advanced Models

As we mentioned above, our method mainly uses Contrastive Learning to obtain strong signal weights and fuse them to the Rocchio model to achieve the improvement of the final retrieval results, so comparison with the basic Rocchio model is primary and necessary. In addition, to verify the effectiveness of our method, we also compared the results with the advanced models (PRoc2 and KRoc), which are based on the Rocchio model for improvement.

Since the average value is better able to test the overall level of the model[6, 18]. Table 4.2 and Table 4.3 below show the comparison of our model with values of MAP and P@10 for the baseline model and the advanced models on different TREC datasets, and the average value is calculated based on the results under 10,20,30,50 different query expansion terms. All the results in these tables are obtained by performing 2-fold cross-validation. The bolded value in this table indicates the best on that dataset.

As shown in Table 4.2, with the baseline model and the advanced models, the MAP value of our model increases with the feedback terms number. The “Avg” row shows the significant improvement of our model compared to the Rocchio model, specifically, not only that, the CLRoc model has a great improvement on each test dataset, achieving 19.78%, 11.52%, 15.57%, 7.28%, 16.85% and 7.53% improvement on the AP90, AP88-89, Disk4&5, WT2G and GOV2 collections, respectively. But we also achieved the best value on the test set compared to all three models.

As shown in Table 4.3, the values of P@10 of our model are compared with the baseline model as well as the advanced models. Similar to the results presented in MAP, our model still achieves a significant boost and the best results on each dataset compared to the baseline model and the advanced models, with the highest percentage of improvement at 17.85% on the dataset WT10G.

Table 4.2: Evaluation of MAP Metrics: CLRoc Model vs. Baseline and Strong Baseline Models

Model	Tf	AP90	AP88-89	DISK4&5	WT2G	WT10G	GOV2
Rocchio	10	0.2858	0.2908	0.2307	0.3249	0.2064	0.3232
	20	0.2884	0.2938	0.2328	0.3253	0.2076	0.3263
	30	0.2899	0.2950	0.2337	0.3255	0.2077	0.3276
	50	0.2926	0.2965	0.2349	0.3258	0.2088	0.3296
	Avg	0.2892	0.2940	0.2330	0.3254	0.2076	0.3267
PRoc2	10	0.3031	0.3128	0.2506	0.3328	0.2212	0.3264
	20	0.3132	0.3176	0.2572	0.3333	0.2258	0.3314
	30	0.3204	0.3200	0.2588	0.3406	0.2248	0.3364
	50	0.3243	0.3212	0.2618	0.3344	0.2226	0.3401
	Avg	0.3153	0.3179	0.2571	0.3353	0.2236	0.3336
KRoc	10	0.3287	0.3131	0.2563	0.3338	0.2174	0.3303
	20	0.3367	0.3172	0.2621	0.3357	0.2166	0.3403
	30	0.3357	0.3214	0.2657	0.3348	0.2162	0.3416
	50	0.3383	0.3222	0.2654	0.3451	0.2125	0.3419
	Avg	0.3349	0.3185	0.2624	0.3374	0.2157	0.3385
CLRoc	10	0.3449*	0.3243*	0.2622*	0.3445*	0.2381*	0.3472*
		(+20.68%)	(+11.52%)	(+13.65%)	(+6.03%)	(+15.36%)	(+7.43%)
	20	0.3461*	0.3272*	0.2651*	0.3475*	0.2413*	0.3487*
		(+20.01%)	(+11.37%)	(+13.87%)	(+6.82%)	(+16.23%)	(+6.86%)
	30	0.3468*	0.3290*	0.2741*	0.3491*	0.2452*	0.3536*
		(+19.63%)	(+11.53%)	(+17.29%)	(+7.25%)	(+18.05%)	(+7.94%)
	50	0.3478*	0.3310*	0.2757*	0.3552*	0.2457*	0.3557*
	(+18.87%)	(+11.64%)	(+17.37%)	(+9.02%)	(+17.67%)	(+7.92%)	
Avg	<b>0.3464*</b>	<b>0.3279*</b>	<b>0.2693*</b>	<b>0.3490*</b>	<b>0.2425*</b>	<b>0.3513*</b>	
	(+19.78%)	(+11.52%)	(+15.57%)	(+7.28%)	(+16.85%)	(+7.53%)	

**Note:** **Bold** values indicate optimal results for each dataset.

Table 4.3: Evaluation of P@10 Metrics: CLRoc Model vs. Baseline and Strong Baseline Models

Model	Tf	AP90	AP88-89	DISK4&5	WT2G	WT10G	GOV2
Rocchio	10	0.4468	0.4571	0.4247	0.4920	0.3092	0.5878
	20	0.4426	0.4571	0.4240	0.4940	0.3071	0.6020
	30	0.4426	0.4551	0.4233	0.4940	0.3082	0.6020
	50	0.4447	0.4571	0.4220	0.4900	0.3082	0.6061
	Avg	0.4442	0.4566	0.4235	0.4925	0.3082	0.5995
PRoc2	10	0.4553	0.4633	0.4353	0.5080	0.3286	0.5822
	20	0.4553	0.4755	0.4393	0.5242	0.3245	0.5914
	30	0.4638	0.4735	0.4380	0.5201	0.3337	0.5837
	50	0.4617	0.4653	0.4407	0.5223	0.3235	0.5852
	Avg	0.4590	0.4694	0.4383	0.5187	0.3276	0.5856
KRoc	10	0.4660	0.4653	0.4313	0.5120	0.3133	0.5878
	20	0.4681	0.4755	0.4387	0.5141	0.3133	0.5939
	30	0.4681	0.4673	0.4380	0.5141	0.3133	0.5898
	50	0.4638	0.4714	0.4393	0.5221	0.3112	0.5981
	Avg	0.4665	0.4699	0.4368	0.5156	0.3128	0.5924
CLRoc	10	0.4882*	0.4757*	0.4452*	0.5255*	0.3617*	0.6177*
		(+9.27%)	(+4.07%)	(+4.83%)	(+6.81%)	(+16.98%)	(+5.09%)
	20	0.4933*	0.4761*	0.4503*	0.5257*	0.3645*	0.6281*
		(+11.46%)	(+4.16%)	(+6.20%)	(+6.42%)	(+18.69%)	(+4.34%)
	30	0.4941*	0.4785*	0.4515*	0.5265*	0.3636*	0.6468*
		(+11.64%)	(+5.14%)	(+6.66%)	(+6.58%)	(+17.98%)	(+7.44%)
	50	0.4952*	0.4795*	0.4521*	0.5279*	0.3631*	0.6489*
	(+11.36%)	(+4.90%)	(+7.13%)	(+7.73%)	(+17.81%)	(+7.06%)	
Avg	<b>0.4927*</b>	<b>0.4774*</b>	<b>0.4497*</b>	<b>0.5264*</b>	<b>0.3632*</b>	<b>0.6353*</b>	
	(+10.92%)	(+4.57%)	(+6.20%)	(+6.88%)	(+17.85%)	(+5.98%)	

**Note:** **Bold** values indicate optimal results for each dataset.

Table 4.4: Evaluation of NDCG and MRR Metrics: CLRoc Model vs. Baseline and Strong Baseline Models

Collection	Metric	Rocchio	PRoc2	KRoc	CLRoc
AP90	NDCG	0.6682	0.6689	0.6810	<b>0.6848*</b>
	MRR	0.6106	0.5777	0.6259	<b>0.6442*</b>
AP88-89	NDCG	0.6745	0.6716	0.6812	<b>0.6823*</b>
	MRR	0.5337	0.5415	0.5534	<b>0.5642*</b>
DISK4&5	NDCG	0.6564	0.6580	0.6701	<b>0.6680*</b>
	MRR	0.5903	0.5989	0.6037	<b>0.6089*</b>
WT2G	NDCG	0.7085	0.7093	0.7138	<b>0.7170*</b>
	MRR	0.6830	0.6907	0.6973	<b>0.7164*</b>
WT10G	NDCG	0.5778	0.5838	0.5877	<b>0.6138*</b>
	MRR	0.5207	0.5194	0.5286	<b>0.5342*</b>
GOV2	NDCG	0.7476	0.7534	0.7541	<b>0.7559*</b>
	MRR	0.6544	0.6569	0.7014	<b>0.7368*</b>

**Note:** **Bold** values indicate optimal results for each dataset.

Nowadays, MRR and NDCG have become popular evaluation metrics in the field of IR that further analyze the benefits of our method. In table 4.4, Superscript letters denote statistically significant enhancements compared to the relevant model. (Wilcoxon signed-rank tests,  $p < 0.05$ ), We present the model’s results alongside three baseline models for MRR and NDCG. The findings show that our model consistently outperforms the baselines in terms of NDCG and MRR across most scenarios datasets, Our method chooses query expansion terms using two metrics: weak and strong feedback signals, effectively leveraging Contrastive Learning to enhance strong signals. This approach enables our query expansion terms to consider term frequency, PRF signals, semantic data, and other interaction details.

#### 4.5.2 Validation Against the SOTA PRF Models

To evaluate the performance of our model, we present a comparison of MAP scores with advanced query expansion methods across six TREC datasets. The **MRF model** derives QE terms by analyzing inter-term dependencies. **IF&FB** incorporates relationships derived from

Table 4.5: Evaluation of MAP Metrics: CLRoc Model vs. SOTA PRF Models

Model	AP90	AP88-89	DISK4&5	WT2G	WT10G	GOV2
BM25	0.2738	0.2882	0.2258	0.3192	0.2050	0.3035
MRF	0.2920	0.3088	0.2579	0.3380	0.2214	0.3357
	(+6.65%)	(+7.15%)	(+14.22%)	(+5.89%)	(+8.00%)	(+10.61%)
HAL	0.2810	0.2916	0.2363	0.3285	0.2158	0.3228
	(+2.63%)	(+1.18%)	(+4.65%)	(+2.91%)	(+5.27%)	(+6.36%)
IF&FB	0.2886	0.2971	0.2565	0.3301	0.2180	0.3326
	(+5.41%)	(+3.09%)	(+13.60%)	(+3.41%)	(+6.34%)	(+9.59%)
RM3	0.3082	0.3171	0.2600	0.3326	0.2253	0.3305
	(+12.56%)	(+10.03%)	(+15.15%)	(+4.20%)	(+9.05%)	(+8.90%)
PRoc3	0.3181	0.3179	0.2575	0.3534	0.2256	0.3393
	(+16.18%)	(+10.31%)	(+14.04%)	(+10.71%)	(+10.05%)	(+11.80%)
TF-PRF	0.3074	0.3190	0.2699	0.3448	0.2350	0.3371
	(+12.27%)	(+10.69%)	(+19.53%)	(+8.02%)	(+14.63%)	(+11.07%)
SRoc	0.3271	0.3197	0.2598	0.3444	0.2374	0.3494
	(+19.48%)	(+10.92%)	(+15.04%)	(+7.89%)	(+15.78%)	(+15.11%)
SPRoc2	0.3282	0.3283	0.2642	0.3456	0.2428	0.3412
	(+19.88%)	(+13.91%)	(+16.19%)	(+8.28%)	(+18.45%)	(+12.65%)
CLRoc	<b>0.3478</b>	<b>0.3310</b>	<b>0.2757</b>	<b>0.3552</b>	<b>0.2457</b>	<b>0.3557</b>
	(+27.03%)	(+14.85%)	(+22.10%)	(+11.28%)	(+19.85%)	(+17.20%)

**Note:** **Bold** values indicate optimal results for each dataset.

information flow within the language model framework to select terms for query expansion. The **RM3 model** uses language models to estimate the probability of generating candidate document queries, selecting terms based on their occurrence frequencies. Demonstrating robust performance in various tasks, the **HAL model** identifies expansion terms through multidimensional distance calculations, prioritizing terms with greater semantic proximity to the query. Among probabilistic methods (**PRoc1**, **PRoc2**, and **PRoc3**), **PRoc3** is chosen for its reduced sensitivity to parameter variations. Furthermore, **TF-PRF** selects QE terms by evaluating co-occurrence frequencies and applies normalization similar to PRoc2. Finally, **SRoc** and **SPRoc2** enhance PRF by integrating sentence-level semantics using BERT embeddings.

The experimental configurations for all methods were uniformly maintained to ensure fair comparisons with baseline models such as **BM25**. Table 4.5 highlights the results, with bolded values representing the highest MAP scores for each dataset. A superscript “\*” indicates statistically significant improvements over BM25+Rocchio, validated using the Wilcoxon signed-rank test ( $p < 0.05$ ). The results consistently demonstrate that our proposed model surpasses others in MAP performance. Earlier methods primarily determined term weights based on co-occurrence statistics and local contextual relevance. In contrast, our approach leverages sentence-level semantic information and refines it using contrastive learning techniques. This framework enhances the retrieval process by assigning higher weights to semantically relevant documents, thereby optimizing results from the initial retrieval stage.

For the fairness of the experimental comparison, we strictly control the experimental setting and use consistent parameters throughout the PRF retrieval process. Table 4.5 shows the results, and it can be visually seen that our method is not inferior to the state-of-the-art models on most of the datasets, or even superior to them.

The CLRoc model shows significant enhancements compared to both baseline and advanced models across various datasets, such as AP90, AP88-89, DISK4&5, WT2G, WT10G,

and GOV2. When compared to BM25, CLRoc shows substantial MAP improvements, with gains ranging from 11.28% to 27.03% across these datasets. This highlights the model’s effectiveness in capturing semantic nuances and providing more relevant query expansions, making it a significant advancement over traditional retrieval methods.

In particular, CLRoc outperforms more advanced models such as PRoc3 and SPRoc2, achieving the highest MAP scores across all datasets. For instance, on AP90, CLRoc outperforms SPRoc2 by 6.00%, while on DISK4&5, it achieves a 22.10% improvement over BM25, demonstrating its robustness in handling both generic and domain-specific queries.

Moreover, the model consistently performs well on web-based datasets such as WT2G and WT10G, where it outperforms BM25 by 19.85% on WT10G and achieves marginal gains over strong competitors like SPRoc2. This further underscores CLRoc’s strength in improving recall and precision across different types of collections, making it a highly effective retrieval model for real-world applications.

### 4.5.3 Hyperparameter Impact Analysis

The sensitivity of the hyperparameter  $y$  significantly affects the robustness of our method. Meanwhile,  $x$  acts as a smoothing element that impacts the importance of the expansion terms. Its main role is to regulate the distribution the weak and strong signal weights. To verify its specific impact, we experimented with the values of  $y$  from 0 to 1 interval of 0.05 and we empirically set the  $x$  to 0.8. Figure 4.4 shows the variation of MAP for different alpha values. As can be seen, the CLRoc model performs best on most datasets with hyperparameter  $y$  values from 0.1 to 0.3. Therefore, in practical applications, we recommend setting the  $y$  at 0.1 to 0.3 and the  $x$  at 0.8 to demonstrate the best retrieval results.

Table 4.6: Case Study of CLRoc

CLRoc		Rocchio	
Term	Weight	Term	Weight
merit	1.919750	merit	1.053868
pai	1.494356	pai	1.028808
senior	1.000000	senior	1.013706
chrysler	0.676381	quasar	0.021062
uaw	0.365428	discrim	0.019574
contract	0.343444	award	0.018453
spokesman	0.300804	wellesley	0.016893
base	0.291148	salomon	0.015145
merit	0.278290	card	0.014897
payout	0.250000	court	0.014853
<b>formula</b>	0.250000	bush	0.014321
<b>pay</b>	0.244254	berger	0.013540
<b>supervisor</b>	0.244254	reuter	0.013245
<b>card</b>	0.238887	stanley	0.012881
<b>continent</b>	0.238827	employe	0.012448
<b>worker</b>	0.237655	rothschild	0.012097
<b>employee</b>	0.237353	tunzoo	0.011948
<b>reach</b>	0.231688	chrysler	0.011539
<b>system</b>	0.231688	scholarship	0.011238
<b>sooner</b>	0.231688	public	0.010143

**Note:** **Bold** indicates new terms obtained by our proposed method.

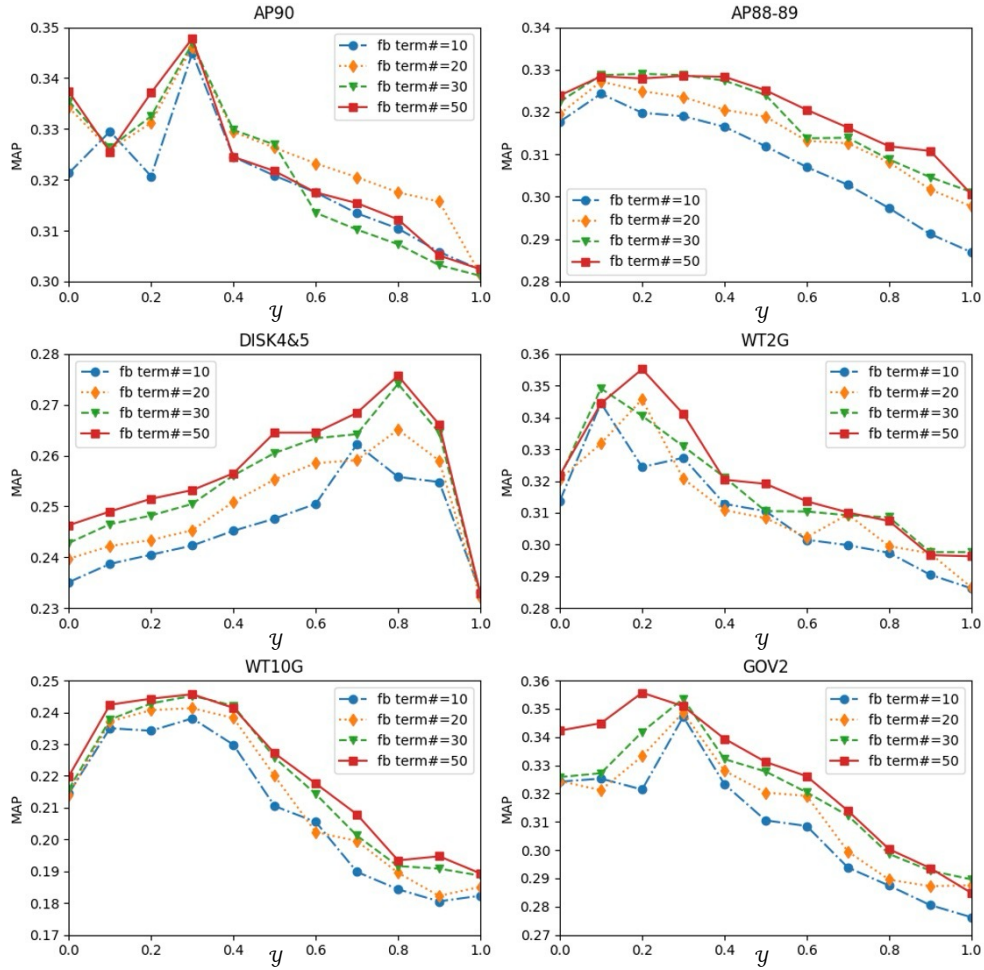


Figure 4.4: Sensitivity of CLRoc

#### 4.5.4 Case study

About the case study, we randomly chose the query “Merit-Pay vs. Seniority” (topic 60 in the AP90 dataset) as an example to compare the extension terms in the proposed model and the baseline. As shown in Table 4.6, the top 20 ranked expansion terms and the corresponding weights generated by the CLRoc and classical Rocchio models. Among them, a number of valid expansion terms were generated in both models, including “merit”, “pay”, “senior”, “chrysler”, and “card”, and “employee”. In addition, many of the terms in the Rocchio model expansion are either missing (e.g., “uaw”, “contract”, “spokesman”, and “base”) or ranked low in this query expansion (e.g., “chrysler” and “employee”). Nonetheless, these

candidate expansion terms hold significance as they convey substantial information about the 'Merit-Pay vs. Seniority' theme. Unlike traditional methods, our CLRoc model considers the semantic meaning of sentences rather than merely splitting the query into "Merit-Pay" and "Seniority." This approach allows it to reveal additional semantic terms that align with the query. In addition, some new terms generated by our approach (e.g., "contract", "spokesman", and "base") are closely related to the topic of "Merit-Pay vs. Seniority". This example demonstrates the advantages of query expansion according to the weak and strong signals relying on Contrast Learning, where the strong feedback signals are obtained by Contrast Learning.

## 4.6 Chapter Summary

This chapter introduced the CLRoc framework, which optimizes document matching by integrating weak and strong relevance signals through contrastive learning. The framework demonstrated significant improvements over baseline and advanced models, effectively enhancing alignment between user intent and retrieved documents.

Despite its success, CLRoc has some limitations. First, the reliance on weak relevance signals from pseudo-relevant documents may still propagate noise, particularly in cases where initial retrieval results contain irrelevant documents. Second, the linear fusion of weak and strong signals might oversimplify the interaction between these two factors, potentially limiting retrieval performance in complex query scenarios.

To address these limitations in future work, improvements could focus on:

- Developing an adaptive weighting mechanism for weak and strong relevance signals based on query context.
- Introducing advanced neural network architectures to capture more nuanced relationships between queries and documents.

- Incorporating user interaction or domain-specific feedback to reduce the reliance on purely pseudo-relevant documents.
- Investigating multi-task learning [108] methods to simultaneously enhance document relevance and query effectiveness understanding.

These advancements aim to further refine document matching and improve retrieval robustness across diverse scenarios.

# Chapter 5

## Optimizing Dense Retrieval via Large Language Model

### 5.1 Chapter Introduction

In recent years, Dense Retrieval has emerged as a powerful method for Information Retrieval, shifting the paradigm from traditional keyword-based approaches to more sophisticated semantic understanding. Dense Retrieval relies on vector representations of documents and queries, typically generated using Deep Learning. Transforming text into continuous embeddings allows for the identification of relevant content based on semantic similarity rather than mere keyword matches. This approach has been particularly effective in capturing complex relationships within text, leading to significant improvements in retrieval accuracy and relevance.

The development of Dense Retrieval has been propelled by advances in neural network architectures, particularly with the introduction of models like BERT and its successors. These models have revolutionized the way embeddings are generated, enabling more nuanced understanding of language and context. As a result, Dense Retrieval methods have gained traction in various applications, from academic research to commercial search engines, as

they provide a more robust solution to the challenges of modern information retrieval.

The landscape of information technology witnessed a remarkable transformation in 2023 with the widespread adoption of Large Language Model (LLM). Model such as OpenAI's ChatGPT, built on transformer architectures, became mainstream, capturing the attention of researchers and developers alike. Their ability to generate human-like text and comprehend complex queries has opened new avenues for enhancing retrieval systems. The sudden popularity of these models can be attributed to their impressive performance across a range of tasks and their potential to significantly improve user interactions with search systems.

In this context, we explore the integration of LLM into our dense retrieval framework. Specifically, during the query encoding phase, we leverage pseudo-relevance and external supplementary information generated by LLM to enhance the original query representation. This approach involves inputting both the augmented information and the original query into the query encoder, allowing for a richer and more contextually informed representation.

The incorporation of insights from LLM aims to capture nuanced semantic relationships and contextual information that may not be fully represented in the original query alone. By improving the quality of the query representation, we anticipate significant enhancements in overall retrieval effectiveness, enabling our framework to deliver more relevant and precise results to users. This combination aims to enhance the retrieval process by integrating insights from LLM with existing Dense Retrieval methods, ultimately improving the relevance and accuracy of retrieval results.

## 5.2 Dense Retrieval

Dense Retrieval [3] is a modern approach in IR that leverages dense vector representations to improve the accuracy and efficiency of retrieving relevant documents or entities. The most typical one is the Dual-Encoder architecture as shown in Figure 5.1. Unlike traditional Sparse Retrieval methods, such as TF-IDF or BM25, Dense Retrieval uses neural networks

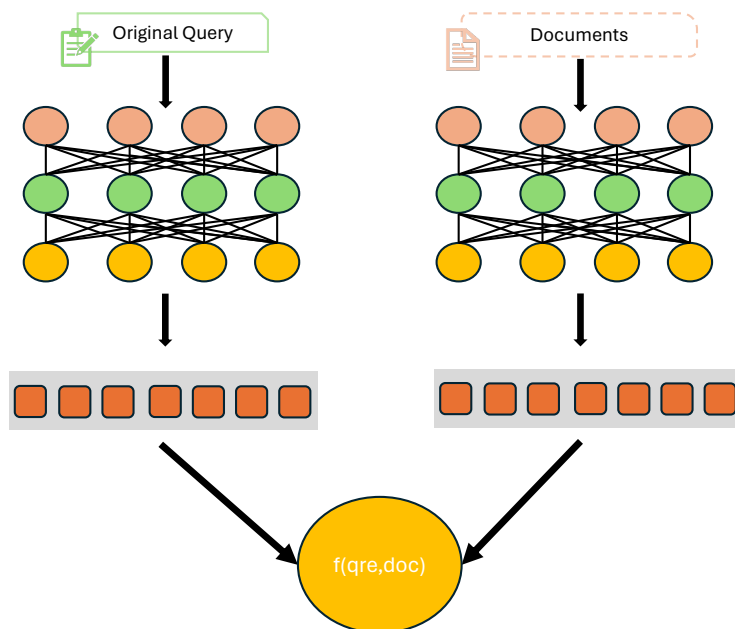


Figure 5.1: Dense Retrieval Dual-Encoder Architecture

to encode queries and documents into dense vectors, allowing for more nuanced matching. This method has shown significant improvements in various retrieval tasks, including open-domain question answering and entity linking. Below, we explore the key aspects of dense retrieval as discussed in the provided research papers.

**Dense Retrieval in Open-Domain Question Answering** Dense Retrieval has been effectively applied to open-domain Question Answering(QA) [98, 109], outperforming traditional sparse methods. The Dual-Encoder framework, which encodes both questions and passages into dense vectors, has demonstrated superior performance, achieving 9%-19% higher top-20 passage retrieval accuracy compared to BM25 systems [110]. This approach allows for retrieving more relevant passages, thereby enhancing the overall performance of QA systems on multiple benchmarks.

**Dense Representations for Entity Retrieval** In the context of entity retrieval, dense representations enable the encoding of mentions and entities into the same vector space. This

facilitates efficient retrieval through approximate nearest neighbour search, eliminating the need for traditional alias tables and re-rankers. The dual encoder model used in this setup has shown to outperform baseline methods and generalizes well across different datasets.

**Hybrid Models** Dense Retrieval models, while effective, can be resource-intensive. To address this, Hybrid Models combining dense and sparse representations have been proposed, offering a balance between precision and computational efficiency [111]. Techniques like constrained clustering and product quantization have been introduced to reduce memory costs and improve search efficiency, enabling fast approximate nearest neighbour searches with compact indexes [112].

ANCE [6] is a boost and popular dense retrieval model, which boosts the BERT-Siamese DR model to outperform all competitive dense and sparse retrieval baselines. It nearly matches the accuracy of sparse-retrieval-and-BERT-reranking using dot-product in the ANCE-learned representation space and provides almost 100x speed-up. This gave us no hesitation in choosing it as the basis of our method.

While Dense Retrieval methods have significantly advanced information retrieval, particularly in tasks like open-domain Question Answering and Entity Retrieval, the evolution of the LLM presents new opportunities for further enhancement. LLM, such as those based on transformer architectures, excel in understanding context and generating coherent text, which can complement the strengths of Dense retrieval systems. Integrating PRF and LLM offers exciting new possibilities for further enhancement. PRF, which utilizes initially retrieved documents to refine query representations, can improve the relevance of retrieval results by providing additional context, enhancing overall retrieval performance, and ultimately creating more effective and user-friendly search solutions.

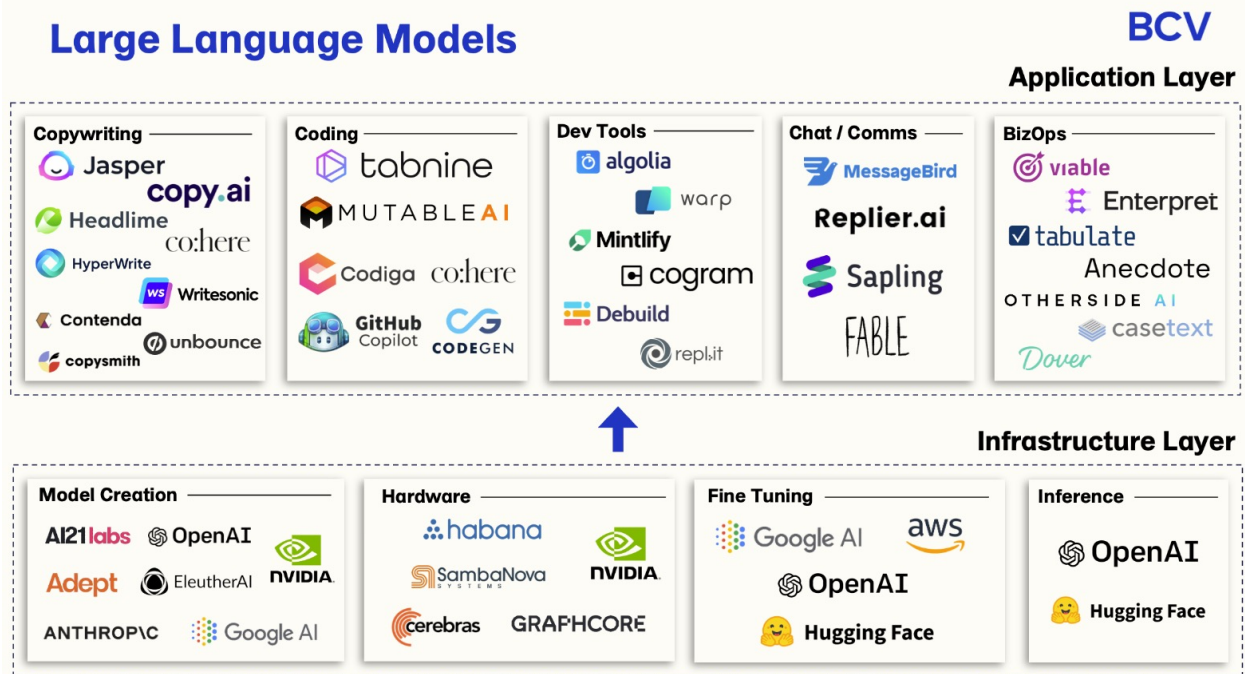


Figure 5.2: Some LLM Applications

### 5.3 Large Language Model

Large Language Models and applications emerge endlessly as shown in Figure 5.2<sup>1</sup> such as OpenAI’s ChatGPT<sup>2</sup>, are advanced neural networks designed to understand and generate human-like text. Built on transformer architectures, these models are trained on vast amounts of textual data, enabling them to grasp complex linguistic patterns, context, and semantics. LLM excel at a variety of tasks, including text completion, summarization, translation, and conversational agents. Their ability to generate coherent and contextually relevant responses has made them invaluable tools in various applications, including information retrieval, where they can enhance query understanding and result relevance.

Large Language Models have revolutionized NLP by leveraging vast amounts of data and parameters to perform complex tasks. When combined with retrieval systems, LLM can enhance its capabilities by accessing external knowledge, thus improving the accuracy and

<sup>1</sup><https://baincapitalventures.com/>

<sup>2</sup><https://chatgpt.com/>

relevance of its outputs. This integration, known as retrieval-augmented LLM, addresses some of the inherent limitations of LLM, such as hallucination and limited domain-specific knowledge.

In the past two years, significant progress has been made both in optimizing Large Language Models and in harnessing their exceptional capabilities to enhance various technologies. Researchers have focused on improving the efficiency and scalability of these models, making them more accessible for practical applications.

By incorporating IR systems, LLM can generate more factual and contextually relevant responses. This approach allows LLMs to access external corpora, providing references that enhance the factual accuracy of their outputs. RETA-LLM [113] is a toolkit designed to facilitate the development of such systems, offering modules for request rewriting, document retrieval, passage extraction, and fact-checking RETLLM [114] introduces a framework that equips LLMs with a read-write memory unit, enabling them to store and recall knowledge explicitly. This memory unit is scalable and interpretable, allowing LLMs to manage time-dependent information effectively, which is crucial for tasks like temporal-based question answering. LLMs can assist in deductive coding, a qualitative research method, by reducing the time and effort required for content analysis. LLM-assisted content analysis (LACA) [115] demonstrates that models like GPT-3.5 can perform coding tasks with accuracy comparable to human coders, thus streamlining the research process. InsightPilot [116] is an LLM-based system that simplifies data exploration by automating the selection of analysis intents and generating intentional queries. This system helps users gain insights from datasets through natural language inquiries, making data analysis more accessible and efficient.

Tang et al [117] introduce the GraphGPT framework, which integrates LLMs with graph structural knowledge through graph instruction tuning. This framework includes a text-graph grounding component to link textual and graph structures and a dual-stage instruction tuning approach with a lightweight graph-text alignment projector. These innovations allow LLMs to comprehend complex graph structures and enhance adaptability across diverse

datasets and tasks. Our framework demonstrates superior generalization in both supervised and zero-shot graph learning tasks, surpassing existing benchmarks.

Currently, the integration of LLMs into traditional systems has yielded impressive results. By leveraging the nuanced understanding and contextual awareness of LLMs, these systems have achieved improved performance and accuracy, demonstrating the transformative potential. This ongoing evolution reflects a broader trend toward creating more sophisticated, intelligent systems capable of addressing complex challenges across diverse domains. The method proposed in this chapter aims to enhance existing Dense Retrieval techniques by leveraging the exceptional capabilities of LLMs and integrating Pseudo-Relevance Feedback.

## 5.4 LLM-PRF: Optimizing Dense Retrieval via Large Language Model

### 5.4.1 Generative External Knowledge Acquisition

Generative Large Language Models play a crucial role in knowledge acquisition. As shown in Figure 5.3, In our method, we select GPT-3.5-turbo as the LLM for experiment and acquire external generative knowledge by integrating the user’s query  $Q_{org}^{llm}$  with a Chain of Thought (CoT) [8] through two distinct approaches. Chain of Thought facilitates structured reasoning by guiding the model through a logical sequence of thought processes. By integrating we enable the model to generate more contextually relevant external knowledge that directly addresses the user’s intent. The first method inputs only the query into the model, represented as:

$$E_1 = \text{LLM}(Q_{org}^{llm}) \tag{5.1}$$

This straightforward approach allows the model to focus directly on the user’s intent. The second method incorporates a set of  $l$  pseudo-relevance documents along with the query,

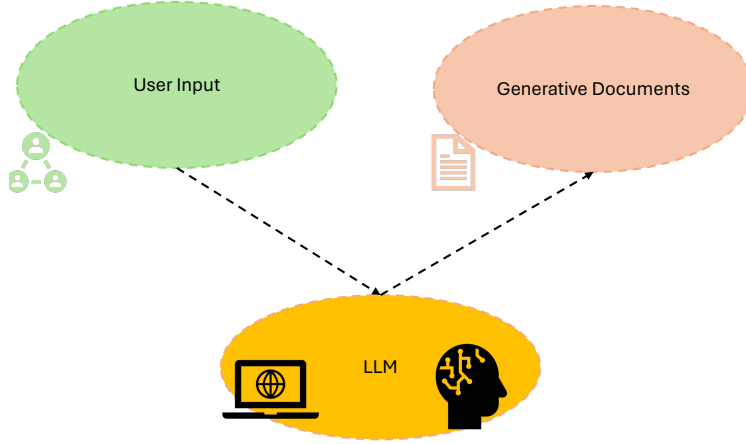


Figure 5.3: Flowchart of Generative Knowledge Acquisition

represented as:

$$E_2 = \text{LLM}(Q_{\text{org}}^{llm} + D_l) \quad (5.2)$$

where  $D_l = \{d_1, d_2, \dots, d_l\}$  is the collection of pseudo-relevance documents. This method enriches the context by providing additional information, potentially leading to more accurate and relevant results. By employing both approaches, we aim to optimize the retrieval process, enhancing both performance and user satisfaction.

#### 5.4.2 Pseudo-Relevance Feedback Injection

As we mentioned above, ANCE is the basic model we use for our method. It is a typical dense retrieval system encodes query and document using a BERT-style encoder and then

calculates the matching score using simple similarity metrics:

$$f(qre, docs) = \text{BERT}^{qre}([\text{CLS}] \text{ } qre \text{ } [\text{SEP}]) \cdot \text{BERT}^{docs}([\text{CLS}] \text{ } docs \text{ } [\text{SEP}]) \quad (5.3)$$

where  $\text{BERT}^{qre}$  and  $\text{BERT}^{docs}$  respectively output their final layer embeddings as the query and the document embedding.

We encode the selected pseudo-relevance documents and inject them into the query encoder to enhance the semantic representation of the query, thus forming a new query representation. Specifically, we first select relevant documents from the initial retrieval, then encode these documents to create supplementary information, and finally combine this with the user’s original query. As shown in the Equation 5.4:

$$\text{Encoder}(qre) = \text{BERT}^{qre}([\text{CLS}] \text{ } qre \text{ } [\text{SEP}]d_1 \text{ } [\text{SEP}]d_2 \text{ } [\text{SEP}] \dots [\text{SEP}]d_k \text{ } [\text{SEP}]) \quad (5.4)$$

where  $D_k = \{d_1, d_2, \dots, d_k\}$  is the collection of pseudo-relevance documents.

### 5.4.3 Dense Retrieval Integration

Not only that, we also need to encode the external knowledge obtained from the LLM as additional effective information into this dense retrieval framework. As we mentioned in Section 5.4.1, we used two different input methods, then after incorporating external knowledge into the encoder, we should also get two variants:

$$\text{Encoder}(a)(qre) = \text{BERT}^{qre}([\text{CLS}] \text{ } qre \text{ } [\text{SEP}]d_1 \text{ } [\text{SEP}] \dots [\text{SEP}]d_k \text{ } [\text{SEP}] [\text{SEP}]E_1 \text{ } [\text{SEP}]) \quad (5.5)$$

$$\text{Encoder}(b)(qre) = \text{BERT}^{qre}([\text{CLS}] \text{ } qre \text{ } [\text{SEP}]d_1 \text{ } [\text{SEP}] \dots [\text{SEP}]d_k \text{ } [\text{SEP}] [\text{SEP}]E_2 \text{ } [\text{SEP}]) \quad (5.6)$$

After changing the encoder method of query, our dense retrieval transitioned from Equation 5.3 to Equations 5.7 and 5.8:

$$LLM - PRF(a) = Encoder(a)(gre) \cdot BERT^{docs}([CLS] docs [SEP]) \quad (5.7)$$

$$LLM - PRF(b) = Encoder(b)(gre) \cdot BERT^{docs}([CLS] docs [SEP]) \quad (5.8)$$

## 5.5 Experimental Setup

### 5.5.1 Experimental Datasets and Analytical Metrics

**Datasets** In the experimental setup of this study, the datasets used are the TREC Deep Learning datasets for paragraph retrieval widely used in retrieval tasks in 2019 and 2020. The TREC 2019 [118] test set contains 43 queries, while the TREC 2020 [119] test set contains 54 queries. The relevance judgments for the two datasets range from 0 (non-relevant) to 3 (highly relevant). It is important to note that when calculating binary relevance (including MAP and Recall), documents labeled as 1 will be treated as non-relevant. Detailed information about the datasets is provided in Table 5.1.

Table 5.1: Validation Datasets for the LLM-PRF

Dataset	Query Count	Average Query Relevance	Number of Documents
TREC DL 2019	43	215.3	8,841,823
TREC DL 2020	54	210.9	8,841,823

This work primarily employs MAP as the primary evaluation metric because it is a standard evaluation metric for most retrieval tasks. Additionally, Recall@1000 (R@1K) and Normalized Discounted Cumulative Gain (NDCG@10) are also utilized to evaluate the feasibility and effectiveness of the proposed methods in this work.

### 5.5.2 Baseline Model

To validate the feasibility and effectiveness of the proposed methods, we choose BM25+Rocchio [46]: The combination of BM25 term weighting and the Rocchio feedback model is a powerful baseline model in PRF, which many researchers use as a strong baseline model for IR evaluation. ANCE [6]: It is a first-stage dense retriever that utilizes information from the corpus to update an artificial neural network index. As we mentioned, our method is based on ANCE, so comparison with it is necessary, and this can also better demonstrate the power of our proposed improvement method.

### 5.5.3 Hyperparameter

We choose GPT-3.5-turbo as the LLM and the experiments in this work were conducted using the Python toolkit Pyserini<sup>3</sup> [120]. Pyserini supports both sparse representations integrated from the Anserini [121] and dense representations integrated from Facebook’s Faiss library [122]. Important hyperparameters include the number of pseudo-relevance documents in the generative model and the number of pseudo-relevance documents in the Dense Retrieval model. Specifically, the parameter  $l$  takes values of 1, 3, 5, and 10. Due to the input length limit of the retrieval model, the maximum value of the parameter is 10, with a step size of 1. The parameter  $k$  takes values from 1 to 10 with a step size of 1. All experiments in this work were conducted on a server equipped with two RTX 3090 24GB.

## 5.6 Experimental Results and Analysis

### 5.6.1 Validation Against with Baseline Models

We perform a comprehensive comparison with the baseline models. The detailed comparison results are shown in Table 5.2. Our proposed models, LLM-PRF(a) and LLM-PRF(b),

---

<sup>3</sup><https://github.com/castorini/pyserini>

demonstrate significant improvements over traditional models like BM25 and BM25+Rocchio across both datasets, TREC DL 2019 and TREC DL 2020. LLM-PRF(a) achieves the highest performance in MAP and NDCG@10 on TREC DL 2019, indicating superior ranking quality at higher positions, while LLM-PRF(b) shows competitive results with slightly better R@1000 scores, reflecting improved recall. On TREC DL 2020, LLM-PRF(b) outperforms all models in terms of MAP and R@1000, highlighting its ability to retrieve more relevant documents overall. This shows that our methods effectively enhance retrieval performance across different evaluation metrics. In comparison to ANCE, the proposed models show significant improvements.

For TREC DL 2019, LLM-PRF(a) achieves a 10.50% improvement in NDCG@10 over ANCE (from 0.648 to 0.7158), while LLM-PRF(b) shows an 8.06% improvement. In terms of R@1000, LLM-PRF(b) improves by 6.80% compared to ANCE (from 0.755 to 0.8065), and LLM-PRF(a) shows a 6.40% improvement.

For TREC DL 2020, LLM-PRF(a) improves NDCG@10 by 7.46% (from 0.646 to 0.6942), while LLM-PRF(b) achieves an 8.11% increase. In R@1000, both models show significant gains, with LLM-PRF(a) improving by 6.75% and LLM-PRF(b) by 6.70% over ANCE

Table 5.2: Evaluation of MAP, NGCD@10 and R@1000 Metrics: LLM-PRF Model vs. Baseline Models

Dataset	TREC DL 2019			TREC DL 2020		
	MAP	NDCG@10	R@1000	MAP	NDCG@10	R@1000
BM25	0.3013	0.5058	0.7500	0.2856	0.4796	0.7860
BM25+Rocchio	0.3474	0.5275	-	0.3102	0.4893	-
ANCE	0.3710	0.6452	0.7554	0.4076	0.6458	0.7764
LLM-PRF(a)	<b>0.4594</b>	<b>0.8096</b>	0.8034	0.4542	<b>0.6984</b>	<b>0.8286</b>
LLM-PRF(b)	0.4566	0.7052	<b>0.8065</b>	<b>0.4566</b>	0.6955	0.8280

**Note:** **Bold** values indicate optimal results for each metric.

## 5.6.2 Validation Against Strong Models

We further evaluate the feasibility and effectiveness of the proposed methods with several strong dense retrieval models. ME-BERT [111] uses hard negative sampling [123] for training and selects the vectors of the first 8 terms in each document as the final document representation, employing multi-vector document encoding. DE-BERT [111] is a single-vector version of ME-BERT. Dense Passage Retrieval (DPR) [110] is an efficient semantic matching-based retrieval model for open-domain question answering tasks, which improves the overall performance of the question answering task. LTRe [124] generates hard negative samples using document embeddings from existing dense retrieval models. ColBERT E2E [125] is an end-to-end retrieval method based on the ColBERT [126] dense retrieval method. Table 5.3 compares the proposed methods with dense retrieval models in terms of NDCG@10 and R@1000. Bold font indicates the best results for the proposed methods across different evaluation metrics.

The comparison in Table 5.3 shows that the proposed methods, LLM-PRF(a) and LLM-PRF(b), outperform several strong dense retrieval models across most metrics for both the TREC DL 2019 and TREC DL 2020 datasets. LLM-PRF(a) achieves the highest score in NDCG@10 for TREC DL 2019 with a score of 0.7158, significantly outperforming models such as ME-BERT and ColBERT E2E, which score 0.687 and 0.693, respectively. Similarly, LLM-PRF(b) achieves the best result in R@1000 for TREC DL 2019, with a score of 0.8065, indicating its strength in retrieving a larger number of relevant documents. On TREC DL 2020, LLM-PRF(a) also leads in NDCG@10 with 0.6942, while LLM-PRF(b) achieves the best results in R@1000 for both datasets. These results demonstrate that the integration of LLM with PRF effectively enhances retrieval performance compared to traditional dense retrieval models, particularly in ranking and recall metrics.

In comparison to the strongest model ColBERT E2E, the proposed LLM-PRF models demonstrate further improvements. For TREC DL 2019, LLM-PRF(a) improves NDCG@10

by 3.29% (from 0.693 to 0.7158), while LLM-PRF(b) achieves a 1.04% increase. In terms of R@1000, LLM-PRF(b) shows a 2.22% improvement (from 0.789 to 0.8065), while LLM-PRF(a) improves by 1.81%.

For TREC DL 2020, LLM-PRF(a) sees a slight improvement in NDCG@10, with a 1.08% increase (from 0.687 to 0.6942), and LLM-PRF(b) achieves an improvement of 1.65%. In terms of R@1000, LLM-PRF(a) edges out ColBERT E2E with a 0.30% improvement (from 0.826 to 0.8284), while LLM-PRF(b) remains competitive with a marginal improvement of 0.02%.

Despite ColBERT E2E being the strongest dense retrieval model, these results show that the LLM-PRF models consistently outperform it across various metrics, particularly in terms of NDCG@10 and R@1000, demonstrating the effectiveness of integrating LLM with PRF to further enhance retrieval performance.

Table 5.3: Evaluation of MAP, NGCD@10 and R@1000 Metrics: LLM-PRF Model vs. Strong Dense Retrieval Models

Models	TREC DL 2019		TREC DL 2020	
	NDCG@10	R@1000	NDCG@10	R@1000
BM25	0.506	0.750	0.480	0.786
BM25+RM3	0.518	0.800	0.482	0.822
DPR	0.600	-	0.557	-
DE-BERT	0.639	-	-	-
ME-BERT	0.687	-	-	-
LTRe	0.675	-	-	-
ANCE	0.648	0.755	0.646	0.776
ColBERT E2E	0.693	0.789	0.687	0.826
LLM-PRF (a)	<b>0.716</b>	0.803	0.694	<b>0.828</b>
LLM-PRF (b)	0.700	<b>0.807</b>	<b>0.698</b>	0.828

**Note:** Bold values indicate optimal results for each metric.

## 5.7 Chapter Summary

This chapter proposed the LLM-PRF framework, which integrates Large Language Models (LLMs) into the dense retrieval process to enhance query representation. By leveraging the contextual reasoning capabilities of LLMs, the framework significantly improved retrieval performance compared to traditional and state-of-the-art dense retrieval models.

However, the approach has notable limitations. First, the computational cost of integrating LLMs, especially during query encoding, can be prohibitive for real-time applications. Second, the reliance on pre-trained LLMs may lead to suboptimal performance in domain-specific tasks without further prompting. Third, LLMs may occasionally generate irrelevant or misleading context, impacting retrieval accuracy.

To address these limitations in future work, improvements could focus on:

- Developing lightweight LLM variants or compression techniques to reduce computational overhead. Incorporating domain adaptation methods to prompt LLMs for specific retrieval tasks.
- Enhancing the interpretability of LLM outputs to mitigate the inclusion of irrelevant terms in query representation.
- Exploring hybrid approaches that combine LLMs with traditional retrieval techniques for balanced efficiency and effectiveness.

These enhancements aim to make LLM-PRF more practical for real-world applications while maintaining its semantic depth and retrieval accuracy.

# Chapter 6

## Conclusion and Future Work

### 6.1 Summary of Contributions

This thesis focuses on the critical challenge of understanding user sparse queries in information retrieval systems. To address this challenge, three models are proposed from different perspectives and presents three innovative models—CNRoc, CLRoc, and LLM-PRF—designed to enhance retrieval performance in this context. Each model leverages distinct yet complementary approaches, integrating Semantic Network, Contrastive Learning, and Large Language Model to improve query expansions within the Pseudo-Relevance Feedback framework.

The **CNRoc model** enriches query expansions by incorporating external conceptual knowledge from ConceptNet. By addressing the limitations of traditional Pseudo-Relevance Feedback methods, which often rely solely on positional and frequency-based information, CNRoc captures nuanced meanings of query terms through a semantic network. This results in contextually relevant and semantically rich query expansions. Rigorous evaluations on standard TREC datasets demonstrate that CNRoc significantly improves key performance metrics, such as MAP, MRR, and NDCG, underscoring its ability to bridge the gap between sparse user queries and relevant documents.

The **CLRoc model** introduces a novel probabilistic framework that combines weak and strong relevance signals through Contrastive Learning. Traditional Pseudo-Relevance Feedback methods primarily focus on weak signals derived from term frequency, which may not fully capture the complexities of user intent. CLRoc addresses this by integrating strong relevance signals generated through Contrastive Learning mechanisms, enhancing the alignment between user queries and the most relevant documents. Extensive experiments validate CLRoc’s superiority over various baseline models across multiple TREC datasets, highlighting its potential to optimize document selection in response to sparse queries.

The **LLM-PRF model** integrates Large Language Model with Pseudo-Relevance Feedback to enhance dense retrieval systems. By utilizing external knowledge generated by LLMs during the query encoding phase, LLM-PRF enriches query representations with greater semantic depth. The framework employs a Chain of Thought reasoning process to gain contextually relevant insights, improving the system’s understanding of user intent. Comprehensive evaluations on TREC DL 2019 and TREC DL 2020 datasets indicate that LLM-PRF outperforms traditional models (BM25 and ANCE) and consistently exceeds state-of-the-art dense retrieval models (ColBERT E2E) across multiple metrics.

In summary, this thesis provides a comprehensive exploration of how these three models—CNRoc, CLRoc, and LLM-PRF collectively address the limitations of existing PRF methods in handling sparse queries. By focusing on understanding user intent in the context of sparse queries, this research lays the groundwork for future advancements, ensuring that users can effectively find the information they seek and make informed decisions.

## 6.2 Future Work

While this thesis has successfully introduced and validated three novel models: CNRoc, CLRoc, and LLM-PRF for enhancing Pseudo-Relevance Feedback in IR. The applications of our proposed framework and methods can also be extended to diverse practical domains,

such as Genomics and chemical IR research [127–129], recommendation systems [130, 131] and Web mining research [132]. There are several promising avenues for future research. In particular, integrating advanced techniques such as Retrieval-Augmented Generation (RAG), generative AI models (GenAI), and intelligent agents presents significant opportunities to further enhance the capabilities of information retrieval systems.

**Integration of RAG:** Future research could explore RAG techniques, which combine retrieval-based methods with generative models to produce contextually accurate and informative responses. By integrating CNRoc and CLRoc with a RAG framework, we can leverage the strengths of both retrieval and generation. The refined query expansions from CNRoc and the strong relevance signals from CLRoc could guide the retrieval process within a RAG model, ensuring that the generative component accesses the most relevant and semantically enriched information. This integration could substantially enhance the quality of outputs, especially in complex query scenarios where traditional methods may fall short.

**Application of Gen AI Models:** The rapid advancements in generative AI, such as models like GPT and BERT, open new possibilities for improving query expansion and document retrieval. Future work could focus on embedding generative AI models directly into the PRF process, enabling them to dynamically generate query expansions or predict relevant documents based on sparse user inputs. By combining generative AI with the semantic enrichment from CNRoc and the relevance optimization of CLRoc, we could create a more adaptive and responsive retrieval system capable of handling ambiguous queries effectively.

**Development of Intelligent Agents [133]:** Developing intelligent agents capable of autonomously managing and optimizing the information retrieval process is another promising direction. These agents could leverage the advanced capabilities of CNRoc, CLRoc, and LLM-PRF, integrating them into a broader framework that includes real-time decision-making, user interaction, and adaptive learning. Equipped with these models, an intelligent agent could continually refine its retrieval strategies based on user feedback, learning from

interactions to improve performance over time. Additionally, incorporating RAG and generative AI could enable these agents to provide comprehensive support, including retrieving relevant documents and generating tailored summaries or responses.

**Addressing Semantic Noise and Model Robustness:** Future work should also prioritize enhancing the robustness of these models, particularly against semantic noise introduced by external knowledge sources like ConceptNet. Although the current models demonstrate effectiveness in refining semantic information, further research is needed to develop sophisticated techniques for mitigating the impact of irrelevant or misleading data. Ensuring the robustness and reliability of intelligent agents across diverse domains and user queries will be crucial as their complexity increases.

**Exploring Cross-Domain Applications:** Finally, extending the application of these models beyond traditional information retrieval tasks to domains such as healthcare, legal research, or financial analysis could provide valuable insights into their generalizability and effectiveness. The combination of RAG, generative AI, and intelligent agents could be tailored to meet specific industry needs, enhancing decision-making and information access in those fields [32, 32–36, 38, 60, 109, 134–140].

In conclusion, the integration of RAG, generative AI, and intelligent agents into the PRF process holds significant potential for advancing the field of information retrieval. Building on the foundations established by CNRoc, CLRoc, and LLM-PRF, future research can explore these cutting-edge technologies to develop more intelligent, adaptive, and effective retrieval systems that better meet user needs.

# Bibliography

- [1] GuihongCao, Jian-Yun Nie, Jianfeng Gao and Stephen Robertson, “Selecting good expansion terms for pseudo-relevance feedback,” in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 243–250, 2008.
- [2] Frederick Wilfrid Lancaster and Emily Gallup, “Information retrieval on-line,” tech. rep., 1973.
- [3] Jiajia Wang, Jimmy Xiangji Huang, Xinhui Tu, Junmei Wang, Angela Jennifer Huang, and Md Tahmid Rahman Laskar and Amran Bhuiyan, “Utilizing BERT for Information Retrieval: Survey, Applications, Resources, and Challenges,” *ACM Computing Surveys*, vol. 56, no. 7, pp. 1–33, 2024.
- [4] Efthimis Nefthimiadis, “Query expansion,” *Annual review of information science and technology (ARIST)*, vol. 31, pp. 121–87, 1996.
- [5] Hugo Liu and Push Singh, “ConceptNet—a practical commonsense reasoning tool-kit,” *BT technology journal*, vol. 22, no. 4, pp. 211–226, 2004.
- [6] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed and Arnold Overwijk, “Approximate nearest neighbor negative contrastive learning for dense text retrieval,” *arXiv preprint arXiv: 2007.00808*, 2020.

- [7] Yizheng Huang and Jimmy Xiangji Huang, “A survey on retrieval-augmented text generation for large language models,” 2024.
- [8] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le and Denny Zhou, “Chain-of-thought prompting elicits reasoning in large language models,” *Advances in neural information processing systems*, vol. 35, pp. 24824–24837, 2022.
- [9] Gerard Salton and Chris Buckley, “Improving retrieval performance by relevance feedback,” *Journal of the American society for information science*, vol. 41, no. 4, pp. 288–297, 1990.
- [10] Vannevar Bush, “As we may think,” *Atlantic Monthly*, July, 1945.
- [11] Hans Peter Luhn, “A statistical approach to mechanized encoding and searching of literary information,” *IBM Journal of research and development*, vol. 1, no. 4, pp. 309–317, 1957.
- [12] Gerard Salton, Anita Wong and Chung-Shu Yang, “A vector space model for automatic indexing,” *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975.
- [13] Melvin Earl Maron and John Larry Kuhns, “On relevance, probabilistic indexing and information retrieval,” *Journal of the ACM (JACM)*, vol. 7, no. 3, pp. 216–244, 1960.
- [14] Jay M Ponte and W Bruce Croft, “A language modeling approach to information retrieval,” in *ACM SIGIR Forum*, vol. 51, pp. 202–208, ACM New York, NY, USA, 2017.
- [15] Thorsten Joachims, “Transductive learning via spectral graph partitioning,” in *Proceedings of the 20th international conference on machine learning (ICML-03)*, pp. 290–297, 2003.

- [16] Karen Sparck Jones, “A statistical interpretation of term specificity and its application in retrieval,” *Journal of documentation*, vol. 28, no. 1, pp. 11–21, 1972.
- [17] Stephen E Robertson and K Sparck Jones, “Relevance weighting of search terms,” *Journal of the American Society for Information science*, vol. 27, no. 3, pp. 129–146, 1976.
- [18] Stephen E Robertson and Steve Walker, “Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval,” in *SIGIR’94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University*, pp. 232–241, Springer, 1994.
- [19] Fuchun Peng, Jimmy Xiangji Huang, Dale Schuurmans and Nick Cercone, “Investigating the relationship between word segmentation performance and retrieval performance in Chinese IR,” in *COLING 2002: The 19th International Conference on Computational Linguistics*, 2002.
- [20] Jiashu Zhao, Jimmy Xiangji Huang and Shicheng Wu, “Rewarding term location information to enhance probabilistic information retrieval,” in *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR’12)*, pp. 1137–1138, 2012.
- [21] Ben He, Jimmy Xiangji Huang and Xiaofeng Zhou, “Modeling term proximity for probabilistic information retrieval models,” *Information Sciences.*, vol. 181, no. 14, pp. 3017–3031, 2011.
- [22] Baiyan Liu, Xiangdong An and Jimmy Xiangji Huang, “Using term location information to enhance probabilistic information retrieval,” in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR’15)*, pp. 883–886, Association for Computing Machinery, Inc, 12 2015.

- [23] Fanghong Jian, Jimmy Xiangji Huang, Jiashu Zhao and Tingting He, “A new term frequency normalization model for probabilistic information retrieval,” in *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pp. 1237–1240, ACM, 2018.
- [24] Zheng Ye and Jimmy Xiangji Huang, “A simple term frequency transformation model for effective pseudo relevance feedback,” in *SIGIR 2014 - Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 323–332, Association for Computing Machinery, 2014.
- [25] Fanghong Jian, Jimmy Xiangji Huang, Jiashu Zhao, Zhiwei Ying and Yuqi Wang, “A topic-based term frequency normalization framework to enhance probabilistic information retrieval,” *Computational Intelligence*, vol. 36, pp. 486 – 521, 2020.
- [26] Yang Liu, Xiaohui Yu Jimmy Xiangji Huang and Aijun An, “Combining integrated sampling with SVM ensembles for learning from imbalanced datasets,” *Inf. Process. Manag.*, vol. 47, no. 4, pp. 617–631, 2011.
- [27] Yoav Freund, Raj Iyer, Robert E Schapire and Yoram Singer, “An efficient boosting algorithm for combining preferences,” *Journal of machine learning research*, vol. 4, no. Nov, pp. 933–969, 2003.
- [28] Donald Metzler and W Bruce Croft, “A markov random field model for term dependencies,” in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 472–479, 2005.
- [29] Xing Wei and W Bruce Croft, “Lda-based document models for ad-hoc retrieval,” in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 178–185, 2006.

- [30] Jiashu Zhao, Jimmy Xiangji Huang and Ben He, “CRTER: Using cross terms to enhance probabilistic information retrieval,” in *Proceeding of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2011, Beijing, China, July 25-29, 2011*, pp. 155–164, ACM, 2011.
- [31] Jiashu Zhao, Jimmy Xiangji Huang and Zheng Ye, “Modeling term associations for probabilistic information retrieval,” *ACM Trans. Inf. Syst.*, vol. 32, no. 2, pp. 7:1–7:47, 2014.
- [32] Qin Chen, Qinmin Hu, Jimmy Xiangji Huang and Liang He, “CA-RNN: Using context-aligned recurrent neural networks for modeling sentence similarity,” in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 265–273, AAAI Press, 2018.
- [33] Xiaoshi Yin, Jimmy Xiangji Huang, Zhoujun Li and Xiaofeng Zhou, “A survival modeling approach to biomedical search result diversification using wikipedia,” *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 6, pp. 1201–1212, 2013.
- [34] Jimmy Xiangji Huang, Jun Miao and Ben He, “High performance query expansion using adaptive co-training,” *Inf. Process. Manag.*, vol. 49, no. 2, pp. 441–453, 2013.
- [35] Mariam Daoud and Jimmy Xiangji Huang, “Modeling geographic, temporal, and proximity contexts for improving geotemporal search,” *J. Assoc. Inf. Sci. Technol.*, vol. 64, no. 1, pp. 190–212, 2013.
- [36] Jimmy Xiangji Huang, Fuchun Peng, Dale Schuurmans, Nick Cercone and Stephen E Robertson, “Applying machine learning to text segmentation for information retrieval,” *Information Retrieval*, vol. 6, pp. 333–362, 2003.

- [37] Xing Tan, Jimmy Xiangji Huang and Aijun An, “Ranking documents through stochastic sampling on bayesian network-based models: A pilot study,” in *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR’16)*, pp. 961–964, Association for Computing Machinery, Inc, 12 2016.
- [38] Jimmy Xiangji Huang, Stephen Robertson, Nick Cercone and Aijun An, “Probability-based Chinese text processing and retrieval,” *Computational Intelligence*, vol. 16, pp. 552–569, 12 2000.
- [39] Zheng Ye and Jimmy Xiangji Huang, “A learning to rank approach for quality-aware pseudo-relevance feedback,” *Journal of the Association for Information Science and Technology*, vol. 67, pp. 942–959, 12 2016.
- [40] IJsbrand Jan Aalbersberg, “Incremental relevance feedback,” in *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 11–22, 1992.
- [41] Amanda Helen Spink, *Feedback in information retrieval*. Rutgers The State University of New Jersey, School of Graduate Studies, 1993.
- [42] Jun Miao, Jimmy Xiangji Huang and Zheng Ye, “Proximity-based rocchio’s model for pseudo relevance feedback,” in *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR’12)*, pp. 535–544, ACM Press, 2012.
- [43] Min Pan, Quanli Pei, Yu Liu, Teng Li, Ellen Anne Huang, Junmei Wang and Jimmy Xiangji Huang, “SPRF: A semantic pseudo-relevance feedback enhancement for information retrieval via ConceptNet,” *Knowledge-Based Systems*, vol. 274, p. 110602, 2023.

- [44] Min Pan, Jimmy Xiangji Huang, Tingting He, Zhiming Mao, Zhiwei Ying and Xinhui Tu, “A simple kernel co-occurrence-based enhancement for pseudo-relevance feedback,” *Journal of the Association for Information Science and Technology*, vol. 71, pp. 264–281, 12 2020.
- [45] Zheng Ye, Ben He, Jimmy Xiangji Huang and Hongfei Lin, “Revisiting Rocchio’s relevance feedback algorithm for probabilistic models,” *Information Retrieval Technology: 6th Asia Information Retrieval Societies Conference, AIRS 2010, Taipei, Taiwan, December 1-3, 2010. Proceedings 6*, pp. 151–161, 2010.
- [46] Joseph John Rocchio, “The smart retrieval system: Experiments in automatic document processing,” *Relevance feedback in information retrieval*, pp. 313–323, 1971.
- [47] Dong Zhou, Mark Truran, Jianxun Liu and Sanrong Zhang, “Collaborative pseudo-relevance feedback,” *Expert Systems with Applications*, vol. 40, no. 17, pp. 6805–6812, 2013.
- [48] Claudio Carpineto, Renato De Mori, Giovanni Romano and Brigitte Bigi, “An information-theoretic approach to automatic query expansion,” *ACM Transactions on Information Systems (TOIS)*, vol. 19, no. 1, pp. 1–27, 2001.
- [49] K. Collins-Thompson, “Reducing the risk of query expansion via robust constrained optimization,” in *Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 837–846, 2009.
- [50] Zheng Ye, Jimmy Xiangji Huang and Hongfei Lin, “Finding a good query-related topic for boosting pseudo-relevance feedback,” *Journal of the American Society for Information Science and Technology*, vol. 62, no. 4, pp. 748–760, 2011.
- [51] Karthik Raman, Raghavendra Udupa, Pushpak Bhattacharya and Abhijit Bhole, “On improving pseudo-relevance feedback using pseudo-irrelevant documents,” in *Advances*

- in Information Retrieval: 32nd European Conference on IR Research, ECIR 2010, Milton Keynes, UK, March 28-31, 2010. Proceedings 32*, pp. 573–576, Springer, 2010.
- [52] Amit Singhal, Mandar Mitra and Chris Buckley, “Learning routing queries in a query zone,” in *Proceedings of the 20th annual international ACM SIGIR conference on research and development in information retrieval*, pp. 25–32, 1997.
- [53] Xuanhui Wang, Hui Fang and ChengXiang Zhai, “A study of methods for negative relevance feedback,” in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 219–226, 2008.
- [54] Pierpaolo Basile, Annalina Caputo and Giovanni Semeraro, “Negation for document re-ranking in ad-hoc retrieval,” in *Conference on the Theory of Information Retrieval*, pp. 285–296, Springer, 2011.
- [55] Kyung Soon Lee, W Bruce Croft and James Allan, “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*, pp. 235–242, 2008.
- [56] Yang Xu, Gareth JF Jones and Bin Wang, “Query dependent pseudo-relevance feedback based on wikipedia,” in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pp. 59–66, 2009.
- [57] Hussein Hazimeh and ChengXiang Zhai, “Axiomatic analysis of smoothing methods in language models for pseudo-relevance feedback,” in *Proceedings of the 2015 international conference on the theory of information retrieval*, pp. 141–150, 2015.
- [58] Jun Miao, Jimmy Xiangji Huang and Jiashu Zhao, “TopPRF: A probabilistic framework for integrating topic space into pseudo relevance feedback,” *ACM Transactions on Information Systems (TOIS)*, vol. 34, no. 4, pp. 1–36, 2016.

- [59] Mozhdeh Ariannezhad, Ali Montazer-alghaem, Hamed Zamani and Azadeh Shakery, “Iterative estimation of document relevance score for pseudo-relevance feedback,” in *European Conference on Information Retrieval*, pp. 676–683, Springer, 2017.
- [60] R. Jothilakshmi and N. Shanthi, “Combining multiple term selection methods for automatic query expansion in pseudo relevance feedback using rank score method,” *Asian Journal of Research in Social Sciences and Humanities*, vol. 7, no. 1, pp. 910–922, 2017.
- [61] Ali Montazer-alghaem, Hamed Zamani and Azadeh Shakery, “Term proximity constraints for pseudo-relevance feedback,” in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1085–1088, 2017.
- [62] Jagendra Singh and Aditi Sharan, “A new fuzzy logic-based query expansion model for efficient information retrieval using relevance feedback approach,” *Neural Computing and Applications*, vol. 28, pp. 2557–2580, 2017.
- [63] SK Michael Wong, Wojciech Ziarko and Patrick CN Wong, “Generalized vector spaces model in information retrieval,” in *Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 18–25, 1985.
- [64] Chengxiang Zhai and John Lafferty, “Model-based feedback in the language modeling approach to information retrieval,” in *Proceedings of the tenth international conference on Information and knowledge management*, pp. 403–410, 2001.
- [65] Yuanhua Lv and ChengXiang Zhai, “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*, pp. 1895–1898, 2009.

- [66] Tao Tao and ChengXiang Zhai, “Regularized estimation of mixture models for robust pseudo-relevance feedback,” in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 162–169, 2006.
- [67] Mostafa Dehghani, Hosein Azarbonya, Jaap Kamps, Djoerd Hiemstra and Maarten Marx, “Luhn revisited: Significant words language models,” in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pp. 1301–1310, 2016.
- [68] Runjie Zhu, Xinhui Tu and Jimmy Xiangji Huang, “Deep learning on information retrieval and its applications,” in *Deep learning for data analytics*, pp. 125–153, Elsevier, 2020.
- [69] Mohan Timilsina, Brian Davis, Mike Taylor and Conor Hayes, “Towards predicting academic impact from mainstream news and weblogs: A heterogeneous graph based approach,” in *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 1388–1389, IEEE, 2016.
- [70] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton and Greg Hullender, “Learning to rank using gradient descent,” in *Proceedings of the 22nd international conference on Machine learning*, pp. 89–96, 2005.
- [71] Bhaskar Mitra and Nick Craswell, “Neural models for information retrieval,” *arXiv preprint arXiv:1705.01509*, 2017.
- [72] Fernando Diaz, Bhaskar Mitra and Nick Craswell, “Query expansion with locally-trained word embeddings,” *arXiv preprint arXiv:1605.07891*, 2016.
- [73] DwaiPAYAN Roy, Debjyoti Paul, Mandar Mitra and Utpal Garain, “Using word embeddings for automatic query expansion,” *arXiv preprint arXiv:1606.07608*, 2016.

- [74] Giorgos Akrivas, Manolis Wallace, Giorgos Andreou, Giorgos Stamou and Stefanos Kollias, “Context-sensitive semantic query expansion,” in *Proceedings 2002 IEEE International Conference on Artificial Intelligence Systems (ICAIS 2002)*, pp. 109–114, IEEE, 2002.
- [75] Lina Yao, Quan Sheng, Yongrui Qin, Xianzhi Wang, Ali Shemshadi and Qi He, “Context-aware point-of-interest recommendation using tensor factorization with social regularization,” in *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pp. 1007–1010, 2015.
- [76] Ayyoob Imani, Amir Vakili, Ali Montazer and Azadeh Shakery, “Deep neural networks for query expansion using word embeddings,” in *Advances in Information Retrieval: 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14–18, 2019, Proceedings, Part II 41*, pp. 203–210, Springer, 2019.
- [77] Min Pan, Junmei Wang, Jimmy Xiangji Huang, Angela J Huang, Qi Chen and Jinguang Chen, “A probabilistic framework for integrating sentence-level semantics via BERT into pseudo-relevance feedback,” *Information Processing & Management*, vol. 59, no. 1, p. 102734, 2022.
- [78] Ike Vayansky and Sathish AP Kumar, “A review of topic modeling methods,” *Information Systems*, vol. 94, p. 101582, 2020.
- [79] David M Blei, Andrew Y Ng and Michael I Jordan, “Latent dirichlet allocation,” *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.
- [80] Xing Yi and James Allan, “A comparative study of utilizing topic models for information retrieval,” in *Advances in Information Retrieval: 31th European Conference on IR Research, ECIR 2009, Toulouse, France, April 6-9, 2009. Proceedings 31*, pp. 29–41, Springer, 2009.

- [81] Yuanyuan Zhang, James Z Wang and Pradip K Srimani, “Semantic graph based pseudo relevance feedback for biomedical information retrieval,” in *Proceedings of the 7th International Conference on Computational Systems-Biology and Bioinformatics*, pp. 48–52, 2016.
- [82] Muhammad Nabeel Asim, Muhammad Wasim, Muhammad Usman Ghani Khan and Waqar Mahmood, “Improved biomedical term selection in pseudo relevance feedback,” *Database*, vol. 2018, p. bay056, 2018.
- [83] Ronald J Tallarida, Rodney B Murray, Ronald J Tallarida and Rodney B Murray, “Chi-square test,” *Manual of pharmacologic calculations: with computer programs*, pp. 140–142, 1987.
- [84] Kezban Dilek Onal, Ye Zhang, Ismail Sengor Altingovde, Md Mustafizur Rahman, Pinar Karagoz, Alex Braylan, Brandon Dang, Heng-Lu Chang, Henna Kim and Quinten McNamara, “Neural information retrieval: At the end of the early years,” *Information Retrieval Journal*, vol. 21, pp. 111–182, 2018.
- [85] Yizheng Huang and Jimmy Xiangji Huang, “Exploring ChatGPT for next-generation information retrieval: Opportunities and challenges,” in *Web Intelligence*, no. Preprint, pp. 1–14, IOS Press, 2024.
- [86] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, “Retrieval-augmented generation for knowledge-intensive nlp tasks,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.
- [87] Jimmy Xiangji Huang, Jun Miao and Ben He, “High performance query expansion using adaptive co-training,” *Inf. Process. Manag.*, vol. 49, pp. 441–453, 2013.
- [88] Xiaobin Li, Stan Szpakowicz and Stan Matwin, “A WordNet-based algorithm for word sense disambiguation,” in *IJCAI*, vol. 95, pp. 1368–1374, 1995.

- [89] Robyn Speer, Joshua Chin and Catherine Havasi, “ConceptNet 5.5: An open multilingual graph of general knowledge,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, 2017.
- [90] M Ross Quillian Allan M Collins, “Retrieval time from semantic memory,” *Journal of Verbal Learning and Verbal Behavior*, vol. 8, pp. 240–247, 1969.
- [91] M. F. Porter, “An algorithm for suffix stripping,” *Program*, vol. 40, pp. 211–218, 12 2006.
- [92] James P Callan, W.Bruce Croft and John Broglio, “TREC and TIPSTER experiments with inquiry,” *Information Processing Management*, vol. 31, pp. 327–343, 12 1995.
- [93] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng and Grégoire Mesnil, “Learning semantic representations using convolutional neural networks for web search,” in *Proceedings of the 23rd international conference on world wide web*, pp. 373–374, 2014.
- [94] Jun-Tao Guo, Yang Xiang, Zhi Guan and Yan-Hong He, “Papain-catalyzed aldol reaction for the synthesis of trifluoromethyl carbinol derivatives,” *Journal of Molecular Catalysis B: Enzymatic*, vol. 131, pp. 55–64, 2016.
- [95] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero and Larry Heck, “Learning deep structured semantic models for web search using clickthrough data,” in *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pp. 2333–2338, 2013.
- [96] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan and Xueqi Cheng, “Text matching as image recognition,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, 2016.
- [97] Sebastian Hofstätter, Hamed Zamani, and Bhaskar Mitra, Nick Craswell and Allan Hanbury, “Local self-attention over long text for efficient document retrieval,” in *Pro-*

- ceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2021–2024, 2020.
- [98] Qixuan Zhang, Xinyi Weng, Guangyou Zhou, Yi Zhang and Jimmy Xiangji Huang, “ARL: An adaptive reinforcement learning framework for complex question answering over knowledge base,” *Inf. Process. Manage.*, vol. 59, May 2022.
- [99] Tianyu Gao, Xingcheng Yao and Danqi Chen, “SimCSE: simple contrastive learning of sentence embeddings,” in *Conference on Empirical Methods in Natural Language Processing*, 2021.
- [100] Ting Chen, Simon Kornblith, Mohammad Norouzi and Geoffrey Hinton, “A simple framework for contrastive learning of visual representations,” in *International conference on machine learning*, pp. 1597–1607, PMLR, 2020.
- [101] Mayura Kulkarni and Shubhangi Kale, “Information retrieval based improvising search using automatic query expansion,” in *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, pp. 1226–1230, IEEE, 2021.
- [102] Hang Li, Ahmedand Mourad, Bevan Koopman and Guido Zuccon, “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*, pp. 2154–2158, 2022.
- [103] Stephen E Robertson, Steve Walker, M M Beaulieu, Mike Gatford and Alison Payne, “Okapi at TREC-4,” *Nist Special Publication Sp*, pp. 73–96, 1996.
- [104] Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba and Stefanie Jegelka, “Debiased contrastive learning,” *arXiv preprint arXiv:2007.00224*, 2020.

- [105] Haobo Wang, Ruixuan Xiao, Yixuan Li, Lei Feng, Gang Niu, Gang Chen and Junbo Zhao, “PiCO+: Contrastive label disambiguation for robust partial label learning,” *arXiv preprint arXiv:2201.08984*, 2022.
- [106] Yuning You, Tianlong Chen, Yang Shen and Zhangyang Wang, “Graph contrastive learning automated,” in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, vol. 139 of *Proceedings of Machine Learning Research*, pp. 12121–12132, PMLR, 2021.
- [107] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra and Stefanie Jegelka, “Contrastive learning with hard negative samples,” *arXiv preprint arXiv:2010.04592*, 2020.
- [108] Chao Huang, Jiahui Chen, Lianghao Xia, Yong Xu, Peng Dai, Yanqing Chen, Liefeng Bo, Jiashu Zhao and Jimmy Xiangji Huang, “Graph-enhanced multi-task learning of multi-level transition dynamics for session-based recommendation,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, pp. 4123–4130, 2021.
- [109] Guangyou Zhou, Zhiwen Xie, Zongfu Yu and Jimmy Xiangji Huang, “DFM: A parameter-shared deep fused model for knowledge base question answering,” *Information Sciences*, vol. 547, pp. 103–118, 12 2021.
- [110] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen and Wen-tau Yih, “Dense passage retrieval for open-domain question answering,” *arXiv preprint arXiv:2004.04906*, 2020.
- [111] Yi Luan, Jacob Eisenstein, Kristina Toutanova and Michael Collins, “Sparse, dense, and attentional representations for text retrieval,” *Transactions of the Association for Computational Linguistics*, 2021.
- [112] Jingtao Zhan, Qingyao Ai, Yiqun Liu, Jiaxin Mao, Xiaohui Xie, Jinghui Zhang and Shaoping Ma, “Disentangled modeling of domain and relevance for adaptable dense retrieval,” *arXiv preprint arXiv:2208.05753*, 2022.

- [113] Jiongnan Liu, Jiajie Jin, Zihan Wang, Jiehan Cheng, Zhicheng Dou and Ji-Rong Wen, “RETA-LLM: A retrieval-augmented large language model toolkit,” *arXiv preprint arXiv:2306.05212*, 2023.
- [114] Ali Modarressi, Ayyoob Imani, Mohsen Fayyaz and Hinrich Schütze, “RET-LLM: Towards a general read-write memory for large language models,” *arXiv preprint arXiv:2305.14322*, 2023.
- [115] Robert Chew, John Bollenbacher, Michael Wenger, Jessica Speer and Annice Kim, “LLM-Assisted Content Analysis: Using large language models to support deductive coding,” *arXiv preprint arXiv:2306.14924*, 2023.
- [116] Pingchuan Ma, Rui Ding, Shuai Wang, Shi Han and Dongmei Zhang, “Demonstration of InsightPilot: An LLM-Empowered Automated Data Exploration System,” *arXiv preprint arXiv:2304.00477*, 2023.
- [117] Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Lixin Su, Suqi Cheng, Dawei Yin and Chao Huang, “GraphGPT: Graph Instruction Tuning for Large Language Models,” *arXiv preprint arXiv:2310.13023*, 2023.
- [118] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos and Ellen M. Voorhees, “Overview of the TREC 2019 deep learning track,” *arXiv preprint arXiv:2003.07820*, 2020.
- [119] Nick Craswell, Bhaskar Mitra, Emine Yilmaz and Daniel Campos, “Overview of the TREC 2020 deep learning track,” *arXiv preprint arXiv:2102.07662*, 2021.
- [120] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep and Rodrigo Nogueira, “Pyserini: A python toolkit for reproducible information retrieval research with sparse and dense representations,” in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2356–2362, 2021.

- [121] Peilin Yang, Hui Fang and Jimmy Lin, “Anserini: Enabling the use of lucene for information retrieval research,” in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR ’17, (New York, NY, USA), p. 1253–1256, Association for Computing Machinery, 2017.
- [122] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini and Hervé Jégou, “The Faiss library,” *arXiv preprint arXiv:2401.08281*, 2024.
- [123] Daniel Gillick, Sayali Kulkarni, Larry Lansing, Alessandro Presta, Jason Baldrige, Eugene Ie and Diego Garcia-Olano, “Learning dense representations for entity retrieval,” *arXiv preprint arXiv:1909.10506*, 2019.
- [124] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Min Zhang and Shaoping Ma, “Learning to retrieve: How to train a dense retrieval model effectively and efficiently,” *arXiv preprint arXiv:2010.10469*, 2020.
- [125] Xiao Wang, Craig Macdonald, Nicola Tonellotto and Iadh Ounis, “Pseudo-relevance feedback for multiple representation dense retrieval,” in *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, pp. 297–306, 2021.
- [126] Omar Khattab and Matei Zaharia, “ColBert: efficient and effective passage search via contextualized late interaction over Bert,” in *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 39–48, 2020.
- [127] Mihai Lupu, Florina Piroi, Jimmy Xiangji Huang, Jianhan Zhu and John Tait, “Overview of the TREC 2009 chemical IR track,” in *Proceedings of The Eighteenth Text REtrieval Conference, TREC 2009, Gaithersburg, Maryland, USA, November 17-*

- 20, 2009, vol. 500-278 of *NIST Special Publication*, National Institute of Standards and Technology (NIST), 2009.
- [128] Mihai Lupu, Jimmy Xiangji Huang, Jianhan Zhu and John Tait, “TREC-CHEM: large scale chemical information retrieval evaluation at TREC,” *SIGIR Forum*, vol. 43, no. 2, pp. 63–70, 2009.
- [129] Jimmy Xiangji Huang, Ming Zhong and Luo Si, “York university at TREC 2005: Genomics track,” in *Proceedings of the Fourteenth Text REtrieval Conference, TREC 2005, Gaithersburg, Maryland, USA, November 15-18, 2005*, vol. 500-266 of *NIST Special Publication*, National Institute of Standards and Technology (NIST), 2005.
- [130] Chao Huang, Jiahui Chen, Lianghao Xia, Yong Xu, Peng Dai, Yanqing Chen, Liefeng Bo, Jiashu Zhao and Jimmy Xiangji Huang, “Graph-enhanced multi-task learning of multi-level transition dynamics for session-based recommendation,” in *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 4123–4130, AAAI Press, 2021.
- [131] Lianghao Xia, Chao Huang, Yong Xu, Jiashu Zhao, Dawei Yin and Jimmy Xiangji Huang, “Hypergraph contrastive collaborative filtering,” in *SIGIR ’22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pp. 70–79, ACM, 2022.
- [132] Jimmy Xiangji Huang, Nick Cercone and Aijun An, “Comparison of interestingness functions for learning web usage patterns,” in *Proceedings of the 2002 ACM CIKM International Conference on Information and Knowledge Management, McLean, VA, USA, November 4-9, 2002*, pp. 617–620, ACM, 2002.

- [133] Michael Wooldridge and Nicholas R Jennings, “Intelligent agents: Theory and practice,” *The knowledge engineering review*, vol. 10, no. 2, pp. 115–152, 1995.
- [134] Hajer Ayadi, Mouna Torjmen-Khemakhem, Mariam Daoud, Jimmy Xiangji Huang and Maher Ben Jemaa, “MF-Re-Rank: A modality feature-based re-ranking model for medical image retrieval,” *Journal of the Association for Information Science and Technology*, vol. 69, pp. 1095–1108, 12 2018.
- [135] Hajer Ayadi, Mouna Torjmen-Khemakhem, Mariam Daoud, Jimmy Xiangji Huang and Maher Ben Jemaa, “Mining correlations between medically dependent features and image retrieval models for query classification,” *Journal of the Association for Information Science and Technology*, vol. 68, pp. 1323–1334, 12 2017.
- [136] Md. Tahmid Rahman Laskar, Jimmy Xiangji Huang and Enamul Hoque, “Contextualized embeddings based transformer encoder for sentence similarity modeling in answer selection task,” in *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pp. 5505–5514, European Language Resources Association, 2020.
- [137] Xiaoshi Yin, Jimmy Xiangji Huang and Zhoujun Li, “Mining and modeling linkage information from citation context for improving biomedical literature retrieval,” *Information Processing Management*, vol. 47, pp. 53–67, 12 2011.
- [138] Xiaoshi Yin, Jimmy Xiangji Huang, Zhoujun Li and Xiaofeng Zhou, “A survival modeling approach to biomedical search result diversification using wikipedia,” *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 6, pp. 1201–1212, 2013.
- [139] Jimmy Xiangji Huang and Qinmin Hu, “A bayesian learning approach to promoting diversity in ranking for biomedical information retrieval,” in *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Infor-*

*mation Retrieval, SIGIR 2009, Boston, MA, USA, July 19-23, 2009* (J. Allan, J. A. Aslam, M. Sanderson, C. Zhai, and J. Zobel, eds.), pp. 307–314, ACM, 2009.

- [140] Zhiwei Ying, Jimmy Xiangji Huang, Jie Zhou, Fanghong Jian and Tingting He, “DSPF: a digital signal processing based framework for information retrieval,” *IEEE Access*, vol. 7, pp. 110235–110248, 12 2019.

# Appendix A

## Published Papers

1. Min Pan, Teng Li, Yu Liu, **Quanli Pei**, Ellen Anne Huang, Jimmy Xiangji Huang. “A semantically enhanced text retrieval framework with abstractive summarization”. In: *Computational Intelligence*. 2024.
2. Min Pan, Shuting Zhou, Teng Li, Yu Liu, **Quanli Pei**, Angela Jennifer Huang, Jimmy Xiangji Huang. “Utilizing passage-level relevance and kernel pooling for enhancing BERT-based document reranking”. In: *Computational Intelligence*. 2024.
3. Min Pan, **Quanli Pei**, Yu Liu, Teng Li, Ellen Anne Huang, Junmei Wang, Jimmy Xiangji Huang. “SPRF: A semantic Pseudo-relevance Feedback enhancement for information retrieval via ConceptNet”. In: *Knowledge-Based Systems*. 2023.
4. **Quanli Pei**, Yulong Chen, Yu Liu, Teng Li, Wei Zhang, Wenrui Xiong, Xinpin Jiang, Min Pan. “Data information prediction based on deep fusion GRU-Stacking”. In: *2022 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*. 2022.
5. Min Pan, Yu Liu, **Quanli Pei**, Huixian Mao, Aoqun Jin, Sheng Huang, Yinhan Yang. “A Multi-Dimensional Semantic Pseudo-Relevance Feedback Information Retrieval Model”. In: *2022 IEEE/WIC/ACM International Joint Conference on Web*

*Intelligence and Intelligent Agent Technology (WI-IAT). 2022.*