THE UNIVERSITY OF QUEENSLAND

AUSTRALIA

AI-driven Automated Systematic Reviews

Shuai Wang

Master of Engineering Science

0000-0002-0726-5250

A thesis submitted for the degree of Doctor of Philosophy at

The University of Queensland in 2025

School of Electrical Engineering and Computer Science

Abstract

A systematic review (SR) is a type of literature review that appraises and synthesises the work of primary research studies to answer one or more research questions. However, the process is time-consuming and labour-intensive; a systematic review typically takes more than two years, requires a median of 1,110 hours of effort, and can cost in excess of $350K. Researchers face steep challenges in identifying and screening relevant studies from vast and continually growing literature pools, often leading to lengthy review processes that can stall critical decision-making. Despite the availability of advanced search tools, current approaches can retrieve an overwhelming number of potential studies, with few genuinely relevant ones hidden among them.

This dissertation investigates the potential of artificial intelligence (AI) to alleviate the burden of SRs. In particular, it examines how AI-driven methods can streamline or assist with systematic review creation without sacrificing the accuracy on which SRs depend. This thesis investigates approaches that had been missed in prior research—many of which attempt to apply the most recent advances in information retrieval and large language models (LLMs). In particular, it empirically evaluates: (1) the possibility of using external knowledge during systematic review creation—such as a set of example studies (seed studies)—to determine whether integrating this knowledge with existing AI methods could provide any additional benefit, (2) AI-assisted automated query generation, (3) automatic identification of relevant documents, which is typically identified through screening using studies retrieved by the query.

Through an extensive set of experiments on multiple existing systematic review collections, this thesis concludes that:

- **Utilising Seed Studies Yields Better Automation:** Leveraging seed studies is confirmed to be useful for automating systematic review creation, including assisting with the creation of queries and enabling ranked screening (so that relevant studies can be identified faster).

- **AI-Assisted Query Generation Improves Query Effectiveness:** AI-driven approaches to query formulation (such as the use of generative LLMs) produce more precise initial searches, minimising irrelevant studies without missing important ones. However, caution is advised due to the inherent stochasticity of generative models. Additionally, the use of AI can effectively suggest structured terms (e.g., MeSH terms) to refine systematic review queries.

- **Automated/Ranked Screening Is Possible but Requires Further Exploration:** Neural methods for ranking or automatic classification can be used for screening and may assist the current systematic review pipeline in reducing costs. Nevertheless, these methods may not yet reach their full potential, and further research is needed to enhance their effectiveness.

Overall, these findings demonstrate the promise of integrating AI-driven methods into systematic review workflows. By using seed studies, automating query generation through cutting-edge language models, and applying automated screening techniques, researchers can substantially reduce the time

and effort required to identify the most relevant literature. Yet, the results also underscore the importance of continued exploration and careful implementation, particularly regarding the variability of generative models and the evolving capabilities of neural ranking approaches. Moreover, while the technical innovations presented here are demonstrated within the context of systematic reviews, they have broader applicability to other professional tasks that rely on complex Boolean queries, such as patent, legal, financial, and news searches. Taken together, this thesis lays a foundation for more efficient, cost-effective, and robust systematic review processes, while highlighting avenues for future innovation in AI-enhanced evidence-based medicine.

Declaration by author

This thesis is composed of my original work, and contains no material previously published or written by another person except where due reference has been made in the text. I have clearly stated the contribution by others to jointly-authored works that I have included in my thesis.

I have clearly stated the contribution of others to my thesis as a whole, including statistical assistance, survey design, data analysis, significant technical procedures, professional editorial advice, financial support and any other original research work used or reported in my thesis. The content of my thesis is the result of work I have carried out since the commencement of my higher degree by research candidature and does not include a substantial part of work that has been submitted to qualify for the award of any other degree or diploma in any university or other tertiary institution. I have clearly stated which parts of my thesis, if any, have been submitted to qualify for another award.

I acknowledge that an electronic copy of my thesis must be lodged with the University Library and, subject to the policy and procedures of The University of Queensland, the thesis be made available for research and study in accordance with the Copyright Act 1968 unless a period of embargo has been approved by the Dean of the Graduate School.

I acknowledge that copyright of all material contained in my thesis resides with the copyright holder(s) of that material. Where appropriate I have obtained copyright permission from the copyright holder to reproduce material in this thesis and have sought permission from co-authors for any jointly authored works included in the thesis.

Publications included in this thesis

1. [254] **Shuai Wang**, Hang Li, Harrisen Scells, Daniel Locke and Guido Zuccon, MeSH Term Suggestion for Systematic Review Literature Search *In Proceedings of the 25th Australasian Document Computing Symposium*, pp. 1-8, 2021.

2. [261] **Shuai Wang**, Harrisen Scells, Ahmed Mourad and Guido Zuccon, Seed-Driven Document Ranking for Systematic Reviews: A Reproducibility Study, *In Proceedings of the 44th European Conference on Information Retrieval*, pp. 686-700, 2022.

3. [258] **Shuai Wang**, Harrisen Scells, Justin Clark, Bevan Koopman and Guido Zuccon, From Little Things Big Things Grow: A Collection with Seed Studies for Medical Systematic Review Literature Search, *In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 3176-3186, 2022.

4. [259] **Shuai Wang**, Harrisen Scells, Bevan Koopman and Guido Zuccon, Automated MeSH term suggestion for effective query formulation in systematic reviews literature search, *In Intelligent Systems with Applications*, pp. 200141, 2022.

5. [260] **Shuai Wang**, Harrisen Scells, Bevan Koopman and Guido Zuccon, Neural Rankers for Effective Screening Prioritisation in Medical Systematic Review Literature Search, *In Proceedings of the 26th Australasian Document Computing Symposium*, pp. 1-10, 2022.

6. [262] **Shuai Wang**, Harrisen Scells, Bevan Koopman and Guido Zuccon, MeSH Suggester: A Library and System for MeSH Term Suggestion for Systematic Review Boolean Query Construction, *In Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pp. 1176-1179, 2023.

7. [264] **Shuai Wang**, Harrisen Scells, Bevan Koopman and Guido Zuccon, Can ChatGPT write a good boolean query for systematic review literature search?, *In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1426-1436, 2023.

8. [263] **Shuai Wang**, Harrisen Scells, Bevan Koopman, Martin Potthast and Guido Zuccon, Generating Natural Language Queries for More Effective Systematic Review Screening Prioritisation, *In Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, pp. 73-83, 2023.

9. [266] **Shuai Wang**, Harrisen Scells, Shengyao Zhuang, Martin Potthast, Bevan Koopman and Guido Zuccon, Zero-shot Generative Large Language Models for Systematic Review Screening Automation, *In Proceedings of the 46th European Conference on Information Retrieval*, pp. 73-83, 2024.

10. [269] **Shuai Wang**, Harrisen Scells, Bevan Koopman and Guido Zuccon, Reassessing Large Language Model Boolean Query Generation for Systematic Reviews. *In Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 3296–3305, 2025.

Other publications during candidature

1. [255] **Shuai Wang**, Shengyao Zhuang and Guido Zuccon, BERT-based Dense Retrievers Require Interpolation with BM25 for Effective Passage Retrieval , *In Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, pp. 317-324, 2021.

2. [136] Hang Li*, **Shuai Wang***, Shengyao Zhuang, Ahmed Mourad and Guido Zuccon, To Interpolate or not to Interpolate: PRF, Dense and Sparse Retrievers , *In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2495-2500, 2022.

3. [253] **Shuai Wang** and Guido Zuccon, Balanced Topic Aware Sampling for Effective Dense Retriever: A Reproducibility Study , *In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2542-2551, 2023.

4. [78] Lukas Gienapp, Harrisen Scells, Niklas Deckers, Janek Bevendorff, **Shuai Wang**, Johannes Kiesel, Shahbaz Syed, Maik Fröbe, Guido Zuccon, Benno Stein, Matthias Hagen and Martin Potthast, Evaluating Generative Ad Hoc Information Retrieval , *In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1916-1929, 2024.

5. [265] **Shuai Wang**, Ekaterina Khramtsova, Shengyao Zhuang and Guido Zuccon, FeB4RAG: Evaluating Federated Search in the Context of Retrieval Augmented Generation , *In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 763-273, 2024.

6. [267] **Shuai Wang**, Shengyao Zhuang and Guido Zuccon, Large Language Models Based Stemming for Information Retrieval: Promises, Pitfalls and Failures , *In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2492-2496, 2024.

7. [190] David Rau, Hervé Déjean, Nadezhda Chirkova, Thibault Formal, **Shuai Wang**, Vassilina Nikoulina and Stéphane Clinchant, BERGEN: A Benchmarking Library for Retrieval-Augmented Generation, In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7640–7663, 2024.

8. [191] David Rau*, **Shuai Wang***, Hervé Déjea and Stéphane Clinchant, Context Embeddings for Efficient Answer Generation in RAG , *In Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, 2024.*

9. [147] Sean MacAvaney, Adam Roegiest, Aldo Lipani, Andrew Parry, Bj¨orn Engelmann, Christin Katharina Kreutz, Chuan Meng, Erlend Frayling, Eugene Yang, Ferdinand Schlatt, Guglielmo Faggioli, Harrisen Scells, Iana Atanassova, Jana Friese, Janek Bevendorff, Javier Sanz-Cruzado, Johanne Trippas, Kanaad Pathak, Kaustubh Dhole, Leif Azzopardi, Maik Fr¨obe, Marc Bertin, Nishchal Prasad, Saber Zerhoudi, **Shuai Wang**, Shubham Chatterjee, Thomas Jaenich, Udo Kruschwitz, Xi Wang, Zijun Long, Report on the Collab-a-thon at ECIR 2024 *In ACM SIGIR Forum, 2024.*

10. [234] Shuoqi Sun, Shengyao Zhuang, **Shuai Wang** and Guido Zuccon, An investigation of prompt variations for zero-shot llm-based rankers. In *European Conference on Information Retrieval*, pages 185–201. Springer, 2025.

11. [271] **Shuai Wang**, Shengyao Zhuang, Bevan Koopman and Guido Zuccon, ReSLLM: Large Language Models are Strong Resource Selectors for Federated Search , In *Companion Proceedings of the ACM on Web Conference 2025*, pages 1360–1364, 2025.

12. [72] Maik Fröbe, Andrew Parry, Harrisen Scells, **Shuai Wang**, Shengyao Zhuang, Guido Zuccon, Martin Potthast and Matthias Hagen, Corpus Subsampling: Estimating the Effectiveness of Neural Retrieval Models on Large Corpora, *In Proceedings of the 47th European Conference on Information Retrieval*, pp. 453-471 , 2025.

13. [270] **Shuai Wang***, Shengyao Zhuang*, Bevan Koopman and Guido Zuccon, 2D Matryoshka Training for Information Retrieval, *In Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp, 3125–3134, 2025.

14. [276] Zheng Yao, **Shuai Wang** and Guido Zuccon, Pre-training vs. Fine-tuning: A Reproducibility Study on Dense Retrieval Knowledge Acquisition,*In Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp, 3276–3285, 2025.

Other publications submitted & under-review

1. [281] Shengyao Zhuang*, **Shuai Wang***, Fabio Zheng, Bevan Koopman and Guido Zuccon, Starbucks: Improved Training for 2D Matryoshka Embeddings, *Submitted to ECIR, 2026.*

2. [268] **Shuai Wang**, Harrisen Scells, Bevan Koopman and Guido Zuccon, AutoBool: Reinforcement-Learned LLM for Effective Automatic Systematic Reviews Boolean Query Generation, *Submitted to ACL ARR October, 2025.*

Statement of Contribution

This thesis includes several publications that are co-authored with other researchers. In accordance with the University of Queensland's authorship policy (PPL 4.20.04), I hereby outline my contribution to each of the included works.

For publications listed in the front matter what are included in this thesis, I am the first author. I was primarily responsible for the conception and design of the studies, development of methods and systems, implementation of experiments, data analysis, and drafting of the manuscripts. I also coordinated the submission and revision processes.

My co-authors contributed in the following capacities:

- **Harrisen Scells**, **Bevan Koopman**, and **Guido Zuccon** provided supervision, conceptual feedback, and critical revisions across all publications.
- **Hang Li**, **Daniel Locke**, **Ahmed Mourad**, **Justin Clark**, **Martin Potthast**, and **Shengyao Zhuang** contributed to selected works through discussion of methodology, system implementation, evaluation strategy, or manuscript editing.

I affirm that the work presented in each publication represents my original contributions, and I accept full responsibility for the content of the versions included in this thesis.

This research was supported by an Australian Government Research Training Program Scholarship.

Statement of parts of the thesis submitted to qualify for the award of another degree

No works submitted towards another degree have been included in this thesis

Research involving human or animal subjects

No animal or human subjects were involved in this research.

*These authors contributed equally to this work

Acknowledgements

I am grateful for all the support, guidance, and encouragement I have received throughout my PhD.

Firstly, my heartfelt thanks go to my supervisors, Prof. Guido Zuccon, A. Prof. Bevan Koopman, and Dr. Harrisen Scells. Their constant support, insightful guidance, and genuine care have been instrumental in shaping my academic growth. Guido, despite his incredibly demanding schedule, consistently made time for me—often dedicating entire afternoons to deep discussions about research directions, ideas, and methods. I vividly recall numerous occasions when he stayed with us until midnight during submission deadlines, reviewing each paper and pushing us towards excellence. Guido's dedication, patience, and genuine care for his students not only inspired me academically but also taught me invaluable lessons about commitment, integrity, and passion for research. Harry offered extraordinary mentorship, especially during the early stages of my PhD. He invested countless hours in daily meetings, patiently teaching me the intricacies of academic writing and research methodology—often providing handwritten, line-by-line feedback on my drafts. Bevan's feedback was always precise and insightful, profoundly influencing my thinking and the quality of my work.

I owe special thanks to Dr Shengyao Zhuang, whose recommendation and guidance first introduced me to this group. It was during INFS7410 course at UQ that my passion for information retrieval truly began, and following the course, he kindly facilitated my initial internship focusing on the interpolation of dense and sparse rankers. I am deeply grateful to my wonderful friends and colleagues at ielab: my coffee mate Katya; always supportive Linh; Joel, the best chair in the world; sweet and caring Max; coding master Hang; Ismail&falmily with his unforgettable lamb; and JJ, the best chit-chat partner, and of course many others in the lab. Your friendship, encouragement, and the countless shared moments—through highs and lows—have made this journey truly unforgettable.

My research experience was enriched by visiting and collborating with Professor Martin Potthast at Leipzig University, whose expertise greatly broadened my academic perspective. I also met Maik and Ferdi at the time, who gave me lots of help during this visit (also my supervisor Harry) Later, during my internship at Naver Labs Europe, I benefited immensely from the supervision of Stephane Clinchant and established a lasting research connection with David Rau. I'm deeply grateful to all my mentors and collaborators during these periods.

I owe my deepest gratitude to my family—mom, Junli Guo; dad, Chaohui Wang; and sister, Meng Wang. You've always been my safe haven, offering endless love and support. My six years overseas before the PhD placed considerable burdens on our family, yet my mom supported me wholeheartedly and without hesitation. Your sacrifices and unwavering belief gave me strength—I am truly blessed and deeply thankful to have you by my side.

Finally, to Christine Lau, who unexpectedly entered my life during the final stages of my PhD. Falling in love with you has truly been the most beautiful surprise—you've shown me kindness beyond measure and warmth that touches me deeply every day. Your generosity, understanding, and genuine care constantly remind me how lucky I am. You deserve nothing but the very best, and I promise to always strive to be that for you.

Financial support

Shuai Wang is supported by a UQ Earmarked PhD Scholarship and this research is funded by the Australian Research Council Discovery Project DP210104043. This research was supported by an Australian Government Research Training Program Scholarship.

Keywords

Systematic Review, Technology-assisted Review, Boolean Query Formulation, Screening Automation, Screening Prioritisation, Domain-specific Search, Information Retrieval, Natural Language Processing

Australian and New Zealand Standard Research Classifications (ANZSRC)

ANZSRC code: 460508, Information retrieval and web search, 100 %

Fields of Research (FoR) Classification

FoR code: 0807, Library and Information Studies, 100 %

To My Mom, Dad and Christine

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Systematic reviews are the cornerstone of evidence-based healthcare, offering rigorous evaluations of literature to address focused research questions. Their methodology is critical across various domains—particularly in medicine and clinical practice—where they inform guidelines, support decision-making, and shape policy by systematically identifying, appraising, and synthesising evidence from diverse studies. For example, a systematic review might explore the high-impact question, "What is the effect of novel gene-editing therapies on patient survival and quality of life in advanced cancers?" This question is of high importance because advanced cancers remain a leading cause of mortality worldwide, and uncovering whether these cutting-edge treatments can significantly extend life and enhance patient wellbeing has profound implications for clinical practice, research investment, and overall healthcare strategies. By aggregating and analysing data from multiple research studies, systematic reviews provide the most robust and comprehensive evidence available, often guiding pivotal decisions in medical practice and healthcare policy.

However, as depicted in Figure 1.1, achieving this thorough standard is often associated with substantial costs in terms of time and labour [92, 112, 148, 153]. For instance, median value of the total time taken for a systematic review in the medical field can require 1,110 hours of expert effort, spanning the formulation of queries, screening of literature, and data synthesis [21, 112]. This intensive demand mainly arises from the need to ensure that reviews are exhaustive and unbiased, covering all relevant studies to avoid skewed outcomes that could misinform clinical practices or policy decisions.

1.1 How Systematic Reviews are Created

Systematic reviews are typically multi-staged, as illustrated in Figure 1.1. The process begins with defining a precise research question and developing a protocol that outlines the objectives, inclusion criteria, and methodology of the review. Next, a comprehensive search strategy is formulated, typically relying on Boolean queries to retrieve potentially relevant articles. Crafting these queries is inherently challenging, as they must strike a balance between sensitivity (capturing all relevant studies) and specificity (excluding irrelevant ones). This task is further complicated by the fact that

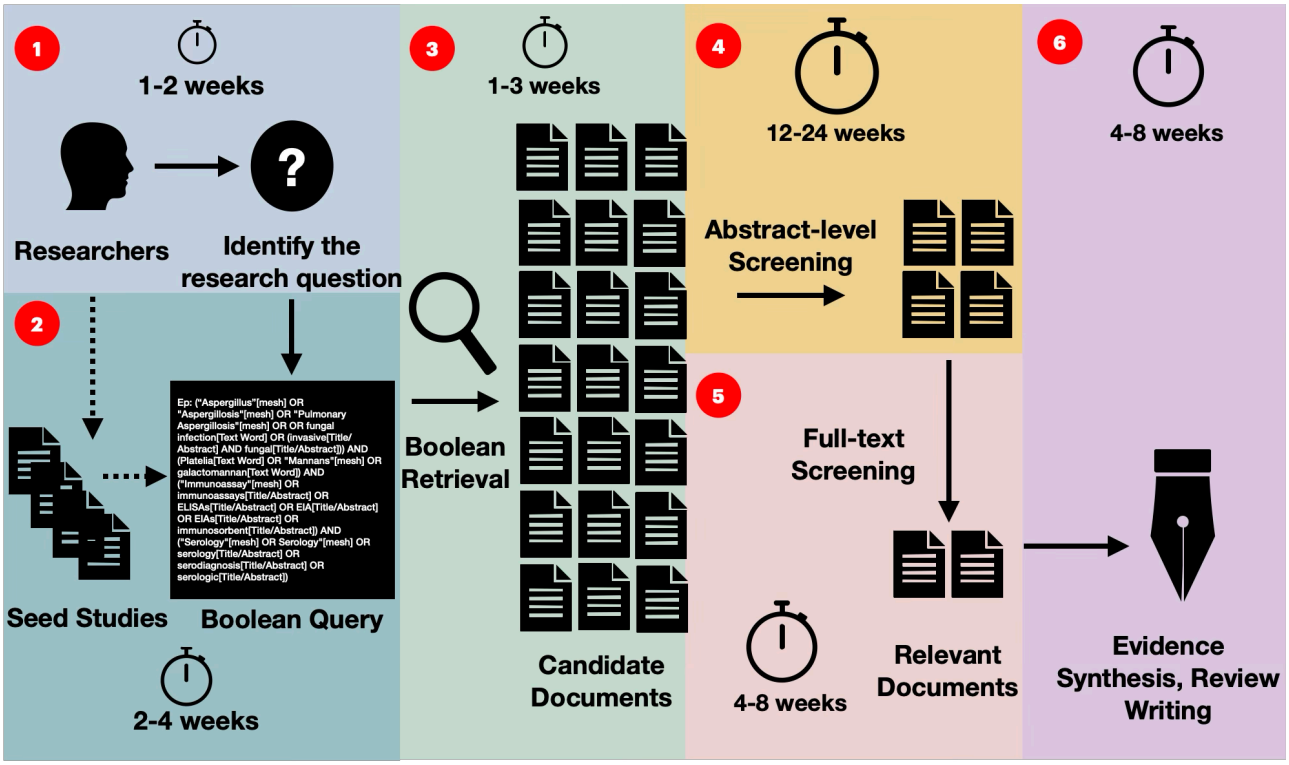


Figure 1.1: Overview of systematic review process, with approximate time estimation for each step.

those formulating the queries often lack full knowledge of the structure of the search space or the features that distinguish relevant from non-relevant documents. To mitigate this, researchers frequently begin with a small set of highly relevant studies—referred to as seed studies—which help ensure critical literature is captured and inform iterative refinements to the search terms and logic. Once candidate studies have been identified, they undergo a screening process at the title and abstract level to exclude clearly irrelevant records. Studies that pass this initial screening are then assessed in full (full-text screening) to confirm their eligibility based on predefined inclusion criteria, resulting in the final set of studies included in the review. Following screening, data extraction is performed to collect key information—such as outcomes, interventions, and study characteristics—from the included studies. Researchers also evaluate the methodological quality or risk of bias of each study to assess its reliability and relevance. Finally, the extracted data are synthesised—often via meta-analysis when appropriate—to generate overarching conclusions or evidence-based recommendations. Note the summary in this section provides only an *abstract overview* of the broader process, a more detailed description of how systematic reviews are created is introduced in Section 2.1.

Given the substantial time, workload, and cost involved at each stage, there is a pressing need for more efficient methodologies that can streamline systematic reviews without compromising the rigor and thoroughness that make them so valuable.

1.2 Focus of this Thesis

Amidst the challenges outlined above, this thesis investigates Artificial Intelligence (AI) methodologies designed to accelerate systematic reviews without compromising their quality or depth. In particular, it

targets the substantial time and human labour costs associated with the screening phase. To streamline this process, the research focuses on three strategic integrations of AI:

1. **Leveraging Additional Knowledge:** Building on established systematic review procedures to refine search strategies and enhance *screening prioritisation*. In this thesis, we concentrate on the use of *seed studies*. These seed studies inform the formulation of search queries and help rank or classify the literature so that studies with a higher likelihood of relevance are reviewed first. By prioritising items likely to be most pertinent, researchers can reduce the immediate screening burden and maintain a high level of thoroughness.

2. **Improving the Search Process:** Although current systematic reviews rely predominantly on Boolean queries, formulating these search strategies can be both complex and time-consuming. This thesis explores AI-driven approaches that can (partially) automate the creation of intricate Boolean queries, ensuring that relevant studies are accurately captured while minimising the inclusion of irrelevant material. By incorporating these techniques into the search phase, researchers can reduce the volume of literature requiring full review at later stages, thereby minimising the overall screening workload.

3. **Implementing AI-Driven Ranking and Classification Systems:** Finally, we introduce ranking and classification methods designed to automate and accelerate the screening process. By employing AI algorithms to classify studies for their relevance, these systems can significantly curtail the time-intensive manual screening often required in systematic reviews, freeing researchers to focus on higher-value tasks such as analysis and synthesis.

1.3 Contributions

Based on the three strategic AI integrations identified above, the contributions of this thesis are categorised into three components, (1) Exploiting Seed Studies, (2) Enhancing Boolean Search, and (3) Automating the screening process. The three components are split into three Parts in this thesis, detailed as follows:

1.3.1 Exploiting Seed Studies

Seed studies are a set of *priori known* studies—typically identified by systematic review researchers or Information Specialists—that are used to guide the formulation and refinement of the search strategy. Although the use of seed studies is not standardised in the systematic review process, it is a widely accepted practice among researchers to leverage these known studies to help develop effective search strategies, such as the creation of a Boolean query, or use as a guide for inclusion/exclusion during systematic review creation.

While seed studies have been explored in previous methodologies to aid in systematic review screening, such experiments often simulate the presence of seed studies using sampled included studies.

This approach can lead to over-optimism regarding the effectiveness of seed studies because included studies are selected post hoc and are inherently biased towards relevance. Consequently, simulating seed studies with included studies does not accurately represent the initial challenges and uncertainties faced during the search stages of a systematic review.

To address this gap, this thesis investigates the effectiveness of seed studies and seed-driven approaches by creating a new systematic review collection comprising actual seed studies. This collection aims to provide a more realistic assessment of seed studies' impact on the development of search strategies. The results of this investigation are detailed in Part 1, and include an analysis of how these real seed studies differ in utility compared to the simulated seed studies (sampled included studies).

Further discussion and findings related to this approach can be found in the following key publications, which are integral to understanding the advancements in using seed studies for systematic reviews:

1. **Shuai Wang**, Harrisen Scells, Ahmed Mourad and Guido Zuccon, Seed-Driven Document Ranking for Systematic Reviews: A Reproducibility Study, *In Proceedings of the 44th European Conference on Information Retrieval*, pp. 686-700, 2022.

2. **Shuai Wang**, Harrisen Scells, Justin Clark, Bevan Koopman and Guido Zuccon, From Little Things Big Things Grow: A Collection with Seed Studies for Medical Systematic Review Literature Search, *In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 3176-3186, 2022.

1.3.2 Enhancing Boolean Search

Boolean searches are fundamental to systematic reviews, leveraging logical operators (e.g., AND, OR, NOT) to retrieve publications aligned with a specific research question. As an illustration, for the question *What is the effect of novel gene-editing therapies on patient survival and quality of life in advanced cancers?*, one could construct a Boolean query similar to the example shown below.[1]

```
((("Neoplasms"[MeSH] OR cancer*[tiab] OR carcinoma*[tiab] OR tumor*[tiab])
AND (advanced[tiab] OR metastatic[tiab] OR "stage iv"[tiab]))
AND
(("Gene Editing"[MeSH] OR CRISPR[tiab] OR "CRISPR-Cas9"[tiab]
OR TALEN[tiab] OR ZFN[tiab] OR "Base Editing"[tiab] OR "Gene Therapy"[tiab]))
AND
(("Survival"[MeSH] OR "Mortality"[MeSH] OR "Quality of Life"[MeSH]
OR "overall survival"[tiab] OR "progression-free survival"[tiab]
OR "treatment outcome"[tiab] OR QoL[tiab])))
```

[1]This query is provided solely for illustrative purposes; its effectiveness has not been validated.

Designing an effective Boolean search query is a critical step in the systematic review process—not just a matter of correct syntax or technical formatting. A well-constructed query determines which studies are retrieved and, consequently, which evidence is considered. To improve both precision and recall, Boolean searches often incorporate controlled vocabularies such as Medical Subject Headings (MeSH) and apply field-specific restrictions—for example, limiting search terms to the title and abstract using tags like [tiab]. The quality of the search query directly influences the comprehensiveness and relevance of the retrieved literature, ultimately shaping the validity and reliability of the entire review.

This thesis introduces AI-based methods to refine query formulation, improving both search precision and efficiency, as detailed in Part 2. Specifically, two automated approaches are proposed in this thesis: (1) using generative large language models to directly formulate Boolean queries from research topics or questions, and (2) automatically suggesting MeSH terms during query construction. These developed methods demonstrate that leveraging AI-driven tools can expedite Boolean query formulation and potentially yield higher-quality systematic reviews. The following publications illustrate the methods and findings:

1. **Shuai Wang**, Harrisen Scells, Bevan Koopman and Guido Zuccon, Can ChatGPT write a good boolean query for systematic review literature search?, *In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1426-1436, 2023.

2. **Shuai Wang**, Hang Li, Harrisen Scells, Daniel Locke and Guido Zuccon, MeSH Term Suggestion for Systematic Review Literature Search *In Proceedings of the 25th Australasian Document Computing Symposium*, pp. 1-8, 2021.

3. **Shuai Wang**, Harrisen Scells, Bevan Koopman and Guido Zuccon, Automated MeSH term suggestion for effective query formulation in systematic reviews literature search, *In Intelligent Systems with Applications*, pp. 200141, 2022.

4. **Shuai Wang**, Harrisen Scells, Bevan Koopman and Guido Zuccon, MeSH Suggester: A Library and System for MeSH Term Suggestion for Systematic Review Boolean Query Construction, *In Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pp. 1176-1179, 2023.

1.3.3 Optimising Screening

Candidate documents retrieved from a Boolean query can be extensive and vary in relevance. AI-based screening methods help streamline subsequent stages of systematic review by directing reviewers to the most pertinent studies and filtering out irrelevant material. In this thesis, two core approaches are explored.

(1) Ranking of retrieved documents (screening prioritisation). Systematic reviews typically involve multi-stage screening, conducted by multiple reviewers in parallel. By prioritising highly relevant records first, reviewers can begin full-text assessments sooner and potentially reduce the overall time to complete the review. Early access to relevant studies also allows for faster synthesis and report drafting.

(2) Automatic classification of document relevance. Beyond prioritisation, this approach automatically identifies and excludes irrelevant or unrelated documents from the candidate pool. As a result, the manual screening workload is reduced, and reviewers can focus on evaluating a smaller, more relevant subset of studies.

Both approaches are discussed in Part 3 and are disseminated based on the following publications:

1. **Shuai Wang**, Harrisen Scells, Bevan Koopman and Guido Zuccon, Neural Rankers for Effective Screening Prioritisation in Medical Systematic Review Literature Search, *In Proceedings of the 26th Australasian Document Computing Symposium*, pp. 1-10, 2022.

2. **Shuai Wang**, Harrisen Scells, Bevan Koopman, Martin Potthast and Guido Zuccon, Generating Natural Language Queries for More Effective Systematic Review Screening Prioritisation, *In Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, pp. 73-83, 2023.

3. **Shuai Wang**, Harrisen Scells, Shengyao Zhuang, Martin Potthast, Bevan Koopman and Guido Zuccon, Zero-shot Generative Large Language Models for Systematic Review Screening Automation, *In Proceedings of the 46th European Conference on Information Retrieval*, pp. 73-83, 2024.

1.4 Structure of the Thesis

This thesis is organised thematically into three main parts, each addressing a core challenge in the automation of systematic reviews using Artificial Intelligence (AI). The chapters follow a sequential progression but are grouped according to shared methodological focus and research objectives. A visual summary is provided in Figure 1.2.

Chapter 2 introduces the principles of systematic reviews and surveys current approaches to automation, highlighting key bottlenecks—particularly in search and screening—and motivating the need for AI-driven solutions. **Part I** (Chapters 3–4) focuses on the use of seed studies to enhance search and screening, including a reproduction of existing methods and the creation of a new dataset. **Part II** (Chapters 5–6) addresses the formulation of Boolean queries, presenting an AI-based query generation method and a tool for MeSH term suggestion. **Part III** (Chapters 7–8) turns to the screening phase, proposing AI-based ranking and classification techniques for prioritisation and relevance prediction. Finally, **Chapter 9** summarises the thesis contributions and outlines future research directions.

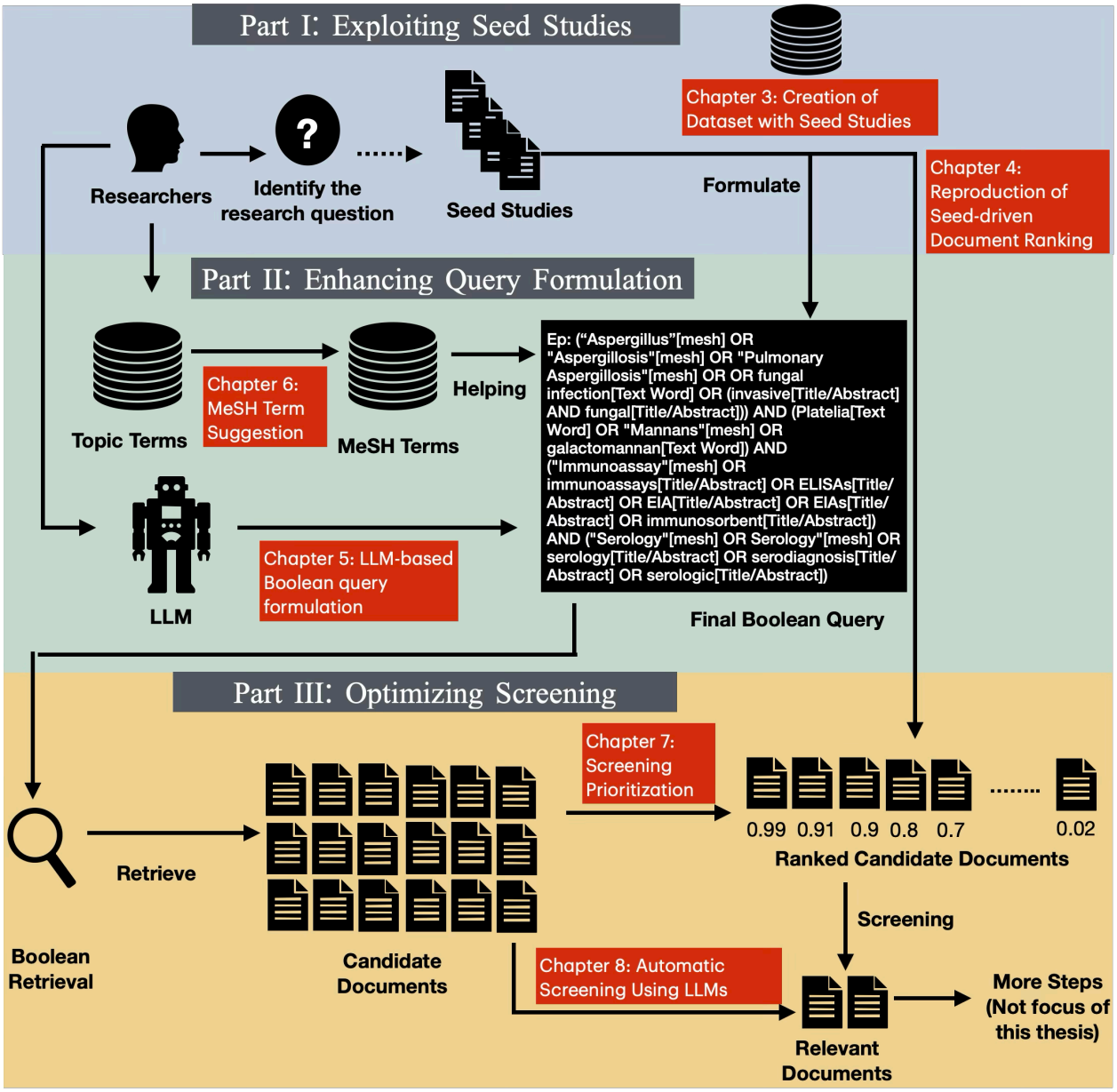


Figure 1.2: Thematic structure of the thesis, organised into three main parts.

Chapter 2

Background and Literature Review

The primary goal of a systematic review is to synthesise existing evidence in a transparent, reproducible manner, thereby minimising bias and maximising the reliability of conclusions. However, creating a systematic review is time-consuming, costly, and complex, involving multiple interdependent stages [3, 26, 150, 168]. Figure 2.1 presents a high-level overview of these stages.

To streamline this process, researchers have explored various automation methods, particularly for tasks such as query formulation [9, 212, 213, 214, 215, 216, 254] and screening prioritisation [49, 62, 128, 132, 183, 208, 227, 261, 282]. In this chapter, Section 2.1 offers a detailed overview of the systematic review process. Section 2.2 then discusses existing automation methods designed to facilitate systematic reviews, and Section 2.3 introduces curated datasets used to evaluate systematic review automation systems.

2.1 Systematic Review Process

The following subsections detail each stage of the systematic review process, highlighting where automation can play a significant role. Specifically, a systematic review involves five steps: Problem Identification, Boolean Query Formulation, Title–Abstract and Full-Text Screening, Quality Assessment and Data Extraction, and Reporting and Discussion.

Although this thesis concentrates on the first three steps, the latter stages are introduced here for completeness and could serve as future research directions.

2.1.1 Problem Identification with PICO Elements

The first stage involves precisely defining the scope and objectives of the review. This phase commonly uses the **PICO** framework [219]:

- **P (Population/Participants):** Define the specific population or participant group(s) under investigation (e.g., adults with a certain condition, software projects of a given size, users in a particular environment).

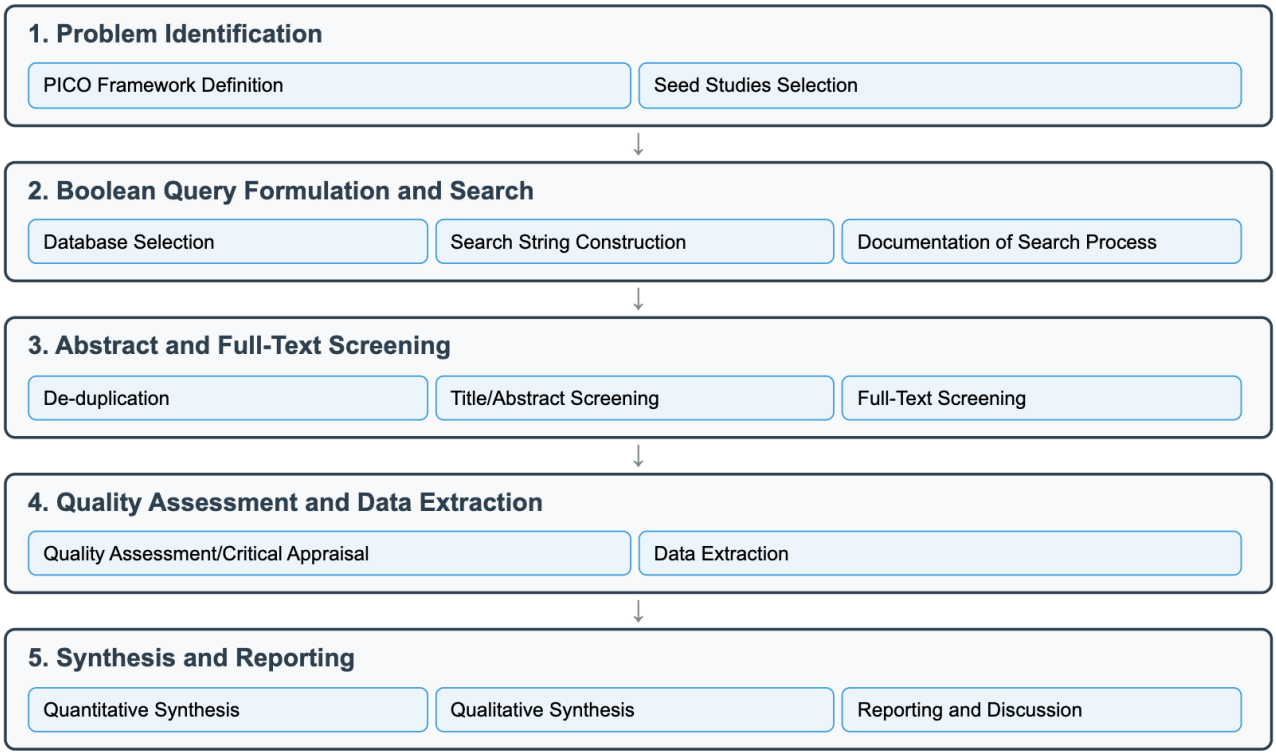


Figure 2.1: A high-level overview of different stages of the systematic review process.

- **I (Intervention/Exposure):** Clearly identify the intervention, treatment, or exposure to be studied (e.g., a new algorithm, a clinical procedure, a management strategy).
- **C (Comparison/Control):** State any comparisons with other interventions or controls (e.g., existing best practices, placebo, alternative methods).
- **O (Outcome):** Specify the outcomes or endpoints of interest (e.g., system performance, clinical improvement, user satisfaction).

These components help narrow down relevant studies and ensure consistency. Formulating a precise research question (or set of questions) is critical; for instance, a typical PICO question might be: *"In population P, does intervention I, compared to control C, lead to outcome O?"*. For example, consider the research question (as introduced in Chapter 2): *"What is the effect of novel gene-editing therapies on patient survival and quality of life in advanced cancers?"* In this case, the PICO components are defined as follows: **P**: Adults with advanced cancers; **I**: Novel gene-editing therapies (e.g., CRISPR-based treatments); **C**: Conventional treatment protocols; and **O**: Patient survival and quality of life measures.

Seed Studies Identification

As part of problem identification, researchers may select a set of **seed studies**. These are pre-identified, exemplary works known to be highly relevant, often seminal or high-quality papers in the domain [38, 96, 242]. The use of seed studies is somewhat related to explicit relevance feedback mechanisms (e.g., Rocchio [196]) in the IR domain, where relevant documents guide query refinement [46, 116, 127].

Active learning using explicit relevance feedback has also been extensively studied in the area of technology-assisted reviews [2, 47, 48, 49, 126, 128, 279, 282], which are high-recall tasks such as systematic review literature search, legal search, and patent search.

However, the difference between explicit relevance feedback and seed studies is subtle but important. Seed studies:

1. May not have been formally assessed as relevant to the final systematic review when provided to an information specialist.
2. Are not necessarily aligned with *all* of the inclusion criteria of a systematic review; for instance, seed studies might be non-randomised research or pilot studies but still provide important domain insight.

Despite this caveat, seed studies play a crucial role in systematic review literature search, particularly for:

- **Query formulation and validation:** Information specialists often use seed studies to identify or confirm relevant search terms (including synonyms, acronyms, related concepts).
- **Snowballing (citation chasing):** References within seed studies can reveal additional potentially relevant papers.
- **Evaluating search completeness:** If known seed studies are missing in initial search results, the search strategy may be suboptimal and, therefore, need refinement.

2.1.2 Boolean Query Formulation and Search

Once the research question is formulated according to the PICO elements (often in a medical context) or seed studies, the next step is to develop a comprehensive search strategy [34, 81, 98, 112, 173]. This is critical for identifying all relevant studies and minimising the risk of publication bias. The following subsections detail the recommended steps.

Database Selection

In medical systematic reviews, selecting the most pertinent bibliographic databases or digital libraries is paramount [24, 35, 151, 247]. Commonly used sources include:

- **PubMed/MEDLINE:** A primary database for biomedical and life sciences literature, leveraging both MeSH (Medical Subject Headings) and free-text searching [273]. PubMed currently contains over *35 million* citations; MEDLINE itself includes approximately *27 million* curated references.
- **EMBASE:** Covers a broad range of biomedical research and includes Emtree (EMBASE's subject headings) for controlled vocabulary [71]. EMBASE comprises roughly *37 million* records from more than *8,500* journals published in multiple languages.

- **Cochrane Library:** A key resource for systematic reviews, clinical trials, and evidence-based medicine [238]. It integrates several databases, including the Cochrane Database of Systematic Reviews (with around *8,000* reviews) and CENTRAL, which contains over *1 million* records of controlled clinical trials.

- **Web of Science, Scopus:** General academic databases that also include substantial biomedical literature [122]. The Web of Science Core Collection indexes over *100 million* records, while Scopus covers approximately *70–80 million* records from peer-reviewed literature, conference proceedings, and trade publications.

- **PsycINFO, CINAHL:** Highly relevant for reviews involving psychology or nursing/allied health perspectives, respectively [35]. PsycINFO holds over *4 million* records in psychology and related disciplines, and CINAHL includes around *6 million* records focusing on nursing and allied health literature.

Relying on just one database risks overlooking potentially relevant studies, as no single database comprehensively indexes all biomedical literature. To minimise this risk, most guidelines (e.g., PRISMA, Cochrane Handbook) advise searching multiple databases to ensure extensive coverage and reduce the likelihood of publication bias [34, 81, 98, 112, 173]. By consulting a variety of databases, systematic reviewers can capture both established research and emerging evidence that may only be indexed in specific or specialised resources. This multifaceted approach strengthens the quality of the review by maximising the pool of studies for possible inclusion.

Search String Construction

A robust search strategy in medical systematic reviews typically combines relevant free-text terms with controlled vocabulary terms (e.g., MeSH in PubMed, Emtree in EMBASE) [19, 157, 214]. Boolean operators (`AND`, `OR`, `NOT`) structure these terms. Researchers may also incorporate truncation (e.g., `child*` to capture *child*, *children*, *childhood*), wildcards, or proximity operators where allowed [99].

What is MeSH? Medical Subject Headings (MeSH) is the U.S. National Library of Medicine's (NLM) controlled vocabulary used primarily for indexing articles in PubMed/MEDLINE [67]. It organises biomedical concepts in a hierarchical tree structure, featuring both broad and narrow terms. Each term in MeSH represents a specific concept in medicine, such as diseases, chemicals, or procedures, ensuring consistency across the literature even when different authors use varied terminologies. MeSH indexing aids in retrieving all relevant articles related to a concept by automatically mapping synonyms and related terms to a single heading. Additionally, MeSH includes subheadings (qualifiers) that allow for more focused queries—for example, specifying particular aspects like complications, diagnosis, or therapy. This depth and structure make MeSH an indispensable tool for constructing comprehensive and precise search strategies in systematic reviews.

- **Controlled Vocabulary Terms:** Identify appropriate MeSH headings in PubMed or Emtree terms in EMBASE corresponding to the core concepts (P, I, C, O). For instance:

```
"Neoplasms"[MeSH] OR "Cancer"[Text Word]
```

- **Synonyms and Variations:** Include alternative or lay terms to maximise coverage. For example, if targeting the population of *advanced cancers*, consider using variations like *metastatic cancer* or *stage iv cancer*.
- **Boolean Operators:** Combine terms systematically, using `OR` to group synonyms and `AND` to intersect concepts. For example:

```
("breast cancer"[MeSH] OR "mammary carcinoma"[Text Word])
AND ("chemotherapy"[MeSH] OR "drug therapy"[Text Word])
```

- **Search Limits:** Apply filters (e.g., publication date range, language, or study design) only when justified. Overly restrictive limits may exclude critical evidence.

Example of a Complex Systematic Review Query

Below is an illustrative example of a combined Boolean query in PubMed for a review addressing the research question "What is the effect of novel gene-editing therapies on patient survival and quality of life in advanced cancers?"—in which the PICO components are integrated as follows: the Population (P) consists of adults with advanced cancers, the Intervention (I) involves novel gene-editing therapies (e.g., CRISPR-based treatments), the Comparison (C) is represented by conventional treatment protocols, and the Outcome (O) focuses on patient survival and quality of life outcomes; this query employs both MeSH terms and free-text terms along with Boolean operators and parentheses for clarity, applies the NOT operator to exclude undesired publication types (such as reviews), animal studies, or pediatric populations outside the scope, and limits results by language (English) and publication date—here is the complete query [1]:

```
((("Neoplasms"[MeSH] OR cancer*[tiab] OR carcinoma*[tiab] OR tumor*[tiab])
AND (advanced[tiab] OR metastatic[tiab] OR "stage iv"[tiab]))
AND
(("Gene Editing"[MeSH] OR CRISPR[tiab] OR "CRISPR-Cas9"[tiab]
OR TALEN[tiab] OR ZFN[tiab] OR "Base Editing"[tiab]
```

[1] This query is provided solely for illustrative purposes; its effectiveness has not been validated.

```
OR "Gene Therapy"[tiab]))
AND
(("Survival"[MeSH] OR "Mortality"[MeSH]
OR "Quality of Life"[MeSH]
OR "overall survival"[tiab] OR "progression-free survival"[tiab]
OR "treatment outcome"[tiab] OR QoL[tiab])))
NOT
("Review"[Publication Type] OR "Animals"[MeSH] OR "Child"[MeSH])
LIMIT: English[lang]
AND ("2000/01/01"[Date - Publication] : "2015/12/31"[Date - Publication])
```

In this query:

- The first grouping targets the **Population (P)**—focusing on advanced cancers—by including both MeSH terms and free-text variations.
- The second grouping captures the **Intervention (I)**, detailing novel gene-editing therapies.
- The third grouping specifies the **Outcomes (O)** related to survival and quality of life.
- The `NOT` operator excludes publication types and populations outside of the review scope (e.g., reviews, animal studies, pediatric populations).
- Publication language is limited to English and the publication date range is set to include studies from `"2000/01/01"` onward, ensuring relevance to contemporary literature.

Before finalising the search strategy, a pilot search helps confirm that known *seed studies* (high-quality clinical trials, seminal papers, etc.) appear among the results [69]. If any key seed studies do not appear, researchers refine search terms, add synonyms, or adjust controlled vocabulary. This iterative process ensures the final strategy is sufficiently sensitive to capture the body of evidence relevant to the medical research question [9, 94, 211, 217].

Why We Are Interested in Applying AI-Approaches for Systematic Review Boolean Query Formulation and Search?

Unlike conventional search tasks—where AI-based approaches have been widely applied for quick, ad-hoc searches focused on topical relevance—systematic reviews demand exhaustive coverage, stringent quality standards, and full transparency in study selection and evaluation. This methodology aims to minimise bias, ensure reproducibility, and synthesise evidence for clinical or scientific decision-making. Therefore, systematic review search strategies require more careful term expansion, frequent use of controlled vocabularies, and iterative testing to guarantee that all relevant articles are captured. These

unique demands motivate the development of new AI-driven approaches specifically tailored to the requirements of systematic reviewing, which is a key focus of this thesis.

Documentation of the Search Process

To ensure replicability, a thorough record of each search strategy is maintained. Good documentation enhances transparency, allowing other researchers to replicate the search or update it in the future [13]. Steps that should be documented include:

- Names of databases searched and the exact date(s) of search.
- Complete search strings used, including Boolean operators.
- Any filters or limits (language, publication year, document type) applied.

Moreover, such detailed documentation enables us to rerun searches during this thesis work and to create and update systematic review collections, thereby enhancing the reproducibility and currency of evidence syntheses.

2.1.3 Title-Abstract and Full-Text Screening

After retrieving the initial search results, the screening stage aims to filter out irrelevant or low-quality studies based on pre-established inclusion and exclusion criteria (derived from the PICO elements). In practice, *title and abstract screening* and *full-text screening* may be partially overlapping or conducted in parallel, especially when initial abstracts are unclear or when secondary reviewers are available to expedite the process. Moreoever, abstracts are generally publicly available, making the initial screening process relatively efficient. However, full-text reviews often require additional effort to obtain articles that may be behind paywalls or restricted access. This discrepancy necessitates a two-stage screening process. In large-scale reviews, some researchers may begin full-text screening concurrently for those studies where detailed information is available, effectively reducing the overall timespan by limiting full-text review to a smaller, pre-filtered set of potentially relevant records. Nonetheless, the systematic screening structure typically follows the steps below. This stage is known to contribute the most to the overall costs of a systematic review [156], highlighting the importance of IR-based methods such as *screening prioritisation* [49, 62, 128, 132, 183, 208, 227, 261, 282] to reduce the manual burden.

De-duplication

Multiple databases often return overlapping sets of results [23, 88, 189]. Therefore, the first step is to remove duplicate records using reference management software or specialised screening tools (e.g., EndNote, Mendeley, Zotero, Rayyan, or Covidence). This prevents redundancy and saves time in subsequent screening stages.

Title and Abstract Screening

Researchers then review the titles and abstracts of each unique record:

- If an abstract clearly matches the inclusion criteria (e.g., it involves the correct population and intervention) and does not violate any exclusion criteria, it is promoted to the next screening stage.
- If it is clearly irrelevant, it is excluded.
- In cases of uncertainty, the study is *included* (or flagged for further review) to avoid premature exclusion.

Typically, two or more reviewers conduct title and abstract screening independently to reduce selection bias, with disagreements resolved through discussion or by involving a third reviewer [231].

Full-Text Screening

Studies that pass the title and abstract screening are retrieved for full-text review [104, 120]. Researchers examine the entire text against the inclusion and exclusion criteria, focusing on aspects such as:

- **Research focus:** Does the article address the precise research question?
- **Methodological relevance:** Does the study employ valid and appropriate methods?
- **Population, intervention, outcomes:** Are these consistent with the PICO elements?

Any reason for exclusion is documented to maintain a transparent audit trail. The final output is a refined set of *included studies* for further analysis. In many systematic reviews, this stage can happen in parallel with continued title and abstract screening, particularly if multiple reviewers or screening teams are available. This parallel or iterative approach can expedite the overall process while still maintaining methodological rigor.

Snowballing (Citation Chasing)

Although the Boolean query strategy forms the core of the initial literature search, many systematic reviews also employ **snowballing** or **citation chasing** [37] to maximise comprehensiveness. Snowballing is performed:

- **Backward:** Checking the references listed in each included article to identify older or foundational studies.
- **Forward:** Checking which newer articles cite the included studies, capturing recent research that may not have been indexed under the same terms.

Snowballing can be critical; some investigations show that up to 51% of the included studies in a systematic review may be identified this way [82]. However, snowballing remains relatively under-explored in IR research [182]. As an additional retrieval strategy, it can substantially improve recall, especially when certain relevant studies are not retrieved by the initial Boolean query.

2.1.4 Quality Assessment and Data Extraction

With the final pool of eligible studies established, the next phase ensures each study is appraised systematically and valuable information is extracted in a structured way.

Quality Assessment (Critical Appraisal)

Researchers evaluate the methodological rigor and the risk of bias in included studies [222]. Standardised tools help maintain consistency. Examples include:

- **Jadad Scale or Cochrane Risk of Bias tool:** commonly used for clinical trials [110].
- **Critical Appraisal Skills Programme (CASP):** checklists covering qualitative and quantitative studies [142].

Quality assessment can result in a numeric score, a rating scale (e.g., low, medium, or high risk of bias), or a descriptive summary of strengths and weaknesses. Studies that fall below a minimum quality threshold may be excluded or flagged for sensitivity analysis [249].

Data Extraction

Researchers develop a data extraction form or spreadsheet to systematically capture relevant details [112]. Common elements include:

- **Bibliographic Information:** Title, authors, publication year, venue/journal.
- **Study Context and Design:** Type of study (e.g., experiment, quasi-experiment, survey, case study), sample size, setting.
- **Interventions and Comparisons:** Detailed description of the intervention, any control or comparison groups.
- **Measurement Outcomes:** Key variables, scales, or metrics, along with results (e.g., effect sizes, performance metrics, thematic findings).
- **Conclusions and Limitations:** Authors' main conclusions and any limitations noted.

This structured approach ensures consistency across reviewers and facilitates a unified data corpus for the synthesis stage. Pilot testing the data extraction form on a subset of studies helps calibrate reviewers, ensuring reliability in the extracted data.

2.1.5 Synthesis and Reporting

The final step involves synthesising and presenting the evidence in a coherent and meaningful way, guided by the original research questions [65].

Quantitative Synthesis (Meta-Analysis)

For studies reporting numerical data on the same or comparable outcome measures, a meta-analysis may be appropriate [163]. Techniques include:

- **Effect Size Calculation:** Converting study outcomes into standardised metrics (e.g., standardised mean difference, odds ratio) to allow pooling across studies [241].
- **Heterogeneity Assessment:** Examining the degree of variability between study results (e.g., using I^2 or Cochran's Q) [221].
- **Publication Bias Analysis:** Visualising using funnel plots or conducting tests such as Egger's regression test [220].

Meta-analysis yields an overall pooled estimate, potentially increasing statistical power and precision.

Qualitative Synthesis (Narrative/Thematic Synthesis)

In domains where interventions and outcomes differ significantly, or where qualitative data is predominant, a meta-analysis may be unsuitable [112, 144, 178]. Instead, researchers employ:

- **Narrative Synthesis:** Summarising and describing patterns, differences, and themes across studies without applying statistical pooling [187].
- **Thematic Analysis:** Coding key concepts or findings from each study to identify recurring themes, relationships, or conceptual frameworks [162].

This approach enables insights into contextual factors and nuanced perspectives that might otherwise be lost in purely quantitative aggregation [18, 112].

Reporting and Discussion

The final output of a systematic review should transparently describe:

- **Search and Selection Process:** Often illustrated via a PRISMA flow diagram, showing numbers of records identified, screened, and included/excluded with reasons.
- **Study Characteristics:** A summary table outlining key elements of each included study (authors, year, context, design, main findings).
- **Synthesised Findings:** Clear answers to the research questions, supported by evidence from the included studies.

- **Limitations:** Potential biases (e.g., publication bias, language bias, incomplete retrieval of grey literature) and study design limitations.
- **Implications and Recommendations:** How the results inform practice, policy, and future research avenues.

Overall, by adhering to this multi-stage process—ranging from defining a clear research question using PICO and selecting seed studies, to meticulously searching, screening, appraising, and synthesising relevant literature—researchers minimise bias and enhance the trustworthiness of the review. The systematic review process not only identifies what is known but also uncovers areas where further investigation is warranted, thereby guiding future research agendas. Additionally, IR methods (e.g., query formulation, screening prioritisation, and citation chasing) can help reduce the time and costs involved in creating systematic reviews [26, 112, 156].

2.2 Existing Automation Method for Systematic Reviews

Constructing systematic reviews requires manual effort by trained professionals across several phases, as identified in previous section.

Step 1: In Problem Identification where a specific research question is defined; software can help in defining a good question [112].

Step2: In Boolean Query Formulation and Search this is where Boolean queries based on the research question are developed. The query defines which (and how many) results are returned, so their quality greatly impacts all later phases. This is why we focus specifically on query generation. Automated methods have been developed for query formulation [41, 133, 198], query refinement [133, 207] and query exploitation [22, 59].

Step 3: In Title-abstract and Full text screening is where all the titles and abstracts retrieved by the query are screened, manually assessing them for relevance. The primary automation approach to help here is screen prioritisation [6, 10, 11, 36, 43, 63, 64, 100, 117, 129, 155, 158, 165, 166, 209, 225, 226, 274, 277], which involves ranking the studies according to their likely relevance to the research question. Once ranked, automated approaches can define a 'stopping criteria', after which screening does not continue because all relevant documents are likely to be found. Active learning approaches [45, 159] are often employed during screening to help rank include studies to be screened.

Step 4:In Quality assessment and Data Extraction which involves extracting specific details from studies relevant to the review; data extraction methods can help here [102, 232].

Step 5: In Synthesis and Reporting , this is the synthesis of all the evidence into a single coherent review document. Synthesis automation [40, 182, 197, 233, 240] have been developed here; they including text thematic analysis [243] and even text generation [188].

2.2.1 Query Formulation and Refinement

Query formulation is the process of deriving a Boolean query, based on the research question, according to a specific set of guidelines. Two guidelines are typically used for developing a query for systematic reviews, namely the *conceptual* method [38] and the *objective* method [97, 224]. These procedures describe the steps one should take when developing a query.

Conceptual Method

The first procedure is called the *conceptual* method. This procedure begins by identifying high-level concepts aligned with the components of the research question—often mapped to PICO elements (Population, Intervention, Comparison, Outcome). These concepts are typically identified from pilot searches or studies known a priori to be relevant. Synonyms, spelling variations, and controlled vocabulary terms (e.g., MeSH in PubMed or Emtree in EMBASE) are then collected for each concept. The information specialist constructs an initial query using Boolean logic (`AND`, `OR`, `NOT`) and iteratively refines the query to ensure retrieval of seed studies and maximise recall.

Using the research question "What is the effect of novel gene-editing therapies on patient survival and quality of life in advanced cancers?", a conceptual formulation may proceed as:

- Identify the main concepts: `advanced cancers`, `gene-editing therapies`, `survival and quality of life`.
- Find controlled vocabulary and synonyms: e.g., `"Neoplasms"[MeSH]`, `CRISPR`, `"Quality of Life"[MeSH]`.
- Combine terms using Boolean logic and validate against seed studies.

Objective Method

The second procedure is called the *objective* method. Like the conceptual method, it typically begins with a small set of seed studies—potentially relevant articles found through prior knowledge or initial exploratory searches. However, the next steps differ in that term selection is driven by statistical analysis rather than purely domain expertise.

Specifically, text mining or frequency analysis is applied to the titles, abstracts, and metadata (e.g., MeSH terms) of the seed set to identify commonly occurring and distinctive terms. These terms are then manually reviewed by the information specialist and incorporated into the Boolean query.

For the same research question, the objective method may proceed as:

- Select 5–10 known relevant studies on CRISPR and advanced cancer.
- Use tools such as PubReMiner or TF-IDF to extract top candidate terms from the titles, abstracts, and MeSH annotations.
- Manually group and combine terms, verify inclusion of seed studies, and adjust accordingly.

While both procedures can lead to similar Boolean queries, they differ significantly in how terms are selected and validated. The conceptual method relies on human expertise and domain familiarity, while the objective method introduces empirical support through automated analysis of real-world examples. These approaches can also be used in combination for greater robustness.

Naturally, these methods require a considerable amount of time (as the information specialist must perform multiple pilot searches and spend time validating the search) [117, 192] and are prone to human error [200, 202]. To this end, Scells et al. [216] investigated automating these two query formulation procedures. The main finding of this line of research was that computationalising these procedures could not match the effectiveness of humans; however, further manual refinement of the automatically generated queries dramatically improved retrieval effectiveness.

Automatic query refinement was developed based on the observed benefit from manual query refinement. Such methods take an initial human authors query and apply a series of transformations (adding terms or clauses) to make the query more effective [133, 207]. In combination with query visualisation [207] tools, these query refinement tools were able to improve the initial query.

Learnings from the query formulation approach drive two clear directions for using ChatGPT to automate query formulation: the first is to allow ChatGPT to generate queries however it sees fit, and the second is to guide ChatGPT by prompting it to follow the instructions of the conceptual or objective procedures. We refer to the first method as *unguided* and the second method as *guided*. Learnings from the query refinement work made us hypothesise that providing an existing query to ChatGPT and asking for a refinement could be beneficial. Note that both Unguided and Guided approaches are investigated as part of this thesis, detailed in Chapter 5.

Prompt Engineering for automatic Boolean query formulation

Prompt engineering is the process of guiding a generative language model to perform a particular task. In some respect, prompt engineering can be seen as a way to guide a generative language model through natural language [194]. A popular way of guiding model output through prompt engineering is for text-to-image generative models [140, 171]. This ‘zero-shot’ or ‘few-shot’ approach to tasks with generative language models has also achieved state-of-the-art results on several natural language tasks [25]. More recently, prompt engineering has been applied to natural language tasks for medicine, such as question answering [138, 145]. The use of generative language models for Boolean query formulation or refinement is relatively under-explored. However, prior work has examined related directions, such as using generative models for query expansion [42, 105]. Note that none of the existing studies has investigated the application of these methods for creating Boolean queries specifically in the context of systematic literature review search—prior to our research.

2.2.2 Systematic Review Screening Automation

Medical systematic reviews follow a standardised process to ensure consistency and quality. The most laborious part of the process is the screening of documents [223]. These documents are retrieved using

a complex Boolean query that attempts to precisely encode the information need required to answer the research question of the systematic review and to ensure the reproducibility of the review (i.e., re-running the search in the future should produce the same set of documents) [77, 148, 205].

Research into automating systematic review screening can be grouped into two primary directions, explored in detail below: (1) Screening Prioritisation, and (2) Leveraging Large Language Models (LLMs) for Screening.

Screening Prioritisation

Boolean queries retrieve an unordered set of documents. However, ranking the retrieved documents offers two major advantages:

1. Systematic reviews are often constrained by time or budget, particularly in the case of rapid reviews [154]. When screening all retrieved documents is infeasible, an effective ranking allows reviewers to prioritise the most relevant documents, increasing the chance of finding critical evidence within the available resources.
2. Systematic reviews typically involve a two-stage screening process: initial screening based on title and abstract, followed by full-text screening if the document is deemed potentially relevant. These stages are performed sequentially for each document. Therefore, frontloading relevant documents through effective prioritisation allows full-text screening to begin earlier, enabling downstream processes to proceed in parallel and shortening the overall review timeline.

This ultimately allows the whole systematic review to conclude earlier than if screening prioritisation was not implemented.

To support this need for effective prioritisation, researchers have explored various Information-retrieval-driven approaches to improve how candidate documents are ranked. These approaches range from traditional keyword-matching models to more sophisticated techniques that leverage semantic understanding of queries and documents. Recent advances in pre-trained language models such as BERT [61], RoBERTa [141], and T5 [186] have led to improvements in several downstream tasks, including document ranking [73, 135, 255], question answering [184], and conversational search [68, 184]. These models are typically pre-trained on large-scale corpora (e.g., Wikipedia [61] or PubMed [87, 130, 176]) to learn general-purpose linguistic and semantic features, and then either directly applied (*zero-shot*) or adapted to specific tasks via *fine-tuning* on task-specific examples.

Despite the success of such models in general-purpose ranking tasks, their application to screening prioritisation in systematic reviews—whether in a zero-shot or fine-tuned setting—has not been investigated in prior work. This thesis addresses this gap by exploring how large language models can be leveraged to support screening prioritisation, both with and without task-specific supervision.

In current practice, screening prioritisation for systematic reviews has been primarily studied using two main classes of approaches:

- **One-off Ranking:** Ranking methods that use a static query to generate a one-time ranked list of candidate documents [6, 10, 128, 204, 209, 261].

- **Iterative Ranking:** Methods that incorporate user feedback during the screening process and iteratively re-rank the remaining (unjudged) documents [47, 49, 50, 51].

These two approaches can also be used in tandem—for instance, beginning with a one-off ranking and updating the document order as user feedback is acquired.

In terms of the *queries* that can be used for screening prioritisation or document ranking, different types of data sources have been investigated in the context of systematic review automation. These include: (1) using the title of the review to represent the review topic, (2) using the Boolean query constructed during the search phase, and (3) using a set of seed studies as the basis for ranking.

- **Title-driven screening prioritisation.** Many existing methods rely on the final review title to represent the underlying information need. In this approach, words from the title are extracted and compared lexically to candidate documents using simple lexical matching techniques [113, 114, 115]. In this thesis, we shift the focus toward the potential of neural-based approaches, which remain largely unexplored in the context of systematic review screening prioritisation. A key limitation of title-based approaches is that the final review title is often unavailable or unstable during the early stages of the review process. As such, we treat the use of the final title as a simulated setting, rather than a realistic representation of early-stage systematic review workflows. This perspective is also adopted in our exploration of AI-driven approaches for screening prioritisation, detailed in Chapter 7.

- **Boolean-driven screening prioritisation.** Boolean queries have been used to guide screening prioritisation, often in combination with the review title [6, 8, 10]. In such cases, terms from both the Boolean query and the review title are extracted—typically using a bag-of-words representation—and a lexical similarity score is used to rank candidate documents. A major limitation of this method is that it assumes the availability of the final review title, which may not be known during the screening phase. The only known method designed to rely solely on the Boolean query is the coordination-level fusion (CLF) approach [208], which interprets the logical structure of the Boolean query and performs rank fusion over the ranked lists generated from its atomic components. This allows the method to more effectively leverage the Boolean query's syntax and semantics for prioritisation. In this thesis, we explore an alternative use of Boolean-driven screening prioritisation by transforming Boolean queries into natural language formulations using the generative capabilities of recent language models. This novel approach is also described in Chapter 7.

- **Seed-driven screening prioritisation.** Seed studies—documents known or suspected to be relevant—are frequently used by information specialists during query formulation and validation [38, 96]. Their use in screening prioritisation is conceptually similar to explicit relevance feedback in Information Retrieval, such as the Rocchio algorithm [196]. In general IR settings, relevance feedback has been successfully applied to query expansion [46, 116, 127], active learning [177], and user modelling [106].

However, in systematic reviews, seed studies should not be treated as equivalent to relevant documents. This distinction arises for several reasons: (1) seed studies may not have been formally assessed for relevance at the time they are used; (2) they may not meet the desired study design criteria—such as being randomised controlled trials, which are typically required for inclusion in systematic reviews [2]; and (3) some seed studies may be used as guidance or inspiration for formulating the review topic, rather than as definitive evidence for inclusion. As such, while seed studies can be helpful for query refinement or initial screening prioritisation, their relevance should be interpreted with caution.

Large Language Models for Screening Automation

While screening prioritisation focuses on ranking the documents retrieved by a Boolean query, an alternative and increasingly popular line of research is to automate the screening process itself—typically formulated as a binary classification task to determine inclusion or exclusion.

Screening automation seeks to bypass ranking and instead classify documents directly. The most common approach is to frame the problem as supervised text classification, where a model is trained on a subset of labelled documents to predict inclusion or exclusion [179, 240]. Early efforts employed traditional classifiers such as SVMs [44, 250], while more recent work has leveraged encoder-based LLMs such as BERT and BioBERT [15, 31]. These models are typically fine-tuned on systematic review data and updated incrementally using active learning [11, 30, 63, 64, 158, 226, 251, 275].

Recent advancements in instruction-based large language models (LLMs), such as ChatGPT, have demonstrated strong capabilities in following user instructions to complete complex tasks [79, 89, 199, 264]. These models, which often contain tens of billions of parameters, are pre-trained on large and diverse corpora, enabling them to generate coherent and contextually appropriate responses across a wide range of topics [79]. Several studies have evaluated the effectiveness of ChatGPT and similar models on downstream tasks such as question answering [169, 237] and ranking [107, 235].

Within the context of systematic review automation, recent studies have begun exploring the use of generative LLMs for document screening. Engene et al. [236] evaluated ChatGPT in imbalanced screening scenarios and reported limited effectiveness—a finding consistent with the class imbalance typical of systematic reviews, where included documents comprise only a small fraction of the total corpus. Robinson et al. [195] proposed Bio-SIEVE, a model fine-tuned from the Guanaco checkpoint [60], based on the LLaMA architecture. While Bio-SIEVE achieved higher classification accuracy, it exhibited inconsistent performance across different review topics. Both studies faced notable limitations, including reliance on closed-source models, limited or non-reproducible datasets, and insufficient evaluation metrics.

Prior to this thesis, no work has systematically evaluated open-source LLMs in a zero-shot setting for automated screening using publicly available datasets. Furthermore, existing evaluations have often overlooked key practical requirements—such as managing class imbalance and achieving high

[2] Randomised controlled trials with large sample sizes are generally considered high-quality evidence in systematic reviews, whereas case reports or observational studies may not be suitable for inclusion.

recall—which are essential for real-world systematic review workflows. A thorough investigation of these issues is presented in Chapter 8.

2.3 Datasets for Evaluating Systematic Review Automation

Several datasets are available for evaluating systematic review automation methods. These datasets primarily serve to benchmark automated systematic review screening systems, especially those leveraging machine learning and information retrieval techniques. Typically, these collections consist of academic articles with associated relevance judgments, essential for assessing and comparing automated approaches. Systematic review datasets are often smaller compared to other information retrieval (IR) tasks, such as general domain IR benchmarks like MS MARCO dev and TREC DL [53, 54], due to the complexity involved in creating systematic review datasets. Generally, CLEF TAR datasets are widely utilised for evaluating systematic review automation methods [113, 114, 115]. Additionally, as part of this thesis, we have contributed an extra dataset, known as the Seed Collection [258]. Recently, Kusa et al. [125] introduced a meta-collection named CSMED, synthesising multiple existing systematic review datasets.

2.3.1 CLEF TAR Collections

The CLEF technology-assisted review (TAR) datasets constitute prominent benchmark datasets explicitly designed to evaluate automation in systematic literature screening. CLEF TAR datasets include collections of academic articles paired with expert-generated relevance judgments.

CLEF TAR datasets comprise three sub-collections: CLEF-2017, CLEF-2018, and CLEF-2019, detailed as follows:

CLEF TAR 2017 [113] includes 50 systematic review topics focused on diagnostic test accuracy (DTA), known for their complexity. These topics are divided into 20 training and 30 testing topics.

CLEF TAR 2018 [114] extends the CLEF-2017 dataset by adding 30 additional DTA systematic review topics while excluding eight unreliable topics, resulting in a total of 72 topics.

CLEF TAR 2019 [115] further enhances previous collections by incorporating diverse types of systematic reviews. It introduces 20 additional DTA topics and 40 intervention-type topics, evenly split between 20 training and 20 testing topics.

2.3.2 Seed Collection

The Seed Collection is a dataset specifically developed as part of this thesis to support the evaluation of systematic review automation methods. It comprises 39 systematic review topics and over 50,000 candidate documents with associated relevance judgments.

Unlike previous datasets, the Seed Collection includes real-world seed studies and incorporates documents obtained through snowballing (citation chasing), capturing a more realistic and comprehensive screening scenario. A detailed description of the dataset construction, motivation, and usage is provided in Chapter 4.

2.3.3 CSMED Meta-Collection

The Citation Screening Meta-Dataset (CSMED) [125] is a recently introduced benchmark that unifies nine publicly available datasets to support the evaluation of automated citation screening systems in systematic reviews. It includes 325 systematic reviews across the biomedical and computer science domains and addresses several limitations of previous datasets, such as data leakage, lack of standardised train/test splits, and limited collection sizes. The datasets integrated into CSMED include those from Cohen et al. [45], Howard et al. [101], Alharbi and Stevenson [7], Hannousse and Yahiouche [93], and Scells et al. [205], as well as the CLEF TAR 2017 [113], CLEF TAR 2018 [114], and CLEF TAR 2019 [115] collections. It also includes an updated version of the Alharbi and Stevenson dataset [9].

Note that as CSMED was introduced relatively late in the development of this thesis, it was not used in the experiments but is included here for completeness.

Part I
Exploiting Seed Studies

Information specialists often use *seed studies*—exemplar documents known a priori and often provided by systematic review researchers—to assist in creating effective queries. Seed studies are employed in various ways, such as identifying potential terms or phrases for Boolean query construction and initial validation of search strategies by analysing precision and recall against these documents.

The use of seed studies has garnered attention within the Information Retrieval community to advance systematic review creation, leading to techniques leveraging them for automatic query formulation [216] and screening prioritisation (i.e., ranking retrieved studies) [128, 261]. However, a significant limitation of previous work lies in the collections used: they often lack real-world seed studies—these works circumvent this limitation by using a small portion of the final relevant studies as pseudo seed studies.

While pseudo seed studies are derived from documents known to be relevant and included in a systematic review, we argue that their use may overestimate the effectiveness of any AI methods (such as ranking of retrieved documents) relying on them. Since these documents are inherently biased towards relevance, the evaluation outcomes may not accurately reflect real-world scenarios where seed studies are less well-defined.

This thesis addresses these limitations in two key ways. First, we validate one influential seed-driven approach that prioritises systematic review rankings [128], using the existing methodology with pseudo seed studies; this is shown in Chapter 3. Then, in Chapter 4, we create a new systematic review collection that incorporates real seed studies. This allows us to further validate seed-driven methods in a more realistic setting, providing insights into their performance under practical conditions.

Chapter 3

Reproduction of Seed-driven Document Ranking

This chapter focuses on validating existing methodologies that utilise seed studies for automating systematic review creation. A key task in this context is *screening prioritisation*—ranking retrieved studies based on their likelihood of inclusion—to help reviewers assess the most promising studies first and thereby accelerate the review process. A notable approach, Seed-driven Document Ranking (SDR), proposed by Lee and Sun [128], leverages seed studies specifically for this ranking task during systematic review creation.

However, the challenges present at the time of SDR's publication (in 2018) limited its broader evaluation—available datasets for researching screening prioritisation were scarce. The original SDR was evaluated only using CLEF TAR 2017, which consists of 50 systematic review topics focused on diagnostic test accuracy. Consequently, subsequent research in this domain often omitted SDR from comparative analyses, leaving its generalisability and robustness underexplored. Later in 2018 and 2019, CLEF TAR introduced additional benchmark datasets specifically designed for evaluating technology-assisted review methods, including screening prioritisation. These datasets provided a more standardised foundation for assessing retrieval and ranking techniques. However, by that time, research focus had shifted toward newer approaches, and SDR was largely left out of subsequent evaluations. As a result, there remains an open question regarding how well SDR performs across these more diverse and representative datasets.

We devise the following research questions (RQs) to guide our investigation into why we are interested in reproducing the SDR method:

RQ1 *Does the effectiveness of SDR generalise beyond the CLEF TAR 2017 dataset?* The original study was evaluated on a single dataset of systematic review topics. In this study, we use our replicated implementation of SDR to examine its effectiveness across more recent and diverse topics (CLEF TAR 2017 only contains systematic reviews about diagnostic test accuracy).

RQ2 *What is the impact of using multiple seed studies collectively on the effectiveness of SDR?* The

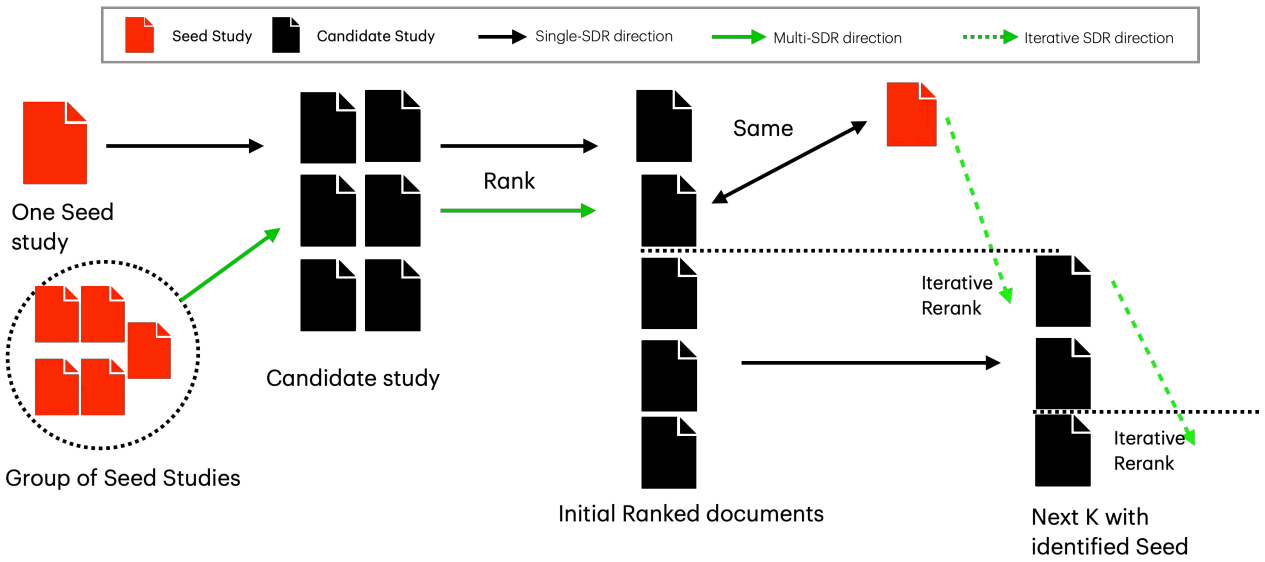


Figure 3.1: Architecture of Single-SDR, Multi-SDR and Iterative-SDR.

original study focused on two aspects: an initial ranking using a single seed study, and an iterative ranking that uses the remaining seed studies one at a time. We investigate the first aspect, focusing on the impact of multiple seed studies (multi-SDR) used collectively to produce an initial ranking.

RQ3 *To what extent do seed studies impact the ranking stability of single- and multi-SDR?* Seed studies have been shown to significantly affect the effectiveness of resulting queries, as demonstrated by Scells et al. [213]. Inspired by this finding, we conduct a similar analysis to measure the variance in SDR effectiveness under both single- and multi-seed study settings.

The above three research questions (1) demonstrate the **novelty** of SDR method by conducting experiments on more datasets (**RQ1**), and experiments that further explore the method's effectiveness (**RQ2**, **RQ3**), (2) assess the **impact** of SDR on the Information Retrieval community and the broader systematic review community, (3) evaluate the **reliability** of SDR by comparing it to several baselines on publicly available datasets, and (4) make our complete reproduction of SDR publicly **available** for use as a baseline in future work on re-ranking for systematic reviews.

3.1 Replicating Seed-driven Document Ranking

In the original paper by Lee and Sun, two experimental settings for SDR are proposed: an initial ranking of retrieved studies using one seed study (Single-SDR), and iterative re-ranking by updating the query with one seed study at a time (Iterative-SDR) to simulate the manual screening process. Our reproduction focuses on the initial ranking stage, which we extend to scenarios using multiple seed studies simultaneously. This focus is justified for two reasons: (1) screening prioritisation is an accepted practice in systematic review creation, as all studies must still be screened [34]; and (2) an effective initial ranking naturally leads to more efficient re-ranking, as relevant studies are identified earlier. Figure 3.1 provides an overview of SDR approaches, illustrating Single-SDR, Multi-SDR, and Iterative-SDR. While this paper focuses on Single-SDR and Multi-SDR experimental settings,

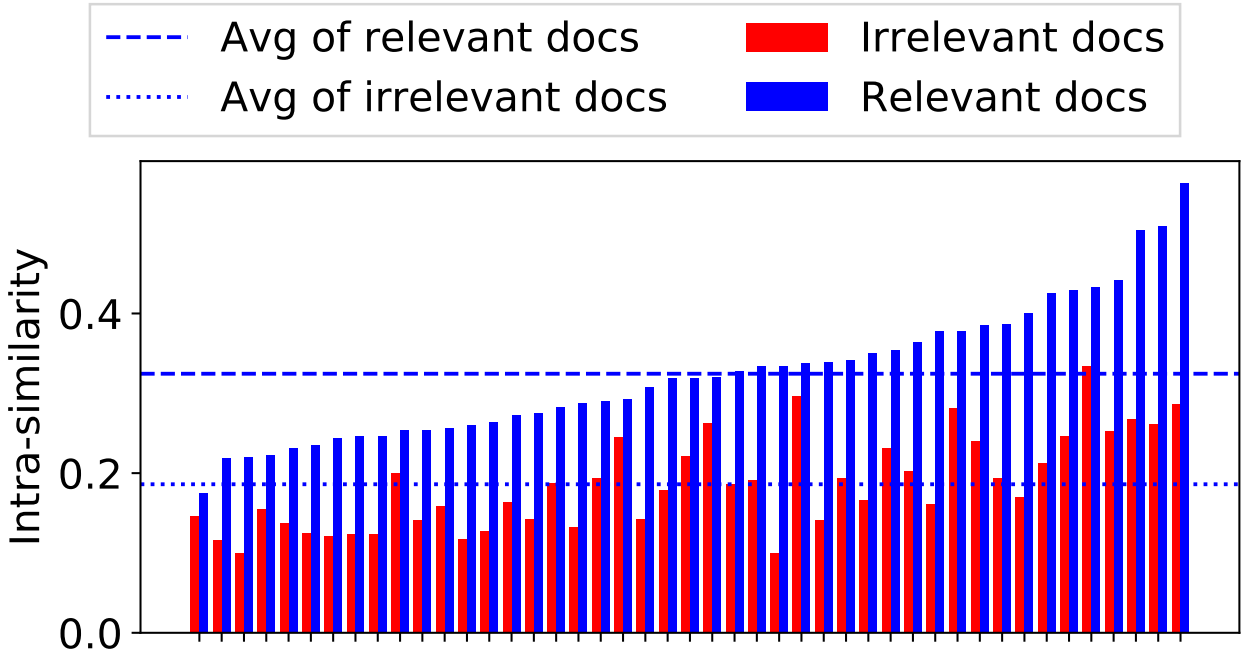


Figure 3.2: Intra-similarity between relevant and irrelevant studies. Each tick on the x-axis represents a review topic from CLEF 2017.

Iterative-SDR is shown for completeness to contextualise our work within the broader SDR framework. The intuition behind SDR is that relevant studies are similar to each other. The original paper makes two key observations to support this idea: (1) relevant studies are more similar to each other than to non-relevant studies; and (2) relevant studies share many *clinical terms*. These observations guide the representation and scoring of documents based on a seed study. We attempt to replicate these observations below to verify that our implementation follows the same methodology and to test whether the underlying assumptions still hold.

Observation 1: *For a given systematic review, its relevant documents share higher pairwise similarity than that of irrelevant documents.*

We find that this observation holds in our reproduction: as shown in Figure 3.2, for every topic, relevant documents exhibit higher pairwise similarity than non-relevant ones. To generate this plot, we followed the original SDR methodology from Lee and Sun [128] and randomly sampled the set of non-relevant documents ten times for each topic so that the number of non-relevant documents matched the number of relevant ones. However, we observe that the absolute values of intra-similarity differ from those reported by Lee and Sun [128]. This discrepancy may be due to differences in data sources and preprocessing. For example, one likely factor is the evolution of PubMed content over time. The original study likely used a snapshot from around 2017, whereas we use the a later version of PubMed (2021). Since the CLEF TAR dataset does not specify the exact date of extraction, we are unable to replicate the original data state precisely.

Observation 2: *Relevant documents for a given systematic review share high commonality in terms of clinical terms.*

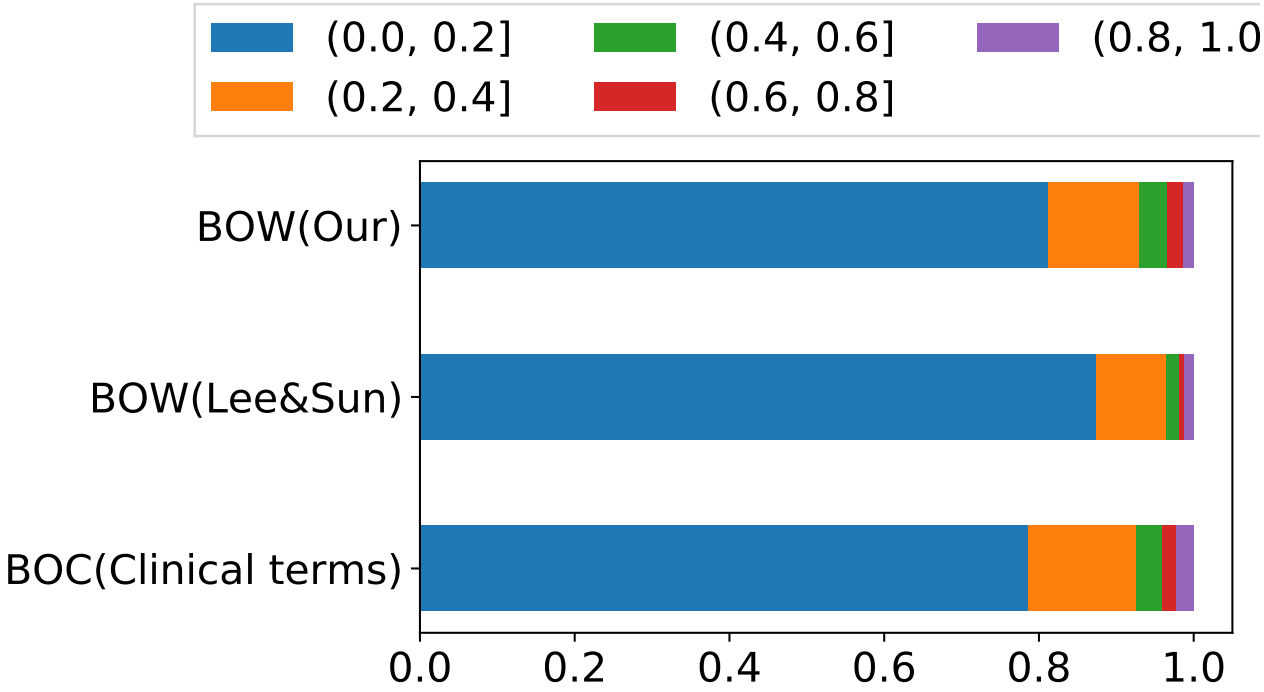


Figure 3.3: Distribution of terms in relevant studies using BOW (Bag of Words) and BOC (Bag of Clinical Terms), as described in Observation 2.

This observation also holds in our reproduction, as shown in Figure 3.3. Following the original study [128], we define a *clinical term* as any term matching an entry in the Unified Medical Language System (UMLS), identified using QuickUMLS. A document is represented either as a bag of words (BOW) with standard stopword removal, or as a bag of clinical terms (BOC), where only UMLS-matching terms are retained. To measure commonality among relevant documents, we compute the normalised document frequency (DocFreq) of each term and group terms into five bins: (0.0, 0.2), [0.2, 0.4), [0.4, 0.6), [0.6, 0.8), and [0.8, 1.0], where a DocFreq of 1.0 indicates the term appears in all relevant documents for a given topic. As reported by Lee and Sun [128], clinical terms (in BOC) appear more frequently in the higher DocFreq bins than generic words (in BOW), indicating stronger term commonality among relevant documents. Our reproduction confirms this trend. While our BOC and BOW distributions do not perfectly match the original plot, the relative patterns are consistent: BOC terms show much greater shared usage across relevant documents than BOW terms. We also found that with minor modifications to the pre-processing pipeline, we achieved slightly higher commonality for BOW terms compared to the original. This is likely due to differences in vocabulary size. When using the original paper's preprocessing, our BOW vocabulary had 1,612,974 terms, and BOC terms accounted for only 4.6% of this space. With our improved preprocessing, BOC terms cover 31.2% of our reduced BOW vocabulary, which is just 14.8% the size of the original. Naturally, a smaller vocabulary increases the likelihood of shared terms, which may partially explain the differences in absolute values. Note that BOC is a subset of BOW.

3.1.1 Document Representation

Given Observation 1 about relevant studies for this task, Lee and Sun chose to represent studies as a 'bag of clinical words' (BOC). They chose to use the Unified Medical Language System (UMLS) as their ontology of clinical terms. UMLS is an umbrella ontology that combines many common medical ontologies such as SNOMED-CT and MeSH. In order to identify UMLS concepts (and therefore the clinical terms) within the studies, Lee and Sun combine the outputs of the NCBO Bioportal [167] API[1] and QuickUMLS [228]. We follow their process as described, however we are not aware if it is not possible to set a specific version for the NCBO API. We use QuickUMLS version 1.4.0 with UMLS 2016AB.

3.1.2 Term Weighting

SDR weights terms based on the intuition that terms in relevant studies are more similar to each other (or occur with each other more frequently) than non-relevant studies. The weight of an individual term in a seed study is estimated by measuring to what extent it separates similar (pseudo-relevant) and dissimilar (pseudo-non-relevant) studies. Formally, each term t_i in a seed document d_s ($t_i \in d_s$) is weighted using the function $\varphi(t_i, d_s) = \ln\left(1 + \frac{\gamma(D_{t_i}, d_s)}{\gamma(D_{\bar{t}_i}, d_s)}\right)$, where D_{t_i} represents the subset of candidate studies to be ranked where t_i appears, and $D_{\bar{t}_i}$ represents the subset of candidate studies to be ranked where t_i does not appear. The average similarity between studies is computed as $\gamma(D, d_s) = \frac{1}{|D|}\sum_{d_j \in D} sim(d_j, d_s)$, where *sim* is the cosine similarity between the vector representations of the candidate study d_j and the seed study d_i. We follow the original implementation and represent studies as tf-idf vectors.

3.1.3 Document Scoring

The original SDR implementation uses the Query Likelihood Model with Jelenik-Mercer smoothing for scoring studies. Typically, this ranking function is derived as indicated by QLM shown in Equation 3.1, where $c(t_i, d_s)$ represents the count of a term in a seed study, $c(t_i, d)$ represents the count of a term in a candidate study, L_d represents the number of terms in a study, $p(t_i|\mathbb{C})$ represents the probability of a term in a background collection, and λ is the Jelenik-Mercer smoothing parameter. To incorporate the term weights as described in Subsection 3.1.2, the original paper includes φ function into the document scoring function as shown in Equation 3.1:

$$score(d, d_s) = \sum_{t_i \in d, d_s} \overbrace{\varphi(t_i, d_s)}^{\text{Term Weight}} \cdot \overbrace{c(t_i, d_s) \cdot \log\left(1 + \frac{1-\lambda}{\lambda} \cdot \frac{c(t_i, d)}{L_d \cdot p(t_i|\mathbb{C})}\right)}^{\text{QLM}} \tag{3.1}$$

where $p(t_i|\mathbb{C})$ is estimated using maximum likelihood estimation over the entire candidate set of studies C. In the original paper, when additional seed studies were ranked in the top-k set of candidate seed studies (denoted as $d_{s'}$), a re-ranking was initiated by expanding each t_i in d_s with the new terms from $d_{s'}$. For our replication study, we only consider the initial ranking of candidate studies, as an

[1] `http://data.bioontology.org/documentation`

abundance of baseline methods can be used as a comparison for this task. It is also arguably the most important step as a poor initial ranking will naturally result in a less effective and less efficient re-ranking.

3.1.4 Multi-SDR

One assumption in the original paper is that only a single seed study can be used at a time for ranking candidate studies. We propose a modification by studying the impact of using multiple seed studies collectively. In practice, it is common for Boolean queries (i.e., the search strategies used to retrieve the set of candidate studies we use for ranking) to be developed with a handful of seed studies, not just a single seed study. We hypothesise that the effectiveness of SDR will increase when multiple seed studies are used. Each relevant study must be used as a seed study for ranking, as the seed studies are not known in any of the collections we used. Therefore the average performance across topics was recorded (i.e., leave-one-out cross-validation). This study follows the methodology for the single-SDR method described in the subsections above. How we adapt single-SDR for a multi-SDR setting, and how we make this comparable to single-SDR is described as follows.

Grouping Seed Studies

To study multi-SDR, we randomly group multiple seed studies together and perform leave-one-out cross-validation over these groups. To account for any topic differences that may impact performance, we use a sliding window across the list of seed studies so that a seed study can appear in multiple groups. The number of seed studies to fill each group was chosen to be 20% of the total seed studies. Rather than use a fixed number of seed studies, choosing different proportions simulates the use of seed studies in practice, i.e., different amounts of seed studies may be known before conducting a review.

Combining Seed Studies for Multi-SDR

The way we exploit multiple seed studies for SDR is, we believe, similar to how Lee and Sun used multiple seed studies in their relevance feedback approach to SDR. We concatenate seed studies together such that the resulting representation can be used directly with the existing single-SDR framework. We acknowledge that there may be more sophisticated approaches to exploit multi-SDR. However, we leave this as future work as it is out of the scope for this reproducibility study. When computing term weights for multi-SDR, we also encountered computational infeasibility for large groups of seed studies. To this end, we randomly sampled the number of irrelevant studies to 50 each time we compute φ.

Comparing Single-SDR to Multi-SDR

Directly comparing the results of multi-SDR to single-SDR is not possible due to the leave-one-out cross-validation style of evaluation used for single-SDR. To address this, we apply an oracle approach

to identify the most effective single-SDR run out of all the seed studies used for a given multi-SDR run in terms of MAP. We then remove the other seed studies used in the multi-SDR run from the oracle-selected single-SDR run so that both runs share the same number of candidate studies for ranking.

3.2 Experimental Setup

3.2.1 Datasets

When the original SDR paper was published, only a single collection with results of baseline method implementations was available. We intend to assess the generalisability of their SDR method on several new collections which have been released since. The collections we consider are:

CLEF TAR 2017 [113] This is the original dataset that was used to study SDR. We include this dataset to confirm that we achieve the same or similar results as the original paper. This collection includes 50 systematic review topics on diagnostic test accuracy – a type of systematic review that is challenging to create. The 50 topics are split into 20 training topics and 30 testing topics. In our evaluation, we removed topics CD010653, CD010771, CD010386, CD012019, CD011549 as they contained only a single or no relevant studies to use as seed studies. For our experiments using multiple seed studies, we further removed topics CD010860, CD010775, CD010896, CD008643, CD011548, CD010438, CD010633, CD008686 due to low numbers of relevant studies.

CLEF TAR 2018 [114] This collection adds 30 diagnostic test accuracy systematic reviews as topics to the existing 2017 collection; however, it also removes eight because they are not 'reliable for training or testing purposes. In total, this collection contains 72 topics. Our evaluation only used 30 additional reviews of the 2018 dataset and removed topics CD012216, CD009263, CD011515, CD011602, and CD010680 as they contained only a single or no relevant studies to use as seed studies. We also removed topic CD009263 because we ran into memory issues when running experiments on this topic due to many candidate documents (approx. 80,000). For our experiments using multiple seed studies, we removed topics CD012083, CD012009, CD010864, CD011686, CD011420 due to low numbers of relevant studies.

CLEF TAR 2019 [115] This collection further develops on the previous years' by also including systematic reviews of different types. From this collection, we use the 38 systematic reviews of interventions (i.e., a different type of diagnostic test accuracy).[2] We use this collection to study the generalisability of SDR on other kinds of systematic reviews. In our evaluation, we removed topics CD010019, CD012342, CD011140, CD012120, CD012521 as they contained only a single or no relevant studies to use as seed studies. For our experiments using multiple seed studies, we further

[2]Although the overview paper claims there are 40 interventions topics, there are two topics that appear in both training and testing splits. However, like the previous datasets, we ignore these splits and combine the training and testing splits.

removed topics CD011380, CD012521, CD009069, CD012164, CD007868, CD005253, CD012455 due to low numbers of relevant studies.

3.2.2 Baselines

The baselines in the original paper included the best performing method from the CLEF TAR 2017 participants, several seed-study-based methods, and variations of the scoring function used by SDR. For our experiments, we compare our reproduction of SDR to all of the original baselines that we have also reproduced from the original paper. The baselines in the original paper include: BM25-{BOW,BOC}, QLM-{BOW,BOC}, SDR-{BOW,BOC}, and AES-{BOW,BOC}. The last method, AES, is an embedding-based method that averages the embeddings for all terms in the seed studies. The AES method uses pre-trained word2vec embeddings using PubMed and Wikipedia, following Lee and Sun [128]. We also include a variation that uses only PubMed embeddings (AES-P). Finally, we include the linear interpolation between SDR and AES, using $\alpha = 0.3$ as the interpolation parameter following Lee and Sun [128].

3.2.3 Evaluation Measures

For comparison to the original paper, we use the same evaluation measures. These include MAP, precision@k, recall@k, LastRel%, and Work Saved over Sampling (WSS). LastRel is a measure introduced at CLEF TAR'17 [113]. It is calculated as the rank position of the last relevant document. LastRel% is the normalised percentage of studies that must be screened in order to obtain all relevant studies. Work Saved Over Sampling; a measure initially proposed to measure classification effectiveness [45], is calculated instead here, by computing the fraction of studies that can be removed from screening to obtain all relevant documents; i.e., $\text{WSS} = \frac{|C| - \text{LastRel}}{|C|}$. Where C is the number of studies originally retrieved (i.e., the candidate set for re-ranking). For precision@k and recall@k, we report much deeper levels of k: the original paper reported $k = \{10, 20, 30\}$; where we report $k = \{10, 100, 1000\}$. Furthermore, we also report nDCG at these k-values, as it provides additional information about relevant study rank positions. We compute LastRel% and WSS using the scripts used in CLEF TAR 2017. For all other evaluation measures we use `trec_eval` (version 9.0.7).

3.2.4 Document Pre-Processing

It is widely known that document pre-processing (e.g., tokenisation, stopword removal, stemming) can have a significant impact on ranking performance [55]. In the original SDR implementation by Lee and Sun [128], Gensim version 3.2.0 was used to preprocess documents. According to the paper, documents were split using whitespace, and stopwords were removed using the NLTK library. However, although the original paper specifies the versions of the libraries used for ranking, it provides limited detail on key preprocessing steps—such as the specific stopword list, tokenisation rules, or punctuation handling.

Table 3.1: Reproduction results of baselines and SDR methods on the CLEF TAR 2017 dataset. For BOW methods, the pre-processing pipeline used by Lee and Sun is denoted by '-LEE'. BOW methods that do not have this demarcation correspond to our pipeline. For AES methods, word2vec PubMed embeddings are denoted by '-P'. AES methods that do not have this demarcation correspond to word2vec embeddings that include PubMed and Wikipedia. Statistical significance (Student's two-tailed paired t-test with Bonferonni correction, $p < 0.05$) between the most effective method (SDR-BOC-AES-P) and all other methods is indicated by †.

Method	MAP	Prec. 10	Prec. 100	Prec. 1000	Recall 10	Recall 100	Recall 1000	nDCG 10	nDCG 100	nDCG 1000	LR%	WSS
Sheffield-run-2[6]	.1706	.1367	.0703	.0156	.1759	.5133	.8353	.2089	.3342	.4465	.4660	.5340
BM25-BOW-LEE	.1710†	.2027†	.0867†	.0195†	.1543	.5118†	.8798†	.2439†	.3419†	.4770†	.4902†	.5098†
BM25-BOW	.1810	.2128†	.0898†	.0200	.1646	.5232†	.8928	.2560	.3534†	.4899†	.4427†	.5573†
BM25-BOC	.1764†	.2145†	.0895†	.0200	.1562	.5215†	.8944	.2539	.3496†	.4871†	.4401†	.5599†
QLM-BOW-LEE	.1539†	.1846†	.0778†	.0184†	.1367†	.4664†	.8508†	.2198†	.3091†	.4454†	.4662†	.5338†
QLM-BOW	.1973	.2360	.0964	.0203	**.1855**	.5464	.9081	**.2827**	.3772	.5100	.3851	.6149
QLM-BOC	.1894	.2330	.0951	.0202	.1809	.5376	.9032	.2771	.3684	.5018	.3936	.6064
SDR-BOW-LEE	.1533†	.1777†	.0780†	.0185†	.1304†	.4710†	.8576†	.2142†	.3088†	.4460†	.4660†	.5340†
SDR-BOW	.1972	.2264	.0952	.0204	.1718	.5398	.9083	.2739	.3728	.5081	.3742	.6258
SDR-BOC	.1953	.2329	**.0974**	**.0206**	.1751	.5530	.9151	.2756	.3751	.5086	.3689	.6311
AES-BOW	.1516†	.1768†	.0785†	.0190†	.1369†	.4611†	.8794†	.2163†	.3106†	.4552†	.4549†	.5451†
AES-BOW-P	.1604†	.1872†	.0809†	.0193†	.1480†	.4954†	.8895†	.2274†	.3255†	.4669†	.4088†	.5912†
SDR-BOW-LEE-AES	.1716†	.2008†	.0870†	.0197	.1484†	.5250†	.8988†	.2389†	.3429†	.4792†	.4148†	.5852†
SDR-BOW-AES	.1958	.2309	.0957	.0203	.1750	.5568	.9163	.2756	.3764	.5090	.3880†	.6120†
SDR-BOC-AES	.1964	.2364	.0972	.0204	.1770	.5699	.9195	.2800	.3813	.5117	.3830†	.6170†
SDR-BOW-LEE-AES-P	.1764†	.2058†	.0883†	.0199	.1570	.5349†	.9081†	.2448†	.3500†	.4865†	.3796†	.6204†
SDR-BOW-AES-P	.1983	.2322	.0961	.0204	.1740	.5673	.9206	.2768	.3812	.5128	.3608	.6392
SDR-BOC-AES-P	**.1984**	**.2369**	.0970	.0205	.1788	**.5737**	**.9241**	.2807	**.3837**	**.5147**	**.3566**	**.6434**

To clarify these settings, we contacted the original authors and confirmed that documents were indeed split using space, and stopwords were removed using NLTK. In our reproduction, we made minor adjustments to the document pre-processing pipeline. First, documents were cleaned to remove punctuation marks. They were then tokenised using Gensim's tokeniser (version 3.2.0), consistent with the original setup. For stopword removal, we used NLTK version 3.6.3 (the latest available at the time of our study), as the original version was not specified. All terms were lowercased for all methods except AES. No stemming was applied in either our reproduction or the original implementation.

3.3 Results

Before we investigate the three research questions of our reproducibility study, we first examine the extent to which we were able to replicate the results of Lee and Sun. In this study, we were unable to exactly replicate the results due to what we believe to be minor differences in document pre-processing and evaluation setup. Despite these difference, the results in Table 3.1 (or in simplified visulisation in Figure 3.4) show a similar performance across the baselines and evaluation measures compared to what Lee and Sun originally reported in their paper for our pre-processing pipeline.

The results observed comparing the document pre-processing pipeline for the BOW representation as described by Lee and Sun (*-LEE) to our document pre-processing pipeline show that the BOW baselines may not have been as strong as if the original authors had performed a similar pipeline as

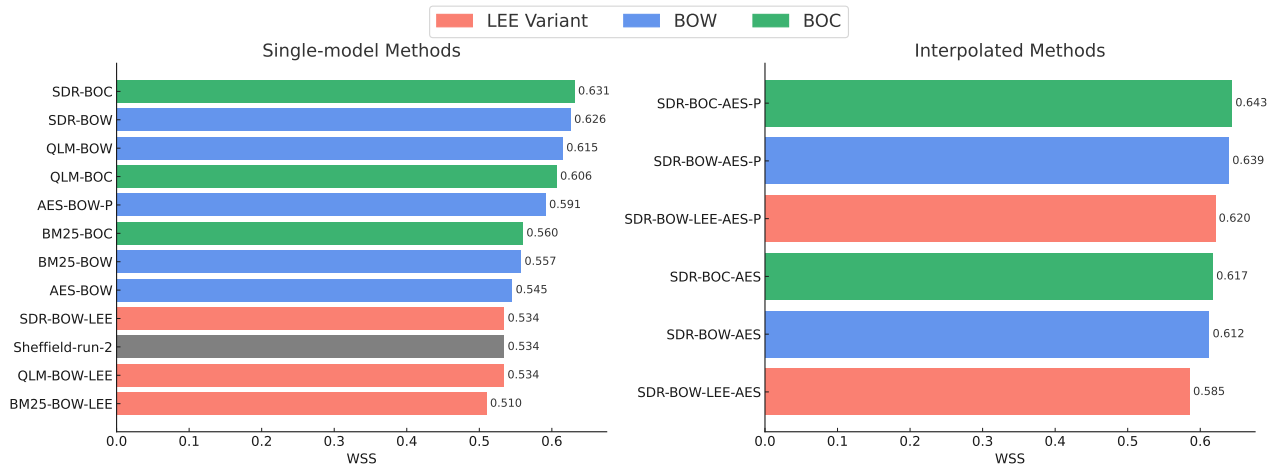


Figure 3.4: A visualisation of the results from Table 3.1 for improved readability. The figure presents WSS scores across methods to highlight performance differences more clearly.

us. We find that although the results comparing their baseline is statistically significant with our best performing method, our baseline is not significantly different. Finally, we find that the SDR-BOW-AES-LEE method, which corresponds to their most effective method, is significantly worse than our most effective method for 2017, SDR-BOW-AES-P.

In terms of the BOC representation we were unable to identify a more effective pipeline for extracting clinical terms. Here, we applied the clinical term extraction tools over *individual terms* in the document (following the pre-processing of Lee and Sun), and not *the entire document*. Although we find this to be counter-intuitive, as tools like QuickUMLS and the NCBO API use text semantics to match n-grams, the result of applying the tools to individual terms has the effect of reducing the vocabulary of a seed study to the key concepts.

Finally, comparing our evaluation setup to Lee and Sun, we find that there were a number of topics in the CLEF TAR 2017 dataset that were incompatible with SDR. Rather than attempting to replicate their results, we simply do not compare their original results with ours, since we do not have access to their run files or precise evaluation setup. Furthermore, when we compare the results we report from to the best performing participant at CLEF TAR 2017 that did not use relevance feedback [6], we remove the same topics from the run file of this participant for fairness. Although this method cannot be directly compared to, we can see that even relatively unsophisticated methods that use seed studies such as BM25-BOW are able to outperform the method by this participant.

3.3.1 Generalisability of SDR

We next investigate the first research question: *Does the effectiveness of SDR generalise beyond the CLEF TAR 2017 dataset?* In Table 3.2, we can see that the term weighting of SDR almost always increases effectiveness compared to using only QLM, and that interpolation with AES can have further benefits to effectiveness. However, we note that few of these results are statistically significant.

Additionally, we find that SDR-AES-P was not always the most effective SDR method. Indeed on the 2019 dataset, SDR-BOW was the most effective. The reason for this may be due to the difference

Table 3.2: Generalisability of results on the CLEF TAR 2017, 2018 and 2019 datasets. Representations used in this table are all BOC. Statistical significance (Student's two-tailed paired t-test with Bonferonni correction, $p < 0.05$) between the most effective method (SDR-AES-P) and other methods is indicated by †.

	Method	MAP 10	Prec. 100	Prec. 1000	Prec. 10	Recall 100	Recall 1000	Recall 10	nDCG 100	nDCG 1000	nDCG	LR%	WSS
2017	QLM	.1894	.2330	.0951	.0202	**.1809**	.5376	.9032	.2771	.3684	.5018	.3936	.6064
	SDR	.1953	.2329	**.0974**	**.0206**	.1751	.5530	.9151	.2756	.3751	.5086	.3689	.6311
	SDR-AES-P	**.1984**	**.2369**	.0970	.0205	.1788	**.5737**	**.9241**	**.2807**	**.3837**	**.5147**	**.3566**	**.6434**
2018	QLM	.2344	.2594	.1130	.0219	.1821	**.6214**	.9104	.3141	.4156	.5312	.3317†	.6683†
	SDR	.2374	.2549	.1136	.0221	.1798	.6176	.9174	.3117	.4163	.5351	.3024	.6976
	SDR-AES-P	**.2503**	**.2688**	**.1161**	**.0222**	**.1957**	.6036	**.9234**	**.3259**	**.4243**	**.5445**	**.2695**	**.7305**
2019	QLM	.2614	.2599	.0881	.0169	.2748	.7032	.9297	.3458	.4700	.5482	.4085	.5915
	SDR	.2790	.2663	**.0899**	**.0169**	**.3048**	.7151	.9337	.3594	.4846	.5602	.3819	**.6181**
	SDR-AES-P	**.2827**	**.2667**	.0898	.0168	.2973	**.7174**	**.9378**	**.3649**	**.4913**	**.5672**	**.3876**	.6124

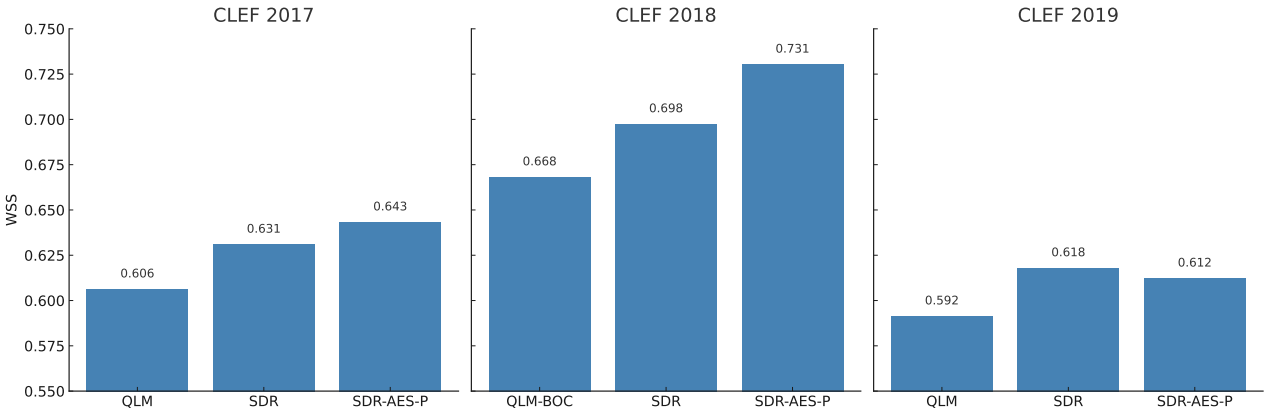


Figure 3.5: A visualisation of the results from Table 3.2 for improved readability. The figure presents WSS scores across methods to highlight performance differences more clearly.

in topicality of the 2019 dataset. This suggests that not only is the method of identifying clinical terms not suitable for these intervention systematic review topics, but that the interpolation between SDR and AES may require dataset-specific tuning.

3.3.2 Effect of Multiple Seed Studies

Next, we investigate the second research question: *What is the impact of using multiple seed studies collectively on the effectiveness of SDR?* Firstly, several topics were further removed for these experiments. Therefore, the results of single-SDR in Table 3.3 are not directly comparable to the results in Tables 3.1 and 3.2. In order to measure the effect multiple studies has on SDR compared to single seed studies, we also remove the same topics for single-SDR.

We find that across all three datasets, compared to single-SDR, multi-SDR can significantly increase the effectiveness. We also find that the largest increases in effectiveness are seen on shallow metrics across all three CLEF TAR datasets. This has implications for the use of SDR in practice, as typically, multiple seed studies are available before conducting the screening process. Therefore, when multiple seed studies are used for the initial ranking process, active learning methods that iteratively rank unjudged studies will naturally be more effective (as more relevant studies are retrieved in the

Table 3.3: Results comparing single-SDR and multi-SDR on the CLEF TAR 2017, 2018, and 2019 datasets. Note that the results for single-SDR are not directly comparable to the above tables as explained in Section 3.1.4. Statistical differences (Student's paired two-tailed t-test, $p < 0.05$) are indicated pairwise between the single- and multi- SDR BOC and BOW methods for each year (e.g., single-SDR-BOC-AES-P vs. multi-SDR-BOC-AES-P for 2017). % Change indicates the average difference between single- and multi-{BOW+BOC}.

	Method	MAP	Prec. 10	Prec. 100	Prec. 1000	Recall 10	Recall 100	Recall 1000	nDCG 10	nDCG 100	nDCG 1000	LR%	WSS
2017	Single-BOC	.3116	.4235	.1463	.0255	.2219	.6344	.9469	.4830	.5330	.6595	.3699	.6301
	Single-BOW	.3098	.4076	.1465	.0255	.2158	.6366	.9472	.4679	.5312	.6566	.3687	.6313
	Multi-BOC	.4554†	.5804†	.1752†	.0272†	.2917†	.7151†	.9661†	.6817†	.6765†	.7835†	.3427	.6573
	Multi-BOW	.4610†	.5910†	.1762†	.0272†	.2951†	.7155†	.9659†	.6924†	.6805†	.7866†	.3450	.6550
	% Change	47.4801	41.0234	20.0132	6.6705	34.1131	12.5557	2.0029	44.5398	27.5202	19.3035	-6.8792	4.0283
2018	Single-BOC	.3345	.4443	.1671	.0285	.2041	.6181	.9280	.5011	.5296	.6551	.2641	.7359
	Single-BOW	.3384	.4433	.1678	.0286	.2062	.6197	.9383	.4955	.5301	.6579	.2577	.7423
	Multi-BOC	.4779†	.6130†	.1979†	.0307†	.2821†	.6997†	.9592†	.7199†	.6823†	.7908†	.2394†	.7606†
	Multi-BOW	.4809†	.6109†	.1978†	.0306†	.2813†	.6968†	.9585†	.7218†	.6835†	.7924†	.2396	.7604
	% Change	42.5011	37.8814	18.1509	7.2657	37.3377	12.8217	2.7561	44.6754	28.8870	20.5797	-8.1919	2.8990
2019	Single-BOC	.3900	.4249	.1285	.0221	.3196	.7261	.9368	.5365	.6164	.6897	.4304	.5696
	Single-BOW	.3925	.4418	.1272	.0222	.3366	.7243	.9386	.5516	.6164	.6916	.4285	.5715
	Multi-BOC	.5341†	.5746†	.1533†	.0243†	.3962†	.7896†	.9622†	.7105†	.7458†	.8091†	.3852†	.6148†
	Multi-BOW	.5374†	.5864†	.1521†	.0244†	.4031†	.7853†	.9616†	.7223†	.7466†	.8114†	.3877†	.6123†
	% Change	36.9305	33.9958	19.3948	9.9327	21.8599	8.5825	2.5819	31.6927	21.0510	17.3213	-10.0189	7.5424

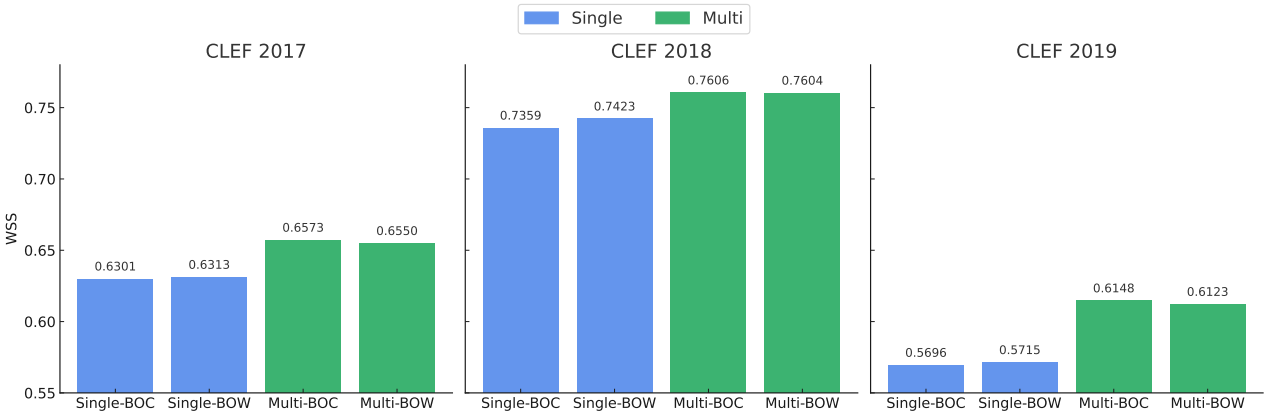
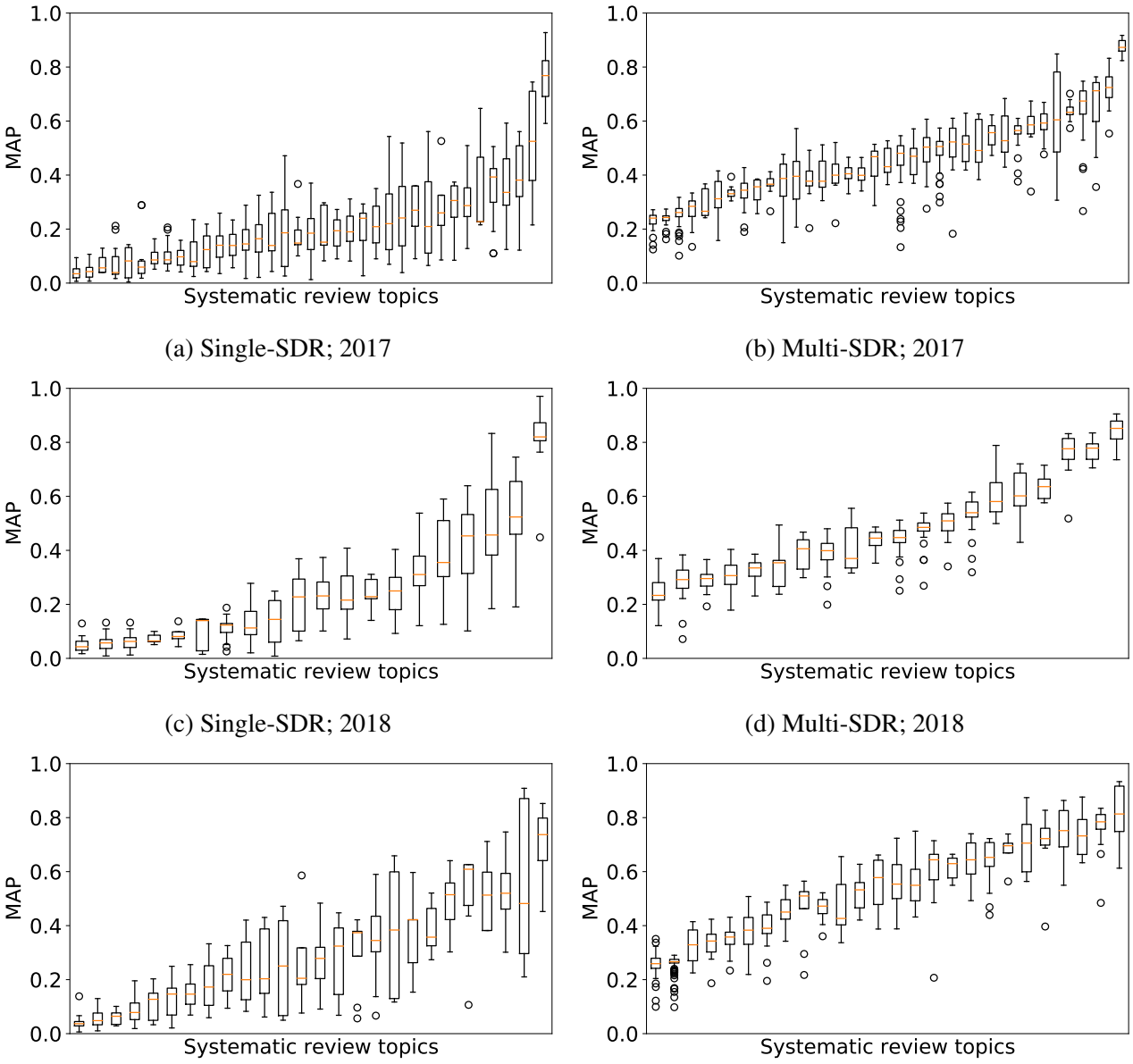


Figure 3.6: A visualisation of the results from Table 3.3 for improved readability. The figure presents WSS scores across methods to highlight performance differences more clearly.

early rankings). However, we argue that the assumption that relevant studies are a good surrogate for seed studies made by Lee and Sun [128] and by others in other work such as Scells et al. [216] may be weak and that methods that utilise relevant studies for this purpose overestimate effectiveness. In reality, seed studies may not be relevant studies. They may be discarded once a Boolean query has been formulated (e.g., they may not be randomised controlled trials or unsuitable for inclusion in the review).

3.3.3 Variability of Seed Studies on Effectiveness

Finally, we investigate the last research question: *To what extent do seed studies impact the ranking stability of single- and multi-SDR?* We investigate this research question by comparing the topic-by-topic distribution of performance for the same results present in Table 3.3. These results are visualised



(a) Single-SDR; 2017

(b) Multi-SDR; 2017

(c) Single-SDR; 2018

(d) Multi-SDR; 2018

(e) Single-SDR; 2019

(f) Multi-SDR; 2019

Figure 3.7: Topic-by-topic distribution of effectiveness (MAP) for the oracle-selected single-SDR-BOC-AES-P method (top figures) versus multi-SDR-BOC-AES-P.

in Figure 3.7. That is, we compare the multi-SDR results to the oracle single-SDR results, described in Section 3.1.4 so that we can fairly compare the variance of one to the other. We find that the variance obtained by multi-SDR is generally higher than that of single-SDR using DTA systematic review topics (Figure 3.7a vs. Figure 3.7b – and Figure 5.1c vs. Figure 3.7d). We compute the mean variance across all topics, and find that the variance of multi-SDR (4.49e-2) is 10.89% higher than single-SDR (4.44e-2) result for the 2017 dataset, and 11.76 % for the 2018 dataset (single: 3.43e-2; multi: 4.17e-2). For the 2019 dataset, we find that the variance of multi-SDR (7.93e-2) is 6.51% lower than single-SDR (8.48e-2).

However, when we randomly sample seed studies from each group for single-SDR, we find that the variance of multi-SDR is significantly lower: 53.2% average decrease across 2017, 2018, and 2019. This suggests that the choice of seed study is considerably more important for single-SDR than for

multi-SDR and that multi-SDR produces much more stable rankings, regardless of the seed studies chosen for re-ranking.

3.4 Summary of Findings

We reproduced the Seed-driven Document Ranking (SDR) method introduced by Lee and Sun [128] on all available CLEF TAR datasets [113, 114, 115]. In our experiments, we simulated seed studies using documents marked as 'included studies.' Our analysis showed that the 2017 and 2018 datasets exhibited similar performance trends, in contrast to the results from the 2019 dataset. We attribute this discrepancy to topical differences between the datasets and suggest that tuning the SDR parameters could help align the 2019 results more closely with those from 2017 and 2018. Through various preprocessing steps, we observed that the bag-of-words (BOW) representation of relevant documents often exhibited a higher overlap of common terms compared to the bag-of-concepts (BOC) representation. We also found that using the BOC representation within SDR improved performance, particularly when term weighting was applied. We showed that multi-SDR consistently outperformed single-SDR. Our experiments used an oracle-based method to select optimal seed studies when comparing multi-SDR with single-SDR. As a result, the actual performance gap between multi-SDR and single-SDR could be even greater than observed. Lastly, regarding the impact of seed studies on ranking stability, our findings indicate that while multi-SDR achieved better effectiveness than single-SDR, it was generally associated with greater variance in performance outcomes.

This chapter contributes to the broader goal of improving seed-driven automation for systematic reviews by evaluating existing methods under controlled, reproducible conditions. However, all experiments in this chapter rely on pseudo-seed studies—documents selected from the final included set—which may not reflect real-world screening scenarios. This limitation motivates the next chapter, where we introduce a new dataset built from actual seed studies used during the early stages of systematic review creation. By evaluating SDR and related methods in a more realistic setting, we aim to better understand their practical reliability and generalisability.

Chapter 4

Creation of Dataset with Seed Studies

Previously, we investigated one effective method named Seed-driven Document Ranking (SDR) and demonstrated its high performance using pseudo seed studies for more effective document ranking during systematic review screening. However, methods employing pseudo seed studies rely on documents known to be relevant and already included in a systematic review, which, as we demonstrate, leads to an overestimation of effectiveness.

To address this issue, we present a new Information Retrieval test collection, named *Seed Collection*, designed specifically for systematic review literature search. The most novel contribution of this collection is the inclusion of real seed studies for each systematic review topic. This test collection facilitates the development of new Information Retrieval methods for systematic review literature searches that depend upon or utilise seed studies. To evaluate the practical impact of real seed studies compared to pseudo seed studies, we reproduce two prominent methods from existing literature: seed-driven automatic query formulation by Scells et al. [213] and seed-driven document ranking by Lee and Sun [128], that is reproduced in Chapter 3. Furthermore, we introduce and explore a technique frequently utilised during systematic review creation yet often overlooked in Information Retrieval research—snowballing (also known as citation chasing). This method is discussed in greater detail in the sections that follow. These three use cases are visualised in Figure 4.1. We make our test collection and the results of all of our experiments and analysis available at `https://github.com/ielab/sysrev-seed-collection`.

4.1 Collection Details

In this section, we discuss the creation and analysis of the collection, including the information needs for systematic review literature search, the attributes of the collection, challenges encountered, relevance judgments, and collection statistics and analysis.

Figure 4.1: High-level overview of the systematic review creation processes that are relevant to our collection, and several use cases of our collection for automating these processes. Also shown are the three use-cases we will demonstrate for our collection: 1 query formulation, 2 screening prioritisation, and 3 ranking with snowballing.

Table 4.1: Attributes of each topic in our collection. PMID refers to 'PubMed identifier', and is used to uniquely identify a study or document in the PubMed database.

Name	Description
ID	Unique ID of a topic.
Link to Review	Link (URL) to the published review.
Title	Title of the systematic review.
Description	Topical summary of the systematic review, drafted during original query formulation
Date restrictions	Date restriction used to retrieve studies.
PubMed query	PubMed Boolean query used to retrieve studies.
Seed studies	PMIDs of seed studies.
Included studies	PMIDs of included studies.
Retrieved studies	PMIDs of studies retrieved by the query.
Snowballed studies	PMIDs of snowballed studies.

4.1.1 Topic Attributes

Our dataset is based on systematic review topics provided by Justin Clark, a senior information specialist at the Bond Institute for Evidence-Based Healthcare. Each systematic review topic is developed using Boolean queries crafted by Justin over the past five years. The original unstructured data is organised into a structured collection comprising 40 topics.

Each topic in the collection includes several attributes, as detailed in Table 4.1. First, each

systematic review is assigned a unique ID. Since the collection comprises completed and published systematic reviews, we also include the title and URL linking to the published review. Additionally, each topic includes a brief description of the search strategy provided by our information specialist co-author.

Beyond these metadata attributes, each topic includes detailed query information, specifically the PubMed Boolean query, the date restrictions applied during searches (i.e., to restrict retrieved studies based on publication dates), and the seed studies utilised to guide query development. Typically, systematic reviews employ multiple queries across various medical literature databases to maximise recall. However, due to subscription barriers, our collection only includes queries from PubMed. This limitation is common across existing collections mentioned in Section 2.3, including ours.

The next set of attributes corresponds to the relevance assessments. The included studies represent studies that are relevant at the full-text level. This means that these studies are assessed as sufficient to be included in the final review. As such, it should be noted that evaluation of an Information Retrieval system using our collection is more *strict* (i.e., it is naturally more challenging to identify those studies that will be included in the final systematic review than those that are potentially relevant at an abstract level). Additionally, we provide the set of retrieved studies that are retrieved by the Boolean query.

The final attribute of each topic corresponds to snowballed studies. We include two sets of snowballed studies for each topic: one corresponding to snowballing the seed studies, and another corresponding to snowballing the retrieved included studies. The second set simulates the process that systematic reviewers often follow to identify additional relevant studies after assessing the initially retrieved set. Note that this is an often overlooked process in Information Retrieval research in this domain. We take this opportunity to conduct an initial investigation into the impact snowballing has on retrieval effectiveness and integrate the comparison of snowballing seed studies into our analysis.

4.1.2 Data Processing

Several attributes of topics require data processing beyond simple extraction. Specifically, the included studies, the retrieved studies, and the snowballed studies require additional processing to be generated.

Included studies. The raw data provided to us does not contain relevance assessment information. Therefore, we manually create the relevance assessments for each topic by extracting the list of studies included in the final analysis of each systematic review. This process is not as straightforward as scraping the reference sections; it requires manually matching the citations used in the analysis to actual references of published studies. In cases where a study is not available in PubMed, we exclude it from our relevance assessments under the assumption that it would not be retrievable in a PubMed-based search.

Retrieved studies. The raw data also does not include the set of studies retrieved by the Boolean queries. To address this, we reproduce the searches for all topics to obtain the retrieved sets. This is automated using the Entrez API [149]. Each Boolean query is executed with its respective date restrictions applied, ensuring that others can reproduce the retrieved document sets for each topic if

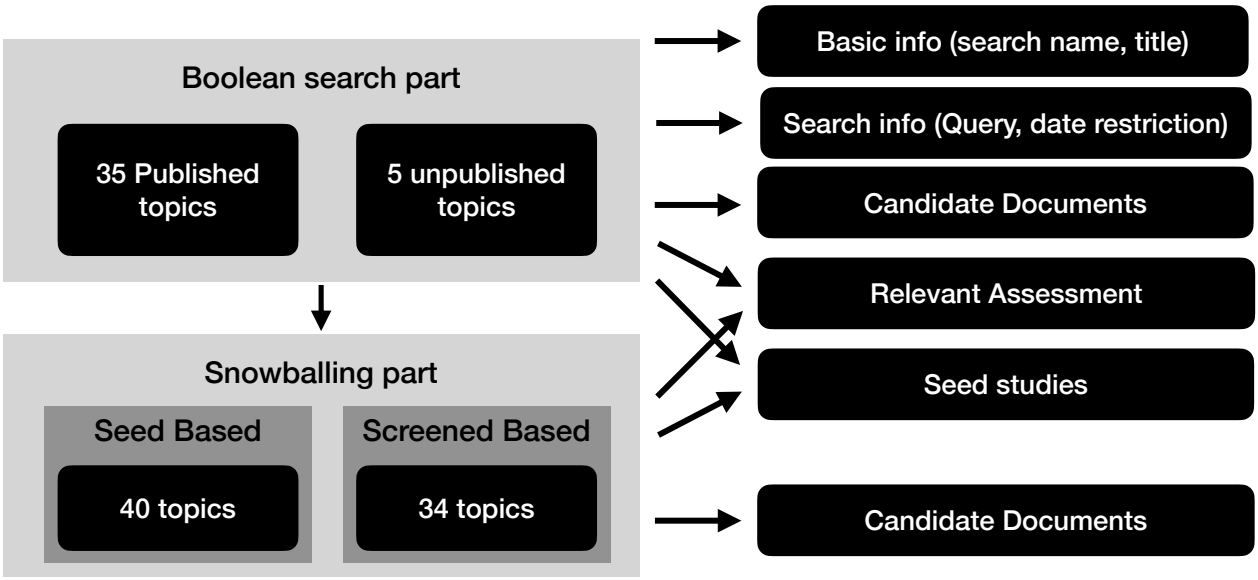


Figure 4.2: Visualised diagram of the seed collection, which consists of main collection part and snowballing part.

needed.

Snowballed studies. Instead of manually performing snowballing, we employ two tools: CitationChaser [91] and SpiderCite [39]. We first use CitationChaser to format the study lists into the RIS format required by SpiderCite.[1] Using SpiderCite, we obtain the cited and citing studies for each input set. Each record in SpiderCite includes a DOI, which we use to retrieve the corresponding entry from PubMed.

4.1.3 Collection Statistics & Analysis

Visualised details of our collection can be viewed in Figure 4.2. Overall, the Seed Collection contains 40 systematic review topics. For each systematic review topic, an average of 15 seed studies are used for query construction, with a median of 12. The average number of included studies per systematic review is 29, with a median of 11.5.

When comparing the overlap between seed studies and included studies, we find that, on average, only 36.3% of seed studies are also included in the final set of included studies, and they make up just 26.1% of all included studies. This finding suggests that, although seed studies are helpful for Boolean query construction, most are disregarded after the query phase. This demonstrates that treating included studies as pseudo seed studies, as in prior work [128, 216], may lead to inaccurate or overoptimistic results, as seed studies and included studies serve fundamentally different purposes in the systematic review creation process.

[1] RIS is a standardised tag format that enables citation management tools to exchange bibliographic data, and it is the only input format supported by SpiderCite.

Topical Similarity Analysis: Comparing Seed Studies and Included Studies.

To better characterise the relationship between seed studies and included studies in our collection, we conduct a topical similarity analysis comparing both document sets to their corresponding systematic review topics. This analysis aims to answer: *Are seed studies and included studies similarly representative of the systematic review topics, or do they capture different aspects of the topic?* Understanding this relationship is crucial for researchers using this collection, as it informs whether methods should use actual seed studies, pseudo seed studies (sampled from included studies), or a combination of both. We measure similarity using three complementary metrics computed against each topic's title (the primary description of the systematic review's focus):

- **Jaccard similarity**: Measures term overlap between document text and topic title, capturing lexical similarity.
- **Term coverage**: Calculates the proportion of key terms from the topic title that appear in the document set, indicating how comprehensively the documents cover topic terminology.
- **Semantic similarity (BERT score)**: Uses contextual embeddings to measure semantic relatedness beyond exact term matches, capturing conceptual similarity.

For each topic, we compute these metrics for both the set of seed studies and the set of included studies, then compare their topical similarity. The analysis reveals whether seed studies (documents identified early by information specialists to guide query formulation) are more, less, or equally representative of the topic compared to included studies (documents that survived the full screening process and met all inclusion criteria).

Aggregate Similarity: Comparable Performance Despite Different Roles. Despite their different roles in the systematic review creation process, seed studies and included studies demonstrate remarkably comparable topical similarity to systematic review topics. Mean Jaccard similarity is 0.0558 ± 0.0173 for seed studies versus 0.0539 ± 0.0183 for included studies, a difference of only -0.0019. Mean term coverage shows similarly minimal difference: 0.3017 ± 0.1162 for seed studies versus 0.3005 ± 0.1240 for included studies (difference: -0.0012). Semantic similarity scores (BERT) also show negligible aggregate difference (0.4722 vs 0.4802 on average). These results suggest that information specialists successfully identify topically relevant exemplar documents as seed studies, even though many of these documents ultimately do not meet the strict inclusion criteria for the final systematic review. The comparable similarity scores indicate that seed studies serve their intended purpose: providing broad topical coverage and relevant terminology for query construction, rather than strictly adhering to the final inclusion criteria.

Topic-Level Variation: Context-Dependent Relationships. However, the aggregate view masks substantial topic-level variation in the relationship between seed and included study similarity. Figure 4.3 illustrates the per-topic differences in Jaccard similarity and semantic similarity between seed

and included studies. For 23 topics (57.5%), included studies showed higher Jaccard similarity to the topic, while seed studies demonstrated higher similarity in 17 topics (42.5%). The magnitude of differences varies considerably across topics. Topic 47 ("Effectiveness and sustainability of deprescribing for hospitalised older patients near end of life") shows the largest negative difference (-0.0200), where seed studies captured broader terminology around deprescribing and end-of-life care, while only 2 studies ultimately met the strict inclusion criteria focusing on sustainability of deprescribing interventions. This suggests that information specialists used seed studies to explore the broader conceptual space before the screening process narrowed to the specific intervention characteristics required by the review. Conversely, Topic 46 ("Dementia caregiving in the Middle East and North Africa") shows the largest positive difference (+0.0215), where included studies demonstrate higher topical similarity. For this topic, seed studies captured broader terminology around dementia and caregiving concepts, while the screening process successfully narrowed to studies meeting the geographic constraint (Middle East and North Africa), resulting in included studies with more precise topical alignment to the specific geographic focus of the review. Analysis of topics with large differences reveals patterns related to topic specificity. Topics focusing on highly specific interventions or narrow clinical populations (e.g., Topic 64 on metformin in pregnancy, Topic 12 on methenamine hippurate) tend to show included studies with higher similarity, suggesting the screening process successfully narrows to more precise, relevant terminology. In contrast, topics with broader scope or interdisciplinary nature (e.g., Topic 46 on dementia caregiving, Topic 14 on COVID-19 impact on healthcare services) show seed studies capturing a wider range of relevant concepts that may be filtered during screening.

Implications for Collection Users. These topical similarity statistics have important implications for researchers using this collection:

For query formulation methods: Seed studies provide broader terminology coverage and conceptual understanding of the topic domain, as evidenced by their comparable topical similarity despite only 36.3% being ultimately included in the final review. Methods that extract terms or concepts from seed studies for query construction can leverage this broader coverage. However, researchers should be aware that seed studies may include terminology that is topically relevant but ultimately filtered during screening (e.g., broader terms for scoping before applying specific inclusion criteria).

For ranking and screening prioritisation methods: The choice between using actual seed studies versus pseudo seed studies (sampled from included studies) depends on topic characteristics. Our analysis shows no universal superiority of either approach across all topics. For highly specific topics (e.g., specific drug interventions, narrow populations), included studies may provide better training signal as they represent the refined scope. For broader, exploratory topics, seed studies capture terminology and concepts that may be valuable despite being filtered during screening. Methods that claim to use seed studies should evaluate performance on both actual seeds and included studies to understand their true effectiveness.

For comprehensive evaluation: Researchers evaluating retrieval methods should consider the 36.3% overlap and topic-dependent similarity relationships when interpreting results. Methods trained

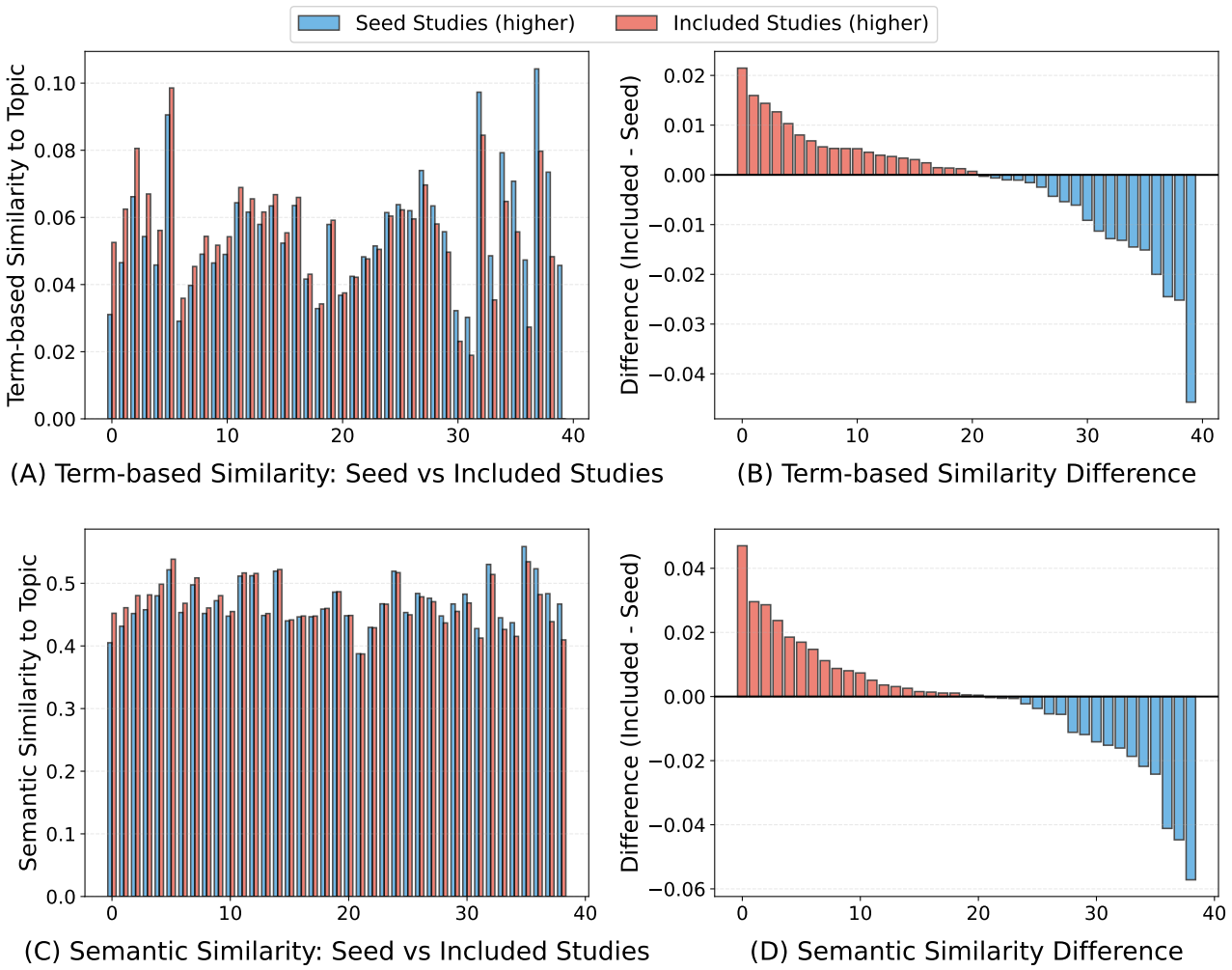


Figure 4.3: Per-topic differences in Jaccard similarity (top) and semantic similarity (bottom) between included studies and seed studies. Positive values (red bars) indicate included studies are more similar to the topic; negative values (blue bars) indicate seed studies are more similar. Topics are sorted by difference magnitude. The substantial variation demonstrates that relative topical similarity is highly topic-dependent. Topics with large positive differences (e.g., Topic 46) tend to involve highly specific interventions, while topics with negative differences (e.g., Topic 18) often involve broader scopes.

or evaluated solely on included studies as pseudo seeds may show inflated performance compared to real-world scenarios where actual seed studies (which are less likely to be included) are used. Our collection enables such realistic evaluation by providing both actual seed studies and included studies.

4.1.4 Searching Analysis

When using Boolean queries with date restrictions to retrieve studies, the mean number of documents retrieved per query is approximately 1,326, with a median of 709. Note that, given all retrieved documents must be manually screened, this presents a substantial screening burden. At an average screening time of 30 seconds per document, this amounts to over 11 hours of continuous labour per query for reviewers [32]. This further motivates the need for effective screening prioritisation techniques, which aim to rank and present the most relevant documents earlier, thereby reducing the manual workload and time to inclusion.

Additionally, we observed that not all included studies were found among the retrieved documents across several topics. In fact, only 75.5% of all included studies across all topics were present in the retrieved sets. There are two primary reasons why some included studies may be missing from the candidate documents. First, some were likely identified through snowballing (i.e., citation chasing). Second, some may have been located through searches in databases other than PubMed—even though such studies may still be indexed in PubMed, they may not have been retrieved due to limitations in the search query.

A notable exception is Topic #18, where none of the included studies were found in the retrieved set. This particular systematic review includes only a single study. In this case, many studies that were initially screened as relevant at the abstract level were later excluded due to a high risk of bias identified during full-text assessment.

4.1.5 Snowballing Analysis

Our collection includes two snowballed sets of studies for each topic. The first set corresponds to snowballed seed studies (**seed-snowballing**), and the second corresponds to simulated snowballing of the included studies in the retrieved set (**screened-snowballing**).

For the seed-snowballing set, we find that 35 topics retrieve at least one included study. The topics that do not retrieve any included studies using this snowballing technique are 46, 52, 53, 66, and 96. The average number of documents snowballed per query is 1,142—generally smaller than the number retrieved from the Boolean search—because fewer seed studies are typically used.

For the screened-snowballing set, only 34 topics retrieve at least one additional included study. The six topics that do not retrieve any additional included studies already contain all included studies prior to snowballing. These topics are 7, 10, 17, 39, 64, and 66. These topics represent an interesting research problem: applying snowballing in such cases would result in wasted time and effort for systematic reviewers. We leave further investigation into determining whether or not to apply snowballing as future work. After removing already screened studies, the screened-snowballing sets contain, on average, 1,000 studies.

4.1.6 Effectiveness comparison of retrieval methods

Finally, we aim to investigate the effectiveness of different search methods when used independently and in combination. The results of this analysis are presented in Figure 4.4. These plots show that using the seed studies alone (i.e., without searching or snowballing) to match against the included studies set obtains the lowest recall but the highest precision. Snowballing the seed studies increases recall but dramatically reduces precision.

This result indicates that, although seed studies are highly relevant to the final set of included studies, relying solely on them is insufficient to meet the high-recall requirements of systematic reviews. In other words, while seed studies provide a strong starting point, comprehensive search strategies remain essential to ensure that the majority of relevant literature is identified.

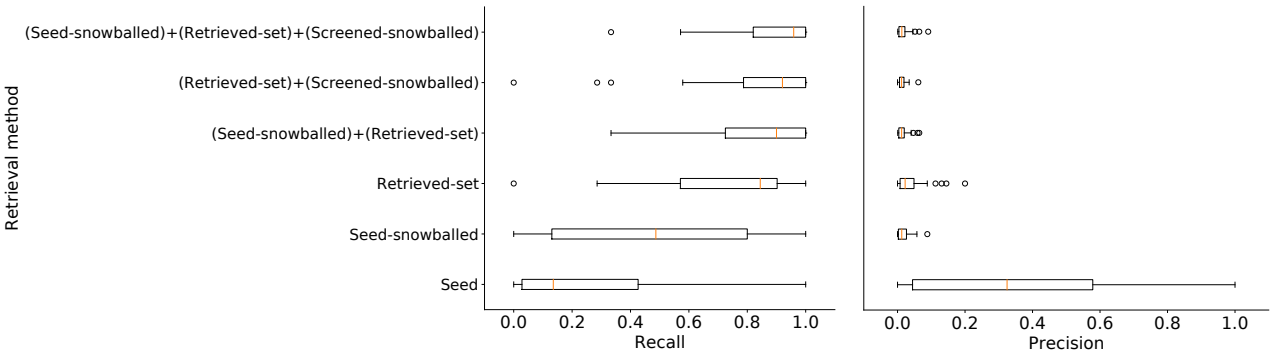


Figure 4.4: Recall and precision box plots of included studies for different retrieval methods. The methods listed include seed studies (seed), the Boolean query results (retrieved set), the two snowballing sets (seed-snowballed, i.e., snowballing applied to seed studies; and screened-snowballed, i.e., snowballing applied to retrieved included studies). These sets of studies are also combined in numerous ways, as indicated by '+'.

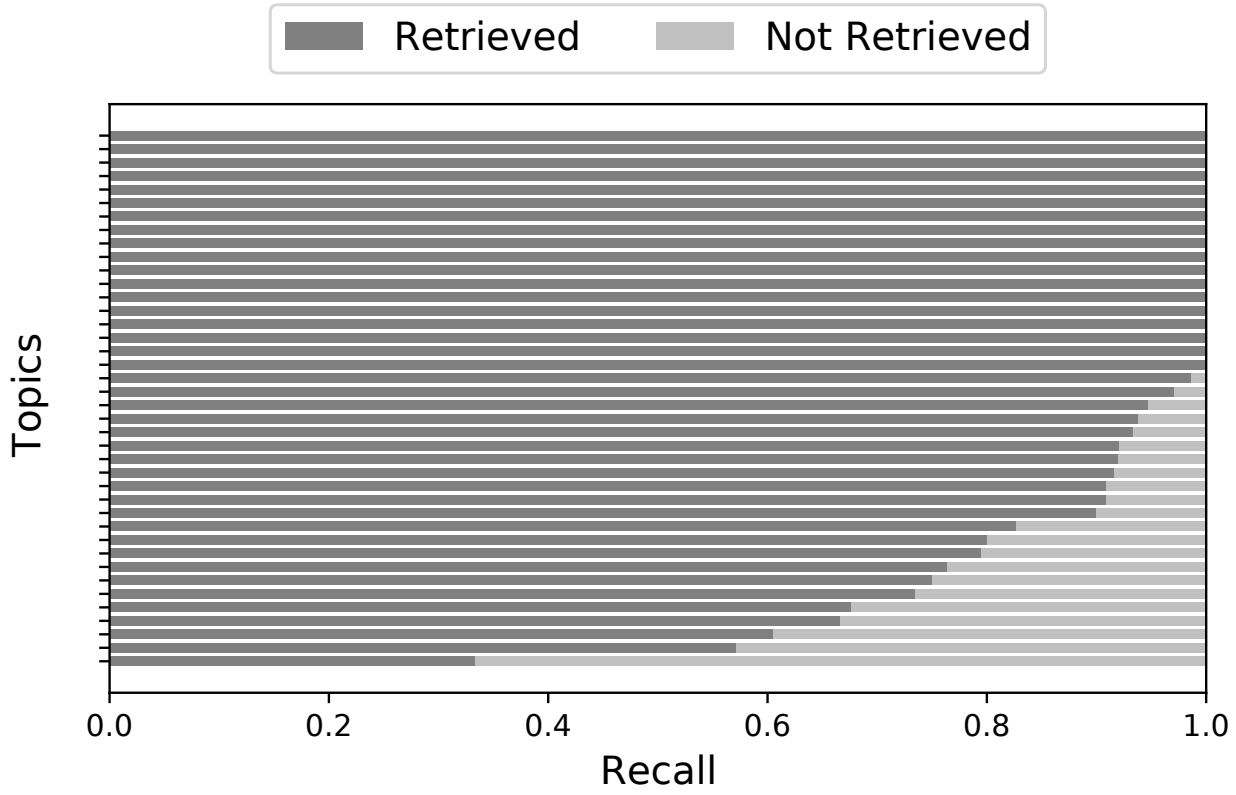


Figure 4.5: Recall distribution of all of the topics given the combined set of studies that includes retrieved studies, the seed-snowballing set, and the screened-snowballing set.

Many topics retrieve almost all included studies with the Boolean query. Yet, combining the retrieved studies with other methods further improves recall while lowering precision. Combining both snowballing methods with the retrieved studies obtains the highest recall and lowest precision. However, total recall is still not achieved. Thus, we analyse the recall for different queries in Figure 4.5.

We find that for some topics, the recall is low. We investigate these topics and find that these systematic reviews use several medical literature databases other than PubMed to retrieve studies. This results in the inclusion of studies that exist in the PubMed database but are not retrieved by the

PubMed query. We also randomly select some topics that reach total recall and find that, even though some of them still use a combination of multiple medical literature databases, they all use PubMed as one of the search sources. Thus, it is possible to create a more effective query for these topics. We leave this problem for future work.

4.2 Use Case 1: Query Formulation

We begin our demonstration of the use-cases of our collection with a reproduction of two automatic query formulation methods that use seed studies, see 1 in Figure 4.1. Following previous research, we compare both automatic query formulation methods when using seed studies and pseudo seed studies [216].

4.2.1 Methods & Experimental Settings

The query formulation experiments are based on two existing methods from the literature [213, 214]. These methods are fully automated adaptations of manual or semi-automated procedures that information specialists use in practice. The first is called the *conceptual method* and is what the majority of information specialists use to formulate queries [38]. The automated conceptual method [214] takes as input a preliminary string for identifying salient terms (we use the title of the systematic review). The seed studies are then used to optimise the coverage of different combinations of terms expanded from those in the title. The second is called the *objective method* and is a more recent procedure that takes a statistical approach to query formulation [96]. At a high level, the automated objective method [213] first identifies and ranks salient terms from seed studies using term frequency statistics of the seed studies and a background collection. Next, terms are filtered and added to Boolean clauses by tuning these statistics to a held-out portion of seed studies. Given that the objective method relies on a held-out portion and the conceptual does not, we run both methods for three iterations using different arrangements of seed studies so that both methods use the same set of seed studies each iteration. We run these three iterations twice: once for the real seed studies and once for the pseudo seed studies.

One other aspect of the automatic versions of the two query formulation methods is the notion of an *instantiation*. In other words, the inclusion or exclusion of different aspects of Boolean queries (e.g., including or excluding MeSH, or including or excluding phrases). To this end, we perform experiments for only the most effective instantiation of each method (Conceptual/Phrase, and Objective/Phrase/Recall/MeSH). We refer the reader to the original study [213] for a comprehensive description of all experimental settings and implementation details that we have used.)

4.2.2 Results & Analysis

The results of our automatic query formulation reproduction study using our collection are presented in Table 4.2. We report precision, recall, and average number of studies retrieved.

Table 4.2: Effectiveness of queries formulated using pseudo seed studies (Pseudo) and seed studies (Seed). Also included are retrieval results of the queries for each topic (Original queries). Percentage differences (Δ) compared to Original queries are shown for Precision and Recall. Oracle experiments and the evaluation measures optimised can be seen in brackets. Oracle refers to the ability of a generated query using the single best seed study that can retrieve a better set of included studies compared to all other seed studies; the oracle is selected using either precision or recall.

	Method	Precision	Recall	Avg. Retrieved	Δ Precision (%)	Δ Recall (%)
	Original queries	0.01748	0.73659	1,326	/	/
Objective	Pseudo	0.00005	0.23457	746,193	−99.71	−68.15
	Seed	0.00024	0.31659	806,760	−98.63	−57.02
	Pseudo (oracle precision)	0.00015	0.22151	346,023	−99.14	−69.93
	Seed (oracle precision)	0.00573	0.34188	779,851	−67.22	−53.59
	Pseudo (oracle recall)	0.00014	0.50142	1,550,669	−99.20	−31.93
	Seed (oracle recall)	0.00572	0.51923	1,209,792	−67.28	−29.51
Conceptual	Pseudo	0.00664	0.27273	433,113	−62.01	−62.97
	Seed	0.00093	0.26781	362,968	−94.68	−63.64
	Pseudo (oracle precision)	0.00682	0.20940	30,961	−60.98	−71.57
	Seed (oracle precision)	0.00203	0.29202	360,549	−88.39	−60.36
	Pseudo (oracle recall)	0.00657	0.37334	667,398	−62.41	−49.32
	Seed (oracle recall)	0.00183	0.41382	848,223	−89.53	−43.82

Comparison of Seed Studies and Pseudo Seed Studies

We first investigate the differences between using seed studies and pseudo seed studies for the objective method. Using real seed studies to formulate queries is more effective (in both precision and recall) than using pseudo seed studies for the objective method. One possible reason seed studies produce better queries is that pseudo seed studies may have been identified through snowballing (and therefore were never originally retrieved using terms present in the Boolean query).

For the conceptual method, the results indicate that seed studies produce less effective queries than pseudo seed studies. We observe a noticeable decrease in precision and a small decrease in recall when using seed studies compared to pseudo seed studies. However, while all queries using the objective method retrieve at least one study across all three iterations, queries from 17 topics constructed using the conceptual method do not retrieve any studies whatsoever. These results suggest that the automatic conceptual method generally produces less effective queries than the objective method, regardless of whether seed or pseudo seed studies are used.

Comparison of Oracle-Setting Based Results

Given the possible issues with query formulation, and the fact that certain (pseudo) seed studies may impact the effectiveness of the resulting queries, we also investigate an oracle approach to identifying the most effective queries across the three iterations. The oracle process can be thought of as simulating the selection of an effective query by choosing from a list of all possible candidate queries, each drafted using a single seed study.

When using the oracle query, the results show a similar trend to those obtained from the objective method above: queries constructed using seed studies achieve higher precision and recall than those using pseudo seed studies. For the conceptual method, the results show that although using seed studies still results in lower precision, the recall is higher than when pseudo seed studies are used. Again, as 17 topics across all three iterations do not retrieve any studies, this outcome may be biased toward those topics that retrieve more than one study.

Comparison to Original Queries

Finally, we compare the results of the two automatic query formulation methods to the retrieval results of the original queries for each topic in our collection. The original queries are highly effective, achieving dramatically higher precision and recall. The average number of studies retrieved is also considerably lower than with the automatic methods.

This highlights that expert-crafted Boolean queries used in real systematic reviews are not only more precise and comprehensive but also more efficient in reducing the screening burden. It also suggests that a better automatic method—capable of approximating the quality of expert-designed queries—could significantly improve the effectiveness and efficiency of systematic review literature searches compared to both objective and conceptual methods by Scells et al. [211].

4.2.3 Query Formulation Findings

From the automatic query formulation experiments, we find that:

- For the objective method, higher effectiveness—both in terms of recall and precision—can be achieved when queries are constructed using seed studies. In fact, pseudo seed studies are detrimental to the query formulation procedure.
- For the conceptual method, queries may require manual modification to prevent biased results. We encountered similar findings to previous research by Scells et al. [216], where several topics retrieved no studies. We leave further investigations for future work.

4.3 Use Case 2: Screening Prioritisation

We continue with another possible use case for this collection: Seed-driven Document Ranking (SDR), which uses seed studies for screening prioritisation (i.e., ranking the set of retrieved studies), see 2 in Figure 4.1. The technique was originally proposed by Lee et al. [128] and is reproduced by us (details are discussed in Chapter 3).

In this section, we investigate the effectiveness of SDR methods by comparing the use of seed studies with pseudo seed studies. Note that SDR refers both to the specific task of screening prioritisation and to a particular method of screening prioritisation called the *SDR method*. We make this distinction clear by referring to `SDR` as the task and the `SDR method` as the ranking function.

4.3.1 Methods & Experimental Setup

The SDR method, originally proposed by Lee et al. [128], ranks retrieved studies by leveraging the observation that terms in relevant documents tend to be more similar than those in irrelevant ones. Each document is scored using term weights based on inter-study similarity, combined with term likelihoods estimated via the query likelihood model (QLM).

Following the experiments in Chapter 3, we use the **Bag of Words (BOW)** representation, which has been shown to outperform clinical-term-based alternatives. In this section, we extend those experiments by comparing real seed studies from our collection to pseudo seed studies within the SDR framework.

Single Pseudo Seed Study

Firstly, we assume a single included study as the available seed study for each systematic review topic, following the original SDR study [128]. As there are multiple included studies for each topic, the overall effectiveness of the retrieval methods is calculated using the average performance obtained by treating each included study as a pseudo seed study in turn.

This leave-one-out cross-validation strategy provides a more reliable and unbiased estimate of performance across all included studies in a systematic review topic [201].[2]

Multiple Pseudo Seed Studies

Using multiple pseudo seed studies is more effective than using a single seed study for SDR, as shown in Chapter 3. For our collection, we also perform SDR using multiple pseudo seed studies.

We adopt the same experimental settings as in Chapter 3 to evaluate the effectiveness of SDR with multiple pseudo seed studies. Specifically, we follow the same seed study grouping strategy, in which 20% of the included studies are selected using a sliding-window approach. The selected groups are then combined by concatenating their titles and abstracts to serve as input to the retrieval methods. Effectiveness for each topic is calculated as the average across all groups.

Using the real seed studies in our collection, we can now realistically evaluate the effectiveness of SDR. In this experiment, all seed studies for each topic are combined by concatenating their titles and abstracts, similar to the multiple pseudo seed study setup. The combined text is used as input to the SDR method, and all included studies are used for evaluation.

Retrieval Methods

Apart from the original SDR method proposed by Lee and Sun [128], we also evaluate several baseline methods: BM25, a query likelihood model (QLM), and a word embedding-based model (AES). As in previous SDR studies, we use two pre-trained word embeddings for AES: one trained on PubMed and Wikipedia, and another trained on PubMed only. Additionally, we include the fusion method used in

[2]Topic #18 has only one included study. We exclude this topic from the evaluation in all experiments.

Table 4.3: Results of baselines and SDR methods on our collection in three experimental settings: single pseudo seed studies, multiple pseudo seed studies, and seed studies. In the header, P refers to 'precision' and R refers to 'recall'. For AES methods, word2vec PubMed embeddings are denoted by '-P'. AES methods that do not have this demarcation correspond to word2vec embeddings, including PubMed and Wikipedia. Statistical significance (Student's two-tailed paired t-test with Bonferonni correction, $p < 0.05$) between SDR method and all other methods is indicated by †.

	Method	MAP	Precision			Recall			nDCG			LR%
			10	100	1000	10	100	1000	10	100	1000	
Single Pseudo	BM25-BOW	.1214†	.1539†	.0696†	.0156	.1226†	.4049†	.6855†	.1846†	.2760†	.3764†	.6915†
	QLM-BOW	.1671	**.2190**	.0829	.0168	**.1646**	.4572	.7059	**.2636**	.3377	.4291	.5966
	SDR-BOW	.1661	.2165	.0833	.0168	.1632	.4638	.7055	.2599	.3386	.4286	.6117
	AES-BOW	.1501	.1932	.0807	.0169	.1361	.4349	.7022	.2345	.3141	.4110	.6365
	AES-BOW-P	.1527	.1965	.0831	.0170	.1361	.4455	.7049	.2349	.3184	.4134	.6017
	SDR-BOW-AES	.1698	.2185	.0864	.0172	.1573	.4710	.7060	.2615	.3437	.4328	.6007
	SDR-BOW-AES-P	**.1709**	.2165	**.0877**	**.0172**	.1531	**.4726**	**.7114**	.2596	**.3446**	**.4337**	**.5820**
Multiple Pseudo	BM25-BOW	.2045†	.2253†	.0712†	.0154	.1888†	.4281†	.6792	.3145†	.3817†	.4758†	.7073†
	QLM-BOW	.3429	.4629	.1222	**.0190**	.2767	.5531	**.7251**	.5934	.5520	.6174	**.5255**
	SDR-BOW	**.3457**	**.4726**	**.1231**	.0189	**.2798**	**.5630**	.7226	**.6027**	**.5574**	**.6196**	.5483
	AES-BOW	.2731†	.3415†	.1059†	.0183	.2228†	.5041†	.7147	.4466†	.4727†	.5529†	.6156†
	AES-BOW-P	.2810†	.3488†	.1094†	.0185	.2260†	.5202†	.7179	.4546†	.4834†	.5598†	.5743
	SDR-BOW-AES	.3374	.4464†	.1201	.0189	.2666	.5517	.7211	.5791†	.5465†	.6131	.5672
	SDR-BOW-AES-P	.3344	.4341†	.1200	.0189	.2634	.5538	.7235	.5662†	.5439	.6103	.5426
Seed Studies	BM25-BOW	.1153†	.1300†	.0595†	.0151	.1067†	.3404†	.6333†	.1516†	.2375†	.3439†	.7591†
	QLM-BOW	**.2294**	**.2850**	.0932	.0169	.2016	.4614	.6706	**.3370**	**.3867**	**.4655**	**.5749**
	SDR-BOW	.2289	.2800	**.0945**	.0169	**.2028**	**.4695**	.6712	.3253	.3860	.4619	.6012
	AES-BOW	.1784†	.2250	.0830†	.0169	.1644	.4243	.6657	.2570	.3276†	.4184†	.6390
	AES-BOW-P	.1835†	.2275	.0860	.0169	.1614	.4354	.6667	.2656	.3381†	.4248†	.6008
	SDR-BOW-AES	.2201	.2650	.0927	**.0171**	.1976	.4615	**.6724**	.3151	.3790	.4581	.5979
	SDR-BOW-AES-P	.2198	.2600	.0912	.0169	.1972	.4484	.6680	.3122	.3732	.4549	.5750

the original paper, where the SDR method is interpolated with AES using the same parameter setting ($\alpha = 0.3$).

Evaluation Measures

We evaluate the different arrangements of seed studies and retrieval methods using rank-based measures. We use the same evaluation metrics as in Chapter 3. In addition to Mean Average Precision (MAP), which measures the overall ranking effectiveness, we also report precision, recall, and nDCG at cut-offs $\{10, 100, 1000\}$. We also include the Last Relevant Percentage (LR%), which reflects the proportion of documents that must be screened in order to identify all included studies.

4.3.2 Results & Analysis

The results of using a single pseudo seed study, multiple pseudo seed studies, and real seed studies are shown in Table 4.3.

Comparison between Single Pseudo Seed and Multiple Pseudo Seeds

For single pseudo seed studies, the SDR method improves effectiveness on deep evaluation metrics (i.e., precision@$\{100, 1000\}$, recall@$\{100, 1000\}$, and nDCG@$\{100, 1000\}$). SDR-BOW-AES-P is the most effective method, except for shallow evaluation measures (i.e., precision@10, recall@10, and

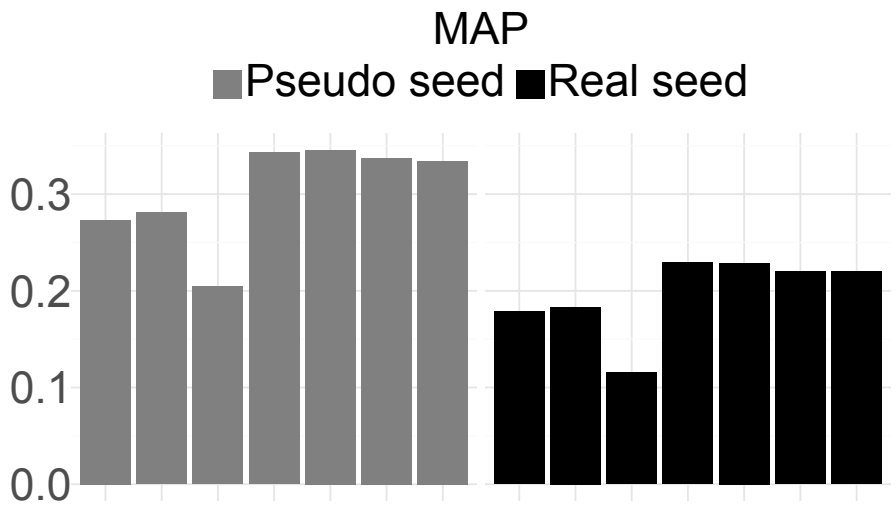


Figure 4.6: Comparison between real seed and pseudo seed studies across ranking methods (x axis are different methods shown in the same order as in Table 4.3). We present only the MAP scores, as all other evaluation measures we considered showed the same trend.

nDCG@10), where QLM achieves the highest performance. This may be because embedding-based methods help alleviate vocabulary mismatch, improving recall but often at the cost of early precision.

For multiple pseudo seed studies, SDR achieves the highest effectiveness across most evaluation metrics, except precision@1000, recall@1000, and LR%. Interestingly, the fusion method does not outperform the individual retrieval methods. This may be due to the poor performance of the AES method, highlighting an open challenge: automatically determining when fusion is beneficial. Overall, using multiple pseudo seed studies consistently outperforms using a single pseudo seed study.

All these findings are in line with our previous reproduction study (Chapter 3), with the exception that fusion does not yield improvements here.

Comparison between Pseudo Seed and Real Seed

Finally, we observe that using real seed studies versus multiple pseudo seed studies has a substantial impact on effectiveness. Although seed studies can outperform single pseudo seed studies in some cases, their effectiveness is significantly lower than that of multiple pseudo seed studies. One possible explanation is that seed studies are not always included in the final systematic review (while pseudo seed studies, by definition, are). Using a non-relevant seed study could degrade performance. This effect is illustrated in Figure 4.6, where the effectiveness of using all seed studies is closer to that of using a single pseudo seed study than to that of using multiple pseudo seed studies.

Furthermore, while SDR and fused SDR methods perform best when using multiple pseudo seed studies, QLM is the most effective method when using real seed studies. We hypothesise that this is because seed studies are often used in practice for term extraction, a process better aligned with QLM than with embedding-based semantic matching.

In conclusion, pseudo seed studies are not representative of real seed studies, and this distinction has important implications for seed-driven retrieval (SDR). This underscores the importance of test collections that include real seed studies, such as the one provided in this work.

Table 4.4: Results of SDR methods on our collection when the seed-snowballing set and retrieved studies are combined. Denotations are identical to those in the caption of Table 4.3.

	Method	MAP	Precision			Recall			nDCG			LR%
			10	100	1000	10	100	1000	10	100	1000	
seed-snow+search	BM25-BOW	.0936†	.0949†	.0421†	.0137	.0943†	.3372†	.7767†	.1175†	.2074†	.3469†	.7389†
	QLM-BOW	**.2672**	.2744	.0813	.0163	.2453	**.5483**	.8612	.3490	**.4269**	**.5326**	**.5236**†
	SDR-BOW	.2663	**.2897**	**.0815**	.0159	**.2592**	.5390	.8529	**.3503**	.4212	.5244	.5672
	AES-BOW	.1886†	.1949†	.0690	**.0165**	.1815†	.4619†	.8595	.2466†	.3366†	.4668†	.5921
	AES-BOW-P	.1950†	.2026†	.0700	.0162	.1906†	.4755	.8530	.2548†	.3459†	.4700†	.5514
	SDR-BOW-AES	.2572	.2513†	.0787	.0164	.2297†	.5210	**.8675**	.3271†	.4136	.5271	.5521
	SDR-BOW-AES-P	.2546	.2487†	.0779	.0164	.2343	.5224	.8666	.3266	.4121	.5245	.5298

4.3.3 Screening Prioritisation Findings

From the SDR experiment, we found that:

- Using pseudo seed studies produces overly optimistic results compared to using real seed studies (i.e., the evaluation scores are higher than those obtained with real seed studies). This is likely because pseudo seed studies serve as explicit relevance feedback, whereas real seed studies function more like pseudo relevance feedback.

- The effectiveness of different ranking models (e.g., BM25, QLM, SDR, AES) depends on the type of seed studies used. This may be attributed to differences in term weighting—relevant terms are more likely to appear in pseudo seed studies but may be absent or less prominent in real seed studies.

4.4 Use Case 3: Ranking With Snowballing

Our final demonstration of the use cases of our collection introduces a new technique developed: ranking with snowballing (see 3 in Figure 4.1).

In Section 4.3, we investigated the effectiveness of SDR using real seed studies versus pseudo seed studies. We found that unlike pseudo seed, when real seed studies are used, the QLM method outperforms the SDR method across most evaluation measures. In this experiment, we investigate whether similar results arise when the seed-snowballing set is combined with the retrieved studies.

Our collection provides two snowballing sets for each topic. Using the seed studies and these snowballing sets, we examine the impact of snowballing within the SDR framework. Specifically, we investigate:

1. The effectiveness of SDR on a combined set of seed-snowballing and retrieved studies (i.e., integrating the seed-snowballing set into the retrieved set and then ranking);

2. The effectiveness of ranking post-screening (i.e., simulating the ranking of the screened-snowballing set using the screened retrieved studies as input).

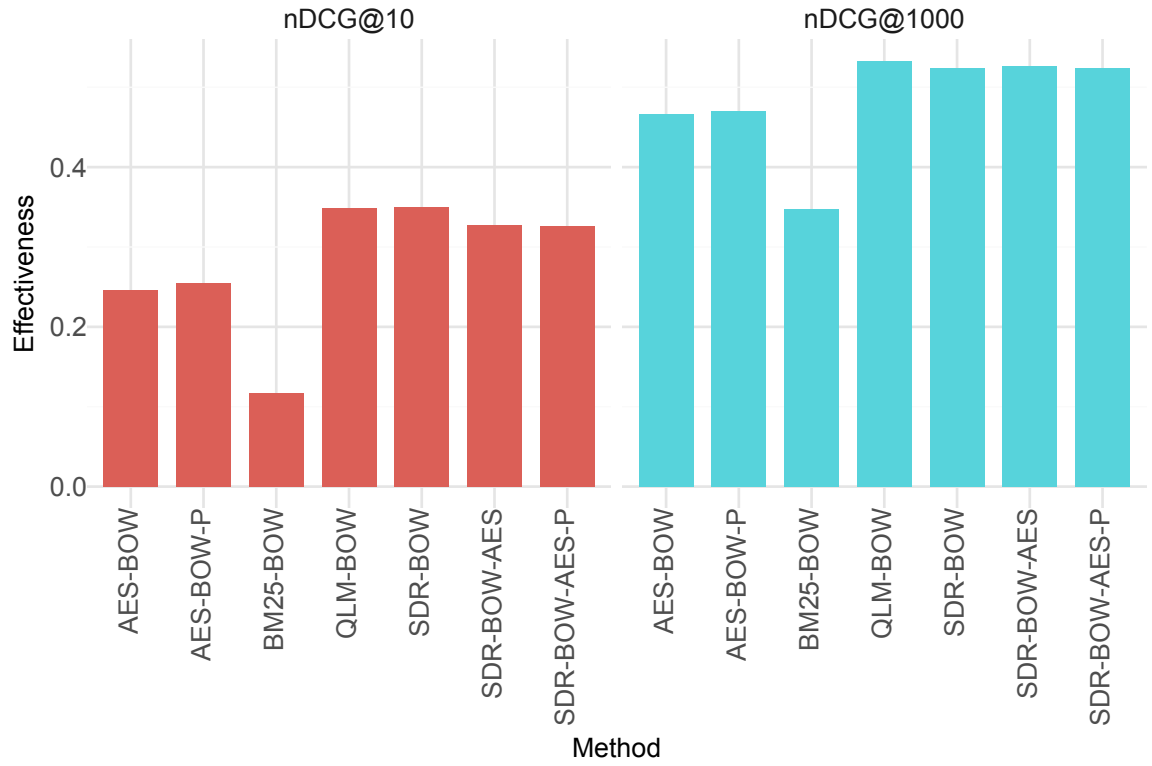


Figure 4.7: Comparison between nDCG@10 and nDCG@1000 for the results in Table 4.4.

4.4.1 Ranking on Combined seed-snowballing and retrieved studies

As the first experiment, we combine the retrieved studies and the seed-snowballing set and investigate the effectiveness of ranking method on this combined set.

Results for this experiment are shown in Table 4.4. The most effective methods are QLM and SDR, with SDR more effectiveness on shallow measures and more effectiveness on deeper measures. These results are highlighted further in Figure 4.7. When comparing these combined results with the seed studies result from Table 4.3, we found that all methods can achieve higher recall{@100,@1000} and LR%, which suggests that adding the seed snowballing set can significantly increase the number of relevant studies retrieved.

Despite the fact that there are more studies overall when combining the seed-snowballing set with retrieved studies, it is worth it in terms of the overall improvement in effectiveness. In practise, it is beneficial to combine the results of seed-snowballing with retrieved studies for screening prioritisation.

4.4.2 Ranking on Screened snowballing document

As a second experiment, we simulate the process of screening prioritisation for the screened-snowballing set. The included studies from the retrieved set, along with the seed studies, are used as input for SDR. Given the difference in performance between the two sets of input studies (i.e., retrieved included studies and seed studies), one could consider applying different term weighting functions depending on the source of the study. However, we leave this investigation for future work. We perform our evaluation on the included studies that do not appear in the retrieved set. As a result, the outcomes of this

Table 4.5: Results of SDR methods on our collection when using the screened-snowballing set. Denotations are identical to those in the caption of Table 4.3.

	Method	MAP	Prec@100	Recall@100	nDCG@100	LR%
screened-snowball	BM25-BOW	0.0208†	0.0100	0.2045	0.0707†	0.4606†
	QLM-BOW	**0.1279**	0.0212	0.3090	**0.2104**	0.3317
	SDR-BOW	0.0921	0.0218	0.3055	0.1829	0.3514
	AES-BOW	0.0474	0.0218	0.3167	0.1436	0.3412
	AES-BOW-P	0.0602	0.0206	0.3111	0.1530	0.3329
	SDR-BOW-AES	0.0898	**0.0226**	**0.3199**	0.1858	0.3257
	SDR-BOW-AES-P	0.0871	0.0218	0.3149	0.1815	**0.3214**

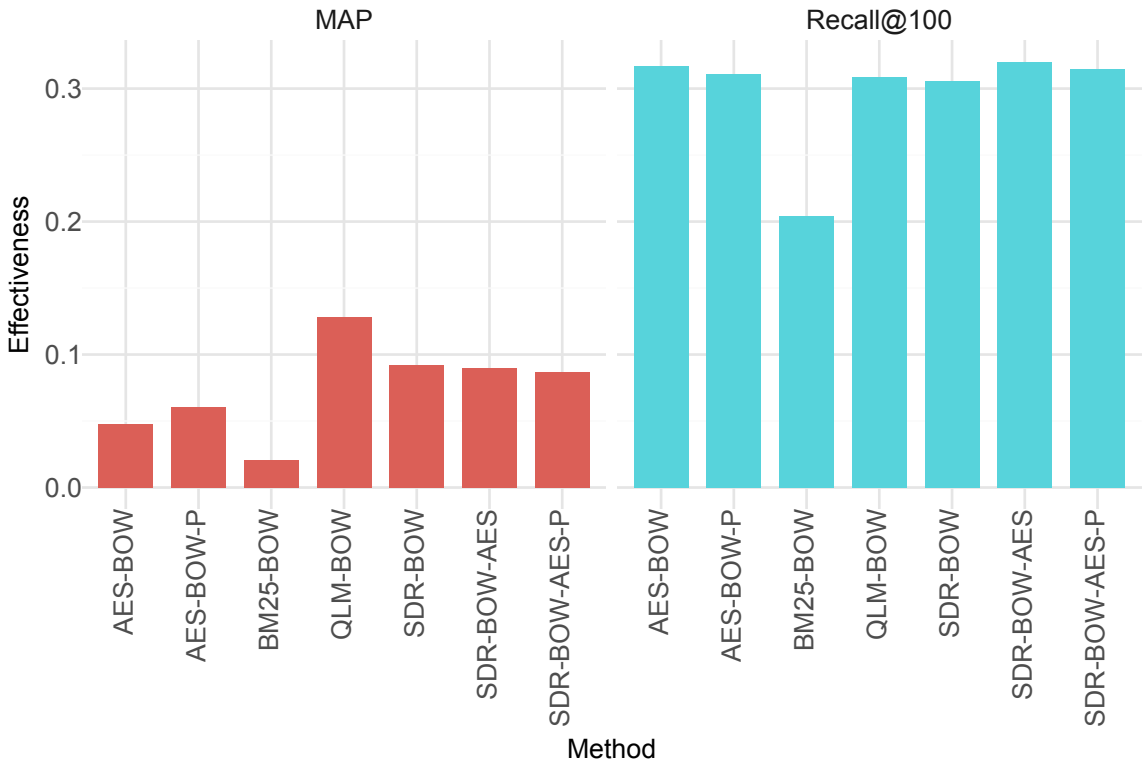


Figure 4.8: Comparison between MAP and recall@100 for the results in Table 4.5.

experiment are not directly comparable to those from the earlier screening prioritisation experiments presented in this paper. We report a subset of the evaluation measures as described in Section 4.3.1.

Results for this experiment are shown in Table 4.5. The two best-performing methods are QLM and AES. QLM achieves the highest effectiveness for MAP and nDCG@100, while the fusion methods perform best for precision@100, recall@100, and LR%. These results are visualised in Figure 4.8. These findings demonstrate that QLM is particularly effective for the SDR task when ranking screened-snowballing studies.

4.4.3 Ranking With Snowballing Findings

From the ranking with snowballing experiments, we find that:

- Combining the seed-snowballing set with retrieved studies is beneficial for SDR.

- QLM and AES perform best for ranking screened-snowballing documents. However, there remains room for future work in determining the optimal combination of study sets to use for SDR.

4.5 Summary of Findings

We present a test collection to properly evaluate systematic review literature search methods that use seed studies. Along with the test collection, we provide a detailed analysis of this resource.

Firstly, we find that only a small portion of seed studies are actually included in a systematic review. This is because not all seed studies provided to an information specialist satisfy the inclusion criteria for the review; some are supplied to offer a broader understanding of the topic. Thus, the use of pseudo seed studies drawn from the included studies does not reflect all the properties of actual seed studies used in practice. Secondly, we investigate the impact of seed studies by reproducing two existing methods that rely on seed studies [128, 211], but which are originally evaluated using pseudo seed studies (i.e., a sample from the included studies). Our experiments show that using pseudo seed studies overestimates the effectiveness of these methods.

The test collection also enables an investigation of the differences between snowballing seed studies and screened studies. We find that ranking a combined list of seed-snowballing documents and retrieved candidate documents is beneficial to exisiting SDR ranking method.

This test collection makes two important contributions: (1) it enables a more realistic evaluation of methods that use seed studies, as all included studies can be used for relevance assessment—eliminating the need to hold out a portion as pseudo seed studies; and (2) it provides realistic data to develop and train new methods that utilise seed studies.

Part II
Enhancing Query Formulation

In evidence-based medicine, the construction of a high-quality systematic review requires researchers to examine all available evidence related to a specific research question. Studies that meet the inclusion criteria are selected, evaluated, and synthesised. To gather the initial set of potentially relevant documents, Boolean queries are developed to search medical databases. Boolean queries offer reproducibility, explainability, and the ability to filter out irrelevant articles [148], thereby reducing the workload of unnecessary document assessments.

However, constructing an effective Boolean query is challenging due to the complexity of accurately representing a research question using Boolean logic. Even experienced systematic review researchers often produce suboptimal queries [95]. To address this, automatic Boolean query formulation and refinement methods have been developed to support researchers in creating better queries for systematic reviews [16, 95, 181, 212, 213, 214, 215, 216].

Boolean Query Formulation refers to the task of creating a Boolean query from scratch, often using one or more "seed" documents—example studies relevant to the systematic review. **Boolean Query Refinement** refers to improving an existing Boolean query, for example by modifying initial Boolean clauses and operators to increase precision (i.e., reducing the number of irrelevant documents retrieved) while maintaining recall, or by suggesting specific terms (such as synonyms, keywords, or MeSH terms) to append to the query. Although several techniques have demonstrated effectiveness and have been integrated into tools that support query creation [133, 207, 217, 256], automated methods still fall short of generating high-quality Boolean queries.

This part introduces the AI methodology developed during my candidature to automate the creation of Boolean queries, a critical component of systematic reviews. Our approach comprises two distinct yet complementary strategies: direct Boolean query formulation and refinement using Large Language Models (LLMs), and Boolean query refinement through MeSH term suggestion.

The first strategy, detailed in Chapter 5, uses LLMs to automatically generate Boolean queries. This method leverages the generative capacity of LLMs to produce queries that are both syntactically correct and contextually aligned with specific systematic review topics. The effectiveness of this approach is evaluated through experiments that test different prompts and model configurations to assess whether current LLMs can reliably generate high-quality Boolean queries for systematic reviews.

On the other hand, Chapter 6 describes the second strategy, which supports Boolean query refinement by suggesting Medical Subject Headings (MeSH terms). MeSH terms are part of a controlled vocabulary maintained by the U.S. National Library of Medicine and are updated annually. They are used to index articles in biomedical databases such as PubMed, helping to standardise terminology and improve search precision. Identifying appropriate MeSH terms can be challenging for information specialists, particularly given their specificity and evolving structure. To address this, the thesis formalises the task of MeSH term suggestion for systematic reviews using the CLEF TAR datasets (described in Chapter 2). Existing methods are benchmarked against this task, and BERT-based models are introduced to improve the effectiveness of MeSH term suggestion.

Chapter 5

LLM-based Boolean Query Formulation

Advances in text generation models have significantly improved the effectiveness of task-based question-answering systems and the quality of responses aligned with user instructions [79, 137]. One of the first milestones in this development is ChatGPT, the model developed by OpenAI and first released commercially in December 2022. At the time, the model was recognised as one of the most advanced text generation systems available, achieving state-of-the-art results across a variety of natural language processing tasks [137]. Despite its strengths, ChatGPT operates as a commercial model with a largely opaque, black-box nature; its architecture, training data, and operational details are not publicly disclosed. This lack of transparency presents challenges for researchers and practitioners seeking to understand, replicate, or validate its performance.

This chapter begins by introducing our methodology for prompting LLMs to generate Boolean queries for systematic reviews. Section 5.1 outlines our prompting strategies, including both unguided and guided approaches for query formulation and refinement. Section 5.2 then describes the experimental setup, including datasets, models, and evaluation measures. For results, we first evaluate the feasibility of using ChatGPT via its web interface for Boolean query generation, as described in Section 5.3. Based on the insights gained from this initial study, Section 5.4 expands the evaluation to a broader set of LLMs—both proprietary API-based models and open-source alternatives [244, 245]—to assess the generalisability of our findings.

5.1 Prompting Methods for Generating Boolean Queries

We investigate the use of LLMs to generate Boolean queries for systematic review literature search. The basic mechanism employed by LLMs is to take an input sequence of text (called a *prompt*), process it through the model, and output the next token in the sequence. This process is repeated iteratively to generate a complete response. Typical LLMs rely on the Transformer architecture [248] and are trained on large-scale text corpora, enabling them to learn patterns in language usage. During generation, the model draws on these learned patterns to produce text similar to what it has seen during training.

A key component in interacting with LLMs is the design of the prompt used to instruct the model.

Table 5.1: Prompts for unguided prompt query formulation

	ID	Prompt
Simple	p1	For a systematic review titled "{review_title}", can you generate a systematic review Boolean query to find all included studies on PubMed for the review topic?
Detailed	p2	You are an information specialist who develops Boolean queries for systematic reviews. You have extensive experience developing highly effective queries for searching the medical literature. Your specialty is developing queries that retrieve as few irrelevant documents as possible and retrieve all relevant documents for your information need. Now you have your information need to conduct research on {review_title}. Please construct a highly effective systematic review Boolean query that can best serve your information need.
	p3	Imagine you are an expert systematic review information specialist; now you are given a systematic review research topic, with the topic title "{review_title}". Your task is to generate a highly effective systematic review Boolean query to search on PubMed (refer to the professionally made ones); the query needs to be as inclusive as possible so that it can retrieve all the relevant studies that can be included in the research topic; on the other hand, the query needs to retrieve fewer irrelevant studies so that researchers can spend less time judging the retrieved documents.
With Example	p4	You are an information specialist who develops Boolean queries for systematic reviews. You have extensive experience developing highly effective queries for searching the medical literature. Your specialty is developing queries that retrieve as few irrelevant documents as possible and retrieve all relevant documents for your information need. You are able to take an information need such as: "{example_review_title}" and generate valid pubmed queries such as: "{example_review_query}". Now you have the information need to conduct research on "{review_title}", please generate a highly effective systematic review Boolean query for the information need.
	p5	You are an information specialist who develops Boolean queries for systematic reviews. You have extensive experience developing highly effective queries for searching the medical literature. Your specialty is developing queries that retrieve as few irrelevant documents as possible and retrieve all relevant documents for your information need. A professional information specialist will extract PICO elements from information needs in a common practice in constructing a systematic review Boolean query. PICO means Patient/ Problem, Intervention, Comparison and Outcome. PICO is a format for developing a good clinical research question prior to starting one's research. It is a mnemonic used to describe the four elements of a sound clinical foreground question. You are able to take an information need such as: "{example_review_title}" and you generate valid pubmed queries such as: "{example_review_query}". Now you have your information need to conduct research on "{review_title}". First, extract PICO elements from the information needs and construct a highly effective systematic review Boolean query that can best serve your information need.

We iteratively design a series of prompts with increasing complexity to generate Boolean queries for systematic review literature search, including those that incorporate example generation. We also experiment with guided prompts, which divide the query generation task into multiple steps and incorporate insights from existing objective methods for query formulation [96, 213, 216]. Prompts are designed for two tasks related to Boolean query generation: the formulation task and the refinement task.

5.1.1 Unguided Prompts for Query Formulation

Unguided Query Formulation prompts refer to prompts that instruct the LLM to formulate a systematic review Boolean query in a single step (i.e., without interaction or iteration). Table 5.1 presents the five prompts we develop for the Boolean query formulation task. These prompts are grouped into three categories: *simple*, *detailed*, and *with example*. A *simple* prompt uses a single sentence to briefly describe the Boolean query formulation task to the LLM. This type of prompt reflects a common use case in which users may not be familiar with constructing high-quality prompts. A *detailed* prompt provides a background scenario that explains what systematic review Boolean query formulation involves and what is expected for an effective query. Finally, a *with example* prompt includes an additional Boolean query example, offering more context to help the LLM better understand the task.

The difference between the two *detailed* prompts ('p2' and 'p3') lies only in how the background context is presented; they are simply two alternative ways of framing the same task, without any intended distinction in meaning or instructional content. For the *with example* prompts ('p4' and 'p5'), 'p5' further incorporates a Boolean query formulation strategy that explicitly identifies PICO elements. PICO stands for Patient/Problem, Intervention, Comparison, and Outcome, and is commonly used to frame research questions and construct high-quality systematic review Boolean queries [66]. The motivation for including PICO element extraction in the prompt is to help the LLM understand the logical structure that underlies the example query.

Each Boolean query formulation corresponds to a specific systematic review topic from the collection. In each formulation, we replace {`review_title`} with the title of the published systematic review, which typically serves as the representation of the topic in established systematic review test collections such as CLEF TAR and the Seed Collection [113, 261].

For prompts that require an example (e.g., p4, p5), a Boolean query formulation example is provided to guide the generation of queries for any given systematic review topic. This requires two placeholders: {`example_review_title`} and {`example_review_query`}. In these example-based prompts, we replace {`example_review_title`} with the title of the selected example review, and {`example_review_query`} with the corresponding Boolean query. For instance, for the example review titled "Thromboelastography (TEG) and rotational thromboelastometry (ROTEM) for trauma-induced coagulopathy in adult trauma patients with bleeding," the following Boolean query is inserted:

```
(Thrombelastography[mesh:noexp] OR
(thromboelasto*[All Fields] OR thrombelasto*[All Fields] OR
ROTEM[All Fields] OR \tem international"[All Fields] OR
(thromb*[All Fields] AND elastom*[All Fields]) OR
(rotational[All Fields] AND thrombelast[All Fields])) OR
(Thrombelastogra*[All Fields] OR Thromboelastogra*[All Fields] OR
TEG[All Fields] OR haemoscope[All Fields] OR haemonetics[All Fields] OR
(thromb*[All Fields] AND elastogra*[All Fields])))
```

The example Boolean query formulation is sampled from the test collection and is always different from the target review topic for which the Boolean query is being generated. In this study, we consider two strategies for selecting example reviews:

1. **High-Quality Example (HQE):** We group all systematic reviews topics from the test collections, and rank them by the effectiveness of their manually formulated Boolean queries to retrieve included studies. Effectiveness is assessed primarily by recall, followed by precision; In the end, we select the top-1 as the HQE.

2. **Related Example (RE):** We use a monoBERT architecture to select the most semantically or topically related example review from the test collection [164] to act as and example Boolean query formulation. To do this, we concatenate the title of the target review with each candidate example title and input these pairs into PubMedBERT—a domain-specific BERT model pre-trained on PubMed abstracts [86]. The model produces a classification score indicating the semantic relatedness between the two titles. The top-ranked example is selected for inclusion in the prompt.

5.1.2 Guided Prompts for Query Formulation

We also design a multi-step prompt that follows the logic of one of the current state-of-the-art automated query formulation methods, namely the objective method [216]. This method relies on identifying statistically salient terms from a small set of seed studies, often using tools like PubReMiner or TF-IDF to extract candidate terms, which are then categorised and structured into a Boolean query. A full description of this method is provided in in Section 2.2.1 in Chapter 2.

Table 5.2 presents our guided prompt design. Specifically, we follow a four-step pipeline to generate a Boolean query. In the first step, query terms are identified from a seed study. In the second step, these terms are classified into four categories: terms related to health conditions (A), treatments (B), study design types (C), and other terms (N/A). In the third step, terms within the same category are grouped using the 'OR' operator, and the category groups are combined using the 'AND' operator. Finally, ChatGPT is instructed to refine the query by adding additional terms, such as MeSH terms. This guided prompting approach enables finer control over the terminology used in Boolean query formulation and breaks the complex task into multiple, more manageable subtasks.

5.1.3 Unguided Prompts for Query Refinement

Unguided Boolean query refinement prompts refer to prompts that provide an LLM with a review topic and a corresponding Boolean query, and instruct the model to modify the query to improve search effectiveness. These prompts can be used by reviewers to refine queries they have created or be integrated into a more complex automated pipeline. Table 5.3 presents the query refinement prompts we develop.

Table 5.2: Example designed guided prompt for query formulation.

Step	Prompt	Example Answer
Step1	Follow my instructions precisely to develop a highly effective Boolean query for a medical systematic review literature search. Do not explain or elaborate. Only respond with exactly what I request. First, Given the following statement and text from a relevant study, please identify 50 terms or phrases that are relevant. The terms you identify should be used to retrieve more relevant studies, so be careful that the terms you choose are not too broad. You are not allowed to have duplicates in your list. statement: "Prevalence of Differentiated Thyroid Cancer in Autopsy Studies Over Six Decades: A Meta-Analysis" Text: Ten occult carcinomas of the thyroid gland were found in 274 unselected autopsies at the Pathology Service, Hospital José Joaquín Aguirre, between December 1980 and March 1983. This is the lowest incidence among the most recent published series. The smallest papillary microcarcinomas presented most frequently as nonencapsulated nonsclerosing tumors without inflammatory infiltrate or fibrosis, suggesting that they may represent the early stages of development. Our data show a relatively high and similar frequency of papillary microcarcinomas in surgical and autopsy series, but do not demonstrate risk factors for clinical evolution.	1. Differentiated thyroid cancer 2. Prevalence 3. Autopsy studies 4. Meta-analysis 5. Occult carcinomas 6. Thyroid gland 7. Incidence 8. Environmental factors 9. Etiology 10. Low risk ... 50. Parameters.
Step 2	For each item in the list you created in Step 1, classify it into three categories: terms relating to health conditions (A), terms relating to a treatment (B), terms relating to types of study design (C). When an item does not fit one of these categories, mark it as (N/A). Each item needs to be categorised into (A), (B), (C), or (N/A).	1. (A) Differentiated thyroid cancer 2. (N/A) Prevalence 3. (C) Autopsy studies 4. (C) Meta-analysis 5. (A) Occult carcinomas 6. (A) Thyroid gland 7. (N/A) Incidence 8. (N/A) Environmental factors 9. (N/A) Etiology 10. (N/A) Low risk ... 50. (N/A) Parameters.
Step 3	Using the categorised list you created in step 2, create a Boolean query that can be submitted to PubMed which groups together items from each category. For example: ((itemA1[Title/Abstract] OR itemA2[Title/Abstract] or itemA2[Title/Abstract]) AND (itemB1[Title/Abstract] OR itemB2[Title/Abstract] OR itemB3[Title/Abstract]) AND (itemC1[Title/Abstract] OR itemC2[Title/Abstract] OR itemC3[Title/Abstract]))	(((differentiated[Title/Abstract] OR thyroid[Title/Abstract] OR carcinoma[Title/Abstract] OR papillary[Title/Abstract] OR microcarcinoma[Title/Abstract]) AND (cancer[Title/Abstract] OR incidence[Title/Abstract] OR etiology[Title/Abstract] OR risk[Title/Abstract] OR gender[Title/Abstract] OR hormonal[Title/Abstract] OR nodular[Title/Abstract] OR goiter[Title/Abstract] OR Hashimoto's[Title/Abstract] OR malignancy[Title/Abstract] OR concomitant[Title/Abstract] OR tumor[Title/Abstract] OR infiltrate[Title/Abstract] OR fibrosis[Title/Abstract] OR development[Title/Abstract] OR frequency[Title/Abstract])) AND (autopsy[Title/Abstract] OR surgical[Title/Abstract] OR material[Title/Abstract] OR series[Title/Abstract] OR specimens[Title/Abstract] OR cases[Title/Abstract])).
Step 4	Use your expert knowledge to refine the query, making it retrieve as many relevant documents as possible while minimising the total number of documents retrieved. Also add relevant MeSH terms into the query where necessary, e.g., MeSHTerm[MeSH]. Retain the general structure of the query, however, with each main clause of the query corresponding to a PICO element. The final query still needs to be executable on PubMed, so it should be a valid query.	(((differentiated thyroid cancer[MeSH] OR "differentiated thyroid"[All Fields] OR "thyroid carcinoma"[All Fields] OR "papillary microcarcinoma"[All Fields]) AND (prevalence[All Fields] OR incidence[MeSH] OR "etiology of"[All Fields] OR "risk factors"[All Fields] OR gender[All Fields] OR hormonal[All Fields] OR "nodular goiter"[All Fields] OR "Hashimoto's thyroiditis"[MeSH] OR malignancy[MeSH] OR "concomitant lesion"[All Fields] OR tumor[All Fields] OR infiltrate[All Fields] OR fibrosis[All Fields] OR "early stages of development"[All Fields] OR frequency[All Fields])) AND (autopsy[MeSH] OR surgical[All Fields] OR material[All Fields] OR series[All Fields] OR specimens[All Fields] OR cases[All Fields]))

We categorise the prompts into two types: *simple* and *with example*. Similar to the unguided prompts for query formulation, the *simple* prompt 'p6' consists of a brief instruction describing the task. The *with example* prompt 'p7' includes an example that illustrates what constitutes a successful query refinement. When prompting an LLM to refine queries, we replace {`initial_query`} with the Boolean query to be refined. For prompts that include an example, we additionally replace {`example_review_title`} with the title of the example review topic, {`example_review_initial_query`} with the initial query of the example topic, and {`example_review_refined_query`} with the refined version of that query.

Table 5.3: Prompts for unguided prompt query refinement

	ID	Prompt
Simple	p6	For a systematic review seed Boolean query: "{initial_query}", This query retrieves too many irrelevant documents and too few relevant documents about the information need: "{review_title}", Please correct this query so that it can retrieve fewer irrelevant documents and more relevant documents.
With Example	p7	For a systematic review seed Boolean query: "{example_review_initial_query}" ,This query retrieves too many irrelevant documents and too few relevant documents about the information need: "{example_review_title}", therefore it should be corrected to: "{example_review_refined_query}". Now your task is to correct a systematic review Boolean query: "{initial_query}" for information need "{review_title}", so it can retrieve fewer irrelevant documents and more relevant documents.

5.2 Experimental Setup

5.2.1 Datasets

Our experiments are conducted using two collections: the CLEF Technology-Assisted Reviews (TAR) collections [113, 114, 115] and the Seed Collection (refer to Chapter 4). For CLEF TAR, we use the 2017 and 2018 editions, as both consist exclusively of diagnostic test accuracy (DTA) systematic reviews. Note that some topics from CLEF TAR 2017 are duplicated in the 2018 collection; after removing these duplicates, we obtain 72 unique review topics. The Seed Collection contains 40 topics, each accompanied by the seed studies originally used by systematic review researchers to guide Boolean query formulation. For both collections, each topic includes the review title, the Boolean query used for literature retrieval, and the relevance assessments of the retrieved documents.

The *with example* prompts use topic CD010438 from the CLEF TAR collection as the high-quality example (HQE), selected for its simple structure that is easier for LLMs to follow.

The guided prompting method is based on the objective prompting approach, which requires at least one seed study as input for query formulation. Therefore, we do not apply guided prompting to the CLEF TAR collection, as it does not include seed studies.

5.2.2 Models

The experiments are designed around four groups of models, each chosen to represent a distinct category of language model availability— including commercial API-based models, managed API deployments, and open-source models that can be run locally.

1. **The ChatGPT Interface (as available in late 2022):** The initial investigation into the capabilities of LLMs to generate systematic review Boolean queries was conducted from December 2022 to January 2023 [264], aligned with the SIGIR 2023 submission deadline (January 31, 2023). At that time, ChatGPT was only accessible via its user interface; OpenAI had not yet released API access or provided versioning information. This limited transparency illustrates

the "black-box" nature of commercial LLMs, making it difficult to replicate experiments or understand what influences model behaviour.

2. **ChatGPT API-based Models:** Following the initial study, we updated our experimental design to leverage the improved accessibility and version control offered by the ChatGPT API. These experiments were conducted during the thesis writing period. The API enables controlled interaction by allowing model selection and random seed specification, improving reproducibility. We evaluate the following API-accessible models: `gpt-3.5-turbo-1106` (GPT3.5-1), `gpt-3.5-turbo-0125` (GPT3.5-0), `gpt-4-1106-preview` (GPT4), and `gpt-4o` (GPT4o-m). Model selection is informed by a recent reproducibility study of our earlier work [229, 264]. We also evaluate the recently released `O1` model, which shows strong reasoning performance across various NLP benchmarks.

3. **Mistral API-based Models:** Beyond ChatGPT, we also assess two Mistral-based models accessed via managed APIs for automatic Boolean query formulation. These include: **Mistral-7B-v0.2 (Mistral-S):** A compact, 7-billion-parameter model updated to version 0.2, designed for balanced performance across general NLP tasks [108]. **Mixtral-8x7B-v0.1 (Mistral-L):** A Mixture-of-Experts (MoE) model that combines eight specialised 7B-parameter submodels. Although it activates only 2 experts (14B total) during inference, it retains efficiency while handling more complex reasoning tasks [109].

4. **Open-Source Local Models:** We also evaluate several open-source models in a fully offline setting to assess performance without relying on commercial APIs. These include: **Mistral-7B-v0.2 (Mistral):** The local deployment of the Mistral model, enabling complete customisation and independence from external services. **Zephyr-7B-beta (Zephyr):** An experimental 7B-parameter model designed for improved logical reasoning and instruction-following [246]. **Llama3.1-8B-Instruct (Llama3.1):** A refined 8B instruction-tuned model from the Llama3 series by Meta, with enhanced training and decoding strategies for better alignment and text generation [80].

5.2.3 Three-Step Validation

After obtaining Boolean queries from the LLMs, we implemented a three-step validation process to ensure syntactic correctness and quality for systematic review use. This validation is essential to filter out incorrectly formatted or ineffective queries.

1. **LLM-based extractor:** Since LLM-generated outputs may include explanations or reasoning alongside the Boolean query, we employed an LLM-based extractor to isolate the query component. For consistency, we always used GPT-3.5-turbo-0125 with a temperature of 0.

2. **Rule-based checker:** The extracted queries were passed through a rule-based checker to ensure compliance with Boolean syntax. We enforced the following rules: (1) Every bracket must

be properly closed. (2) Only three Boolean operators are allowed: `NOT`, `AND`, and `OR`. (3) Each operation must use a single operator at a time; consecutive operators (e.g., `AND OR`) are considered invalid. (4) Each operator must be preceded by a term or an opening bracket.

3. **Boolean query validator:** We used PubMed's Entrez API [203] to verify whether the automatically generated Boolean queries returned a reasonable number of documents. A query was considered valid if it retrieved between 1 and 1,000,000 documents. All queries were executed using the original search date to maintain consistency with the time the topics were created.

Whenever a query failed validation, a new attempt was made by regenerating the prompt response. To avoid excessive regeneration, we capped the process at 20 attempts per topic. If no valid query was produced within this limit, the last generated query was retained, regardless of its validity.

5.2.4 Evalution

To evaluate the generated Boolean queries, we execute them using PubMed's Entrez API to retrieve PubMed IDs [28]. We assesse the effectiveness of each formulated Boolean query based on the retrieved set of documents, using standard evaluation measures including precision, recall, and F-measure (F1 and F3). These metrics are consistent with those used in Chapter 4.

5.3 Initial Evaluation Using the ChatGPT Interface

To explore the feasibility of using LLM for Boolean query formulation in systematic reviews, we first conducted an initial investigation using the ChatGPT interface. This preliminary study aimed to assess the capabilities of ChatGPT in generating and refining Boolean queries. To guide our exploration, we formulated the following research questions:

RQ1: How does ChatGPT compare to previous automatic methods for formulating and refining Boolean queries in systematic reviews?

RQ2: To what extent do prompt designs influence the effectiveness of Boolean queries generated by ChatGPT for systematic reviews?

RQ3: What is the impact of guiding ChatGPT through multi-step prompts that mimic existing automated Boolean query generation methods?

RQ4: What are the limitations and challenges of using ChatGPT for Boolean query formulation in systematic reviews?

The following sections present the results of our experiments involving both unguided and guided Boolean query generation using the ChatGPT interface.

Table 5.4: Unguided prompt query formulation results. Statistical significant differences (Student's two-tailed, paired t-test with Bonferonni correction, $p < 0.05$) between p4 and all other methods are indicated by *.

		Prompts	Precision	F1	F3	Recall
CLEF TAR	Baselines	Manual	0.0207*	0.0290*	0.0481*	0.8317*
		Conceptual	0.0015*	0.0027*	0.0101*	0.6997*
		Objective	0.0002*	0.0005*	0.0023*	**0.9128***
	Simple	p1	0.0543	0.0500	0.0590	0.1293*
	Detailed	p2	**0.1166**	0.0654	0.0696	0.1310*
		p3	0.0844	0.0443	0.0497*	0.1175*
	With Example	p4	0.0752	0.0642	**0.0847**	0.5035
		p5	0.0958	**0.0717**	0.0844	0.3335*
Seed Collection	Baselines	Manual	0.0367	**0.0651***	**0.1099***	**0.7366***
		Conceptual	0.0018			0.4138
		Objective	0.0057			0.5192
	Simple	p1	0.0501	0.0274	0.0298	0.0528
	Detailed	p2	**0.0983**	0.0310	0.0278	0.0394*
		p3	0.0730	0.0329	0.0329	0.0519
	With Example	p4	0.0283	0.0274	0.0374	0.1290
		p5	0.0188	0.0193	0.0271	0.0785

5.3.1 Unguided Prompt Query Formulation

Table 5.4 reports the results of unguided prompt query formulation. These results indicate that queries generated from ChatGPT generally obtain a higher precision compared to the previous automatic query formulation methods, with a trade-off of lower recall. For F-measure, ChatGPT-generated queries are more effective than both the state-of-the-art and originally authored queries on the CLEF collections. However, they are less effective on the Seed collection. Systematic review literature search generally requires high recall to ensure that all relevant evidence can be found. All the ChatGPT-generated queries obtain a lower recall than the baseline methods, suggesting that ChatGPT-generated queries may not be suitable for high-recall retrieval, but rather best suited when time is limited; e.g., for rapid reviews [154].

Using different Simple and Detailed prompts (p1–3) only had a minor impact on effectiveness. For CLEF, p2 was statistically significantly better with respect to precision; otherwise, the prompt type did not have a strong effect. However, we found that prompts that include a high-quality systematic review topic as an example are able to significantly outperform those without, shown as a consistently higher F1, F3 and recall. When comparing the effectiveness of two prompts with examples, we found that asking ChatGPT to generate PICO elements before generating its final Boolean query resulted in Boolean queries with a lower recall but higher precision. Overall, our findings indicate that including a high-quality systematic review query example in the prompt is crucial, while the level of detail in the

Table 5.5: Comparison of result for unguided prompt query generation prompt 'p4' when using a different types of examples. For each collection, two types of example are used, $p4-HQE$ refers to using one high-quality example, while $p4-RE$ refers to using a related query as an example. Statistical significant differences ($p < 0.05$) between the two types of examples are indicated by *.

		Precision	F1	F3	Recall
CLEF TAR	p4-HQE	0.0751	0.0642	0.0872	0.5035
	p4-RE	0.1105(+47.1%)	0.0909(+41.6%)	0.1144(+31.2%)	0.4183(−38.1%)
Seed Collection	p4-HQE	0.0283	0.0274	0.0374	0.129
	p4-RE	0.0351(+24.0%)	0.0140(−48.9%)	0.0139(−62.8%)	0.0161*(−87.6%)

task description may not have a significant impact.

Additionally, we observe a notable difference between the CLEF TAR and Seed Collection when comparing automatically generated queries to manual queries. In the CLEF TAR collection, automatic methods—particularly the objective method—still benefit from achieving relatively high recall, which contributes to their overall effectiveness. In contrast, for the Seed Collection, automatically generated queries consistently exhibit lower recall, leading to reduced overall effectiveness. While we do not have a definitive explanation for this discrepancy, one possible reason is the difference in topic composition: all topics in the CLEF TAR collection are diagnostic test accuracy (DTA) reviews, whereas the Seed Collection includes a mix of both DTA and intervention review topics.

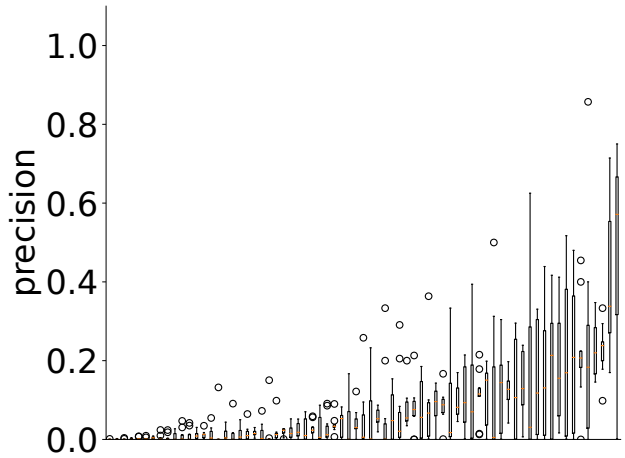
Next, we test the effectiveness when different types of example is used, as described in section 5.1.1. Table 5.5 compares the effectiveness of queries generated using the most relevant topic in the prompt to queries generated using a high-quality example. Using a relevant topic as an example can result in queries with higher precision, but lower recall.

Finally, we study the variability of effectiveness using p4(HQE) – the most effective prompt compared with other prompts, and we re-run the prompt ten times. Variability is shown in Figure 5.1. Recall varied more than precision and F-measures: variance of recall is 12% of its mean value; precision was 7.1%, F-1 6.6% and F-3 7.2%. Our result indicates that the generated queries from the same prompt would mainly differ in the ability to obtain more relevant documents (recall) from the included studies.

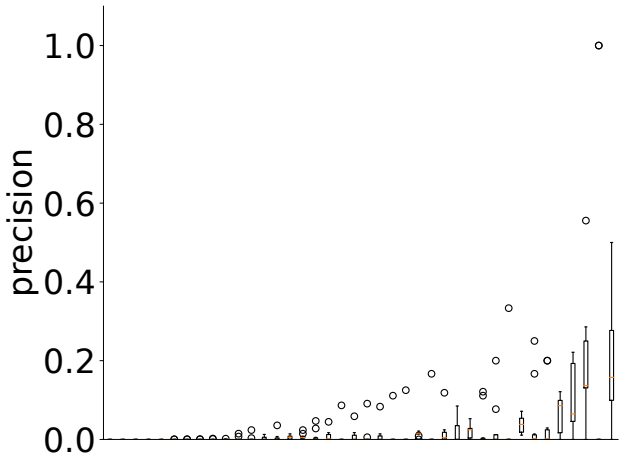
5.3.2 Guided Prompt Query Formulation

Table 5.6 reports the result of guided prompt query formulation. These results suggest that if using a well-chosen seed study, queries generated from guided prompt are far more effective than queries generated from unguided prompts.

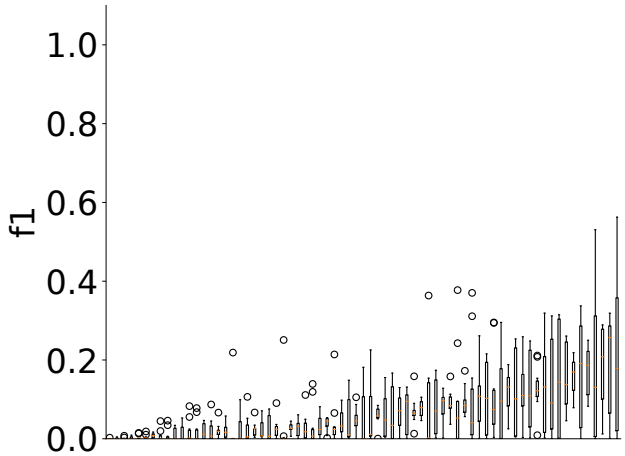
However, like previous experiments in unguided prompt query generation and refinement, the effectiveness varies considerably across runs; and the effectiveness also depends on the seed study being used. Figure 5.2 reports the variability of query effectiveness when different seed studies are used to generate queries. Figure 5.3 reports the variability of effectiveness when the same seed study is used, picked from the best seed study from the first run.



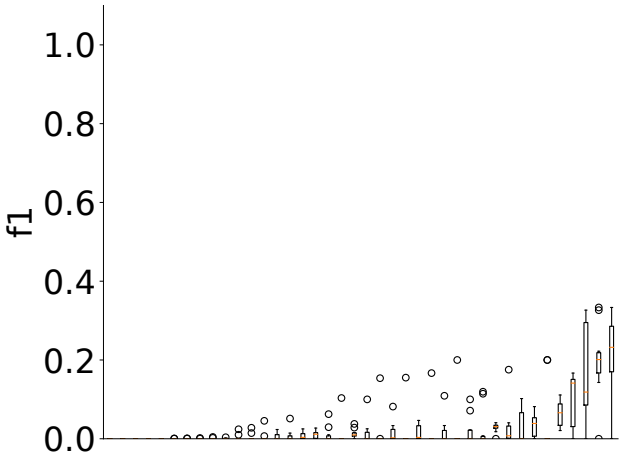
(a) CLEF Precision



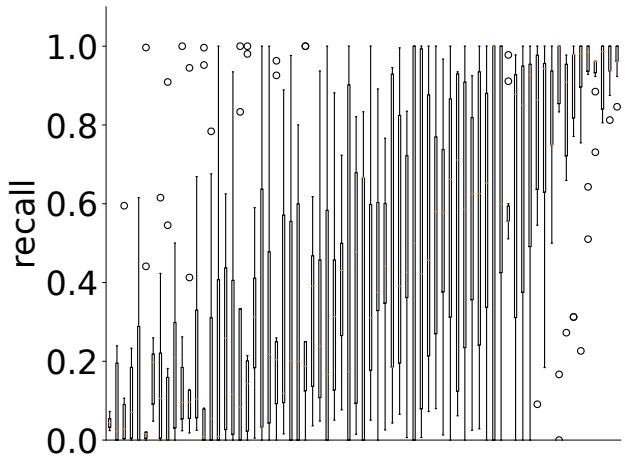
(b) SC Precision



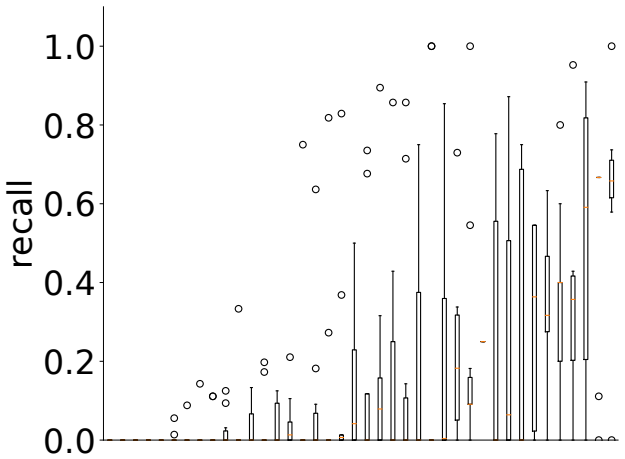
(c) CLEF F1



(d) SC F1



(e) CLEF Recall



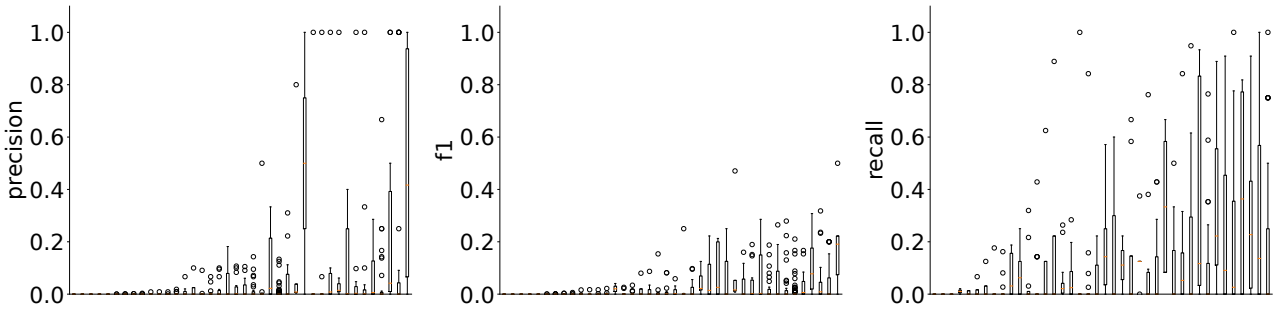
(f) SC Recall

Figure 5.1: Topic-by-topic variability for the effectiveness of 10 iterative runs in unguided prompt query formulation using p4. *CLEF* indicates CLEF TAR collection and *SC* indicates seed collection.

From the variability graph, we see that query generation using guided prompt is not stable across different seed studies. Furthermore, even when the same seed study is used to generate multiple queries, there is a high degree of variability. The range of precision and recall for some topics can span from 0 to 1, especially when the average effectiveness is high.

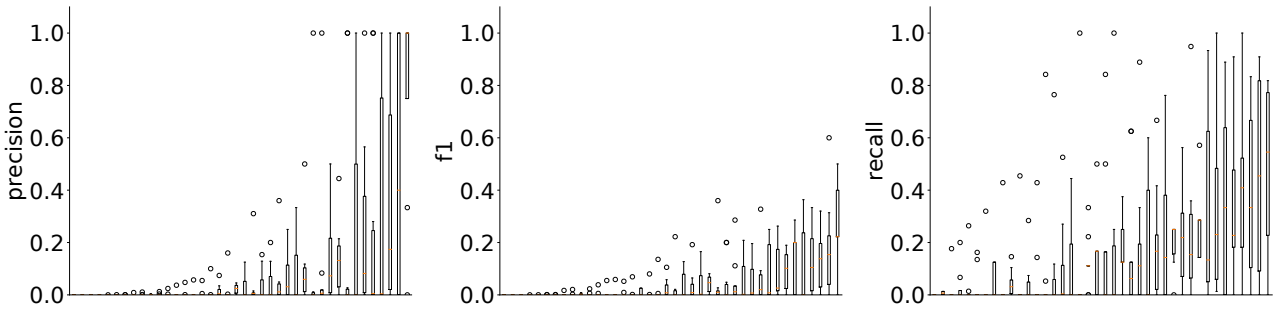
Table 5.6: Guided prompt query formulation on Seed Collection, compared with unguided prompt query generation ‘p4’; Statistical significant differences (p ¡ 0.05) between guided prompt and unguided prompt is indicated by *.

Prompts	Precision	F1	F3	Recall
p4	0.0284	0.0274	0.0374	0.1290
Guided	0.0993(+249.6%)	0.0492(+79.6%)	0.0565(+51.1%)	0.5171(+301.6%)*



(a) Precision (b) F1 (c) Recall

Figure 5.2: Topic-by-topic variability for the effectiveness of using different seed studies for guided prompt query formulation.



(a) Precision (b) F1 (c) Recall

Figure 5.3: Topic-by-topic variability for the effectiveness of 10 iterative runs using the same seed study in guided prompt query formulation.

5.3.3 Boolean Query Refinement

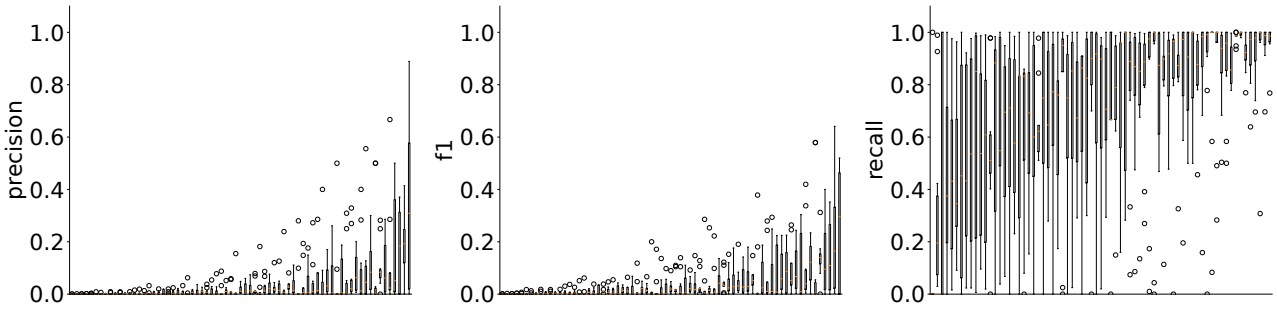
Table 5.7 suggests that ChatGPT is capable of improving the effectiveness of systematic review literature search queries. ChatGPT for query refinement leads to an increase in precision and F-measure, while obtaining a lower recall. Therefore, it is crucial to first create a query with high recall, and then use ChatGPT to refine it.

Using automatically formulated queries from the objective queries obtained the highest effectiveness when ChatGPT was used to refine the queries. Refinement yielded an 11% drop in recall, but considerable gains in precision, F1, and F3.

Variability is studied by running the best method, refined-objective run, ten times. We show the variability of the query refinement in Figure 5.4. There is less variance in query refinement than in query formation (Figure 5.1). This is understandable, given the query structure is already provided by the seed query, whereas query formulation must be done from scratch from the title of the review.

Table 5.7: Result table for query refinement on CLEF TAR collection. For a refinement method, 'p6-Manual', 'p6' indicates the prompt used to generate the refined query; 'Manual' indicate the seed queries used for ChatGPT to refine. For each query refinement method, statistical significant differences ($p < 0.05$) between refined prompt and seed queries are indicated by *. Percentage changes from seed to refined queries are shown in parentheses.

Prompts	Precision	F1	F3	Recall
Manual	0.0207	0.0290	0.0481	0.8317
p6-Manual	0.0795* (+284.1%)	0.0597* (+105.9%)	0.0802* (+66.7%)	0.5060* (-39.2%)
Conceptual	0.0014	0.0027	0.0100	0.6996
p7-conceptual	0.0022 (+57.1%)	0.0039 (+44.4%)	0.0069 (-31.0%)	0.2699* (-61.4%)
Objective	0.0002	0.0005	0.0023	**0.9128**
p7-Objective	0.0460* (+22900.0%)	0.0471* (+9320.0%)	0.0652* (+2736.5%)	0.8115* (-11.1%)
p4	0.0751	0.0642	0.0872	0.5035
p7-p4	**0.1162** (+54.7%)	**0.0772** (+20.2%)	**0.0921** (+5.6%)	0.3179* (-36.9%)



(a) Precision (b) F1 (c) Recall

Figure 5.4: Topic-by-topic variability for the effectiveness of 10 iterative runs in unguided prompt query refinement.

5.3.4 Answer to Research Questions in Initial Evaluation

This section summarises the key findings of our initial evaluation using the ChatGPT interface, based on the research questions introduced.

RQ1: *How does ChatGPT compare to previous automatic methods for formulating and refining Boolean queries in systematic reviews?*

ChatGPT performs competitively with prior automatic methods, particularly improving precision by reducing irrelevant documents. However, these gains often come at the cost of lower recall—an important consideration in systematic reviews.

RQ2: *To what extent do prompt designs influence the effectiveness of Boolean queries generated by ChatGPT for systematic reviews?*

Prompt design significantly impacts retrieval effectiveness of Boolean Query. Prompts that include example queries tend to outperform simple instructions, especially in terms of recall. The type of example matters: related examples improve precision, while unrelated high-quality examples improve recall. In contrast, adding PICO-based instructions often reduces recall and yields mixed results on

precision, suggesting that over-structuring the prompt may be detrimental.

RQ3: *What is the impact of guiding ChatGPT through multi-step prompts that mimic current state-of-the-art automated Boolean query generation methods?*

Guided prompts based on the objective method lead to more effective queries overall, improving both recall and precision. Decomposing the task into manageable steps helps the model generate more structured and accurate Boolean queries. However, this approach also introduces additional complexity and does not resolve core challenges such as the variability in the effectiveness of the generated Boolean queries.

RQ4: *What are the limitations and challenges of using ChatGPT for Boolean query formulation in systematic reviews?*

Three key limitations emerge: (1) over 55% of MeSH terms generated by ChatGPT are invalid, undermining the reliability of the queries—particularly for recall-oriented tasks; (2) considerable variability in outputs across repeated runs, even with identical prompts, which introduces reproducibility concerns and requires costly manual evaluation to identify high-performing queries; and (3) the black-box nature of the ChatGPT interface, which limits transparency and prevents deeper inspection or customisation of the generation process. These challenges motivate our later investigation into more transparent and controllable alternatives, including open-source language models.

5.4 Generalising Boolean Query Generation to Other LLMs

While our initial study using the ChatGPT interface produced promising results, it also revealed key limitations—such as limited controllability, non-deterministic outputs, and lack of transparency. These issues raise concerns about reproducibility and the practical deployment of ChatGPT-generated Boolean queries in systematic review workflows.

To address these concerns and test the generalisability of our findings, we extend our investigation to a broader set of LLMs, including both proprietary API-based models and open-source alternatives. Open-source models—providing access to model weights, decoding parameters, and training details—are especially valuable in research contexts requiring reproducibility, transparency, and fine-grained customisation. They also allow for secure deployment and task-specific fine-tuning, offering a controllable alternative to commercial black-box systems.

This phase of the study examines whether insights from the ChatGPT interface generalise to modern, more configurable LLMs. Unlike the ChatGPT interface (circa late 2022), newer models—whether accessed via APIs or open-source implementations—provide full control over system-level instructions, prompt formatting, and output structure (e.g., JSON). This flexibility enables a deeper investigation into how model architecture, prompt design, and output constraints influence the effectiveness of Boolean query generation.

We address the following research questions:

Table 5.8: Summary of evaluated LLMs and their shorthand labels.

Model Type	Model Name	Shorthand Label
ChatGPT Interface	/	ChatGPT
ChatGPT API-based	`gpt-3.5-turbo-1106`	GPT3.5-1
	`gpt-3.5-turbo-0125`	GPT3.5-0
	`gpt-4-1106-preview`	GPT4
	`gpt-4o`	GPT4o-m
	`O1`	O1
Mistral API-based	`Mistral-7B-v0.2`	Mistral-S
	`Mixtral-8x7B-v0.1`	Mistral-L
Open-Source (Local)	`Mistral-7B-v0.2`	Mistral
	`Zephyr-7B-beta`	Zephyr
	`Llama3.1-8B-Instruct`	Llama3.1

RQ1: Do findings from the ChatGPT interface generalise to other LLMs?

RQ2: How do different LLMs compare in generating valid and effective Boolean queries using the same prompts, and how sensitive are they to prompt variations?

RQ3: What is the impact of separating system instructions from user prompts and enforcing structured output formats (e.g., JSON) on query effectiveness?

For clarity, we refer to the evaluated LLMs using shorthand labels as summarised in Table 5.8. The evaluated models span three categories: ChatGPT API-based models, Mistral API-based models, and open-source models run locally. Each model is assigned a consistent shorthand label, which we use throughout the remainder of this section for concise reference.

5.4.1 Unguided Prompt Query Formulation

Unguided query formulation results on the CLEF TAR and Seed collections are presented in Tables 5.9 and 5.10, respectively. These tables include a comparative analysis between multiple extended LLMs and the ChatGPT interface. The results indicate that most LLM variants achieve higher effectiveness in terms of both recall and precision compared to ChatGPT interfaces (ChatGPT).

Across the tested LLMs and prompts, the most effective model or prompt depends significantly on the evaluation metric and dataset. Zero-shot formulation prompts (p1–p3) generally achieve higher precision but lower recall compared to one-shot prompts (p4–p5), resulting in mixed F-measure outcomes. When compared with the ChatGPT interface, one-shot prompts yield superior effectiveness on seven models for F1 and six models for F3 metrics on the CLEF TAR collection. For the Seed collection, one-shot prompts outperform alternatives on two models for both F1 and F3.

Evaluating all LLMs under identical prompts, model o1 (the most recent reasoning-focused model) consistently achieves higher recall than other variants, except for the prompt p4 scenario, where o1 performs marginally lower than Llama3.1 on the CLEF TAR collection and GPT3.5-1 on the Seed

Table 5.9: Results on 71 CLEF topics for all query-formulation prompts. For each evaluation metric, **bolded** values indicate the highest value among all prompts for a given large language model, and coloured values indicate the highest effectiveness among all model variations within each prompt. A paired t-test with Bonferroni correction (p ¡ .05) is performed: *a* indicates statistical significance relative to the **bolded** value, *b* relative to the gray-celled value, and *c* relative to the manual baseline.

	Prompt	GPT3.5-1	GPT3.5-0	GPT4	GPT4o-m	Mistral-S	Mistral-L	Mistral	Zephyr	Llama3.1	o1	ChatGPT
						Manual = .0209						
Precision	p1	**.1470**^{c}	.1451^{c}	.0854^{a}	.1393^{c}	.0139^{a,b}	.0384^{a,b}	.0415^{b}	.0792	.1069^{c}	**.1361**^{c}	.0543^{b}
	p2	.1430^{c}	**.1501**^{c}	**.1707**^{c}	.1294^{c}	**.1008**	**.1329**^{c}	**.1233**^{c}	.1088^{c}	.1097^{c}	.1324^{c}	**.1166**^{c}
	p3	.1249^{c}	.1051^{c}	.1348^{c}	**.1409**^{c}	.0617^{b}	.1019^{c}	.0732^{c}	.1057	**.1153**^{c}	.1078^{c}	.0844^{c}
	p4	.1064^{c}	.1006^{c}	.1333^{c}	.0662^{b}	.0844	.0965^{c}	.0624^{b}	.1135^{c}	.0685	.1060^{c}	.0752
	p5	.0934^{c}	.1190^{c}	.1372^{c}	.0971^{c}	.0803^{c}	.1109^{c}	.0904^{c}	**.1319**^{c}	.0801	.1189^{c}	.0958^{c}
						Manual = .8436						
Recall	p1	.3419^{a,c}	.2529^{a,b,c}	.1183^{a,b,c}	.3235^{a,c}	.0406^{a,b,c}	.1533^{a,b,c}	.0754^{a,b,c}	.0410^{a,b,c}	.1899^{a,b,c}	.4341^{a,c}	.1293^{a,b,c}
	p2	.3643^{c}	.2808^{a,b,c}	.2595^{b,c}	.2878^{a,b,c}	.1829^{a,b,c}	.2484^{a,b,c}	.0895^{a,b,c}	.1554^{b,c}	.3753^{a,c}	.5252^{a,c}	.1310^{a,b,c}
	p3	.4357^{b,c}	.4120^{b,c}	.3112^{b,c}	.3343^{a,b,c}	.1137^{a,b,c}	**.4304**^{b,c}	.2049^{a,b,c}	.1691^{b,c}	.3741^{a,b,c}	**.6545**^{c}	.1175^{a,b,c}
	p4	**.5108**^{c}	**.4862**^{c}	.3149^{b,c}	**.5509**^{c}	**.4831**^{c}	.3615^{b,c}	.4345^{c}	.2655^{b,c}	**.5811**^{c}	.5438^{a,c}	**.5035**^{c}
	p5	.4138^{c}	.4654^{c}	**.3200**^{b,c}	.4576^{c}	.3174^{b,c}	.3191^{b,c}	**.4833**^{c}	**.2904**^{b,c}	.4740^{c}	.5377^{a,c}	.3335^{b,c}
						Manual = .0298						
F1	p1	.1083^{c}	.0970^{c}	.0551^{a,b}	.1020^{c}	.0118^{a,b}	.0381^{a,b}	.0226^{a,b}	.0252^{a,b}	.0665^{b}	.1335^{c}	.0500^{b}
	p2	**.1083**^{c}	.0981^{c}	.1016^{c}	.1073^{c}	.0536^{b}	.0839^{b,c}	.0414^{a,b}	.0593^{b}	.0875^{c}	.1427^{c}	.0654^{b,c}
	p3	.1053^{c}	.0922^{c}	.1043^{c}	**.1108**^{c}	.0363^{b}	.0845^{b,c}	.0446^{a,b}	.0401^{a,b}	**.1045**^{c}	.1381^{c}	.0443^{b}
	p4	.0996^{c}	.0818^{b,c}	**.1135**^{c}	.0525^{a,b}	.0617^{b}	.0731	.0586^{b}	.0734^{b,c}	.0673^{b}	.1316^{c}	.0642^{b}
	p5	.0798^{b,c}	**.1017**^{c}	.1092^{c}	.0809^{b,c}	**.0666**^{b}	**.0937**^{b,c}	**.0860**^{b,c}	**.0928**^{c}	.0617^{b}	**.1428**^{c}	**.0717**^{b,c}
						Manual = .0497						
F3	p1	.1311^{c}	.1114^{c}	.0608^{a,b}	.1269^{c}	.0128^{a,b,c}	.0495^{a,b}	.0275^{a,b}	.0265^{a,b}	.0746^{b}	.1686^{c}	.0590^{b}
	p2	.1338^{c}	.1147^{b,c}	.1206^{c}	.1252^{c}	.0616^{b}	.0966^{b}	.0428^{a,b}	.0674^{b}	.1110^{c}	.1859^{c}	.0696^{b}
	p3	**.1363**^{c}	.1181^{b,c}	.1275^{b,c}	**.1341**^{b,c}	.0437^{b}	.1025^{b,c}	.0539^{a,b}	.0534^{a,b}	**.1301**^{b,c}	**.1966**^{c}	.0497^{b}
	p4	.1335^{c}	.1049^{b,c}	**.1392**^{c}	.0631^{a,b}	**.0839**^{b}	.0863^{b}	.0822^{b}	.0877^{b}	.0913^{b}	.1777^{c}	**.0847**^{b}
	p5	.1064^{b,c}	**.1250**^{b,c}	.1364^{b,c}	.1085^{b,c}	.0800^{b}	**.1141**^{b,c}	**.1139**^{b,c}	**.1077**^{b,c}	.0829^{b}	.1959^{c}	.0844^{b}

collection; however, these differences are statistically insignificant. Additionally, model o1 consistently outperforms other models across all prompt variations in terms of F-based metrics on the CLEF TAR collection. Among other models, Llama3.1 and GPT3.5-1 exhibit relatively higher recall in multiple prompt settings, jointly holding the highest recall in five instances (combining CLEF TAR and Seed collections across five prompt variations).

Compared with manually formulated Boolean queries across both collections, the findings reaffirm earlier results from the ChatGPT interface study: automated methods achieve higher precision but significantly lower recall, with the sole exception being model o1 using prompt p3 on the Seed collection. Notably, the gap in recall has substantially narrowed compared to previous results reported by Wang et al. [264], particularly for model o1. The average recall is now approximately 15% lower than manual queries across both collections, marking a considerable improvement over the previously reported gaps (35% for CLEF TAR and 60% for the Seed collection).

5.4.2 Guided Prompt Query Formulation

Table 5.10 additionally presents results from guided query formulation experiments. These findings reinforce the conclusions drawn from the ChatGPT interface: guided prompts, using carefully chosen seed studies, consistently improve query effectiveness compared to unguided prompts. Specifically, selecting the optimal seed study (Guided-best) significantly enhances recall relative to single-step formulation prompts for all evaluated models.

Table 5.10: Results on 40 Seed collection topics for all query-formulation prompts. For each evaluation metric, **bolded** values indicate the highest value among all prompts for a given large language model, and coloured values indicate the highest effectiveness among all model variations within each prompt. A paired t-test with Bonferroni correction (p ¡ .05) is performed: *a* indicates statistical significance relative to the **bolded** value, *b* relative to the coloured value, and *c* relative to the manual baseline. o1 model for Guided prompt were not conducted due to high expenses.

	Prompt	GPT3.5-1	GPT3.5-0	GPT4	GPT4o-m	Mistral-S	Mistral-L	Mistral	Zephyr	Llama3.1	o1	ChatGPT
						Manual = .0341						
Precision	p1	.0475	.0309	.0262^{b}	**.0491**	.0090	.0079	.0155	.0431	**.0605**	**.0827**	.0514
	p2	.0573	.0251	**.0344**	.0151	**.0962**	**.0470**	.0355	.0942	.0599	.0469	.0983
	p3	.0687	**.0757**	.0313	.0370	.0680	.0451	**.0706**	.0255	.0349	.0523	.0730
	p4	.0768	.0602	.0126	.0337	.0142	.0378	.0178	.0282	.0132	.0384	.0284
	p5	**.0932**	.0229	.0011^{c}	.0145	.0611	.0466	.0167	**.1104**	.0202	.0618	.0189
	Guided-best	.0078	.0061	.0209	.0210	.0497	.0292	.0381	.0755	.0158	–	**.0993**
	Guided-combined	.0002^{c}	.0010^{c}	.0022^{c}	.0071	.0038^{c}	.0165	.0015^{c}	.0087	.0022^{c}	–	–
						Manual = .7241						
Recall	p1	.3045^{a,c}	.1318^{a,b,c}	.0600^{a,b,c}	.1930^{a,c}	.0898^{a,b,c}	.1142^{a,b,c}	.0244^{a,b,c}	.0321^{a,b,c}	.2446^{a,c}	.3767^{a,c}	.0542^{a,b,c}
	p2	.1495^{a,c}	.0858^{a,b,c}	.0580^{a,b,c}	.1625^{a,c}	.0644^{a,b,c}	.1614^{a,c}	.0858^{a,b,c}	.0876^{a,b,c}	.2177^{a,c}	.3245^{a,c}	.0394^{a,b,c}
	p3	.2437^{a,b,c}	.1986^{a,b,c}	.0321^{a,b,c}	.0982^{a,b,c}	.1154^{a,b,c}	.2774^{a,b,c}	.0991^{a,b,c}	.0321^{a,b,c}	.2329^{a,b,c}	**.5786**	.0519^{a,b,c}
	p4	.2860^{a,c}	.2078^{a,c}	.0056^{a,b,c}	.2077^{a,c}	.1580^{a,c}	.1180^{a,c}	.1536^{a,c}	.1682^{a,c}	.2251^{a,c}	.2592^{a,c}	.1290^{a,c}
	p5	.2848^{a,c}	.3285^{a,c}	.0231^{a,b,c}	.2746^{a,c}	.2385^{a,c}	.1251^{a,b,c}	.2986^{a,c}	.1928^{a,c}	.3113^{a,c}	.4182^{a,c}	.0785^{a,b,c}
	Guided-best	.6441^{a}	.6328^{a}	.6188^{a}	.3820^{a,b,c}	.6091^{a}	.5042^{a,b}	.5241^{a}	.4434^{a,b,c}	.6070^{a}	–	**.5171**
	Guided-combined	**.7361**	**.7315**	**.6978**	**.5045**^{b}	**.6649**	**.5908**	**.5772**	**.5134**^{b}	**.6801**	–	–
						Manual = .0605						
F1	p1	.0410	.0221^{b}	.0071^{b,c}	**.0441**	.0140^{b,c}	.0118^{b,c}	.0131^{b}	.0167	**.0627**	**.0777**	.0281^{b}
	p2	.0322	.0227	.0111^{c}	.0205	**.0424**	.0323	.0220	.0246	.0436	.0533	.0310
	p3	.0597	**.0525**	.0128	.0198	.0385	**.0357**	**.0439**	.0224	.0421	.0537	.0329
	p4	.0471	.0383	.0047^{c}	.0169	.0123^{c}	.0143	.0190	.0252	.0159^{c}	.0262	.0274
	p5	**.0612**	.0355	.0018^{b,c}	.0184	.0347	.0299	.0253^{c}	**.0698**	.0267	.0561	.0193
	Guided-best	.0139	.0110^{c}	**.0335**	.0304	.0334	.0232	.0300	.0441	.0216	–	**.0492**
	Guided-combined	.0003^{a,c}	.0020^{a,c}	.0044^{c}	.0100	.0046^{c}	.0040^{c}	.0008^{c}	.0083^{a,c}	.0035^{c}	–	–
						Manual = .1024						
F3	p1	.0571	.0266^{b,c}	.0069^{b,c}	**.0534**	.0198^{b,c}	.0175^{b,c}	.0143^{b,c}	.0175^{b,c}	**.0769**	**.1019**	.0306^{b}
	p2	.0388	.0273^{c}	.0108^{c}	.0304^{c}	.0400	.0376	.0211^{c}	.0241^{c}	.0544	.0755	.0278^{c}
	p3	**.0820**	**.0652**	.0141^{b,c}	.0242^{c}	.0416	**.0512**	**.0418**	.0240	.0555	.0852	.0329
	p4	.0566	.0469	.0038^{b,c}	.0228^{c}	.0151^{c}	.0133^{c}	.0243^{c}	.0338^{c}	.0225^{c}	.0366^{c}	.0374
	p5	.0791	.0526	.0027^{b,c}	.0302^{c}	**.0440**	.0375	.0373^{c}	**.0783**	.0352	.0780	.0271^{c}
	Guided-best	.0232^{c}	.0187^{c}	**.0509**	.0420	.0424	.0363	.0360	.0581	.0314^{c}	–	**.0565**
	Guided-combined	.0006^{a,c}	.0036^{a,c}	.0084^{c}	.0132^{c}	.0083^{c}	.0064^{c}	.0011^{c}	.0134^{a,c}	.0057^{c}	–	–

Out of the nine evaluated LLMs (excluding the cost-prohibitive o1 model), six achieve higher recall than the ChatGPT interface. Compared to original manual queries, most models show no statistically significant difference in effectiveness, except for Mistral-S in terms of recall, and GPT3.5-0 for F1. Additionally, GPT3.5-0, GPT3.5-1, and Llama3.1 demonstrate statistically significant differences for F3.

The combined query approach (Guided-combined), which merges queries generated from multiple seed studies using logical OR, achieves recall comparable to the Manual Boolean queries across all tested models, with no statistically significant differences observed. However, this method reduces precision, resulting in statistically significant differences compared to the Original manual queries across all models, excluding Zepher, Mistral-L, and GPT4o-mini. A notable advantage of this combined approach is its pre-hoc nature, removing the necessity to identify a single best seed study as required by the Guided-best method.

Table 5.11: Results on 71 CLEF TAR topics for all query-refinement prompts. For each evaluation metric, **bolded** values indicate the highest value among all prompts for a given large language model, and coloured values indicate the highest effectiveness among all model variations within each prompt. A paired t-test with Bonferroni correction (p ¡ .05) is performed: *a* indicates statistical significance relative to the **bolded** value (highest prompt for each model), *b* relative to the coloured value, and *c* relative to the refinement base, i.e., p6-Manual and p6, p7-conceptual and p7, and p7-objective and objective.

	Prompt	GPT3.5-1	GPT3.5-0	GPT4	GPT4o-m	Mistral-S	Mistral-L	Mistral	Zephyr	Llama3.1	ChatGPT
Precision	Manual=.0209, conceptual=.0015, objective=.0002										
	p6-Manual	.0393^{a,b,c}	.0492^{b,c}	.1211^{c}	**.0812**^{b,c}	**.0873**^{c}	.0839^{a,c}	**.0865**^{c}	**.1234**^{c}	**.1186**^{c}	**.0795**^{c}
	p7-conceptual	**.0982**^{b,c}	**.0742**^{b,c}	**.1581**^{c}	.0547^{b,c}	.0575^{b,c}	**.1411**^{c}	.0596^{b,c}	.0583^{a,b,c}	.0566^{a,b,c}	.0022^{a,b}
	p7-objective	.0460^{a,b,c}	.0242^{a,b,c}	.1100^{a,c}	.0041^{a,b,c}	.0317^{a,b,c}	.0906^{c}	.0237^{a,b,c}	.0416^{a,b,c}	.0135^{a,b,c}	.0460^{b,c}
Recall	Manual=.8436, conceptual=.6996, objective=.9128										
	p6-Manual	**.6653**^{c}	.7057^{c}	.3934^{b,c}	.4828^{a,b,c}	.3944^{a,b,c}	.4346^{b,c}	.4340^{a,b,c}	.2450^{a,b,c}	.3833^{a,b,c}	.5060^{a,b,c}
	p7-conceptual	.4630^{a,b,c}	.4744^{a,c}	.2834^{a,b,c}	.5092^{a,c}	.4735^{a,c}	.3120^{a,b,c}	.5548^{c}	.3004^{a,b,c}	.5827^{a,c}	.2699^{a,b,c}
	p7-objective	.5763^{b,c}	**.7175**^{c}	**.4339**^{b,c}	**.8078**^{c}	**.7386**^{c}	**.5011**^{b,c}	**.6188**^{b,c}	**.5517**^{b,c}	**.7951**^{c}	**.8115**^{c}
F1	Manual=.0298, conceptual=.0027, objective=.0005										
	p6-Manual	.0536^{a,b,c}	**.0648**^{b,c}	.1149^{c}	**.0882**^{c}	**.0715**^{b,c}	.0786^{b,c}	**.0659**^{b,c}	**.0810**^{b,c}	**.0840**^{b,c}	**.0597**^{b,c}
	p7-conceptual	**.0882**^{b,c}	.0579^{b,c}	**.1300**^{c}	.0549^{a,b,c}	.0572^{b,c}	**.0985**^{b,c}	.0467^{b,c}	.0524^{b,c}	.0670^{b,c}	.0039^{a,b}
	p7-objective	.0386^{a,b,c}	.0282^{a,b,c}	.1180^{c}	.0079^{a,b,c}	.0405^{a,b,c}	.0749^{b,c}	.0335^{a,b,c}	.0496^{a,b,c}	.0222^{a,b,c}	.0471^{b,c}
F3	Manual=.0497, conceptual=.0101, objective=.0023										
	p6-Manual	.0794^{b,c}	**.0920**^{b,c}	.1440^{c}	**.1196**^{c}	**.0924**^{b,c}	.1053^{b,c}	**.0834**^{b,c}	**.0957**^{b,c}	**.1103**^{b,c}	**.0802**^{b,c}
	p7-conceptual	**.1149**^{b,c}	.0799^{b,c}	**.1539**^{c}	.0776^{a,b,c}	.0780^{b,c}	**.1209**^{c}	.0644^{b,c}	.0697^{b,c}	.0881^{b,c}	.0069^{a,b}
	p7-objective	.0475^{a,b,c}	.0387^{a,b,c}	.1476^{c}	.0148^{a,b,c}	.0582^{b,c}	.1005^{b,c}	.0494^{a,b,c}	.0652^{b,c}	.0348^{a,b,c}	.0652^{b,c}

5.4.3 Unguided Prompt Query Refinement

Next, we investigate LLMs variances for their ability to refine Boolean queries. Table 5.11 shows the results of unguided Boolean query refinement on the CLEF TAR collection. The results demostrated that the findings on ChatGPT interface generalise to all other LLMs, that is: Boolean query refinement by LLMs increases precision with a drop in recall. Specifically, compared to ChatGPT interface, tested LLMs generally obtain a higher precision with p6-manual and p7-conceptual and a higher recall with p7-conceptual. For p7-objective, the recall value is lower than the ChatGPT interface, while precision is similar.

5.4.4 Impact of Prompt and Output Type

We next investigate prompt and output types, note that is comparison is not suitable for ChatGPT interface as there's no easy way to control the system prompt and output format. Note that by default (in all previous shown result tables), we used plain-text prompts using user prompt only, as system prompts and JSON output were not available for some models.

Table 5.12: Results on CLEF TAR for p4 prompt variations.

Prompt	GPT3.5-1				Mistral-S			
	Precision	F1	F3	Recall	Precision	F1	F3	Recall
Plain-text	**0.1064**	**0.0996**	**0.1335**	0.5108	**0.0844**	0.0617	0.0839	0.4831
JSON-Output	0.0859	0.0943	0.1250	**0.5562**	0.0679	0.0424	0.0530	0.3914
System-JSON	0.0898	0.0901	0.1191	0.5413	0.0839	**0.0656**	**0.0854**	**0.5828**

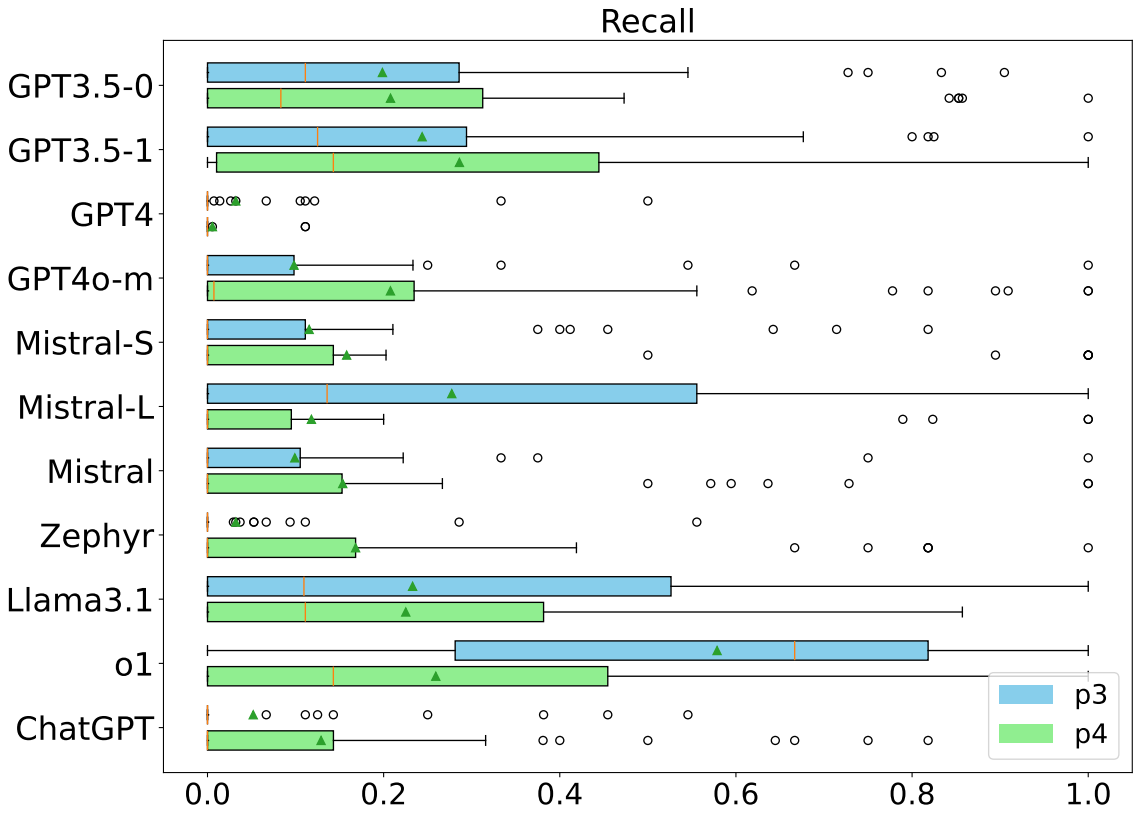


Figure 5.5: Recall variability of formulated Boolean queries using prompts p3 and p4 for the Seed collection.

We specifically investigate the impact of separating prompts and restricting output formats, focusing on GPT3.5-1 and Mistral-S, using p4 as the reference prompt. Table 5.12 compares plain-text, JSON output, and System-JSON settings. The impact of JSON output varies across models. Enforcing JSON output resulted in increased recall but decreased precision and F1-score for GPT3.5-1, while Mistral-S showed a drop in all metrics. For System-JSON, we observed a varied trend: it increased both recall and precision for Mistral-S, but for GPT3.5-1, it led to a decrease in recall and increase in precision.

Overall, none of the above results showed statistical significance. Therefore, we conclude that the use of system prompts and enforced JSON output, despite their varied effects, likely has a minimal impact on the quality of Boolean query formulation.

5.4.5 Variability and Incorrect Formulation

To investigate the variability in Boolean query formulation effectiveness across different models, we select three representative prompts: `p3`, a more effective detailed prompt (compared to `p2`); `p4`, a more effective example-based prompt (compared to `p5`); and the guided prompt, which incorporates seed studies. Notably, `p3` serves as a zero-shot prompt, `p4` as a one-shot prompt, and the guided prompt as a multi-step, seed-driven formulation.

Figure 5.5 illustrates the variability in query effectiveness for each model on the Seed Collection. The figure reveals substantial variance in performance across models, highlighting that certain models

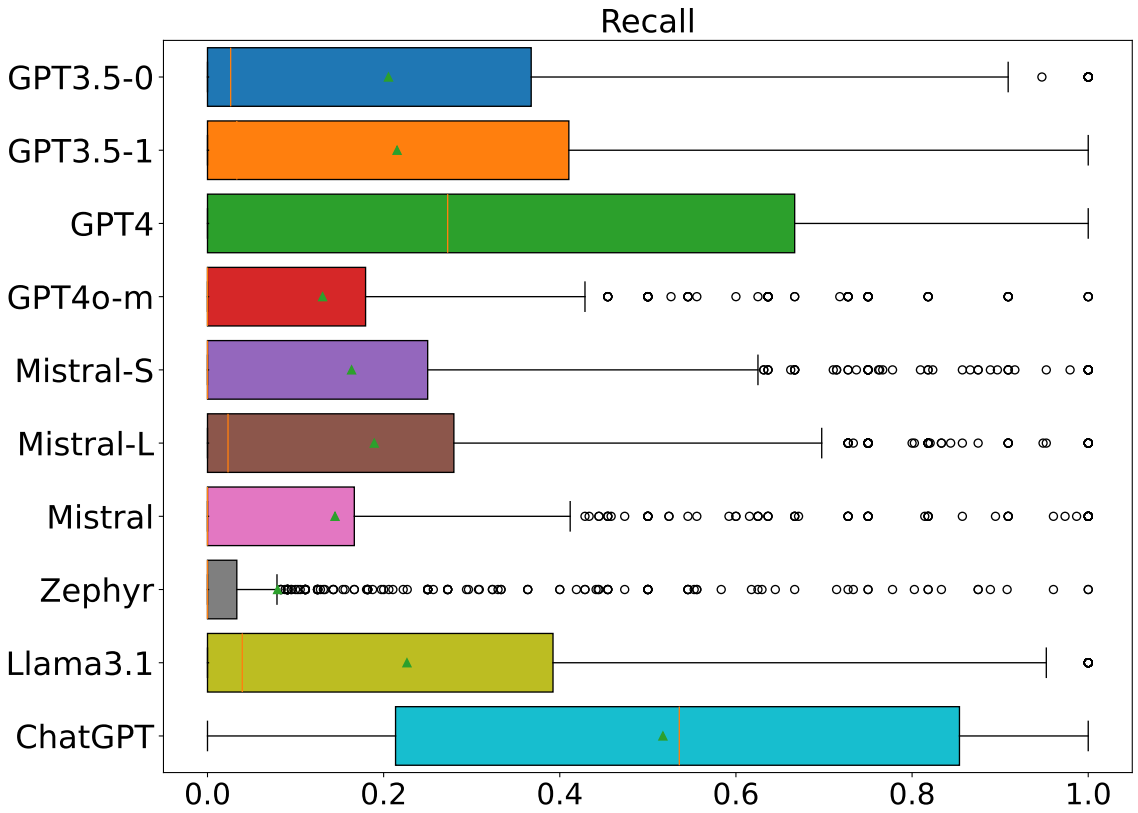


Figure 5.6: Recall variability of all formulated Boolean queries using guided prompt for the Seed collection. Generation from all the seed studies is aggregated together.

are inherently less suited for Boolean query formulation. This suggests that model selection plays a critical role in achieving reliable and effective query generation.

For example, GPT-4 consistently exhibits very low recall across all queries it generates. This suggests that model selection is crucial—certain models, such as GPT-4, should be avoided when aiming for effective Boolean query formulation. In contrast, O1 models demonstrate higher effectiveness compared to all other models.

Regarding prompt selection, while most models benefit from an example to improve Boolean query formulation, certain models, such as Mistral-L and O1, actually perform worse with a one-shot prompt. Overall, the best prompt to use is model dependant, one prompt working on one model a does not necessarily mean it would also achieve high effectiveness on other models.

Figure 5.6 presents the variability in guided Boolean query formulation. These results indicate that, although most queries achieved higher recall and were optimised for recall, their overall effectiveness was still lower compared to the variance observed across different models, with respect to Table 5.10. This highlights the importance of seed selection for guided queries, suggesting that a model capable of generating a single high-quality Boolean query based on a seed study is more effective than one producing multiple relatively average-quality queries across all seed studies.

Figure 5.7 presents the average number of attempts required for each model to generate the first valid Boolean query. For query formulation, the results indicate that Mistral-S, GPT4o-m, and Llama3.1 required a higher number of retries compared to other models. In contrast, the remaining

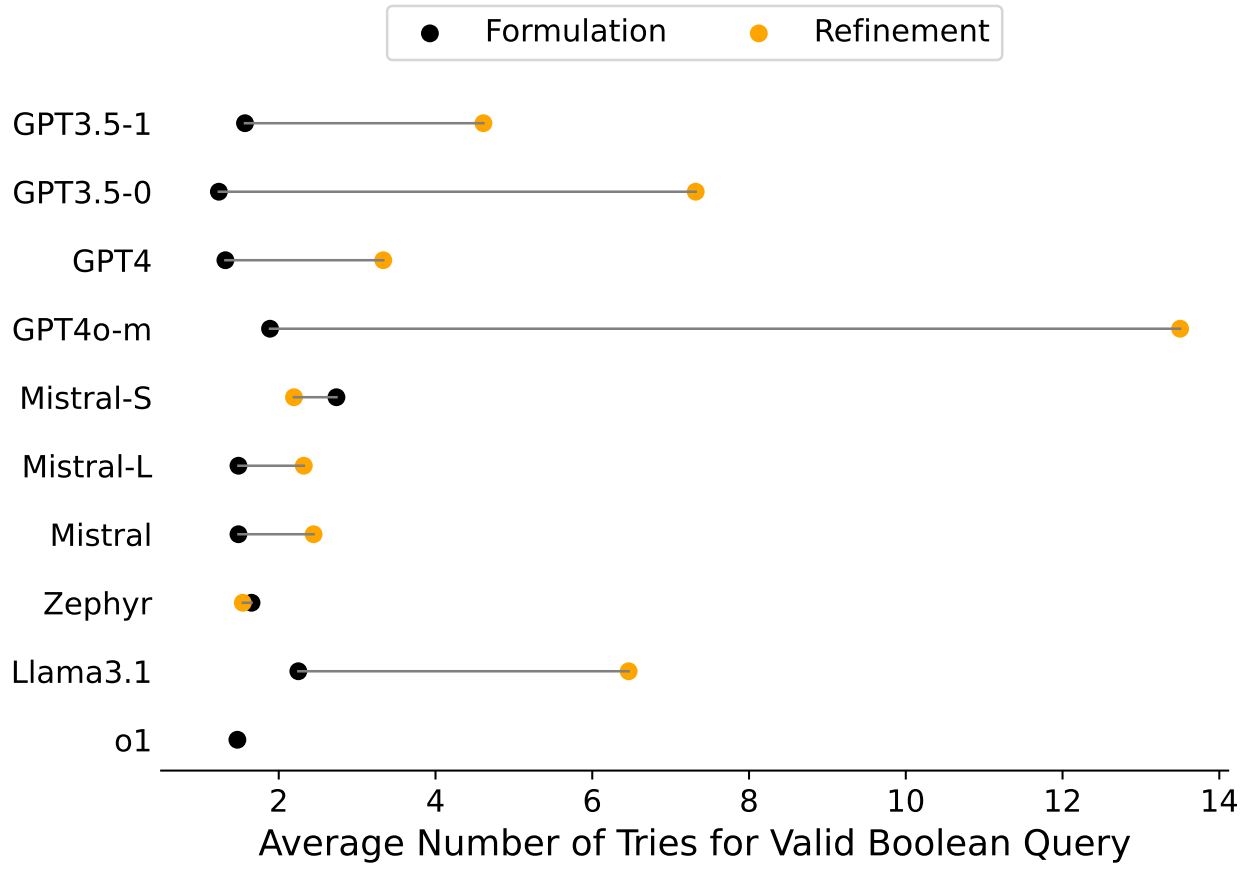


Figure 5.7: Average number of tries per model to generate the first valid Boolean query. o1 not used for query refinement due to high cost.

models consistently produced valid queries with considerably fewer attempts.

Boolean query refinement required more attempts to produce a valid query, likely due to the added complexity of both modifying the existing query and ensuring its correctness. This gap is particularly large for smaller GPT-based models, such as GPT-3.5 and GPT-4o-m, which struggle more with refinement compared to formulation. However, for Mistral-based models and most open-source models (except Llama 3.1), the number of retries for refinement does not appear to be substantially higher than for formulation. Notably, models such as Mistral-S and Zephyr required fewer retries on average for refinement than for initial query formulation.

5.4.6 Answer to Research Questions in Generalisation Study

We summarise our findings across multiple LLMs in relation to the stated research questions:

RQ1: *Do findings from the ChatGPT interface generalise to other LLMs?*

Yes. The trends observed with ChatGPT interface—higher precision but lower recall compared to manual queries—generalise across all tested LLMs. Importantly, newer models such as `o1` narrow the recall gap substantially (from 35–60% down to 15%). Thus, while the general trade-off persists, recent models are better at balancing precision and recall.

RQ2: *How do different LLMs compare in their ability to generate valid and effective Boolean queries using identical prompts, and how sensitive are they to prompt variation?*

Performance varies widely. `o1` consistently achieved the highest recall and F-measure, followed by Llama3.1 and GPT-3.5 variants. GPT-4, in contrast, performed poorly across prompts. Prompt type also mattered: one-shot prompting generally improved recall, though some models (e.g., Mistral-L, `o1`) performed worse under one-shot settings—highlighting model-specific sensitivities to prompt design.

RQ3: *What is the impact of system prompt separation and enforced output formats (e.g., JSON) on query effectiveness?*

Minimal. Experiments with GPT3.5-1 and Mistral-S show no statistically significant differences. Enforcing structured outputs (e.g., JSON) slightly improved recall for GPT3.5-1 but reduced F1-score; Mistral-S saw a general decline. Separating system instructions had mixed effects. These findings suggest that formatting and prompt separation offer little practical benefit for Boolean query tasks.

Additional Observations We observed high variability in output quality across LLMs and generations. Some models frequently produced invalid or suboptimal queries, especially in refinement settings. Small models like GPT-3.5 and GPT-4o-mini often required multiple attempts to yield a usable query. Errors like invalid MeSH terms and syntax issues remained a common source of failure, reinforcing the need for automated validation and possible iterative refinement.

5.5 Case Study

To complement our quantitative evaluation, we conduct a case study to qualitatively examine the characteristics of Boolean queries generated by different LLMs and prompt strategies. This allows us to better understand the practical differences in query structure, term selection, and overall effectiveness. We focus on topic 22 from the Seed collection—chosen due to its high variance in performance across different model and prompt combinations. Specifically, we compare: (1) Boolean queries generated by different LLMs using the same prompt (p4; see Table 5.13), and (2) Boolean queries generated by the same LLM (o1) using different prompts (Table 5.14).

Table 5.13: Comparison of different models generating Boolean queries for topic 22 from the Seed collection, using *p4*.

Model	Boolean Query	P	F1	F3	R
Manual	(”rabies vaccines”[Mesh] OR ((vaccination[Mesh]) AND (rabies[Mesh] OR ”rabies virus”[Mesh])) OR ((Rabies[tiab]) AND (vaccine[tiab] OR Vaccines[tiab] OR Vaccination[tiab] OR Vaccinations[tiab] OR Immunise[tiab] OR Immunize[tiab] OR Immunisation[tiab] OR Immunization[tiab]))) AND (”immunization, secondary”[Mesh] OR ”antibodies, viral”[Mesh] OR ”time factors”[Mesh] OR secondary[tiab] OR booster[tiab] OR boost[tiab] OR boosters[tiab] OR boostability[tiab] OR titer[tiab] OR gmt[tiab] OR ((sero[tiab]) AND (conversion[tiab] OR protection[tiab])) OR seroconversion[tiab] OR seroprotection[tiab]) AND (”administration and dosage”[sh] OR Administer[tiab] OR Administered[tiab] OR Administration[tiab] OR Intramuscular[tiab] OR Intradermal[tiab] OR “pre-exposure”[tiab] OR “pre exposure”[tiab]) NOT (animals[Mesh] NOT humans[Mesh])	.113	.201	.332	.947
GPT3.5-1	(”rabies vaccines”[MeSH Terms] OR ”rabies vaccines”[All Fields] OR ”rabies vaccination”[All Fields] OR ”rabies vaccine”[All Fields] OR ”rabies immunization”[All Fields] OR ”immunization”[MeSH Terms] OR ”immunization”[All Fields]) AND (”immunogenicity”[MeSH Terms] OR ”immunogenicity”[All Fields]) AND (”dose-response relationship, immunologic”[MeSH Terms] OR ”dose-response relationship, immunologic”[All Fields] OR ”dose-response”[All Fields] OR ”dose response”[All Fields] OR ”dose-effect”[All Fields] OR ”dose effect”[All Fields])	.003	.006	.010	.026
GPT3.5-0	(”rabies vaccines”[MeSH Terms] OR ”rabies vaccines”[All Fields] OR ”rabies vaccine”[All Fields] OR rabies[All Fields]) AND (”immunogenicity”[MeSH Terms] OR ”immunogenicity”[All Fields] OR ”immunogenicity”[MeSH Terms] OR ”immunogenicity”[All Fields] OR ”immune response”[MeSH Terms] OR ”immune response”[All Fields]) AND (”dose-response relationship, drug”[MeSH Terms] OR ”dose-response relationship, drug”[All Fields] OR ”dose-response”[All Fields] OR ”dose response”[All Fields] OR ”dose dependent”[All Fields])	.065	.037	.031	.026
GPT4	(rabies[mesh] OR rabies[All Fields]) AND (vaccin*[mesh] OR vaccin*[All Fields] OR immuniz*[mesh] OR immuniz*[All Fields] OR immunisation[All Fields] OR immunization[All Fields]) AND (immunogenic*[mesh] OR immunogenic*[All Fields]) AND (”pre-exposure prophylaxis”[mesh] OR pre-exposure[All Fields] OR ”post-exposure prophylaxis”[mesh] OR post-exposure[All Fields]) AND (systematic[sb] AND (Review[ptyp] OR meta-analysis[ptyp] OR meta-analysis as topic[mesh] OR meta-analysis[All Fields]))	.000	.000	.000	.000
GPT4o-m	immunogenicity[All Fields] AND (rabies[mesh:noexp] OR rabies[All Fields] OR rabies vaccine[All Fields] OR rabies vaccinations[All Fields] OR rabies immunization[All Fields] OR rabies prophylaxis[All Fields]) AND (pre-exposure[All Fields] OR post-exposure[All Fields] OR dose-response[All Fields] OR (vaccination[All Fields] AND response[All Fields]))	.075	.134	.221	.618
Mistral-S	(Rabies[Mesh] OR Rabies Vaccines[Mesh] OR Rabies Immunization[Mesh] OR ”Rabies vaccine”[All Fields] OR ”pre-exposure prophylaxis”[All Fields] OR ”post-exposure prophylaxis”[All Fields] OR ”Rabies PEP”[All Fields] OR ”Rabies POEP”[All Fields] OR rabies prophylaxis[All Fields] OR immunogenicity[All Fields] OR antibody response[All Fields] OR seroconversion[All Fields] OR vaccine efficacy[All Fields] OR ”dose-response”[All Fields]) AND (”Systematic Review”[PB] OR ”Meta-Analysis”[Mesh] OR ”Review Literature as Topic”[Mesh] OR ”Review Literature”[All Fields] OR ”Systematic Literature Review”[All Fields])	.000	.000	.000	.000

Continued on next page

Table 5.13 – continued from previous page

Model	Boolean Query	P	F1	F3	R
Mistral-L	(”Rabies Vaccination”[Mesh] OR ”Rabies Vaccines”[Mesh] OR ”Rabies Vaccines, Inactivated”[Mesh] OR ”Rabies Vaccines, Live”[Mesh] OR ”Rabies Immunization”[Mesh] OR ”Rabies Immunoglobulin, Human”[Mesh] OR ”Rabies Immunoglobulins”[Mesh] OR ”Rabies Virus Neutralizing Antibodies”[Mesh] OR ”Rabies Vaccines, Human Diploid Cell”[Mesh] OR ”Rabies Vaccines, Vertebrate Cell Culture”[Mesh] OR ”Rabies”[All Fields] OR ”rabies vaccines”[All Fields] OR ”rabies vaccine”[All Fields] OR ”rabies vaccination”[All Fields] OR ”rabies immunization”[All Fields] OR ”rabies prophylaxis”[All Fields] OR ”post-exposure prophylaxis”[All Fields] OR ”post-exposure prophylaxis, rabies”[All Fields] OR ”pre-exposure prophylaxis”[All Fields] OR ”pre-exposure prophylaxis, rabies”[All Fields] OR ”rabies immunoglobulin”[All Fields] OR ”rabies immunoglobulins”[All Fields] OR ”rabies ig”[All Fields] OR ”rabies igg”[All Fields] OR ”rabies antibodies”[All Fields] OR ”rabies titer”[All Fields] OR ”rabies serology”[All Fields] OR ”rabies seronegative”[All Fields] OR ”rabies seropositive”[All Fields]) AND (Immunogenicity[All Fields] OR ”immune response”[All Fields] OR ”antibody response”[All Fields] OR ”antibody titer”[All Fields] OR ”antibody production”[All Fields] OR ”immune protection”[All Fields] OR ”immunization response”[All Fields] OR ”vaccine response”[All Fields]) AND (Systematic[All Fields] OR ”Meta-Analysis”[Publication Type] OR ”meta analysis”[All Fields] OR ”meta-analyses”[All Fields] OR ”review”[Publication Type] OR ”review, systematic”[Publication Type] OR ”systematic reviews”[All Fields])	.002	.004	.006	.013
Mistral	(”Rabies Vaccines/” OR ”rabies vaccination”[All Fields] OR ”rabies immunization”[All Fields]) AND (Immunogenicity[Mesh] OR ”antibody response”[All Fields] OR ”immune response”[All Fields] OR ”antigen-antibody complexes”[Mesh] OR ”humoral immunity”[Mesh]) AND (”pre-exposure prophylaxis”[Mesh] OR ”post-exposure prophylaxis”[Mesh] OR ”immunization, postexposure”[Mesh])	.207	.114	.093	.079
Zephyr	(”rabies vaccine”[Mesh] OR (”rabies”[All Fields] AND ”vaccine”[All Fields])) AND ((”immunogenicity”[Mesh] OR ”immunogenicity”[All Fields]) OR (”antibody response”[Mesh] OR ”antibody response”[All Fields])) AND ((”pre-exposure”[All Fields] OR ”pre exposure”[All Fields]) OR (”post-exposure”[All Fields] OR ”post exposure”[All Fields])) AND ((”systematic review”[All Fields] OR ”meta-analysis”[All Fields]))	.000	.000	.000	.000
Llama3.1	(rabies vaccine*[All Fields] OR post exposure prophylaxis[Mesh:noexp] OR pre exposure prophylaxis[Mesh:noexp] OR rabies immunoglobulin[Mesh:noexp] OR (rabies[All Fields] AND immunogen*[All Fields]) OR (rabies[All Fields] AND vaccin*[All Fields]) OR (”rabies in situ”[All Fields] OR post vaccine react*[All Fields] OR vaccine reaction*[All Fields])) AND (immunogenicity[Mesh:noexp] OR immunogenecity[All Fields] OR immune respons*[All Fields] OR ”dose response”[All Fields] OR ”dose response relationship”[All Fields] OR ”dose response curve”[All Fields] OR (immunogen*[All Fields] AND response[All Fields] AND dose[All Fields]))	.046	.083	.138	.421
o1	(rabies[MeSH:noexp] OR ”Rabies Virus”[MeSH:noexp] OR ”Rabies Vaccines”[MeSH:noexp] OR rabies[tiab] OR ”rabies virus”[tiab] OR ”rabies infection”[tiab] OR ”rabies vaccine”[tiab]) AND ((”Immunization, Pre-Exposure”[MeSH:noexp] OR ((pre-exposure*[tiab] OR preexposure*[tiab]) AND (prophylaxis[tiab] OR vaccinat*[tiab] OR immuni*[tiab]))) OR (”Immunization, Post-Exposure”[MeSH:noexp] OR ((post-exposure*[tiab] OR postexposure*[tiab]) AND (prophylaxis[tiab] OR vaccinat*[tiab] OR immuni*[tiab])) OR PEP[tiab])) AND (”Immunogenicity, Vaccine”[MeSH:noexp] OR immunogenic*[tiab] OR (immune[tiab] AND (response*[tiab] OR respons*[tiab])) OR ”antibody formation”[MeSH:noexp] OR seroconversion*[tiab] OR (antibody[tiab] AND (titer*[tiab] OR titre*[tiab] OR level*[tiab] OR concentration*[tiab])) OR ((neutralizing[tiab] OR neutralising[tiab]) AND (antibody[tiab] OR antibodies[tiab]))) AND humans[MeSH Terms]	.141	.241	.372	.816

Topic title: *Immunogenicity after pre- and post-exposure rabies vaccination: A systematic review and dose-response meta-analysis*

5.5.1 Comparison by LLMs

Boolean queries generated by different LLMs vary in structure, term coverage, and specificity. The manually formulated query is the most comprehensive, incorporating MeSH terms, free-text synonyms, and Boolean logic to maximise recall. For example:

```
("rabies vaccines"[Mesh] OR "rabies virus"[Mesh] OR Rabies[tiab] AND
(vaccine[tiab] OR Immunization[tiab])) AND (...)
```

Table 5.14: Comparison of different prompts used to generate Boolean queries for topic 22 from the Seed collection, using o1.

Prompt	Boolean Query	P	F1	F3	R
Manual	(”rabies vaccines”[Mesh] OR ((vaccination[Mesh]) AND (rabies[Mesh] OR ”rabies virus”[Mesh])) OR ((Rabies[tiab]) AND (vaccine[tiab] OR Vaccines[tiab] OR Vaccination[tiab] OR Vaccinations[tiab] OR Immunise[tiab] OR Immunize[tiab] OR Immunisation[tiab] OR Immunization[tiab]))) AND (”immunization, secondary”[Mesh] OR ”antibodies, viral”[Mesh] OR ”time factors”[Mesh] OR secondary[tiab] OR booster[tiab] OR boost[tiab] OR boosters[tiab] OR boostability[tiab] OR titer[tiab] OR gmt[tiab] OR ((sero[tiab]) AND (conversion[tiab] OR protection[tiab])) OR seroconversion[tiab] OR seroprotection[tiab]) AND (”administration and dosage”[sh] OR Administer[tiab] OR Administered[tiab] OR Administration[tiab] OR Intramuscular[tiab] OR Intradermal[tiab] OR “pre-exposure”[tiab] OR “pre exposure”[tiab]) NOT (animals[Mesh] NOT humans[Mesh])	.113	.201	.332	.947
p1	(”rabies”[MeSH Terms] OR rabies[Title/Abstract] OR ”rabies virus”[Title/Abstract]) AND (”rabies vaccines”[MeSH Terms] OR ”rabies vaccine”[Title/Abstract] OR ”rabies vaccination”[Title/Abstract] OR vaccine*[Title/Abstract]) AND (immunogenic*[Title/Abstract] OR ”immune response”[Title/Abstract] OR ”immunological response”[Title/Abstract]) AND ((”pre-exposure prophylaxis”[MeSH Terms] OR ”pre-exposure prophylaxis”[Title/Abstract] OR preexposure[Title/Abstract]) OR (”post-exposure prophylaxis”[MeSH Terms] OR ”post-exposure prophylaxis”[Title/Abstract] OR postexposure[Title/Abstract]))	.222	.326	.427	.618
p2	(”Rabies Vaccines”[Mesh] OR ”rabies vaccine*”[tiab] OR ”rabies vaccination*”[tiab]) AND (”pre-exposure prophylaxis”[tiab] OR ”preexposure prophylaxis”[tiab] OR ”pre exposure prophylaxis”[tiab] OR ”post-exposure prophylaxis”[tiab] OR ”postexposure prophylaxis”[tiab] OR ”post exposure prophylaxis”[tiab]) AND (”immunogenicit*”[tiab] OR ”immune respons*”[tiab] OR ”immunologic respons*”[tiab] OR ”serologic respons*”[tiab] OR ”antibody titer*”[tiab] OR ”antibody titre*”[tiab] OR seroconversion[tiab] OR ”virus neutralizing antibody*”[tiab] OR ”virus neutralising antibody*”[tiab])	.213	.286	.345	.434
p3	((”rabies”[MeSH Terms] OR ”rabies virus”[MeSH Terms] OR rabies[tiab] OR ”rabies virus”[tiab]) AND (”rabies vaccines”[MeSH Terms] OR ”rabies immunoglobulin”[MeSH Terms] OR ”vaccination”[MeSH Terms] OR ”pre-exposure prophylaxis”[MeSH Terms] OR ”post-exposure prophylaxis”[MeSH Terms] OR ”rabies vaccine*”[tiab] OR ”rabies immunoglobulin”[tiab] OR vaccin*[tiab] OR immuniz*[tiab] OR immunis*[tiab] OR ”pre-exposure prophylaxis”[tiab] OR ”post-exposure prophylaxis”[tiab] OR PEP[tiab] OR PrEP[tiab] OR RIG[tiab] OR HRIG[tiab]) AND (immunogenicity[tiab] OR ”immune response*”[tiab] OR ”antibody titer*”[tiab] OR ”antibody titre*”[tiab] OR antibodies[tiab] OR seroconversion[tiab] OR seroconvert*[tiab] OR ”humoral immunity”[tiab] OR ”immune memory”[tiab] OR ”neutralizing antibody*”[tiab] OR ”neutralising antibody*”[tiab] OR ”virus neutralizing antibody*”[tiab])) NOT (animals[MeSH Terms] NOT humans[MeSH Terms])	.073	.136	.239	.987
p4	(rabies[MeSH:noexp] OR ”Rabies Virus”[MeSH:noexp] OR ”Rabies Vaccines”[MeSH:noexp] OR rabies[tiab] OR ”rabies virus”[tiab] OR ”rabies infection”[tiab] OR ”rabies vaccine”[tiab]) AND ((”Immunization, Pre-Exposure”[MeSH:noexp] OR ((pre-exposure*[tiab] OR preexposure*[tiab]) AND (prophylaxis[tiab] OR vaccinat*[tiab] OR immuni*[tiab]))) OR (”Immunization, Post-Exposure”[MeSH:noexp] OR ((post-exposure*[tiab] OR postexposure*[tiab]) AND (prophylaxis[tiab] OR vaccinat*[tiab] OR immuni*[tiab])) OR PEP[tiab])) AND (”Immunogenicity, Vaccine”[MeSH:noexp] OR immunogenic*[tiab] OR (immune[tiab] AND (response*[tiab] OR respons*[tiab])) OR ”antibody formation”[MeSH:noexp] OR seroconversion*[tiab] OR (antibody[tiab] AND (titer*[tiab] OR titre*[tiab] OR level*[tiab] OR concentration*[tiab])) OR ((neutralizing[tiab] OR neutralising[tiab]) AND (antibody[tiab] OR antibodies[tiab]))) AND humans[MeSH Terms]	.141	.241	.372	.816
p5	(”Rabies Vaccines”[Mesh:noexp] OR (rabies[All Fields] AND vaccin*[All Fields]) OR ”rabies vaccine”[All Fields] OR ”rabies immunoglobulin”[All Fields] OR rabipur[All Fields] OR verorab[All Fields] OR imovax[All Fields] OR ”pre-exposure prophylaxis”[All Fields] OR ”post-exposure prophylaxis”[All Fields] OR ”rabies prophylaxis”[All Fields]) AND (”Immunogenicity, Vaccine”[Mesh:noexp] OR immunogenic*[All Fields] OR immunolog*[All Fields] OR (antibod*[All Fields] AND (respons*[All Fields] OR titer*[All Fields] OR titre*[All Fields])) OR seroconversion[All Fields] OR ”immune response”[All Fields] OR (virus[All Fields] AND neutraliz*[All Fields]) OR immunity[Mesh])	.017	.034	.066	1.000

Topic title: *Immunogenicity after pre- and post-exposure rabies vaccination: A systematic review and dose-response meta-analysis*

GPT-3.5 models tend to produce simpler queries with limited term expansion, reducing recall. For instance:

```
("rabies vaccines"[MeSH Terms] OR "rabies vaccination"[All Fields]) AND
("immunogenicity"[MeSH Terms])
```

GPT-4 adds more structure but often over-filters results, introducing excessive constraints (e.g., systematic review-specific criteria), which can hurt retrieval recall. More advanced models such as

GPT-4o and o1 generate more balanced queries by combining effective term expansion with structured Boolean logic. The best-performing model, o1, integrates logic and human-relevant filters effectively:

```
(rabies[MeSH:noexp] OR "Rabies Virus"[MeSH:noexp] OR rabies[tiab]) AND (immune
response[tiab] OR immunogenicity[MeSH]) AND humans[MeSH Terms]
```

Overall, simpler models tend to lack detail, advanced models risk over-filtering, and o1 achieves the best balance between recall and precision.

5.5.2 Comparison by Prompts

Boolean queries also vary significantly with prompt design, even when generated by the same model. Different prompts emphasise different aspects of query construction, affecting recall and precision. For instance, prompts p1 and p2 are concise and structured, prioritising Boolean syntax but limiting term coverage. The query from p1 is:

```
("rabies"[MeSH Terms] OR rabies[Title/Abstract] OR "rabies virus"[Title/Abstract])
AND ("rabies vaccines"[MeSH Terms] OR "rabies vaccine"[Title/Abstract]) AND
(immunogenic*[Title/Abstract])
```

In contrast, p3 and p4 encourage broader term expansion, resulting in more inclusive queries. The query from p4 includes:

```
(rabies[MeSH:noexp] OR "Rabies Virus"[MeSH:noexp]) AND ("Immunization,
Pre-Exposure"[MeSH:noexp] OR pre-exposure*[tiab]) AND (immunogenic*[tiab] OR
"antibody formation"[MeSH:noexp])
```

Prompt p5, which uses the PICO (Patient, Intervention, Comparison, Outcome) structure, produces the most structured queries, balancing precision and recall. Overall, prompts tailored for systematic review specialists (p3, p4, p5) tend to yield more comprehensive queries, whereas simpler prompts (p1, p2) lead to narrower retrievals.

The effectiveness of a prompt ultimately depends on the desired balance between recall and specificity in a systematic review context.

5.6 Summary of Findings

In this chapter, we conducted an extensive evaluation of automated Boolean query formulation and refinement for systematic reviews using both the ChatGPT interface and a range of LLMs.

Our initial evaluation showed that Boolean queries generated by the early version of the ChatGPT interface generally achieved higher precision than those produced by existing automated methods (Objective/Conceptual), though this came at the cost of reduced recall. ChatGPT was also effective in refining pre-existing queries, further improving precision by filtering out irrelevant documents, with only marginal losses in recall. Prompt design had a significant impact on query effectiveness. A

particularly strong positive signal came from including high-quality example queries in the prompt, which substantially improved recall. Additionally, guiding ChatGPT through iterative, structured prompts—modelled after state-of-the-art query generation methods—improved both recall and precision. However, the effectiveness of this approach was highly dependent on the choice of seed studies.

Extending our investigation to multiple LLMs confirmed that the general trends observed with the early ChatGPT interface generalise across models. Newer models notably reduced the recall gap observed in earlier experiments. Model effectiveness varied, with some (e.g., o1, Llama3.1, GPT3.5) consistently outperforming others. Prompt sensitivity was model-dependent, highlighting the importance of tuning prompt strategies to the specific model in use. We found that separating system instructions from user prompts and enforcing structured output formats (e.g., JSON) had minimal impact on query effectiveness, suggesting limited benefit from imposing rigid output constraints.

One persistent challenge across models was the variability in query correctness and effectiveness across repeated generations. In particular, generating valid Boolean queries reliably remained difficult for many models. Frequent issues included incorrect or nonexistent MeSH terms. We observed that 55% of MeSH terms generated by ChatGPT were invalid or not part of the official vocabulary, which negatively affected recall. To address this issue, one possible solution is to aggregate LLM-generated queries with automated MeSH term suggestion methods, which we introduce in the next chapter to improve the reliability and completeness of Boolean queries.

In summary, while LLMs demonstrate strong potential for automating Boolean query formulation in systematic reviews, real-world deployment requires careful attention to model selection, prompt engineering, and validation. Although it remains an open question whether LLMs should be solely responsible for generating queries in systematic review workflows, this chapter highlights their potential—balanced by challenges around variability, robustness, and reproducibility. Regardless of their ultimate role, we see this as a promising direction for future research.

Chapter 6

MeSH Term Suggestion

Boolean queries for systematic reviews often combine free-text keywords with structured vocabulary such as Medical Subject Headings (MeSH) to improve retrieval effectiveness. MeSH is a controlled vocabulary developed by the U.S. National Library of Medicine to conceptually index biomedical literature in databases like PubMed. These terms act as standardised representations of medical concepts, helping disambiguate terminology and enabling more comprehensive literature searches.

Prior research has consistently shown that incorporating MeSH terms into Boolean queries enhances both precision and recall, particularly in complex or domain-specific reviews. However, constructing high-quality MeSH-enhanced queries remains a challenge: The MeSH vocabulary is large and continually expanding—comprising over 29,640 unique terms as of 2022, the time of this study—making it difficult even for expert information specialists to navigate effectively. PubMed attempts to address this complexity through Automatic Term Mapping (ATM), which expands user-supplied free-text terms by mapping them to relevant MeSH entries. While ATM has shown utility in general-purpose search, it suffers from several known limitations: poor handling of acronyms, inconsistent mapping of synonyms, and confusion between MeSH terms and journal names. Critically, its effectiveness in systematic review contexts has not been systematically evaluated.

One might argue that directly generating Boolean queries using large language models (LLMs) inherently enables MeSH term suggestion. However, as discussed in the previous chapter, LLMs often produce invalid or nonexistent MeSH terms, which negatively impacts recall. This observation motivates an alternative approach: rather than relying on LLMs to produce complete queries, can we instead focus on reliably suggesting valid, high-quality MeSH terms to support human- or LLM-assisted query construction?

This chapter investigates that question by focusing exclusively on the task of MeSH term suggestion. Specifically, we explore the feasibility and effectiveness of using both lexical-based methods and modern pre-trained language models(such as BERT, T5, and GPT-style models)to recommend MeSH terms for use in systematic review Boolean queries. We introduce a comprehensive evaluation framework to benchmark these approaches against manually constructed Boolean queries from published systematic reviews. Our experiments show that transformer-based BERT variants, consistently

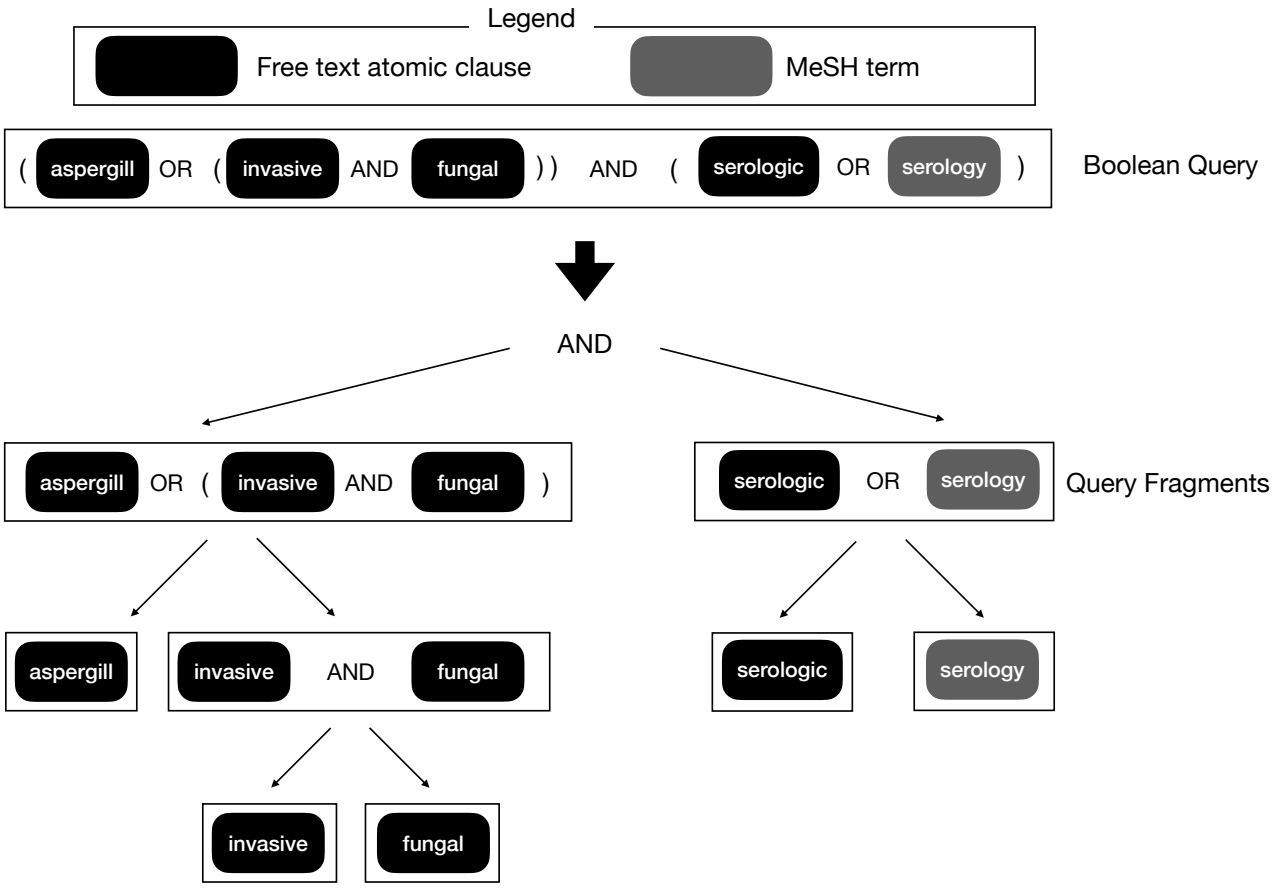


Figure 6.1: Example query showing a **Boolean Query**, two **Query Fragments**, several **Free text atomic clauses**, and a **MeSH term**.

outperform both traditional lexical methods and expert-curated MeSH selections.

Ultimately, the methods proposed in this chapter can be integrated into systematic review tools to assist information specialists in constructing more effective, complete, and consistent Boolean queries, while addressing key limitations observed in the prior chapter.

6.1 MeSH Term Suggestion Task and Framework

We start by formally defining the task of MeSH term suggestion in the context of Boolean query construction for systematic reviews. The goal is to recommend appropriate MeSH terms for Boolean queries that initially contain only free-text terms. These MeSH terms serve to enrich the query, improving retrieval precision and consistency across biomedical databases such as PubMed.

A Boolean query can be represented as a tree structure: internal nodes correspond to Boolean operators (e.g., `AND`, `OR`), and the leaves are atomic clauses—either free-text terms or MeSH terms. Free-text atomic clauses typically describe key biomedical concepts such as diseases, interventions, or population groups. We define each first-level subtree of the Boolean query (i.e., each node directly under the root `AND`) as a *query fragment*. Each fragment typically captures one distinct component of the overall information need [38], and often aligns with a PICO element (i.e., Population, Intervention, Comparison, or Outcome) [218]. These concepts are shown in Figure 6.1.

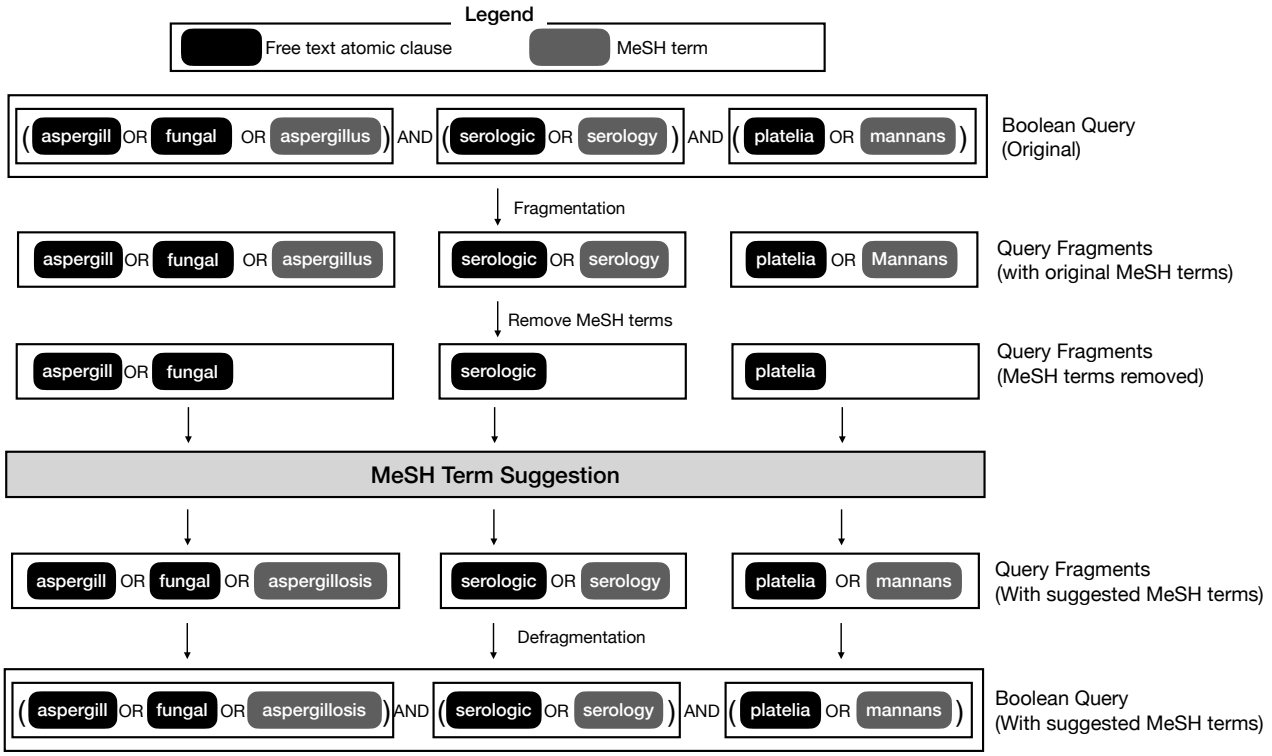


Figure 6.2: Overview of the MeSH term suggestion procedure. Proposed methods using lexical MeSH term retrieval or BERT MeSH term retrieval facilitate the suggestion of MeSH terms. We evaluate each method that suggests MeSH terms in terms of (1) the ability for the suggested MeSH terms to effectively retrieve literature for a defragmented Boolean query , (2) overlap between suggested MeSH terms and MeSH terms included in the original query. Note that the number of MeSH terms suggested for a fragment may be lower or higher than the number of MeSH terms in the original query.

The MeSH term suggestion task is then defined as follows: for each query fragment, identify a set of relevant MeSH terms that can be appended to the fragment. In this work, we treat each query fragment independently and suggest MeSH terms without considering dependencies across fragments. Investigating such dependencies remains an open direction for future work. Figure 6.2 provides an overview of the proposed framework. Starting with an input Boolean query, we apply a *fragmentation* step to extract individual query fragments. To simulate a MeSH-free input, we remove any pre-existing MeSH terms. Then, a MeSH suggestion method is applied to each fragment independently. The resulting augmented fragments (now containing both free-text and MeSH terms) are recombined via *defragmentation*, using the `AND` operator to reconstruct the enriched Boolean query.

6.2 MeSH Term Suggestion Methods

We propose two categories of methods for suggesting MeSH terms: *Lexical methods* and *BERT-based methods*. Lexical methods typically rely on pre-defined term-relationship dictionaries, such as the Unified Medical Language System (UMLS), to support MeSH term suggestion. These approaches are commonly employed in existing systems, including PubMed's Automatic Term Mapping (ATM). However, a key limitation of lexical-based systems is their reliance on manually crafted rules, which are costly to maintain and often brittle. These systems struggle with handling spelling variants, acronyms,

and misspellings, which reduces their robustness and adaptability.

To address these limitations, we also explore methods that leverage pre-trained neural models, specifically BERT, for MeSH term suggestion. Neural models like BERT have demonstrated resilience to the weaknesses of lexical systems [61, 255]. That said, BERT-based approaches introduce their own challenges, most notably the need for large-scale annotated training data. The following sections provide an overview of both lexical and neural approaches to MeSH term suggestion.

6.2.1 Lexical MeSH Term Suggestion

Our lexical-based methods are formulated as a pipeline of three steps: retrieval, ranking, and refinement. The following sections provide a brief overview of each of these steps:

Retrieval

The first step in our MeSH term suggestion pipeline is the **retrieval** of MeSH terms. The retrieval of MeSH terms is facilitated by three different methods:

1. **ATM:** The entire free-text-only query fragment is submitted to the PubMed Entrez API [203] for *automatic term mapping* (ATM). This is the default system used by PubMed for automatically adding MeSH terms to queries.

2. **MetaMap:** Each free-text atomic clause in a query fragment is submitted to MetaMap [14].[1] The results are filtered to include only those entities derived from the MeSH source. All of the mapped MeSH terms are recorded for each of the free-text terms in a query fragment. Additionally, the score is recorded for each MeSH term.

3. **UMLS:** We index UMLS [20][2] into Elasticsearch v7.6. Each free-text atomic clause in the query fragment with MeSH terms removed is submitted to the Elasticsearch index. The results are filtered to include only synonyms of concepts derived from the MeSH source. Additionally, the BM25 score is recorded for each MeSH term.

For the MetaMap and UMLS approaches, the same MeSH term may be retrieved multiple times for a given free-text fragment. To address this, we re-score the MeSH terms using rank fusion (CombSUM) [70]. The intuition for this re-scoring is that frequently retrieved MeSH terms that also obtain high scores from these retrieval methods should be scored highly overall (thus ranked higher than common MeSH terms *or* highly scoring MeSH terms).

Ranking

Once MeSH terms have been retrieved, they are **ranked** using the entity ranking approach described by Jimmy et al. [111], adapting features proposed by Balog [17]. In total, we use eleven entity features.

[1] Version 2018 with options set to default values.

[2] Version 2019AB using the `MRCONSO`, `MRDEF`, `MRREL`, and `MRSTY` tables.

Positive instances correspond to MeSH terms in the original query fragment; negative instances correspond to MeSH terms not in the original query fragment (binary labels). With features and instance labels, we train a learning-to-rank (LTR) model for each retrieval method.

In addition to LTR, we also investigate a rank fusion approach [70], where we combine the normalised MeSH term suggestion scores from each of the three methods to produce a new ranking that incorporates the top-ranked MeSH terms from each method. The motivation is that each method may retrieve different MeSH terms, which may be ranked differently. Therefore, we boost MeSH terms that are retrieved and ranked highly by multiple methods.

Refinement

Finally, we seek to **refine** the list of suggested MeSH terms by determining an appropriate rank cut-off. To achieve this, we use a score-based gain function that measures how much cumulative gain is accumulated across the ranked list. Formally, the cumulative gain CG at rank p is defined as:

$$CG_p = \sum_{i=1}^{p} \text{score}_i \tag{6.1}$$

where each `score` is calculated as $1 -$ normalised score, using min-max normalisation over the scores of all suggested MeSH terms. This inverted scoring assigns lower gain values to higher-ranked terms and higher gain values to lower-ranked terms. The result is that cumulative gain increases slowly at the top of the ranking and more rapidly toward the bottom. This design ensures that the total cumulative gain across the full list is bounded and can be meaningfully used to apply a cut-off.

We introduce a tunable parameter, κ, for each retrieval method, which defines the percentage of total cumulative gain permitted before truncating the list (i.e., refining the ranking). We vary κ from 5% to 95% in increments of 5%. The motivation for this design becomes clear in relation to the gain function: because the top-ranked MeSH term has the lowest (or zero) gain, it is always included in the final suggestion list. This guarantees that at least one MeSH term is suggested for every query fragment.

Note that multiple MeSH terms may share the same score (i.e., they may be tied). To handle this, we adopt a conservative strategy at the cut-off boundary defined by κ. Specifically, we treat all tied MeSH terms as a single unit and include or exclude them as a group, based on whether their combined gain causes the cumulative gain to exceed the threshold. This has the effect of assigning a larger block of gain to tied terms. As a result, tied MeSH terms near the top of the ranking are more likely to be included in the final list, while those near the bottom are more likely to be excluded. In essence, either all tied MeSH terms are kept (if they appear near the top), or none are (if they appear near the bottom).

6.2.2 BERT-based MeSH Term Suggestion

Next, we extend our MeSH term suggestion methods using fine-tuned BERT-based models.

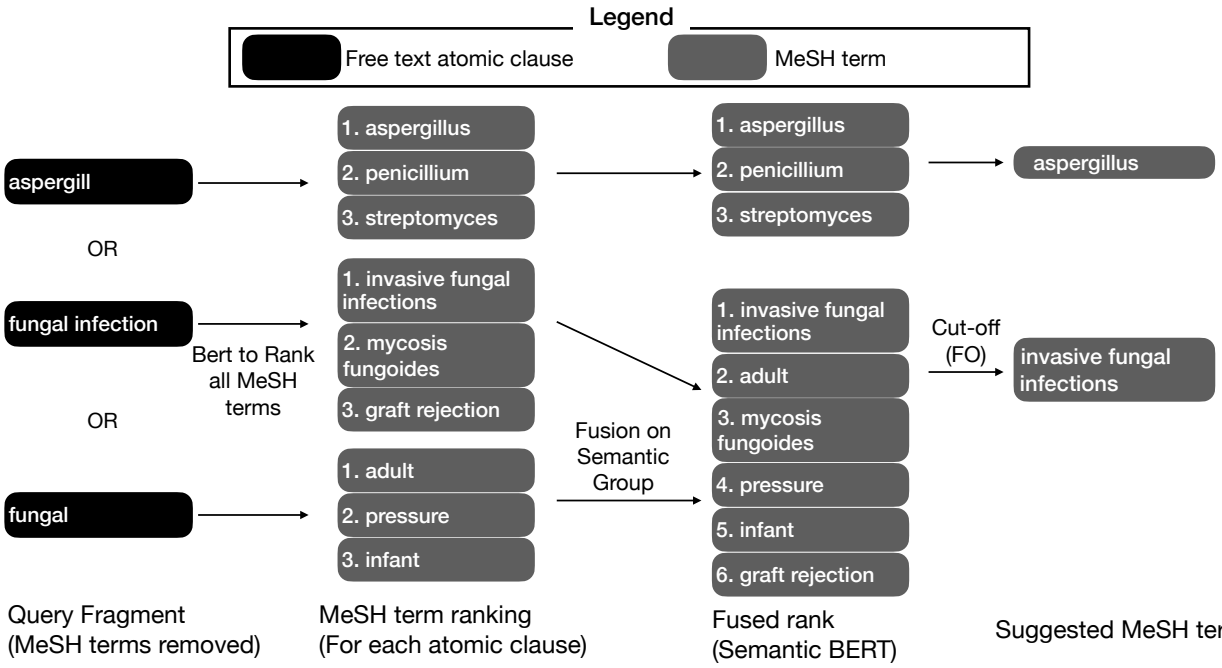


Figure 6.3: Overview of the MeSH term suggestion for the BERT methods. Note that Fusion of MeSH ranks may be optional in the pipeline.

Architecture

We choose BERT models pre-trained in the medical domain—BioBERT [130]—as the backbone for our proposed BERT-based MeSH term suggestion methods. BioBERT is a PLM pre-trained on PubMed abstracts and PubMed Central (PMC)[3] full-text articles using the BERT training architecture [61]. After fine-tuning, BioBERT has achieved state-of-the-art performance on many medical-related tasks, including biomedical named entity recognition, relation extraction, and question answering [130]. We show the architecture of our fine-tuning and inference processes in Figure 6.4.

Ideally, training data closely related to the target task should be used to fine-tune a PLM to achieve the highest effectiveness. In our case, this suggests the use of professionally constructed medical systematic review Boolean queries to fine-tune our model. However, PLMs are typically data-hungry and require a large number of labelled training samples. In systematic review literature search, even though several public datasets are available with Boolean queries—such as the CLEF TAR collections [113, 114, 115], the Seed Collection [258], and the collection from Scells et al. [205]—only 253 unique topics would be available to train the model, which is an insufficient amount to effectively fine-tune a BERT model.

Instead, we create training samples by approximating the target task using data obtained from PubMed. We use the publicly available PubMed baseline to obtain the metadata for all published articles up to the start of 2022.

The metadata contains information such as the title and abstract, but importantly for this work, it also includes author-assigned keywords and the relevant MeSH terms for each article. We use the assigned keywords and MeSH terms for every article in the PubMed dataset to approximate the task of

[3]PubMed Central is the repository containing full-text articles of the open-access part of the PubMed database.

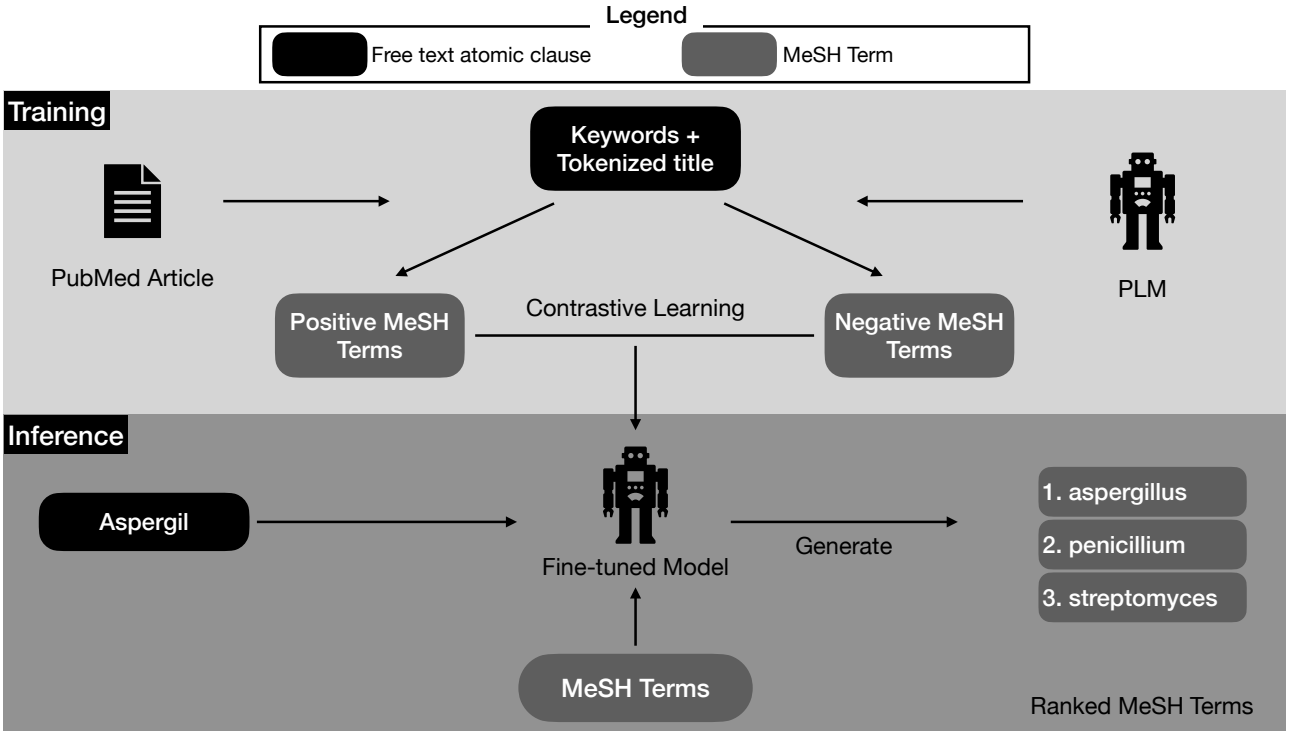


Figure 6.4: Architecture of model fine-tuning and inference.

Table 6.1: Example query fragments with separation of semantic groups. In the example, 'neonatal sepsis', 'neonatal bacteremia' and 'neonatal infections' are grouped to form a semantic group, while 'death' is another semantic group.

MeSH Removed Fragment	neonatal sepsis OR neonatal bacteremia OR neonatal infections OR death			
free text atomic clauses	neonatal sepsis	neonatal bacteremia	neonatal infections	death
semantic group	neonatal sepsis, neonatal bacteremia, neonatal infections			death

MeSH term suggestion. To maximise the amount of training data, we also extract keywords from the title (as not all PubMed articles contain keywords). To tokenise titles, we first use Gensim [121], and then remove stopwords using NLTK [143].

We use Tevatron [76] to develop a dense retriever to suggest MeSH terms. The model is fine-tuned with a localised contrastive loss using triples of $< k_{a,i}, m_a^+, m_a^- >$, where a is a PubMed article, $k_{a,i}$ is the ith keyword in the article, m_a^+ are the MeSH terms for the article, and m_a^- are ten randomly sampled MeSH terms from the MeSH thesaurus. Many MeSH terms contain spaces or punctuation. Our model treats each MeSH term as a unique token in the model vocabulary. Once the model is fine-tuned, we obtain an encoding for all MeSH terms. At inference time, we create an encoding for a keyword to obtain a score using the `[CLS]` token for all MeSH terms. Thus, our method scores and ranks all MeSH terms given a keyword.

Ranking Suggestions

The goal of MeSH term suggestion is to suggest MeSH terms for each query fragment. However, the result from the BERT suggestion method consists of a ranked list of MeSH term suggestion for

each free text atomic clause. We need to combine the rankings for each MeSH term. We formulate this combination task into two steps: (1) choosing how to represent a MeSH term ranking, and (2) choosing where to cut off the ranking. We present an overview of the combination task in Figure 6.3.

First, we choose how to represent a ranking, which means deciding whether MeSH terms should be suggested individually for every free text atomic clause, collectively for every fragment, or using other heuristics to determine how the representation should be computed. We designed three ranking representation methods:

1. **Atomic BERT**: We treat suggestions for each free text atomic clause individually, applying no strategy to combine the suggestions.

2. **Fragment BERT**: We combine all MeSH term rankings for a given query fragment. We apply rank fusion (normalised CombSUM [70]) to all of the free text atomic clauses in a query fragment. For computational reasons, we only use the top 20 MeSH terms for each free text atomic clause.

3. **Semantic BERT**: We semantically group free text atomic clauses and apply the same rank fusion technique as above, but to each group.

   We show an example of a semantic group in Table 6.1. To derive semantic groups, we first take all free text atomic clauses from the fragment and obtain word2vec embeddings for each free text atomic clause. We then compute cosine similarities between all free text atomic clauses to determine if they are semantically related. In our experiments, we apply a threshold of 0.7 on the similarity [4].

   We use a word2vec model pre-trained on PubMed and Wikipedia [160]. There are two reasons we use word2vec rather than BERT for semantic grouping. First, if we apply our proposed BERT model, we note that it is fine-tuned using semantic pairs of free text atomic clauses and MeSH terms; thus, calculating the similarity between two free text atomic clauses can result in a model mismatch. Second, the use of an additional BERT model increases the latency of suggestions at inference time, as each free text atomic clause would need to be encoded twice.

Second, we choose where to cut off the ranking of MeSH terms from the ranking representations. We propose four strategies for ranking cut-off:

1. **First only (FO)**: The first MeSH term of the ranking is selected for each ranking representation.

2. **Same as free text atomic clauses (SA)**: The number of MeSH terms selected equals the number of free text atomic clauses in each fragment (applicable only to **Fragment BERT**).

[4] We set the similarity threshold to 0.7 based on heuristic observations. Cosine similarities above 0.7 typically indicate strong semantic similarity in word2vec spaces, while lower thresholds tended to group unrelated clauses during manual inspection. Conversely, higher thresholds (e.g., above 0.8) were found to be overly restrictive, resulting in very few clause pairs being grouped at all.

3. **Same as original (SO)**: The number of MeSH terms selected equals the number of MeSH terms in the query fragment prior to MeSH removal (applicable only to **Fragment BERT**).

4. **Linear (LN)**: The number of MeSH terms selected is learned using a linear function with respect to the number of free text atomic clauses in the fragments (applicable only to **Fragment BERT**).

6.3 Experimental Setup

This section describes the datasets, preprocessing steps, training settings, and evaluation procedures used to assess the effectiveness of MeSH term suggestion methods.

6.3.1 Datasets

We evaluate our methods using topics from the CLEF TAR tasks from 2017, 2018, and 2019 [113, 114, 115]. These datasets contain professionally constructed Boolean queries designed for systematic review literature search. A total of 15 topics are discarded due to missing MeSH terms,[5] and one additional topic is discarded due to retrieval issues caused by automatic query translation from Ovid Medline to PubMed format [206].[6] After filtering, we use 116 unique topics in total, accounting for partial topic overlap across years.

6.3.2 Preprocessing and Query Fragment Extraction

For each topic, we automatically segment the Boolean query into *query fragments* using the procedure from Scells et al. [206]. Each query fragment contains at least one MeSH term. This results in 311 unique fragments across all queries, averaging 2.68 fragments per query. We manually correct spelling, syntax, and formatting errors in each fragment. We also extract:

- Original MeSH terms,
- Free-text keywords,
- Fragment versions with MeSH terms, and
- Fragment versions with MeSH terms removed.

6.3.3 Training Settings

Lexical Methods. For training the learning-to-rank model used in the lexical methods, we use the training/test splits provided by the CLEF TAR datasets. The 2019 dataset is split by review type—*intervention* and *diagnostic test accuracy (DTA)*—whereas the 2017 and 2018 datasets consist

[5] Discarded topics: **2017**: CD007427, CD010771, CD010772, CD010775, CD010783, CD010860, CD011145; **2018**: CD007427, CD009263, CD009694; **2019**: CD006715, CD007427, CD009263, CD009694, CD011768.

[6] Additional discarded topic: **2017**: CD010276.

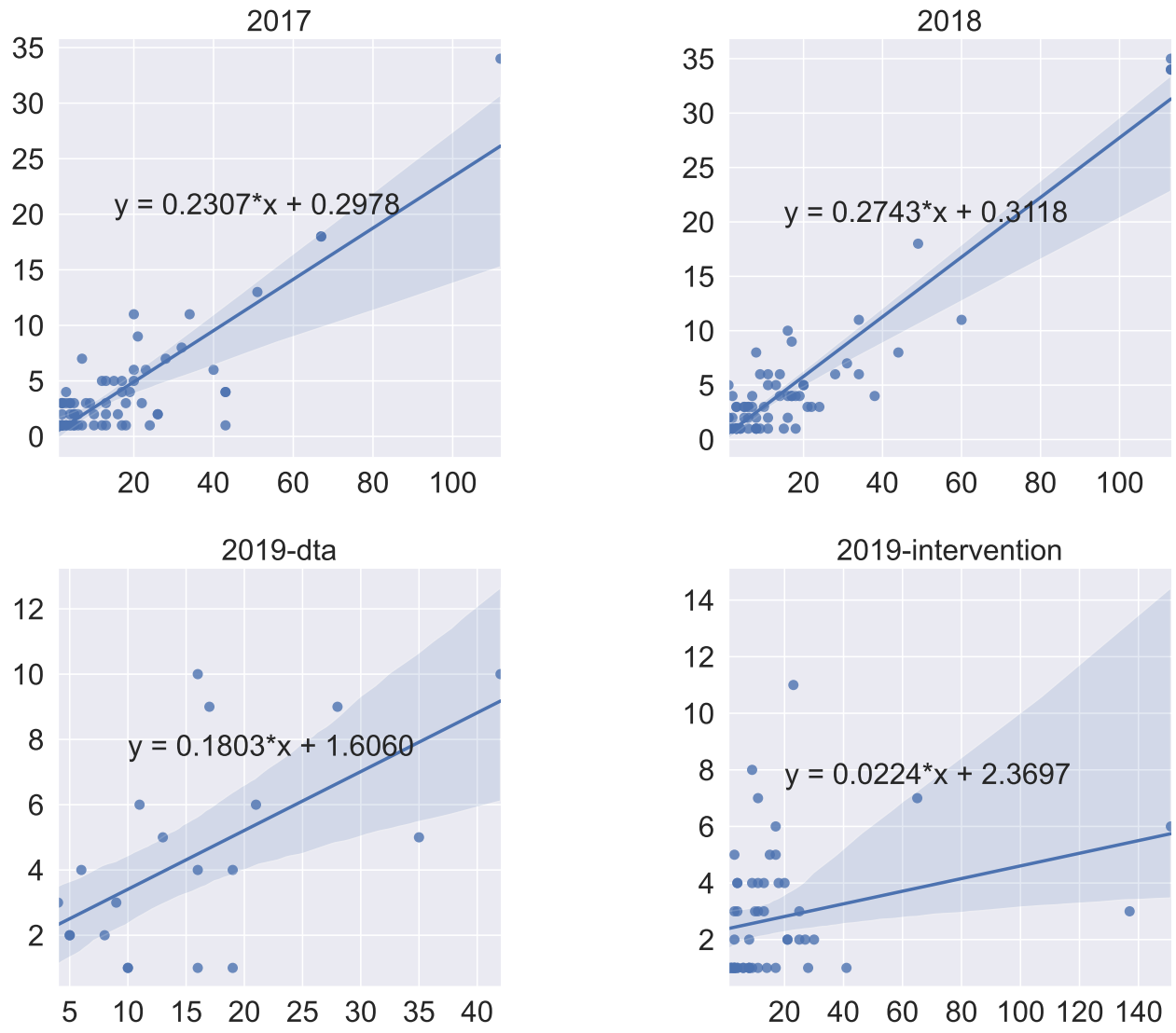


Figure 6.5: Linear regression performed on the number of keywords (x-axis) and the number of MeSH terms (y-axis) in query fragments for training splits of CLEF TAR 2017, 2018, 2019-dta and 2019-intervention.

entirely of DTA topics. We use the ‘quickrank‘ library [29], with LambdaMART configured to optimise nDCG. All other hyperparameters are kept at default values.

BERT Methods. To support the linear cut-off strategy used in the BERT-based suggestion method (see Section 6.2.2), we learn a dataset-specific cut-off function. For this, we extract all query fragments from the CLEF TAR training splits, count the number of free-text atomic clauses and MeSH terms in each fragment, and perform linear regression to fit a predictive model. The learned regression curves are shown in Figure 6.5.

6.3.4 Evaluation

Our evaluation strategy covers both (1) the retrieval effectiveness of queries enriched with suggested MeSH terms, and (2) the overlap between suggested MeSH terms and those originally selected by

expert information specialists.

Retrieval Effectiveness. The ultimate goal of a systematic review search is to retrieve all relevant documents with minimal noise. Therefore, we evaluate retrieval effectiveness using precision, recall, and F_β (with $\beta = \{1,3\}$) based on the documents retrieved by executing the defragmented Boolean queries in PubMed. We use the Entrez API to issue queries, and apply date restrictions to ensure reproducibility, as PubMed is continually updated.

MeSH Term Overlap. In parallel, we evaluate how closely the suggested MeSH terms match those used in the original queries. While the original terms are not necessarily optimal, they are curated by domain experts and provide a useful reference. Overlap is computed using the Jaccard index.

Evaluation Settings. We evaluate lexical methods in two modes:

- **All**: All retrieved MeSH terms are included.
- **Cut**: MeSH terms are filtered using the score-based cut-off described in Section 6.2.1.

We also evaluate all BERT-based suggestion methods, comparing their performance against the original queries and lexical baselines.

6.4 Main Results

This section presents the results of our evaluation of MeSH term suggestion methods, focusing on two core aspects: retrieval effectiveness and suggestion effectiveness. **Retrieval effectiveness** assesses how well Boolean queries augmented with suggested MeSH terms perform in retrieving relevant documents for systematic reviews. We compare both lexical-based and BERT-based approaches against the original expert-curated queries. **Suggestion effectiveness** examines the quality of the suggested MeSH terms themselves by comparing them to the MeSH terms included in the original queries. This provides insight into how closely automated methods approximate expert selections.

The following subsections report quantitative results for both evaluation dimensions, followed by a case study to provide qualitative insight into how suggested terms affect query formulation.

6.4.1 Retrieval Effectiveness of Suggested MeSH Terms

Lexical Methods

The results of the lexical methods are presented in Table 6.2. Unrefined methods generally achieve higher recall than their corresponding refined versions, but at the cost of lower precision. This indicates that adding more MeSH terms to query fragments increases the number of both relevant and irrelevant studies retrieved.

Table 6.2: Search effectiveness of Boolean query using suggested MeSH terms evaluated by precision (P), F1, F3 and recall (R). Lexical methods: For each method, *CUT* indicates cut-off ranks. BERT methods: *FO*, *SA*, *SO*, *LN* indicate different cut-off strategies. No statistical significant differences are detected between the ORIGINAL query and those obtained by the other methods (two-tailed t-test with Bonferroni correction, $p < 0.05$).

	Dataset	2017				2018				2019-dta				2019-intervention			
	Method	P	F1	F3	R	P	F1	F3	R	P	F1	F3	R	P	F1	F3	R
	ORIGINAL	.289	.311	.440	.745	.323	.576	.965	.629	.227	**.421**	**.738**	.966	.165	.212	.309	.471
Lexical Method	ATM	.265	.262	.353	.549	.317	.552	.898	.190	.113	.211	.373	.916	.156	.183	.269	.073
	ATM-CUT	.316	.299	.404	.269	.354	.624	.033	.998	**.243**	.398	.637	.375	.173	.191	.288	.938
	MetaMap	.304	.287	.381	.519	.342	.599	.980	.150	.131	.245	.433	.791	.135	.218	.339	.974
	MetaMap-CUT	.337	.312	.423	.191	.360	.633	.043	.071	.193	.358	.625	.393	.159	.251	.382	.831
	UMLS	.275	.269	.355	.458	.297	.519	.847	.000	.114	.214	.384	.616	.118	.183	.275	.998
	UMLS-CUT	.335	.315	.430	.225	.384	**.681**	**.133**	.963	.174	.305	.508	.381	.173	.191	.295	.638
	Fusion	.218	.227	.300	.712	.284	.495	.800	.455	.103	.192	.342	.075	.109	.173	.263	.212
	Fusion-CUT	.323	.303	.409	.282	.333	.582	.951	.120	.147	.269	.465	.394	.161	.173	.262	.797
BERT Method	Atomic-BERT-FO	.257	.249	.330	**.830**	.289	.488	.795	.523	.092	.173	.310	.870	.070	.126	.219	.587
	Semantic-BERT-FO	.273	.272	.363	.633	.284	.501	.820	.502	.096	.181	.324	.870	.110	.183	.288	.483
	Fragment-BERT-FO	**.342**	**.324**	**.446**	.415	**.382**	.678	.132	.041	.169	.314	.548	.924	**.212**	**.276**	**.422**	.106
	Fragment-BERT-SA	.212	.216	.284	.699	.268	.471	.772	**.652**	.097	.181	.323	**.357**	.076	.137	.235	**.806**
	Fragment-BERT-SO	.265	.250	.335	.593	.328	.588	.991	.258	.129	.243	.433	.987	.176	.238	.358	.431
	Fragment-BERT-LN	.265	.274	.373	.615	.318	.561	.925	.355	.112	.211	.378	.969	.105	.167	.265	.428

When evaluated using F1 and F3, the refined methods consistently outperform their unrefined counterparts across all datasets. Specifically, U-CUT achieves the highest effectiveness on CLEF 2017 and 2018, A-CUT performs best on CLEF 2019-dta, and M-CUT achieves the highest scores on the CLEF 2019-intervention dataset.

In terms of recall, the unrefined fusion method achieves the highest among all lexical suggestion methods. This improvement likely results from combining all MeSH terms suggested by ATM, MetaMap, and UMLS using logical OR, thus maximising coverage. However, this also degrades precision, suggesting that unrefined fusion is not suitable for automatic query formulation. That said, in a semi-automatic setting, information specialists could review and select from the suggested terms to improve precision while retaining recall.

BERT-based Methods

We first compare the effectiveness of BERT-based suggestions with the original Boolean queries. Across all evaluation metrics (Precision, F1, F3, and Recall), BERT methods outperform the original queries on CLEF TAR 2017, 2018, and 2019-intervention. In contrast, performance is generally lower for CLEF TAR 2019-dta, likely due to the small number of topics (only eight unique topics), which limits the reliability of the evaluation.

We then compare BERT suggestions with lexical suggestions. When compared with unrefined lexical methods, BERT suggestions achieve comparable or better results in terms of F1 and F3, with consistent gains across all datasets. Compared to refined lexical methods, BERT suggestions perform similarly on most datasets, but underperform on CLEF TAR 2019-dta, where refined lexical methods achieve better overall effectiveness.

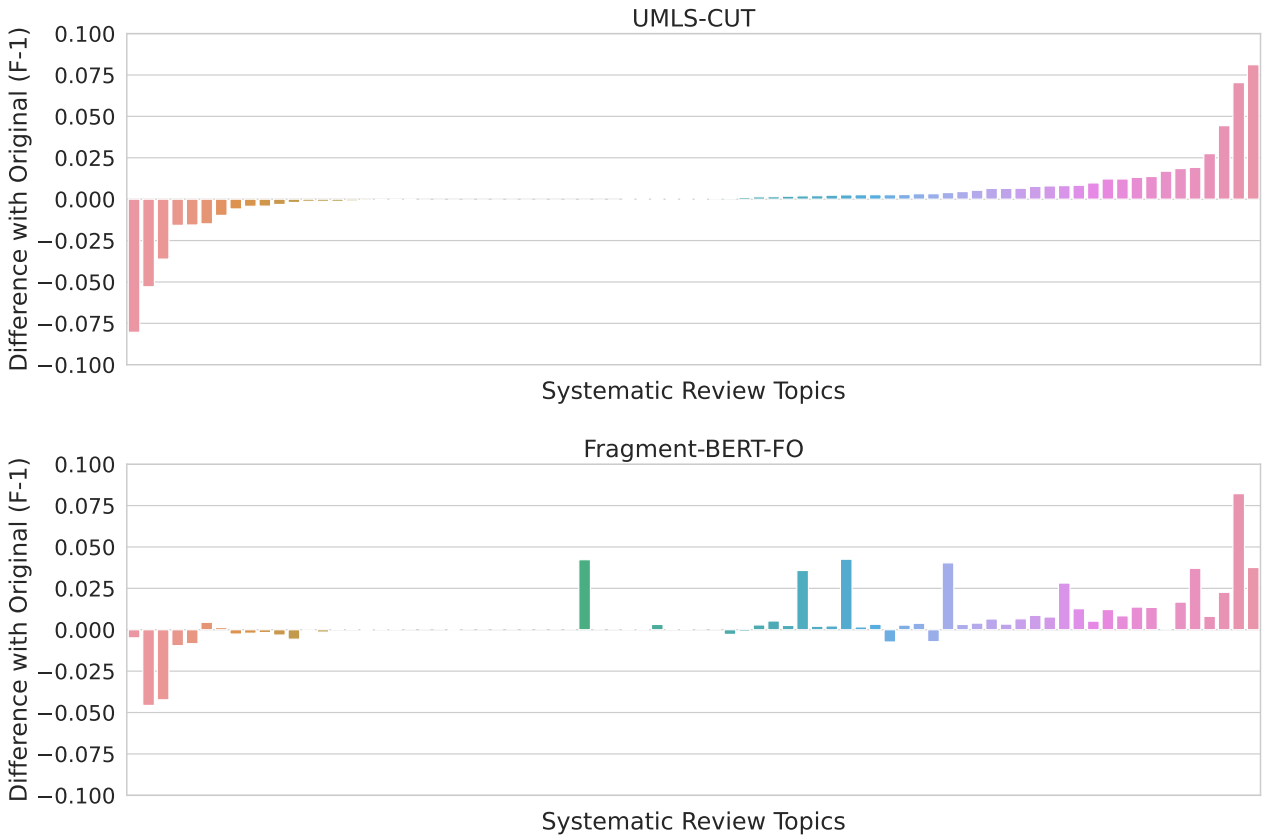


Figure 6.6: The plot shows systematic review topics versus original query effectiveness; each bar represents a topic. The y-axis represents the effectiveness difference between the query with the suggested MeSH terms and the original query. Effectiveness is measured using F1.

In terms of recall, BERT suggestions slightly outperform unrefined lexical methods and substantially exceed refined lexical methods.

As noted in Section 6.4.1, unrefined lexical methods are strong in recall, while refined methods excel in precision and F-measures. BERT-based suggestions combine the strengths of both: achieving recall comparable to unrefined methods and F1/F3 scores similar to refined methods. This suggests that BERT methods may offer a more balanced and effective approach to MeSH term suggestion compared to purely lexical approaches.

Search Stability

We analyse the topic-level stability of retrieval effectiveness for MeSH term suggestion methods. Stability refers to the variance in effectiveness across topics; lower variance implies higher stability.

Figure 6.6 shows the per-topic difference in F1 scores between the original query and the enriched query with suggested MeSH terms. We focus on the best-performing methods for each category: UMLS-CUT (U-CUT) for lexical and Fragment-BERT-FO (F-B-FO) for BERT-based methods. The plots show that for most topics, both U-CUT and F-B-FO outperform or match the effectiveness of the original query. However, a few topics show a drop in effectiveness. These cases may be due to inherent topic difficulty or residual data quality issues, such as undetected spelling errors in free text atomic clauses. Understanding the cause of these failures is a promising direction for future work.

Table 6.3: Jaccard index(Jaccard) values quantifying the overlap between the MeSH terms suggsted by the investigated methods and those in the original query, along with the average number (Num) of MeSH term suggested by each method. In the original queries, there were on average 4.1343 MeSH terms for 2017, 4.8333 for 2018, 4.4000 for 2019-dta, and 2.7547 for 2019-intervention. Lexical methods: *CUT* indicates cut-off ranks. BERT methods: *FO*, *SA*, *SO*, *LN* indicate different cut-off strategies. Two-tailed statistical significance (t-test, $p < 0.05$) with Bonferroni correction between ATM and the other methods is indicated by $*$.

	Dataset	2017		2018		2019-dta		2019-intervention	
	Method	Jaccard	Num	Jaccard	Num	Jaccard	Num	Jaccard	Num
Lexical Method	ATM	0.0999	5.54	0.2368	6.01	0.2117	5.15	0.2356	4.89
	ATM-CUT	0.1995*	2.42	0.1938	2.31	0.2004	2.05	0.2109	1.30
	MetaMap	0.2654*	4.69	0.2218	4.04	0.2163	4.80	0.2069	4.51
	MetaMap-CUT	0.2374*	2.31	0.1964	1.90	0.2241	2.35	0.1981	1.77
	UMLS	0.2243*	8.93	0.2235	7.97	0.1905	7.70	0.2405	7.57
	UMLS-CUT	0.2751*	1.90	0.2424	1.86	0.1986	2.20	0.2050	1.75
	Fusion	0.2165*	11.48	0.2160	10.94	0.1735	10.50	0.2212	9.74
	Fusion-CUT	0.2761*	2.78	0.2742	3.32	0.2508	3.10	0.2909	2.43
BERT Method	Atomic-BERT-FO	0.2532*	12.73	0.3105	12.26	0.1573	11.85	0.2252	13.62
	Semantic-BERT-FO	0.2370*	11.07	0.2963	10.69	0.1654	10.75	0.2219	11.53
	Fragment-BERT-FO	0.3455*	1.00	0.3812*	1.00	0.1681	1.00	0.2235	1.00
	Fragment-BERT-SA	0.2233*	**16.63**	0.2639	**16.49**	0.1790	**15.50**	0.2531	**17.23**
	Fragment-BERT-SO	**0.3921***	4.13	**0.4634***	4.83	0.2574	4.40	**0.3301**	2.75
	Fragment-BERT-LN	0.2780*	5.27	0.2689	3.78	**0.2667**	3.85	0.2415	3.85

6.4.2 Suggestion Effectiveness of Suggested MeSH Terms

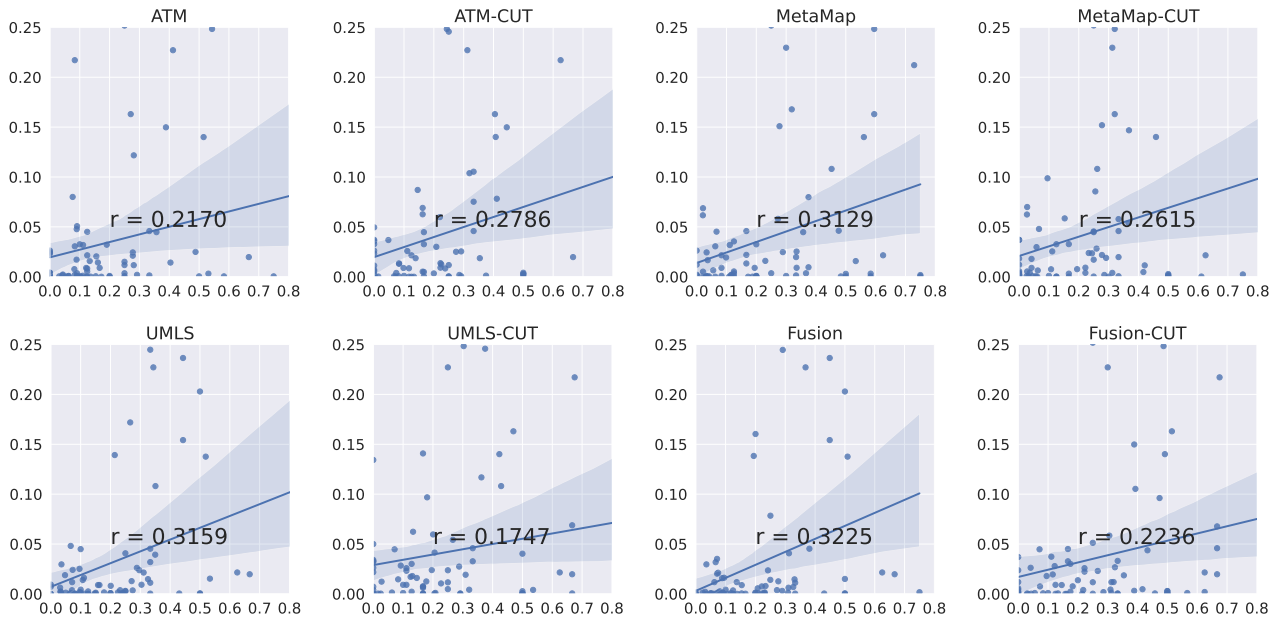


Figure 6.7: Correlation between search effectiveness (F1) and overlap with original MeSH terms (Jaccard index) for lexical-based methods.

Next, we study the overlap between the MeSH terms suggested by the evaluated methods and those included in the original queries. This is reported in Table 6.3, using the Jaccard index as the evaluation metric. One immediate observation is that the overlap for all BERT-based methods is

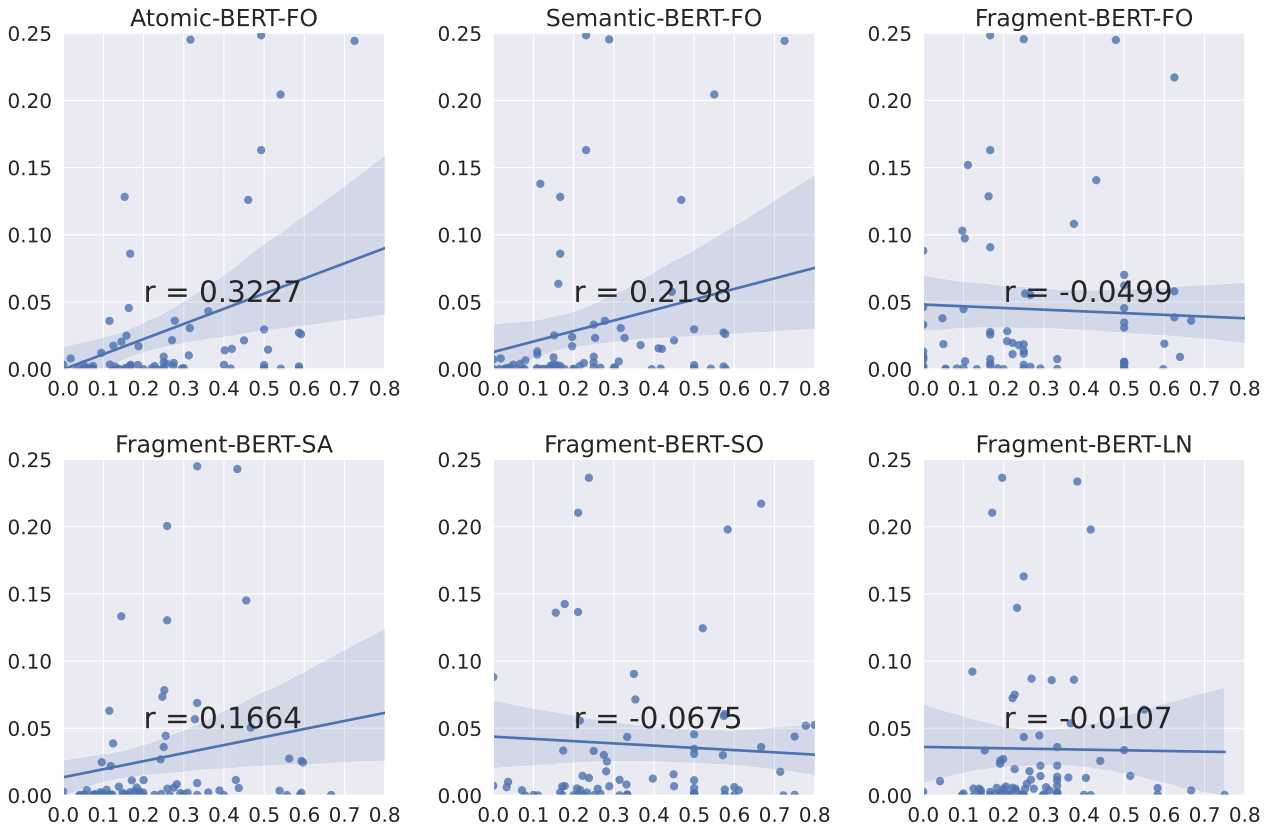


Figure 6.8: Correlation between search effectiveness (F1) and overlap with original MeSH terms (Jaccard index) for BERT-based methods.

considerably higher than that of lexical methods. In each dataset, the highest Jaccard index value is consistently achieved by a BERT-based suggestion method. Notably, applying the SO cut-off strategy to Fragment BERT yields the highest overlap across datasets, suggesting that BERT-based methods tend to recommend MeSH terms that align closely with those chosen by systematic reviewers.

The previous results in Table 6.2 showed that, overall, BERT methods outperform lexical methods in terms of retrieval effectiveness, and also exceed the performance of the original queries, although the differences are not statistically significant. Taken together with the results from Table 6.3, these findings suggest that BERT methods not only propose MeSH terms similar to those found in expert-crafted queries, but also identify terms that are more effective than those proposed by other methods.

To further investigate this, we analyse whether there is a correlation between retrieval effectiveness and the extent to which suggested MeSH terms match those in the original query. This would help determine whether the original MeSH terms should be treated as a gold standard. We again use the Jaccard index to measure the overlap with the original query and use F1 to measure the retrieval effectiveness of the resulting Boolean query.

Figure 6.7 (Lexical) Figure 6.8 (BERT-based) and presents the results of this correlation analysis. We find that, for all lexical methods, search effectiveness is only weakly correlated with overlap. In contrast, this is not the case for BERT methods. This indicates that MeSH terms used in the original queries are not necessarily optimal. In many cases, MeSH terms suggested by BERT but not present in the original query lead to higher retrieval effectiveness. This highlights the potential of BERT-based

models to identify novel but relevant MeSH terms that may be overlooked in expert-authored queries.

6.5 Ablation Studies

This section presents ablation experiments to better understand the components of the BERT-based MeSH term suggestion framework, which was shown to be more effective than lexical methods in previous evaluations. Given its high retrieval and suggestion effectiveness, we focus our ablations on how specific design choices within the BERT framework—namely, ranking representation strategies and cut-off strategies—influence overall performance.

6.5.1 Effect of BERT Ranking Representation Strategies

We compare different ranking representation strategies in BERT, including Atomic BERT, Semantic BERT, and Fragment BERT. All representations are evaluated under the same cut-off strategy to ensure fair comparison.

Among the three, Fragment BERT achieves the highest values for Precision, F1, and F3, though it has the lowest recall. This is expected, as Fragment BERT suggests only one MeSH term per fragment, leading to fewer retrieved documents and thus higher precision. This trade-off mirrors the trend observed in lexical methods, where adding more MeSH terms tends to improve recall but can reduce precision. Between Semantic BERT and Atomic BERT, the Semantic BERT strategy produces higher precision, while recall is slightly lower than that of Atomic BERT. When evaluated using F1 or F3, Semantic BERT consistently outperforms Atomic BERT. Therefore, we consider Semantic BERT a better choice than Atomic BERT for ranking representation.

6.5.2 Effect of Cut-off Strategies

We evaluate four cut-off strategies for selecting MeSH terms from the BERT rankings: First Only (FO), Same as Free text atomic clauses (SA), Same as Original (SO), and Linear (LN). Note that these cut-off strategies are only applicable to Fragment BERT. The FO strategy consistently yields the highest Precision, F1, and F3 across datasets, though it also results in the lowest recall. This again confirms the trade-off between precision and recall as the number of added MeSH terms increases.

Among the other three strategies, SO and LN consistently outperform SA. This suggests that the number of MeSH terms originally included by information specialists (SO), or those estimated by a learned linear function (LN), more accurately reflect effective term set sizes than simply using the number of free text atomic clauses (SA).

6.6 Case Study

Given the findings above, we next investigate the reasons behind highly effective or ineffective results. We select two representative topics from the CLEF TAR 2019-intervention dataset: CD009642, where

Table 6.4: Query fragments in different methods, For Lexical methods: *CUT* indicates cut-off ranks . For BERT suggestion method, *A-B*indicates Atomic BERT, *S-B* indicates Semantic BERT, *F-B* indicates Fragment BERT. In each BERT method, *FO*, *SA*, *SO*, *LN* indicates cut-off strategy used. For each fragment, bold text means MeSH term.

Topic	CD009642		
Fragments	Fragment 1		Fragment 2
ORIGINAL	**Lidocaine** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain** OR **Pain, Postoperative** OR **Postoperative Care** OR **Postoperative Complications** OR (post operative OR postoperative) AND (pain* OR recovery)
ATM	lidocain* OR Lignocain* OR Xylocain*	AND	**Pain** OR (post operative OR postoperative) AND (pain* OR recovery)
ATM-CUT	lidocain* OR Lignocain* OR Xylocain*	AND	**Pain** OR (post operative OR postoperative) AND (pain* OR recovery)
MetaMap	**Lidocaine** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain** OR (post operative OR postoperative) AND (pain* OR recovery)
MetaMap-CUT	**Lidocaine** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain** OR (post operative OR postoperative) AND (pain* OR recovery)
UMLS	**Lidocaine** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain** OR (post operative OR postoperative) AND (pain* OR recovery)
UMLS-CUT	**Lidocaine** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain** OR (post operative OR postoperative) AND (pain* OR recovery)
Fusion	**Lidocaine** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain** OR (post operative OR postoperative) AND (pain* OR recovery)
Fusion-CUT	**Lidocaine** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain** OR (post operative OR postoperative) AND (pain* OR recovery)
Atomic-BERT-FO	**Lidocaine** OR **Xylans** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Postoperative Care** OR **Pain** OR **Recovery of Function** OR (post operative OR postoperative) AND (pain* OR recovery)
Semantic-BERT-FO	**Lidocaine** OR **Xylans** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Postoperative Care** OR **Pain** OR **Recovery of Function** OR (post operative OR postoperative) AND (pain* OR recovery)
Fragment-BERT-FO	**Lidocaine** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain, Postoperative** OR (post operative OR postoperative) AND (pain* OR recovery)
Fragment-BERT-SA	**Lidocaine** OR **Procaine** OR **Xylans** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain, Postoperative** OR **Postoperative Care** OR **Postoperative Period** OR **Postoperative Complications** OR (post operative OR postoperative) AND (pain* OR recovery)
Fragment-BERT-SO	**Lidocaine** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain, Postoperative** OR **Postoperative Care** OR **Postoperative Period** OR **Postoperative Complications** OR (post operative OR postoperative) AND (pain* OR recovery)
Fragment-BERT-LN	**Lidocaine** OR **Procaine** OR **Xylans** OR lidocain* OR Lignocain* OR Xylocain*	AND	**Pain, Postoperative** OR **Postoperative Care** OR **Postoperative Period** OR (post operative OR postoperative) AND (pain* OR recovery)

suggestion methods outperform the original query, and CD004414, where they struggle to match its effectiveness. Query fragments and suggested MeSH terms for these topics are shown in Tables 6.4 and 6.5. Their corresponding search effectiveness scores are reported in Table 6.6.

First, we observe that both the suggested MeSH terms and the resulting search effectiveness are consistent across lexical methods. One exception is UMLS for topic CD004414, which suggests more MeSH terms than ATM and MetaMap, leading to a drop in effectiveness.

In contrast, MeSH terms suggested by BERT methods differ markedly from those produced by lexical methods and the original human-constructed queries. This divergence reveals patterns that provide insights into effective query formulation, addressing the question: what do humans miss or eschew that language models successfully capture, and what can we learn about manual query construction?

Table 6.5: Query fragments in different methods, For Lexical methods: *CUT* indicates cut-off ranks . For BERT suggestion method, *A-B*indicates Atomic BERT, *S-B* indicates Semantic BERT, *F-B* indicates Fragment BERT. In each BERT method, *FO*, *SA*, *SO*, *LN* indicates cut-off strategy used. For each fragment, bold text means MeSH term.

Topic	CD004414		
Fragments	Fragment 1		Fragment 2
ORIGINAL	**Hand** OR hand* OR finger* OR palm*	AND	**Hand Dermatoses** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
ATM	**Hand** OR **Fingers** OR hand* OR finger* OR palm*		**Fingers** OR **Eczema** OR **Hand** OR **Irritants** OR **Occupations** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
ATM-CUT	**Hand** OR hand* OR finger* OR palm*		**Fingers** OR **Eczema** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
MetaMap	**Hand** OR **Fingers** OR hand* OR finger* OR palm*		**Hand** OR **Fingers** OR **Eczema** OR **Occupations** OR **Irritants** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
MetaMap-CUT	**Hand** OR hand* OR finger* OR palm*		**Hand** OR **Fingers** OR **Eczema** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
UMLS	**Fingers** OR **Hand** OR **Palm Oil** OR **Computers, Handheld** OR hand* OR finger* OR palm*		**Eczema** OR **Fingers** OR **Hand** OR **Dermatitis, Atopic** OR **Kaposi Varicelliform Eruption** OR **Retirement** OR **Computers, Handheld** OR **Occupations** OR **Palm Oil** OR **Irritants** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
UMLS-CUT	**Fingers** OR hand* OR finger* OR palm*		**Eczema** OR **Fingers** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
Fusion	**Hand** OR **Fingers** OR **Palm Oil** OR **Computers, Handheld** OR hand* OR finger* OR palm*		**Eczema** OR **Fingers** OR **Hand** OR **Dermatitis, Atopic** OR **Occupations** OR **Kaposi Varicelliform Eruption** OR **Retirement** OR **Irritants** OR **Computers, Handheld** OR **Palm Oil** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
Fusion-CUT	**Hand** OR hand* OR finger* OR palm*		**Eczema** OR **Fingers** OR **Hand** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
Atomic-BERT-FO	**Hand** OR **Fingers** OR **Palm Oil** OR hand* OR finger* OR palm*		**Hand** OR **Eczema** OR **Occupations** OR **Dermatology** OR **Irritants** OR **Dermatitis, Contact** OR **Fingers** OR **Palm Oil** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
Semantic-BERT-FO	**Hand** OR **Fingers** OR **Palm Oil** OR hand* OR finger* OR palm*		**Dermatology** OR **Eczema** OR **Occupations** OR **Irritants** OR **Dermatitis, Contact** OR **Hand** OR **Fingers** OR **Palm Oil** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
Fragment-BERT-FO	**Hand** OR hand* OR finger* OR palm*		**Dermatitis, Contact** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
Fragment-BERT-SA	**Hand** OR **Fingers** OR **Palm Oil** OR hand* OR finger* OR palm*		**Dermatitis, Contact** OR **Dermatitis, Allergic Contact** OR **Hand** OR **Fingers** OR **Eczema** OR **Dermatitis, Atopic** OR **Patch Tests** OR **Skin Diseases** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
Fragment-BERT-SO	**Hand** OR hand* OR finger* OR palm*		**Dermatitis, Contact** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)
Fragment-BERT-LN	**Hand** OR **Fingers** OR **Palm Oil** OR hand* OR finger* OR palm*		**Dermatitis, Contact** OR **Dermatitis, Allergic Contact** OR **Hand** OR (dermat* OR eczema) AND (occupation* OR irritant* OR contact) AND (hand* OR finger* OR palm*)

BERT models capture both lexically similar terms and semantically related concepts that human query constructors often overlook. In the first fragment of topic CD009642, while all lexical methods suggest **Lidocaine**, BERT methods additionally suggest related drugs such as **Procaine** and **Xylans**—pharmacologically related local anesthetics that the original human query did not include. This suggests that when specifying interventions, searchers should consider the broader therapeutic class rather than limiting queries to explicitly named substances. In CD004414, BERT suggests **Patch Tests**, a standard diagnostic method for contact dermatitis that does not appear lexically in the fragment text. This demonstrates that standard diagnostic or assessment methods associated with conditions should be included even when not explicitly mentioned in the review protocol.

Another advantage of BERT methods is their ability to guarantee at least one MeSH term suggestion for each query fragment. Lexical methods, in contrast, rely on rule-based knowledge bases and may return no suggestions when exact matches are not found—for example, ATM produces no output in Fragment 1 of CD009642. Across our dataset, lexical methods failed to produce suggestions for approximately 18% of query fragments, whereas BERT methods provided suggestions for 100% of fragments. This coverage gap partially explains BERT's higher recall and suggests that when manually constructing queries, searchers should systematically verify that every key concept has at least one corresponding MeSH term.

While BERT's semantic flexibility is often beneficial, it may also lead to suboptimal suggestions. In CD004414, several BERT methods suggest **Palm Oil** and **Computers, Handheld** from the lexical match to "palm*", which are semantically incorrect interpretations when the query seeks hand anatomy, not palm oil or handheld devices. This highlights the importance of verifying MeSH term definitions to ensure semantic alignment with the intended concept, particularly for ambiguous terms. Additionally, maintaining clear conceptual boundaries between different query fragments (e.g., population vs. condition vs. intervention) helps avoid unintended Boolean interactions that reduce precision.

These observations suggest that while BERT-based suggestion tools can improve query quality, understanding why they succeed or fail provides valuable knowledge for improving human query formulation practices. Investigating how to guide BERT models to avoid semantic drift while preserving their generalisation capabilities remains an open direction for future work.

6.7 Summary of Findings

In this Chapter, we presented the task of MeSH term suggestion in the context of systematic review literature search—specifically, for enriching Boolean queries. This task builds upon previous works focused on the computational assistance in creating [4, 213, 214, 216] or refining [5, 9, 95, 212] Boolean queries for systematic reviews.

We introduced two classes of methods for MeSH term suggestion: lexical-based methods and BERT-based methods. We conducted a comprehensive evaluation of these methods, comparing the effectiveness of automatically suggested MeSH terms with those originally selected by information specialists. Our findings show that the MeSH terms chosen by specialists were often not the most

Table 6.6: Search effectiveness of the Boolean queries with the suggested MeSH terms evaluated by precision (P), F1, F3 and recall (R). For Lexical methods: *CUT* indicates cut-off ranks. BERT methods: *FO*, *SA*, *SO*, *LN* indicate different cut-off strategies.

Topic ID	CD009642				CD004414			
Method	P	F1	F3	R	P	F1	F3	R
ORIGINAL	0.0088	0.0175	0.0344	1.0000	0.0013	0.0026	0.0052	0.6875
ATM	0.0109	0.0215	0.0421	0.9194	0.0018	0.0035	0.0070	0.3125
ATM-CUT	0.0109	0.0215	0.0421	0.9194	0.0020	0.0040	0.0078	0.3125
MetaMap	0.0109	0.0215	0.0421	0.9194	0.0018	0.0035	0.0070	0.3125
MetaMap-CUT	0.0109	0.0215	0.0421	0.9194	0.0014	0.0027	0.0054	0.3125
UMLS	0.0109	0.0215	0.0421	0.9194	0.0013	0.0025	0.0050	0.3125
UMLS-CUT	0.0109	0.0215	0.0421	0.9194	0.0020	0.0040	0.0078	0.3125
Fusion	0.0109	0.0215	0.0421	0.9194	0.0018	0.0035	0.0069	0.3125
Fusion-CUT	0.0109	0.0215	0.0421	0.9194	0.0014	0.0027	0.0054	0.3125
Atomic-BERT-FO	0.0108	0.0214	0.0418	0.9194	0.0012	0.0024	0.0048	0.3125
Semantic-BERT-FO	0.0108	0.0214	0.0418	0.9194	0.0012	0.0024	0.0048	0.3125
Fragment-BERT-SA	0.0259	0.0504	0.0955	0.9194	0.0012	0.0024	0.0048	0.3125
Fragment-BERT-SO	0.0270	0.0525	0.0993	0.9194	0.0028	0.0055	0.0109	0.3125
Fragment-BERT-LN	0.0276	0.0536	0.1013	0.9194	0.0013	0.0026	0.0052	0.3125

effective, and that more effective terms can often be automatically suggested using our proposed methods.

We also observed that BERT-based methods generally achieved higher effectiveness than lexical-based methods. This advantage is likely due to BERT's ability to capture deeper semantic relationships, rather than relying solely on surface-level lexical similarity. These findings suggest a promising future direction in combining lexical and BERT-based approaches, such as through hybrid architectures or score fusion, to leverage the strengths of both. Such combinations have already shown strong potential in related tasks like ad-hoc retrieval [119, 136, 146, 255].

Finally, we believe there is significant untapped potential in making better use of MeSH entity structure. While our current methods treat MeSH terms as flat labels, the MeSH thesaurus provides much richer information that could guide more informed and accurate suggestions. For instance, the hierarchical structure of MeSH could help constrain suggestions to more relevant subfields, while MeSH entity types (such as descriptors or concepts) could be used to guide model training or filter outputs. In addition, external definitions (e.g., Wikipedia pages linked to each MeSH term) may provide useful context for learning more semantically aware representations.

Another promising direction is to integrate our MeSH suggestion framework with the Boolean query formulation work described in earlier chapters. There, we explored how LLMs can automatically generate full Boolean queries for systematic reviews. However, as noted, LLMs often introduce invalid or non-existent MeSH terms. The methods developed in this chapter could serve as a validation or augmentation layer—helping to correct, refine, or suggest appropriate MeSH terms within automatically generated queries. In this way, MeSH term suggestion can operate as a complementary tool to LLM-based generation, combining the strengths of large language models with domain-specific term

grounding.

Part III
Optimising Screening

In the previous part of this thesis, we explored how Boolean query formulation can be enhanced using LLMs and MeSH term suggestion methods to improve retrieval effectiveness. However, even with improved queries, the resulting document pool often includes a substantial number of irrelevant studies that must still be manually screened. To address this, prior research has developed methods such as screening prioritisation and relevance feedback for active learning. These techniques aim to identify relevant studies earlier in the screening pipeline, thereby reducing manual screening effort in multi-stage review workflows [6, 10, 50, 52, 84, 85, 128, 210, 257, 274].

In recent years, transformer-based approaches have shown strong performance in both ranking and classification tasks [61, 73, 164]. Encoder-based models such as BERT have demonstrated clear improvements in ranking over traditional lexical methods like BM25 [164, 255]. Meanwhile, decoder-based models, or large language models (LLMs), have exhibited impressive capabilities in following instructions and generating relevant outputs [199, 244, 245].

In this part, we investigate the potential of transformer-based models to optimise systematic review screening. We focus on two complementary strategies. First, we explore encoder-based models to enhance screening prioritisation, particularly through query-based ranking (Chapter 7). Second, we examine the use of generative LLMs to automate inclusion and exclusion decisions in a zero-shot setting, reducing the need for extensive labelled data (Chapter 8).

Chapter 7

Screening Prioritisation

As a way to reduce the effort, time, and cost involved in systematic review creation, the research community has explored the adoption of automation tools [170]. One key task where automation can be particularly beneficial is *screening prioritisation*, which involves ranking retrieved documents by their likelihood of relevance. Effective screening prioritisation enables researchers to complete downstream tasks earlier—such as full-text screening—by working in parallel, so that no relevant studies are present beyond the screening cut-off. Screening Prioritisation may also support early stopping by prioritising relevant studies near the top of the ranked list to the remaining screening activities.

A wide range of methods have been proposed to support screening prioritisation [6, 10, 50, 52, 84, 85, 128, 210, 257, 261, 274, 275]. Broadly, these methods differ in the source of supervision used for ranking. Some rely on explicit formulations of the information needs, such as review titles, Boolean queries, or review objectives. Others use interactive learning strategies—such as active learning or relevance feedback—based on partially labelled screening data. Notably, prior to this research, query-driven screening prioritisation has been dominated by traditional lexical ranking techniques, such as BM25 or query likelihood models [180, 280]. In this chapter, we focus exclusively on neural approaches to query-driven screening prioritisation, which remained largely underexplored before our research.

We investigate two types of query input for this task. First, we study the use of **systematic review titles** to drive ranking—an input commonly adopted in prior work [6, 10, 275]. We examine both zero-shot and fine-tuned encoder-based neural models, comparing their performance to traditional methods to quantify the improvements introduced by neural techniques. These experiments are described in Section 7.1.

Second, we address a key limitation of relying on review titles: in practice, systematic reviews often begin with only a working title composed of a few keywords [258], and finalised titles are not always available during early screening stages [57, 98]. Our experiments confirm that using working titles results in a substantial drop in prioritisation effectiveness compared to detailed final titles [83]. Therefore, we turn our attention to **Boolean queries**, which are a core component of systematic review

workflows and are typically available from the outset as the basis for initial retrieval. Unlike review titles, Boolean queries are carefully constructed or iteratively refined early in the review process [1]. We investigate whether these Boolean queries can be translated into ranking-friendly formats using large language models (LLMs) to support screening prioritisation. This second strategy is presented in Section 7.2.

7.1 Screening Prioritisation using Review Titles

In existing systematic review datasets such as CLEF TAR, the review title is commonly used as the query representation for ranking documents, following the standard setup adopted in prior work [6, 10, 275]. The title typically reflects the core intent of the review topic and is often serves as a practical and realistic query proxy for screening prioritisation, especially during the early stages of the review process.

Building on this established setup, we investigate the use of neural ranking models to prioritise relevant documents using only the review title as input. We evaluate both zero-shot and fine-tuned variants of pre-trained language models, comparing their performance against traditional ranking baselines as well as the best-performing CLEF TAR submissions. Additionally, we consider different document representation strategies—using either the title alone or a concatenation of title and abstract—to assess the impact of input length and content on ranking effectiveness.

7.1.1 Method

We adopt a neural document reranking framework based on cross-encoder architectures, considering two settings: (1) a zero-shot setting, where pre-trained language models are applied directly without task-specific fine-tuning, and (2) a fine-tuned setting, where models are further trained on relevance labels derived from existing systematic review datasets. In both settings, we examine different document representation strategies and assess their effect on ranking effectiveness.

Following the monoBERT cross-encoder design [164], each query-document pair is jointly encoded to compute a scalar relevance score. Given a query q (the systematic review title) and a candidate document d, we concatenate them into a single input sequence:

$$\text{Input} = \texttt{[CLS]}\ q\ \texttt{[SEP]}\ d\ \texttt{[SEP]}$$

This input is passed through a BERT-based encoder, and the final hidden state corresponding to the `[CLS]` token is used to compute a relevance score via a linear projection.

We explore two model variants:

- **Zero-shot**: We evaluate five pre-trained models without task-specific fine-tuning: BERT base [61], BERT base fine-tuned on MS MARCO [74], and three domain-specific models trained on biomedical corpora—BioBERT [130], PubMedBERT [87], and BlueBERT [176].

[1] Boolean queries do not enforce ranking of citations beyond the document coordination level. Matching is typically based on Boolean term overlap, not a principled ranking of relevance.

- **Fine-tuned**: We further fine-tune a subset of the better-performing zero-shot models using the training splits from CLEF TAR. Fine-tuning is performed using a localised contrastive loss applied to input triples of the form $\langle q, d^+, \text{set}(d^-)\rangle$, where d^+ is a relevant document and d^- is a set of sampled non-relevant documents. We fine-tune BERT base, BERT+MSMARCO, and BioBERT using the method of Gao et al. [74], with default hyperparameters.

To represent the document d, we concatenate the document title and abstract, separated by a `[SEP]` token:

$$d = \texttt{title [SEP] abstract}$$

This composite representation allows the model to utilise both concise and contextual information from the document, while respecting BERT's maximum input length of 512 tokens. Inputs that exceed this limit are truncated from the end of the abstract.

7.1.2 Experimental Setup

This section outlines the datasets, baseline methods, and fine-tuning details used to evaluate neural rankers for screening prioritisation using review titles.

Dataset and Evaluation

We evaluate our models using three CLEF TAR datasets from 2017 to 2019 [113, 114, 115]. Each dataset comprises systematic review topics with associated document pools retrieved via Boolean queries and annotated with relevance labels.

CLEF TAR 2017 and 2018 focus exclusively on Diagnostic Test Accuracy (DTA) reviews, comprising 20/30 and 50/30 training/testing topics, respectively. The 2019 dataset includes DTA, intervention, prognosis, and qualitative topics; we use only the DTA and intervention subsets due to the lack of training data for the other review types.

For each topic, we use the review title as the query and retrieve document metadata (title and abstract) via PubMed using their PMIDs. We evaluate models using abstract-level relevance labels and report standard CLEF TAR metrics: Mean Average Precision (MAP), rank of the last relevant document (Last_Rel), Recall@1, 5, 10, 20%, and Work Saved over Sampling (WSS) at 95% and 100% [45]. All metrics are computed using the official CLEF TAR 2018 evaluation script.

Baseline Methods

To contextualise the performance of neural rankers, we compare against traditional term-based baselines. These include the Query Likelihood Model (QLM) with Jelinek-Mercer smoothing [278] and BM25, implemented using the Gensim toolkit [193].

We also compare against the strongest known CLEF TAR runs for each dataset, obtained from official campaign submissions. These runs serve as strong baselines or state-of-the-art references in

the literature. Only runs that return full document rankings (i.e., no early cutoffs) are included for fair comparison. Specifically, we use:

- CLEF TAR 2017: `waterloo.A-rank-normal` [50]
- CLEF TAR 2018: `UWB` [51]
- CLEF TAR 2019 DTA: `Sheffield/DTA/DTA_sheffield-Odds_Ratio` [8]
- CLEF TAR 2019 Intervention: `Sheffield/DTA/DTA_sheffield-Log_Likellihood` [6]

Since CLEF TAR participants were allowed to use *iterative ranking* (i.e., relevance feedback during evaluation), we restrict our comparison to submissions that do not rely on such feedback, where this information is available. Specifically, for CLEF TAR 2017 and 2018, we include non-feedback runs identified via task overview papers:

- CLEF TAR 2017: `sheffield.run4` [6]
- CLEF TAR 2018: `shef-general` [10]

For CLEF TAR 2019, where feedback usage could not be determined, we conservatively compare against the top-performing run, acknowledging that it may include feedback.

Fine-tuning Details

For all fine-tuning experiments, we adopt a contrastive learning setup based on positive and negative document triples. Each training instance consists of one relevant document (judged at the abstract level) and nine non-relevant documents, sampled per topic. We use a batch size of 3 and fine-tune all models using the Reranker toolkit [74].

Models are trained for 100 epochs, with checkpoints saved every 100 steps. We report results using the final checkpoint and also monitor convergence across earlier checkpoints. During inference, relevance is computed by concatenating the query (e.g., reformulated Boolean query) with the document representation (title or title+abstract), and feeding the pair into the trained model.

7.1.3 Main Results

Zero-shot Neural Rankers

Table 7.1 presents the effectiveness of various neural models under the zero-shot setting, alongside traditional IR baselines. BERT-M refers to the BERT base model fine-tuned on the MS MARCO dataset.[2]

Across all evaluation metrics and datasets, the traditional baselines—BM25 and QLM—generally outperform the zero-shot neural rankers. Nonetheless, there are a few notable exceptions:

[2]We consider BERT-M as zero-shot since the fine-tuning was performed on a different dataset (MS MARCO) and for a different task (ad hoc web search), which differs from the screening prioritisation task considered here.

Table 7.1: Results obtained using pre-trained language models in a zero-shot setting. Statistical significant differences (Student's two-tailed paired t-test with Bonferonni correction, $p < 0.05$) between BERT and all other methods are indicated by †.

Dataset	Method	LastRel	MAP	Recall (%)				WSS (%)	
				1	5	10	20	95	100
2017	BM25	2999.70	0.1497†	0.0931†	0.2717†	0.3851†	0.5737†	0.3518	0.2520
	QLM	2999.53	**0.1721**†	**0.1071**†	**0.2849**†	**0.4067**†	**0.6340**†	0.3588	0.2588
	BERT	**2898.03**	0.0917	0.0119	0.1052	0.2594	0.4386	0.3565	**0.3082**
	BERT-M	3161.57	0.1247	0.0220	0.1324	0.2682	0.5265	**0.3627**	0.2606
	BioBERT	3358.27	0.0998	0.0301	0.1801	0.2922	0.4372	0.2373†	0.1836†
	BlueBERT	3824.47†	0.0365†	0.0001†	0.0113†	0.0349†	0.0935†	0.0233†	0.0407†
	PubMedBERT	3628.50†	0.0670	0.0156	0.0667	0.1433†	0.2717†	0.1205†	0.1018†
2018	BM25	6095.20	**0.1683**†	**0.0972**†	**0.2947**†	0.4444†	0.6441†	0.4202	0.2336
	QLM	5956.97	0.1660†	0.0896†	0.2904†	**0.4544**†	**0.6456**†	0.4322	0.2540
	BERT	6158.23	0.1249	0.0222	0.1348	0.2960	0.5422	0.4037	**0.2622**
	BERT-M	**5805.47**	0.1398	0.0188	0.1465	0.2997	0.5894	**0.4476**	0.2567
	BioBERT	6696.93	0.0881	0.0236	0.1526	0.2580	0.4059†	0.1499†	0.0836†
	BlueBERT	7204.37†	0.0329†	0.0014†	0.0093†	0.0198†	0.0493†	0.0017†	0.0193†
	PubMedBERT	6955.83†	0.0700†	0.0112	0.0636†	0.1593†	0.3138†	0.1424†	0.0883†
2019-dta	BM25	2722.75	0.1185	0.0479	0.2129	**0.3290**	0.5276	0.3138	0.2080
	QLM	**2318.25**	**0.1223**	**0.0644**	**0.2164**	0.3270	**0.5335**	**0.3470**	**0.2477**
	BERT	2513.88	0.0922	0.0244	0.1318	0.2381	0.3906	0.2577	0.2095
	BERT-M	3233.75	0.0955	0.0105	0.0792	0.1979	0.3793	0.2629	0.1232
	BioBERT	3264.00	0.0810	0.0160	0.1290	0.2294	0.3365	0.1370	0.0950
	BlueBERT	3771.00	0.0688	0.0010	0.0256	0.0526	0.1050	0.0227	0.0160
	PubMedBERT	3330.25	0.1044	0.0335	0.1226	0.2144	0.3119	0.2016	0.0979
2019-int.	BM25	1715.60	0.2112	0.0968	**0.3053**	**0.3989**	**0.5542**	0.3510	0.2955
	QLM	1724.05	**0.2123**	**0.0981**	0.2793	0.3851	0.5110	0.3397	0.2939
	BERT	**1398.55**	0.1603	0.0536	0.2104	0.3282	0.5041	**0.3624**	**0.3330**
	BERT-M	1836.20	0.1769	0.0384	0.1951	0.3545	0.5268	0.3228	0.2663
	BioBERT	1832.85	0.1463	0.0530	0.1346	0.1982	0.3074†	0.1585†	0.1631†
	BlueBERT	2057.00	0.0462	0.0063	0.0275†	0.0513†	0.1066†	0.0083†	0.0361†
	PubMedBERT	1974.25	0.0780	0.0124	0.0502	0.0905†	0.2748†	0.1207†	0.0944†

- The BERT model outperforms the baselines in WSS@100 for all datasets except 2019-dta, in WSS@95 for 2019-intervention, and in Last_Rel for both 2017 and 2019-intervention.
- The BERT-M model yields competitive results in WSS@95 for 2017 and 2018, and in Last_Rel for 2018.

When comparing the different zero-shot neural models, BERT and BERT-M consistently perform better than the domain-specific biomedical variants. Among the latter, BioBERT emerges as the most effective, albeit still underperforming relative to BERT and BERT-M. Notably, BERT-M, despite being fine-tuned on a large-scale web search corpus, does not show significant improvements over BERT, suggesting limited transferability from web search to the screening prioritisation domain.

Overall, these results indicate that zero-shot neural rankers are not yet competitive with traditional retrieval methods for this task. While pre-trained language models capture general semantic signals, they appear to lack the task-specific alignment required for high-quality prioritisation in the systematic review setting.

Table 7.2: Results obtained when using pre-trained language models in the fine-tuned setting. Statistical significant differences (Student's two-tailed paired t-test with Bonferonni correction, $p < 0.05$) between BioBERT-Tuned and all other methods are indicated by †.

Dataset	Method	LastRel	MAP	Recall (%)				WSS (%)	
				1	5	10	20	95	100
2017	BEST-No-Feedback	2382.47	0.2179†	0.1308	0.3325†	0.4993†	0.6877†	0.4880†	0.3946†
	BEST-Iterative	1469.40	**0.3183**	0.1707	**0.5434**	**0.7322**	**0.8863**	**0.7009**	**0.6106**
	BM25	2999.70†	0.1497†	0.0931†	0.2717†	0.3851†	0.5737†	0.3518†	0.2520†
	QLM	2999.53†	0.1721†	0.1071†	0.2849†	0.4067†	0.6340†	0.3588†	0.2588†
	BERT-Tuned	2418.83†	0.2273†	0.1066†	0.4088†	0.5888†	0.7789	0.5435	0.4340†
	BERT-M-Tuned	2475.20†	0.2770	0.1419	0.3727†	0.5843†	0.7707	0.5234†	0.4187†
	BioBERT-Tuned	**1461.67**	0.3078	**0.1845**	0.4903	0.6816	0.8355	0.6530	0.5913
2018	BEST-No-Feedback	5519.20	0.2584†	0.1287†	0.3827†	0.5449†	0.7295†	0.5520†	0.4314†
	BEST-Iterative	**2655.00**	0.3776	0.1854	0.5940	**0.7696**	**0.9149**	**0.7558**	**0.6104**
	BM25	6095.20†	0.1683†	0.0972†	0.2947†	0.4444†	0.6441†	0.4202†	0.2336†
	QLM	5956.97†	0.1660†	0.0896†	0.2904†	0.4544†	0.6456†	0.4322†	0.2540†
	BERT-Tuned	5581.77	0.3467†	0.1981†	0.5028†	0.6772†	0.8196†	0.6188†	0.4121†
	BERT-M-Tuned	5185.50	0.3387†	0.1934†	0.4833†	0.6515†	0.8265†	0.6559	0.4815†
	BioBERT-Tuned	4108.40	**0.4444**	**0.2768**	**0.5975**	0.7574	0.8946	0.7194	0.6103
2019-dta	BEST	2183.50	0.2477	0.1685	0.4391	0.5940	0.7421	0.4899	0.3470
	BM25	2722.75	0.1185†	0.0479	0.2129	0.3290†	0.5276†	0.3138†	0.2080†
	QLM	2318.25	0.1223†	0.0644	0.2164	0.3270†	0.5335†	0.3470†	0.2477†
	BERT-Tuned	1399.38	0.2234	0.1580	0.4390	0.6013	0.7620	0.5870	0.4600
	BERT-M-Tuned	1178.00	0.2535	0.2049	0.4474	0.5904	0.7536	0.6151	0.4997
	BioBERT-Tuned	**852.75**	**0.3177**	**0.2604**	**0.4998**	**0.6710**	**0.8171**	**0.6857**	**0.5845**
2019-int.	BEST	1132.00	0.2929†	0.1655	0.4192†	0.5424†	0.7225†	0.4582†	0.3808†
	BM25	1715.60	0.2112†	0.0968†	0.3053†	0.3989†	0.5542†	0.3510†	0.2955†
	QLM	1724.05	0.2123†	0.0981†	0.2793†	0.3851†	0.5110†	0.3397†	0.2939†
	BERT-Tuned	1374.30	0.2808†	0.1646†	0.3736†	0.5274†	0.6586†	0.3629†	0.3011†
	BERT-M-Tuned	1571.25	0.3343†	0.1614	0.4021†	0.5649†	0.7061†	0.4461†	0.3623†
	BioBERT-Tuned	**706.85**	**0.4559**	**0.2155**	**0.5805**	**0.7374**	**0.8417**	**0.6462**	**0.5794**

Fine-tuned Neural Rankers

Table 7.2 presents the results of fine-tuned neural rankers using the title and abstract (TiAb) representation. We focus on the three best-performing models from the zero-shot experiments—BERT, BERT-M, and BioBERT—as candidates for fine-tuning.

Fine-tuning leads to substantial performance improvements across all three models compared to their zero-shot counterparts in Table 7.1. This is particularly noteworthy given the limited amount of training data available in the CLEF TAR datasets, suggesting that even small-scale supervision can enable pre-trained models to better adapt to the screening prioritisation task.

Among the three models, BioBERT emerges as the most effective after fine-tuning, frequently outperforming both BERT and BERT-M, despite the latter two showing stronger performance in the zero-shot setting. This result underscores the benefit of domain-specific pre-training when combined with task-specific fine-tuning.

Fine-tuned neural rankers also consistently outperform traditional IR baselines (BM25 and QLM)

across all evaluation metrics and datasets. This reinforces the importance of task-adaptive training for neural models in this context, even when only limited relevance labels are available.

We also compare against the best-performing non-feedback runs submitted to the respective CLEF TAR tasks.[3] Fine-tuned models outperform these no-feedback CLEF runs in nearly all cases. Furthermore, BioBERT achieves higher effectiveness than the best feedback-based (iterative) runs from CLEF 2017 and 2018 on Recall@1%, Recall@5%, Average Precision, and WSS@100.

Interestingly, the performance gains of fine-tuned neural rankers are especially pronounced on shallow metrics such as Recall@1% and Recall@5%, while the advantage narrows on deeper metrics. This is consistent with the nature of iterative feedback-based systems, which tend to improve over time but may initially perform poorly in early iterations. In contrast, the fine-tuned neural rankers are highly effective at the top of the ranking from the outset, suggesting their potential integration into interactive or active learning loops to further enhance overall screening efficiency.

7.1.4 Ablation Studies

To better understand the performance characteristics of our fine-tuned neural rankers, we conduct a series of ablation studies. Specifically, we examine: (1) the convergence behaviour of the models during fine-tuning, (2) the impact of different document representations (Title vs. Title+Abstract (TiAB)), and (3) per-topic variation in effectiveness through a comparative analysis with state-of-the-art feedback-based systems. These analyses provide deeper insights into training dynamics, representation choices, and topic-specific strengths and weaknesses, and further clarify the conditions under which fine-tuned neural models excel in the screening prioritisation task.

Model Convergence

We analyse the convergence behaviour of the neural rankers during fine-tuning. Figure 7.1 plots the AP on the test sets across successive training checkpoints. The figure shows that model performance tends to plateau after approximately 100 epochs. This observation is supported by a statistical analysis, which found no significant differences in AP beyond this point (paired two-tailed t-test with Bonferroni correction, $p < 0.05$).

Importantly, the results reported in Table 7.2 correspond to the final checkpoint (after 100 epochs) for each model, but these are not necessarily the points of peak test performance. In several cases, earlier checkpoints achieved slightly higher AP. This suggests that, with an effective early stopping mechanism, even better results could be obtained. A standard approach to identifying the optimal stopping point is to use a held-out validation set. However, the CLEF TAR datasets do not provide validation splits, and carving out a portion of the limited training data for validation would significantly reduce the amount of supervision available for both training and validation, potentially leading to unreliable outcomes. As such, we opted to train on the full training set and report the final checkpoint results for consistency and fairness in comparison.

[3] For CLEF 2019, it is not possible to determine whether the selected runs used feedback.

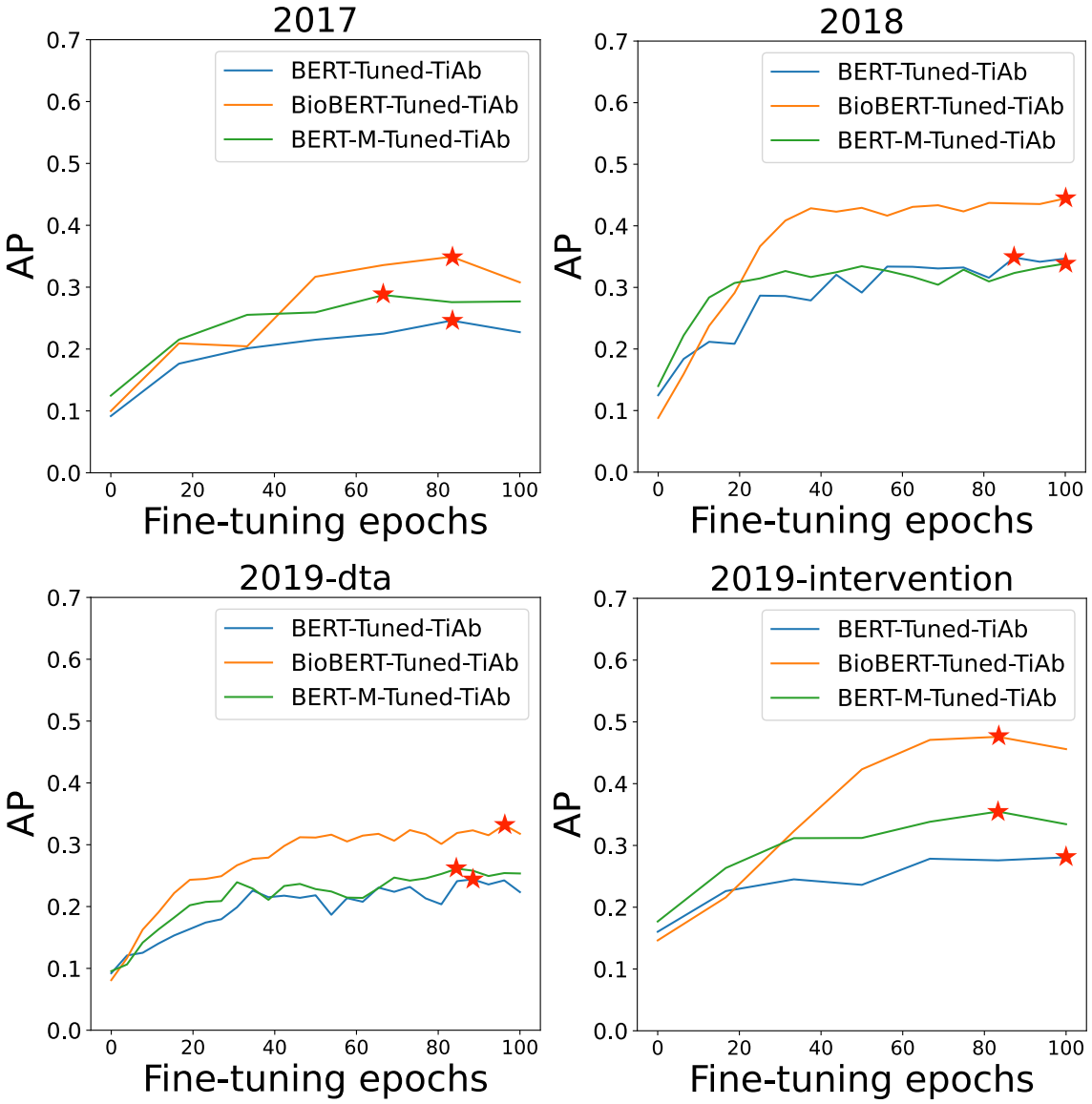


Figure 7.1: Convergence of neural rankers during fine tuning. The y-axis reports AP measured on the test set, while the x-axis corresponds to subsequent fine-tuning steps. AP measurements are taken every 100 training steps. For each neural ranker, the checkpoint with the highest test AP is marked with red ⋆.

Document Representation

We investigate the impact of document representation on the effectiveness of fine-tuned neural rankers. Specifically, we compare two input strategies: using the document title alone (*Title*) and using the title concatenated with the abstract (*TiAb*).[4] Zero-shot models are excluded from this analysis, as they did not demonstrate competitive performance in earlier experiments.

Table 7.3 presents the comparison results. Across all CLEF TAR datasets and evaluation metrics, using the *TiAb* representation consistently outperforms the *Title*-only variant, regardless of the underlying pre-trained language model. The only exception is Recall@1% for the BERT ranker on CLEF TAR 2017, where the difference is negligible. These findings suggest that abstracts provide critical additional context beyond what is captured by titles alone. This highlights the importance of richer

[4]TiAb results here are consistent with those reported in Table 7.2.

Table 7.3: Comparison of using *Title* vs *TiAb* as document representation. Statistical significance (Student's two-tailed paired t-test $p < 0.05$) between representation of two models is indicated by †.

Dataset	Method	LastRel	MAP	Recall (%)				WSS (%)	
				1	5	10	20	95	100
2017	BERT-Tuned-TiAb	**2418.83**	**0.2273**	0.1066	**0.4088**	**0.5888**	**0.7789**	**0.5435**	**0.4340**
	BERT-Tuned-Title	3540.67†	0.2037	**0.1119**	0.3429	0.4819†	0.6468†	0.2626†	0.1700†
	BERT-M-Tuned-TiAb	**2475.20**	**0.2770**	**0.1419**	**0.3727**	**0.5843**	**0.7707**	**0.5234**	**0.4187**
	BERT-M-Tuned-Title	2953.47	0.2299†	0.1082	0.3388	0.5084†	0.7027	0.4361†	0.3043†
	BioBERT-Tuned-TiAb	**1461.67**	**0.3078**	**0.1845**	**0.4903**	**0.6816**	**0.8355**	**0.6530**	**0.5913**
	BioBERT-Tuned-Title	2072.10†	0.2789†	0.1565	0.4176†	0.5937†	0.7919	0.5876†	0.4655†
2018	BERT-Tuned-TiAb	**5581.77**	**0.3467**	**0.1981**	**0.5028**	**0.6772**	**0.8196**	**0.6188**	**0.4121**
	BERT-Tuned-Title	6087.97	0.2652†	0.1303†	0.4054†	0.5667†	0.7328†	0.4898†	0.3004†
	BERT-M-Tuned-TiAb	**5185.50**	**0.3387**	**0.1934**	**0.4833**	**0.6515**	**0.8265**	**0.6559**	**0.4815**
	BERT-M-Tuned-Title	5757.77†	0.2661†	0.1466†	0.3967†	0.5738†	0.7523†	0.5333†	0.3356†
	BioBERT-Tuned-TiAb	**4108.40**	**0.4444**	**0.2768**	**0.5975**	**0.7574**	**0.8946**	**0.7194**	**0.6103**
	BioBERT-Tuned-Title	4928.00†	0.3557†	0.1862†	0.5202†	0.6952†	0.8524†	0.6527†	0.4813†
2019-dta	BERT-Tuned-TiAb	**1399.38**	**0.2234**	**0.1580**	**0.4390**	**0.6013**	**0.7620**	**0.5870**	**0.4600**
	BERT-Tuned-Title	1597.25	0.1851	0.1436	0.3802	0.4993†	0.6708†	0.5247	0.3892
	BERT-M-Tuned-TiAb	**1178.00**	**0.2535**	**0.2049**	**0.4474**	**0.5904**	**0.7536**	**0.6151**	**0.4997**
	BERT-M-Tuned-Title	1798.63	0.1858†	0.1195	0.3250†	0.5229	0.6831	0.5136	0.3729
	BioBERT-Tuned-TiAb	**852.75**	**0.3177**	**0.2604**	**0.4998**	**0.6710**	**0.8171**	**0.6857**	**0.5845**
	BioBERT-Tuned-Title	1091.00	0.2566†	0.1834	0.4761	0.6308	0.7876	0.6244	0.5095
2019-int.	BERT-Tuned-TiAb	**1374.30**	**0.2808**	**0.1646**	**0.3736**	**0.5274**	**0.6586**	**0.3629**	**0.3011**
	BERT-Tuned-Title	1883.85	0.2476	0.0953†	0.3231	0.4609	0.6134	0.3103	0.2763
	BERT-M-Tuned-TiAb	**1571.25**	**0.3343**	**0.1614**	**0.4021**	**0.5649**	**0.7061**	**0.4461**	**0.3623**
	BERT-M-Tuned-Title	1653.65	0.2702†	0.1045†	0.3488	0.5176	0.6891	0.3924	0.3288
	BioBERT-Tuned-TiAb	**706.85**	**0.4559**	**0.2155**	**0.5805**	**0.7374**	**0.8417**	**0.6462**	**0.5794**
	BioBERT-Tuned-Title	1145.50†	0.3677†	0.1694	0.4868†	0.6447†	0.7572†	0.5222†	0.4622

document representations in neural ranking models for the screening prioritisation task.

Topic-by-Topic Analysis

Finally, we conduct a topic-by-topic comparison between the fine-tuned BioBERT ranker and the best iterative (feedback-based) runs submitted to the CLEF TAR tasks to complement the aggregate findings reported in Section 7.1.3 and Table 7.2. While average effectiveness metrics suggest that BioBERT performs competitively with the best iterative systems, these averages are computed over relatively small topic sets, 30 topics each in CLEF TAR 2017 and 2018, and may be influenced by outliers. To investigate this, we perform a detailed per-topic analysis, visualised in the gain-loss plot shown in Figure 7.2.

The results reveal substantial variation across topics. In nearly half of the cases, BioBERT achieves higher effectiveness than the feedback-based systems, despite not using any relevance feedback. This suggests that, for many topics, feedback may not be strictly necessary for achieving strong performance. Iterative methods typically depend on surfacing relevant documents early in the ranking, which makes them sensitive to shallow metrics like Recall@1% and Precision@5. In contrast, BioBERT demonstrates strong early-rank performance even in the absence of feedback. This is evident

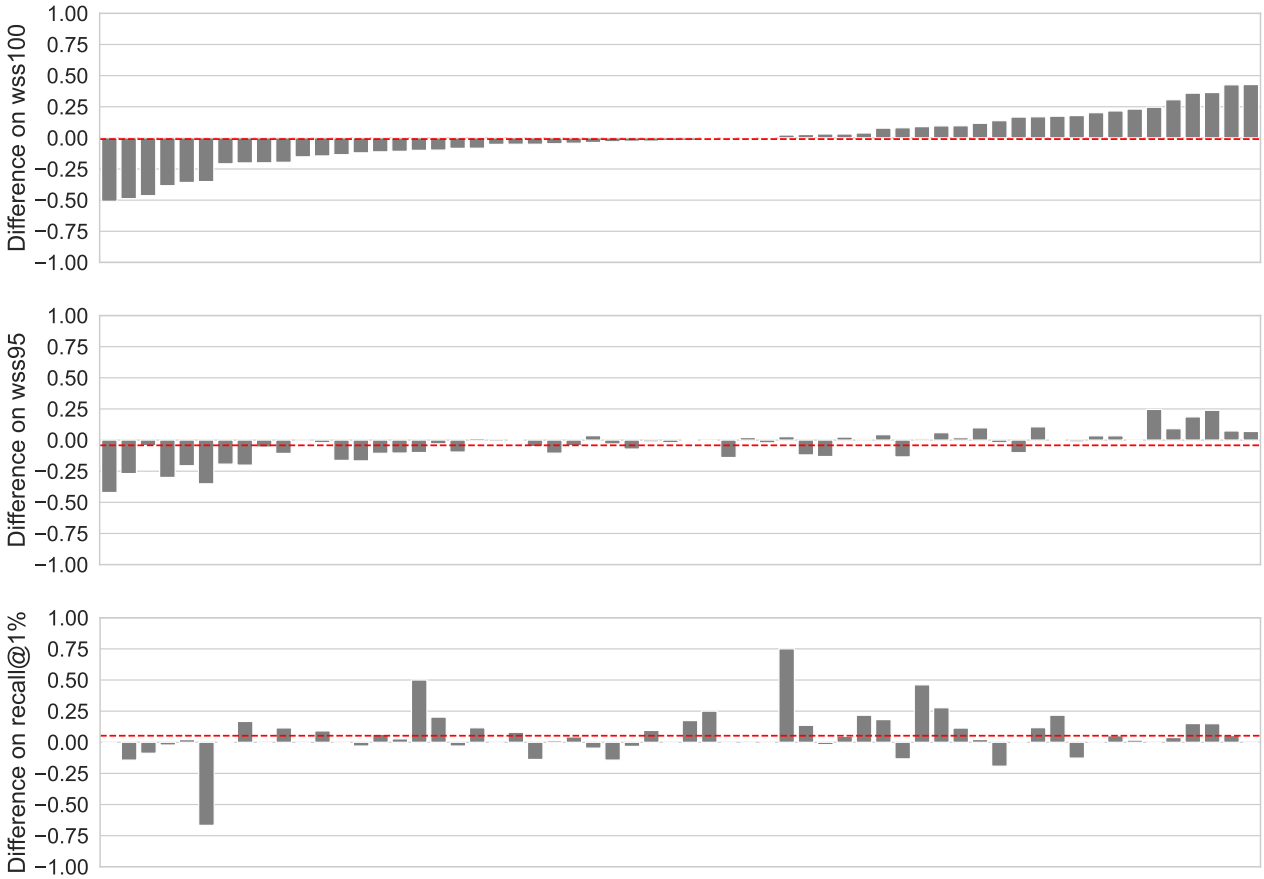


Figure 7.2: Per-topic effectiveness difference between BioBERT fine-tuned model and the best iterative runs for in CLEF TAR 2017 and 2018 (i.e., best active learning result of the year).

from its higher Recall@1% scores on the majority of topics, while WSS@100 (a deeper metric) shows more comparable results across systems.

These findings suggest that neural rankers such as BioBERT can provide high-quality initial rankings and may serve as effective components in hybrid or interactive relevance feedback pipelines. Their ability to produce strong early signals could enhance iterative methods by offering better starting points for learning from user feedback.

7.1.5 Summary of Screening Prioritisation Using Review Titles

This section examined the effectiveness of pre-trained neural rankers for screening prioritisation when using systematic review titles as queries, a common setting in previous screening prioritisation works. We focused on non-iterative approaches and evaluated neural rankers in both zero-shot and fine-tuned configurations.

Our results show that zero-shot neural rankers perform poorly in this setting. However, fine-tuned models—trained on limited relevance data—significantly outperform traditional non-iterative baselines and match or exceed the best iterative methods submitted to CLEF TAR. We also found that representing documents using both title and abstract (TiAb) was essential for strong performance, highlighting the importance of document-side richness even when the query is minimal.

While iterative feedback systems perform well on certain topics, fine-tuned neural rankers typically provide better early-rank effectiveness across most topics. This suggests their suitability for integration into interactive frameworks, where early retrieval quality is critical. Recent work has explored iterative relevance feedback with pre-trained models but suffers from high latency [275]. Alternative feedback methods—either based on input text (with token length constraints) or dense representations (more scalable) [134, 172, 252]—may offer more practical paths forward. Future work should explore adapting these strategies to cross-encoder rankers in the context of screening prioritisation.

In summary, fine-tuned neural rankers using only review titles as queries show strong potential for improving systematic review efficiency and effectiveness, particularly when paired with rich document representations. However, this strategy assumes access to a well-defined and finalised review title—an assumption that often does not hold in real-world review workflows. Early-stage reviews typically begin with only a provisional working title composed of a few keywords [258], and finalised titles may not yet be available [57, 98]. Our results confirm that relying solely on such titles leads to suboptimal performance. Therefore, in the next section, we turn our attention to Boolean queries—an essential and required component of systematic review protocols—as a more robust and informative query source for supporting screening prioritisation.

7.2 Screening Prioritisation Using Boolean Queries

We now explore the use of Boolean queries for screening prioritisation, leveraging the structured information typically available at the outset of systematic reviews. Although Boolean queries are routinely used during the prior retrieval phase, their potential as input to learning-based ranking models has received limited attention [30, 118, 128, 129, 260, 261].

Despite being expressive and detailed, Boolean queries are not directly compatible with BERT-style cross-encoder rankers. This is due to their rigid syntax, reliance on formal operators, and practical constraints related to input length—complex Boolean queries can easily exceed the 512-token limit imposed by these models. To overcome these limitations, we propose transforming Boolean queries into natural language using instruction-following large language models (LLMs). Reformulating structured queries in this way preserves their original intent while enabling integration with neural rankers designed for natural language input.

In this section, we used the only two open-source instruction-following LLMs available at the time of this research—OpenAI's ChatGPT [79][5] and Stanford's Alpaca [239][6]—for the task of rewriting Boolean queries into natural language. Figure 7.3 provides an overview of the proposed pipeline.

7.2.1 Method

Our approach consists of two main stages: (1) transforming Boolean queries into natural language queries using large language models (LLMs), and (2) ranking candidate documents using a cross-

[5]We used OpenAI's GPT-3.5-turbo API with a maximum context length of 4,097 tokens.

[6]The model was fine-tuned using the original Stanford Alpaca setup.

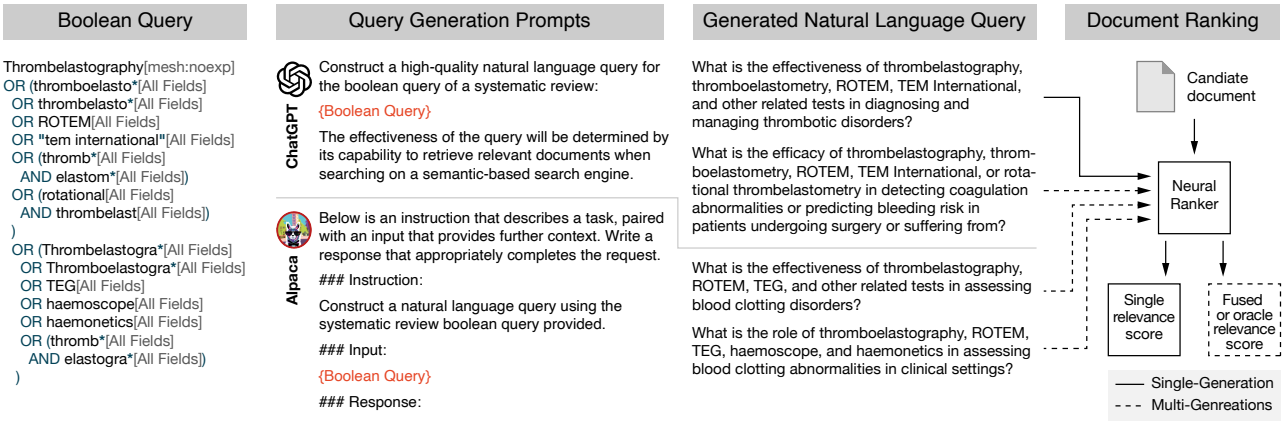


Figure 7.3: Illustration and examples of our screening prioritisation approach: Given a Boolean query, an instruction-based LLM is prompted to generate one or more natural language queries. Then, given a generated query and a candidate document, a neural ranker is used to predict one or more relevance scores for the document. In the latter case, the scores are fused by addition. As a baseline for our experiments, the score that maximises effectiveness is selected by an oracle.

encoder-based neural ranker.

Step 1: Query Generation.

The goal of this stage is to translate complex Boolean queries into natural language queries that accurately capture the underlying information need. Boolean queries are reformulated using model-specific prompts. Preliminary experiments revealed that Alpaca frequently outputs the original Boolean expression when used in a zero-shot setting. To mitigate this, we fine-tuned Alpaca on a dataset of Boolean–natural language query pairs generated by ChatGPT.

To examine how variation in LLM output affects ranking performance, we control the generation stochasticity using the temperature parameter t [131]. Lower values (e.g., $t = 0.0001$) produce more deterministic responses, while higher values yield more diverse outputs. We consider two settings: *Single-Generation*, where only one query is generated per Boolean query, and *Multi-Generations*, where multiple diverse queries are generated.

Step 2: Document Reranking.

Once natural language queries are produced, we rank candidate documents using a cross-encoder neural model. The model architecture and fine-tuning setup follow those described in the previous section on review titles. Specifically, we use a BioBERT-based cross-encoder fine-tuned with localised contrastive loss on training triples of the form $\langle \text{query}, d^+, d^- \rangle$.

Multi-Query Fusion. In the *Multi-Generations* setup, we generate multiple natural language queries per Boolean query. For each query-document pair, we compute relevance scores independently and combine them using one of two strategies:

- **Fusion:** Relevance scores across all queries are summed to produce a final score per document.

- **Oracle:** The single ranked list with the highest mean average precision (MAP) is selected as an upper-bound baseline.

Finally, we also experiment with combining the ranked outputs from the natural language queries and the original Boolean query. For this, we use the CombSUM technique [70], which adds the document relevance scores from each list to produce a final combined ranking.

7.2.2 Experimental Setup

This section describes the setup used to evaluate neural rankers for screening prioritisation using Boolean queries reformulated via LLMs. We first introduce the datasets and evaluation metrics. We then describe the baseline methods used for comparison, followed by details on how we fine-tune the Alpaca model for query rewriting and the BioBERT-based ranker for document scoring.

Dataset and Evaluation

We reuse the CLEF TAR datasets and follow the same evaluation protocol as described for title-based neural rankers in Section 7.1. To broaden our analysis, we also include the Seed Collection—introduced in this thesis and detailed in Chapter 4.

Like CLEF TAR, the Seed Collection includes topic-specific relevance judgments suitable for training and evaluation. A distinctive feature is the presence of a topic "Description," which serves as a proxy for the *working title* often available at the start of a systematic review. This allows us to assess how well Boolean queries (converted to natural language) compare to early-stage titles for prioritisation.

Evaluation follows CLEF TAR conventions: Average Precision (AP), rank of the last relevant document (Last_Rel), Recall@1%, 5%, 10%, and 20%, and Work Saved over Sampling (WSS) at 95% and 100%. Metrics are computed using the official CLEF TAR 2018 evaluation script [114].

Baseline Methods

We compare our approach against several strong baselines. First, we include BM25 and QLM, as reported in Section 7.1, to maintain consistency with prior comparisons. We also benchmark against the most effective participant runs submitted to each CLEF TAR dataset—regardless of whether they use feedback—since earlier experiments showed that fine-tuned neural rankers already match or exceed these systems.[7] We additionally include the CLF method [208], which directly uses Boolean queries for screening prioritisation. Finally, we include a neural baseline that uses the original Boolean query directly, without LLM reformulation, to assess the added value of query rewriting.

[7] Specifically: *sheffield.run4* for CLEF 2017 [6]; *shef-general* for CLEF 2018 [10]; *DTA_sheffield-Odds_Ratio* for CLEF 2019-DTA; and *DTA_sheffieldLog_Likelihood* for CLEF 2019-Intervention [8].

Fine-tuning Details: the Alpaca Query Generator

To fine-tune the Alpaca model for Boolean-to-natural language query rewriting, we first generate training data using ChatGPT in a Single-Generation setting. These ChatGPT outputs serve as gold-standard targets. Alpaca is then fine-tuned on this data using the prompt shown in the second column of Figure 8.1.

Given the complexity and length of Boolean queries, we simplified the input prompts for ChatGPT and increased Alpaca's input limit from 512 to 768 tokens.[8] The model is trained for three epochs with batch size and gradient accumulation both set to 1, following the original Alpaca configuration [239]. At inference, the same prompt is used to convert test-set Boolean queries into natural language form.

Fine-tuning Details: Neural Ranker

The neural ranker is a BioBERT-based cross-encoder fine-tuned using the Reranker toolkit [74], following the same procedure as in Section 7.1. The only change is an increased maximum query length—from 64 to 256 tokens—to accommodate the longer queries produced by Boolean query reformulations.

7.2.3 Main Results

This section presents and analyses the results of our experiments. We begin by evaluating the effectiveness of using a single LLM-generated query (*Single-Generation*), followed by an analysis of multiple generated queries (*Multi-Generations*).

Effectiveness of Single-Generation

To understand the effectiveness of **Single-Generation**, Table 7.4 compares the ranking effectiveness of the generated query to the original Boolean query, our baseline methods, and title-driven methods (where the working title is used to rank candidate documents). We also evaluate the differences in effectiveness of screening prioritisation between queries generated by various generation models.

Boolean vs. Generated Query: Natural language queries consistently outperform the original Boolean queries across most datasets and metrics. The only exception is 2019-DTA under MAP, likely due to its small topic count (eight topics), which increases sensitivity to outliers. Generated queries perform particularly well on deeper metrics like Recall@5%, 10%, 20%, and WSS@95/100, though performance on Recall@1% is slightly lower—suggesting better late-stage filtering but a small trade-off in early precision. Furthermore, fusing the scores of the generated and Boolean queries improves effectiveness further. This is especially evident on CLEF-2017, 2018, and 2019-Intervention, confirming that combining semantic and structural signals enhances ranking performance.

Neural vs. Baselines. Neural rankers using either Boolean or generated queries outperform lexical methods (BM25, QLM) across most settings. The only exceptions—Last_Rel on CLEF-2018 and

[8]This is the maximum supported with batch size 1 across three 80GB Nvidia A100 GPUs.

Table 7.4: Evaluation results for comparing methods for Boolean-driven screening prioritisation by generating natural language queries. We use natural language queries generated by ChatGPT and Alpaca, and the fusions of Boolean+ChatGPT and Boolean+Alpaca. Statistical significant differences (Student's two-tailed paired t-test with Bonferroni correction, $p < 0.05$) between using the Boolean query with the BioBERT (BERT) ranker, and other approaches are indicated by $*$.

Dataset	Query	Ranker	MAP	Last_Rel	Recall (%)				WSS (%)	
					1%	5%	10%	20%	95%	100%
2017	Boolean	BM25	0.114*	3242.73*	0.083*	0.215*	0.324*	0.491*	0.252*	0.188*
	Boolean	QLM	0.122*	3223.40*	0.073*	0.209*	0.325*	0.476*	0.243*	0.195*
	Boolean	CLF	0.217	3028.03*	0.149	0.341*	0.473*	0.671*	0.442*	0.327*
	Best Participation Run		0.218	2382.47*	0.131	0.332*	0.499*	0.688*	0.488*	0.395*
	Boolean	BERT	0.278	1790.87	0.166	0.488	0.656	0.812	0.600	0.536
	ChatGPT	BERT	0.293	1991.17	0.150	0.476	0.643	0.801	0.590	0.501
	Boolean+ChatGPT	BERT	**0.300***	1843.13	0.170	**0.499**	**0.664**	0.823	0.610	0.532
	Alpaca	BERT	0.284	1866.00	0.165	0.435	0.607	0.789	0.591	0.502
	Boolean+Alpaca	BERT	0.295	**1759.23**	**0.171**	0.483	0.663	**0.827***	**0.615**	**0.539**
2018	Boolean	BM25	0.154*	6033.07*	0.082*	0.242*	0.391*	0.563*	0.361*	0.264*
	Boolean	QLM	0.157*	6097.13*	0.080*	0.252*	0.380*	0.557*	0.384*	0.251*
	Boolean	CLF	0.272*	5743.27*	0.152	0.393*	0.546*	0.729*	0.552*	0.411*
	Best Participation Run		0.258*	5519.20	0.129*	0.383*	0.545*	0.729*	0.552*	0.431
	Boolean	BERT	0.353	4830.93	0.202	0.517	0.681	0.845	0.656	0.503
	ChatGPT	BERT	0.381	**4508.93**	0.247*	**0.555***	**0.713***	**0.865**	**0.692***	0.528
	Boolean+ChatGPT	BERT	**0.386***	4603.77*	**0.247***	0.551*	0.705*	0.859*	0.685*	**0.537***
	Alpaca	BERT	0.333	4957.23	0.191	0.493	0.662	0.827	0.640	0.485
	Boolean+Alpaca	BERT	0.365	4628.23	0.220	0.525	0.688	0.849	0.668	0.523
2019-dta	Boolean	BM25	0.125*	2766.88*	0.068	0.163*	0.303*	0.463*	0.299*	0.163*
	Boolean	QLM	0.121*	2614.75*	0.042	0.185*	0.278*	0.432*	0.271*	0.180*
	Best Participation Run		0.248	2183.50	0.168	0.439	0.594	0.742	0.490*	0.347*
	Boolean	BERT	**0.272**	1146.00	0.174	0.419	0.565	0.751	0.651	0.528
	ChatGPT	BERT	0.247	1173.25	0.183	**0.454**	**0.594**	**0.757**	0.660	0.528
	Boolean+ChatGPT	BERT	0.268	**1134.38**	**0.183**	0.446	0.584	0.755	**0.665**	**0.545**
	Alpaca	BERT	0.241	1217.88	0.170	0.483	0.622	0.784	0.666	0.520
	Boolean+Alpaca	BERT	0.251	1146.13	0.173	0.458	0.592	0.783	0.659	0.537
2019-Int.	Boolean	BM25	0.154*	1479.45*	0.070*	0.181*	0.264*	0.417*	0.289*	0.264*
	Boolean	QLM	0.148*	1473.85*	0.041*	0.201*	0.287*	0.450*	0.295*	0.252*
	Best Participation Run		0.293	1132.00	0.165	0.419	0.542	0.722	0.458	0.381*
	Boolean	BERT	0.389	1064.30	0.195	0.458	0.619	0.733	0.557	0.499
	ChatGPT	BERT	0.433*	993.30	0.217	0.487	**0.654**	0.788*	0.573	0.503
	Boolean+ChatGPT	BERT	**0.446***	**975.75**	**0.233**	**0.508***	0.651*	**0.789***	**0.578**	**0.529**
	Alpaca	BERT	0.317	1087.30	0.120	0.337*	0.510*	0.656	0.491	0.448
	Boolean+Alpaca	BERT	0.377	1024.70	0.169	0.460	0.616	0.730	0.551	0.504
Seed Collection	Boolean	BM25	0.087*	990.10*	0.034*	0.140*	0.249*	0.412*	0.252*	0.253*
	Boolean	QLM	0.085*	986.48*	0.018*	0.141*	0.212*	0.397*	0.254*	0.260*
	Working Title	BERT	0.171*	801.05*	0.090*	0.275*	0.350*	0.562*	0.465*	0.450*
	Boolean	BERT	0.199	785.90	0.085	0.248	0.412	0.600	0.481	0.467
	ChatGPT	BERT	0.217	**727.15**	0.082	**0.330***	**0.482***	0.670*	0.530*	0.505
	Boolean+ChatGPT	BERT	0.219	744.78*	0.078	0.279	0.468*	**0.677***	0.525*	**0.506***
	Alpaca	BERT	0.221	780.60	0.083	0.314*	0.473	0.655	0.529*	0.500
	Boolean+Alpaca	BERT	**0.230***	765.93	**0.098**	0.268	0.466*	0.661*	**0.531***	0.505*

Recall@1% on CLEF-2019-DTA—show no statistically significant difference. Compared to the CLF method [208], which also uses Boolean queries, our models consistently achieve higher scores, with statistical significance in most metrics except WSS95 (CLEF-2017) and WSS100 (CLEF-2018).

Compared to the best CLEF participant runs, our methods also perform better overall. While these runs often include richer input (e.g., final titles), our approach relies solely on Boolean queries, highlighting the strong performance of LLM-driven query reformulation.

Boolean vs. Working Titles. Using the Seed Collection, we compare Boolean-based and working-

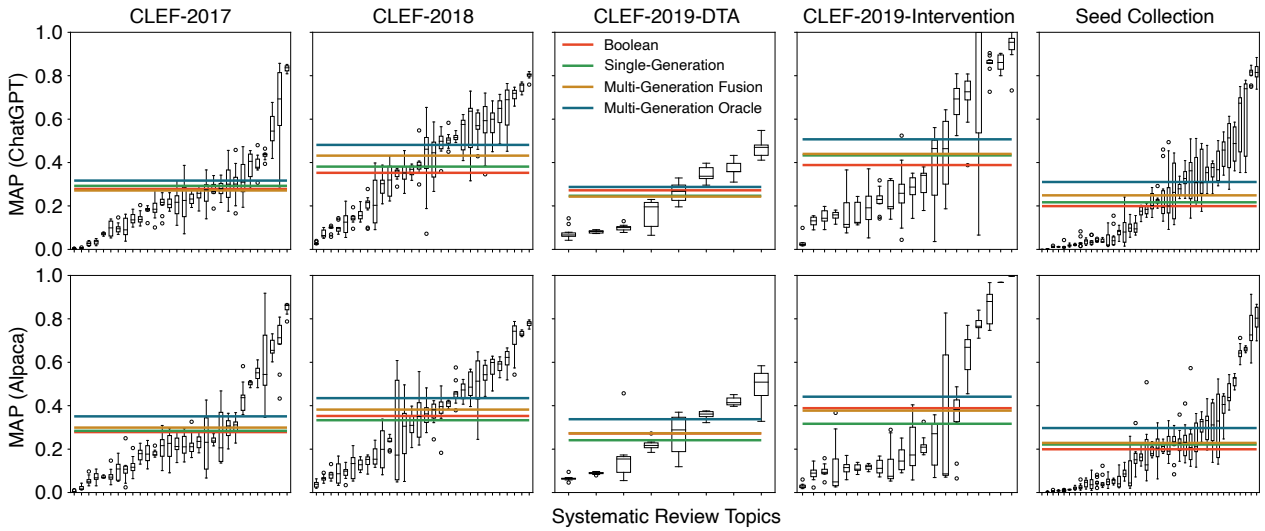


Figure 7.4: Topic-by-topic variability graph for the effectiveness of the Multi-Generations setup, using a single generated natural language query to rank documents. The coloured horizontal lines indicate the average effectiveness of different methods (Boolean, Single-Generation, Multi-Generation Fusion, and Multi-Generation Oracle).

title-based screening. Results show that working titles perform significantly worse—not only compared to generated queries but also to the original Boolean queries and even BM25. This confirms that working titles, while readily available early on, are not sufficient for effective prioritisation.

ChatGPT vs. Alpaca. ChatGPT-generated queries consistently outperform those from Alpaca across most datasets, with a particularly large gap on CLEF-2019-Intervention. This drop in Alpaca's performance may stem from domain mismatch, as the model was fine-tuned primarily on DTA-style queries. Nevertheless, fusing Alpaca-generated queries with the original Boolean queries improves effectiveness on all datasets. In the Seed Collection, this fusion even exceeds the performance of either source individually. These results suggest that a fine-tuned, open-source model like Alpaca can approach the effectiveness of proprietary models like ChatGPT, provided it is trained on data from a similar domain, while offering greater transparency and customisability.

Overall, transforming Boolean queries into natural language with LLMs improves screening prioritisation effectiveness. Generated queries outperform both lexical baselines and direct Boolean inputs, especially when fused. Our models also exceed prior CLEF runs, despite using fewer inputs. While ChatGPT performs best, domain-aligned fine-tuning enables open models like Alpaca to approach its effectiveness, offering a transparent and adaptable alternative.

Variability and Impact of Multi-Generations

Figure 7.4 shows the effectiveness of multiple natural language queries generated by ChatGPT and Alpaca, measured by MAP across topics. We observe substantial variability in effectiveness across generated queries, consistent with findings from earlier work on user- and system-generated query reformulation [139, 161, 174, 211, 214, 215, 216, 259, 283]. Intervention topics exhibit the greatest instability.

Comparing the two models, Alpaca shows greater variance than ChatGPT on DTA topics (e.g., +14.3%, +7.7%, and +166% in CLEF-2017, 2018, and 2019-DTA, respectively). In contrast, for intervention and unclassified topics (e.g., Seed Collection), Alpaca is more stable, showing 28.6%–39.1% lower variance than ChatGPT.

In terms of average effectiveness, fusing multiple generations generally improves performance over Single-Generation, with a few exceptions (e.g., CLEF-2017 and CLEF-2019-DTA for ChatGPT, CLEF-2019-Intervention for Alpaca). Notably, fusing Alpaca's Multi-Generations consistently outperforms the original Boolean query, showing the value of diversified reformulations even from smaller open models.

Finally, Oracle selection over Multi-Generations achieves the highest effectiveness across all settings. This highlights the potential of learning to select or weight generated queries, a promising direction for improving screening prioritisation further.

7.2.4 Ablation Studies

To better understand why natural language queries yield higher effectiveness, and to assess the contributions of fusion and training procedures, we conduct a series of ablation studies.

Generating Titles vs. Generating Natural Language Queries

Our first experiment tests the hypothesis that generating a natural language query from a Boolean query is more effective than generating a systematic review title. We posit that titles often reflect only a narrow subset of the Boolean query's intent, and omitting key information may reduce effectiveness in screening prioritisation. To evaluate this, we compare two approaches: (1) generating a systematic review title from the Boolean query, and (2) generating a natural language query. For document ranking, we train a cross-encoder BioBERT model using the training portion of each dataset, with titles or queries as inputs.

ChatGPT is used to generate both queries and titles in a zero-shot setting. For Alpaca, we fine-tune it on the training titles using the same settings as in Section 7.2.2, and test on the corresponding held-out topics.

The results, presented in Table 7.5, clearly demonstrate that generating titles yields lower effectiveness than generating natural language queries, regardless of whether the generation is done using ChatGPT or Alpaca, with the only exceptions being WSS95 on CLEF-2017 and WSS100 on CLEF-2018 when comparing the Alpaca model. Nevertheless, titles generate using ChatGPT appears to be significantly less effective than when generated through the Alpaca model, with most results showing statistical significance.

Impact of Fusion

We next assess how combining the results of Boolean and generated queries affects effectiveness. Figure 7.5 compares screening performance when using Boolean queries, generated queries, and their

Table 7.5: Results comparing the effectiveness of generating a title (Generating Title) versus generating a natural language query (Generating Natural Query) from the Boolean query of a systematic review for screening prioritisation. Statistical significant differences ($p < 0.05$) between the effectiveness of a generated title versus a generated natural language query are indicated by $*$.

Dataset	Model	Query	AP	WSS95	WSS100
2017	ChatGPT	Generating Natural Query	**0.293**	**0.590**	**0.501**
	ChatGPT	Generating Title	0.140*	0.486*	0.396*
	Alpaca	Generating Natural Query	**0.284**	0.591	**0.502**
	Alpaca	Generating Title	0.270	**0.595**	0.502
2018	ChatGPT	Generating Natural Query	**0.381**	**0.692**	**0.528**
	ChatGPT	Generating Title	0.277*	0.626*	0.491
	Alpaca	Generating Natural Query	**0.333**	**0.640**	0.485
	Alpaca	Generating Title	0.307	0.637	**0.501**
2019-dta	ChatGPT	Generating Natural Query	**0.247**	**0.660**	**0.528**
	ChatGPT	Generating Title	0.175	0.565*	0.504
	Alpaca	Generating Natural Query	**0.241**	**0.665**	**0.521**
	Alpaca	Generating Title	0.164	0.544*	0.458
2019-Int.	ChatGPT	Generating Natural Query	**0.433**	**0.573**	**0.503**
	ChatGPT	Generating Title	0.164*	0.443*	0.404
	Alpaca	Generating Natural Query	**0.317**	**0.491**	**0.448**
	Alpaca	Generating Title	0.232	0.458	0.408
Seed Collection	ChatGPT	Generating Natural Query	**0.217**	**0.530**	**0.505**
	ChatGPT	Generating Title	0.127*	0.494	0.490
	Alpaca	Generating Natural Query	**0.221**	**0.529**	**0.500**
	Alpaca	Generating Title	0.164	0.432*	0.439

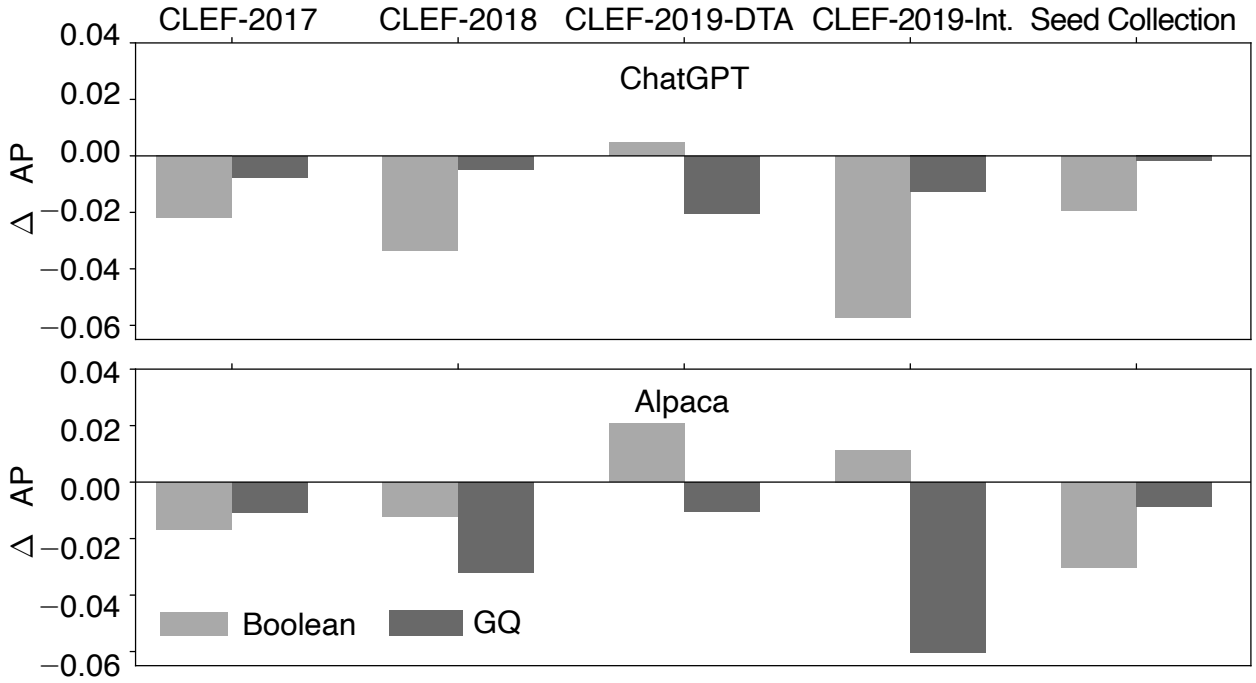


Figure 7.5: Differences in AP from Boolean, Generated Query (GQ) to their fused effectiveness.

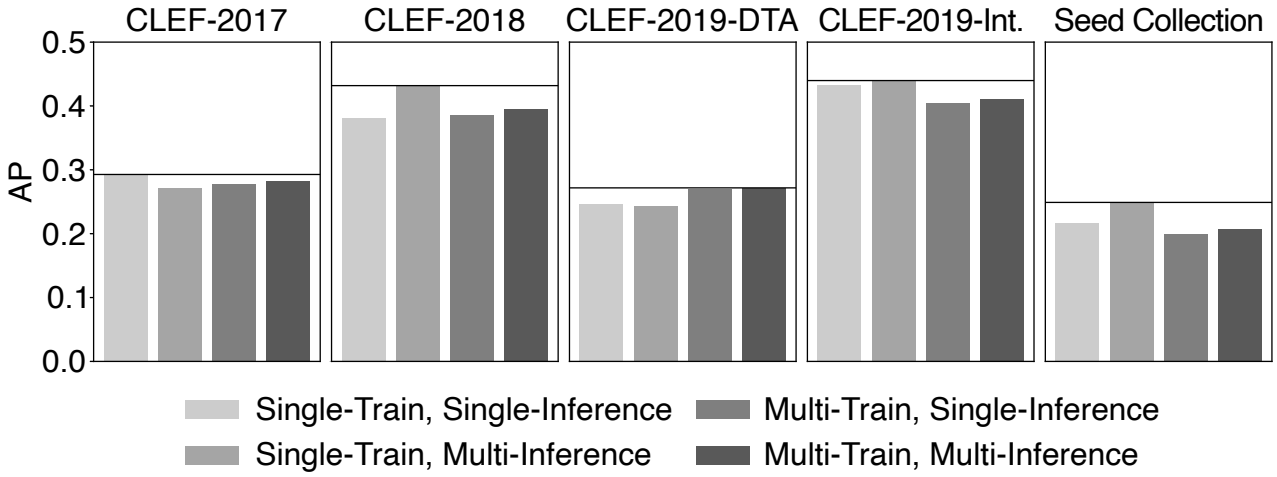


Figure 7.6: Effectiveness when different training and inference settings are used for ranking candidate documents using the generated natural language query from ChatGPT.

fusion.

The results indicate that the fusion, on average, consistently outperforms using the generated query alone, but it is not always more effective than using the Boolean queries alone. The effectiveness of Boolean queries should not be overlooked. When comparing results across the two generation models, we observe that the effectiveness gains over Boolean queries obtained tend to be more stable when ChatGPT is used. Using ChatGPT in query generation may thus contribute to more consistent improvements when the results are combined with those from Boolean queries.

Training the Ranker with Single vs. Multi-Generations

In our main experiments, we train the natural language query-based ranker using Single-Generation outputs for reproducibility, as Multi-Generation setups yield non-deterministic outputs—even with identical prompts. However, we are also interested in understanding how using Single versus Multi-Generations affects the performance of the trained ranker.

To investigate this, we define four training and inference strategies: **Single-Train**, **Multi-Train**, **Single-Inference**, and **Multi-Inference**.

In *Single-Train*, the model is trained using a single generated query per Boolean query. In *Multi-Train*, the model is trained on all generated queries for each Boolean query. In *Single-Inference*, we evaluate the model using a single generated query per topic. In *Multi-Inference*, we use all generated queries at test time and fuse their scores for ranking.

Using identical training parameters across all strategies, we present the screening effectiveness of each setting with ChatGPT queries in Figure 7.6. The results show that training with multiple queries does not consistently improve performance compared to using a single deterministic query. The only exception is CLEF-2019-DTA. However, Multi-Inference consistently improves effectiveness, suggesting that diversity at inference is beneficial. This likely stems from the model being exposed to different phrasings of the same information need, which helps surface relevant documents across varying formulations.

By contrast, training on multiple queries may dilute learning by forcing the model to generalise over diverse inputs for the same topic, which can lead to an averaging effect in the learned relevance patterns. This result suggests that inference-time diversity is more impactful than training-time diversity for improving screening effectiveness.

7.2.5 Summary of Screening Prioritisation using Boolean Queries

Our results demonstrate that transforming Boolean queries into natural language using instruction-based LLMs significantly improves screening prioritisation performance. Below, we summarise our key findings.

Effectiveness of Generated vs. Original Boolean Queries. Generated natural language queries consistently outperform the original Boolean queries across datasets and metrics. This holds for both neural rankers and the CLF model [208], which directly uses Boolean queries. Moreover, combining the rankings from Boolean and generated queries using CombSUM fusion yields further gains, suggesting that these query types offer complementary signals.

Impact of Generation Models. ChatGPT-generated queries generally outperform those from Alpaca, with the performance gap particularly large on CLEF-2019-Intervention. This may be due to Alpaca being fine-tuned primarily on DTA-style queries, while ChatGPT, used in a zero-shot setting, is more domain-agnostic. Despite this, Alpaca shows promise, especially when fine-tuned on in-domain data, highlighting the potential of open-source models for transparent and customisable systems.

Effectiveness of Neural Ranking. Neural rankers consistently outperform traditional lexical methods such as BM25 and QLM, as well as the best-performing CLEF TAR participant runs. This confirms the robustness and general effectiveness of neural approaches for screening prioritisation, even when limited to only the Boolean query as input.

Benefit of Multi-Generations. Generating multiple natural language queries introduces variation in effectiveness, particularly for intervention-style topics. Alpaca shows higher variance on DTA topics but greater stability on intervention topics. Fusion of Multi-Generations improves overall performance compared to Single-Generation alone. Oracle results—selecting the best-performing query—suggest that substantial gains are achievable if effective query selection strategies are employed. This opens a path for future work on query performance prediction in systematic reviews, a largely underexplored area.

Comparison with Working Titles. Using a review's working title as a query results in substantially lower effectiveness compared to using a generated query derived from its Boolean counterpart. While prior work using final review titles showed strong results, our findings reveal that early-stage titles

lack sufficient detail. This highlights the importance of using richer query representations, such as LLM-generated queries from Boolean expressions, when operating at the point of initial screening.

Overall Findings. This section uses Boolean queries as a practical and underutilised input for screening prioritisation. Unlike final review titles, which are investigated in Section 7.1 and used in prior works and alre but only available retrospectively—Boolean queries are typically crafted at the start of systematic reviews and thus represent a realistic signal for ranking candidate studies. While their formal structure and length make them unsuitable for direct use in BERT-style rankers, we show that instruction-tuned LLMs can reformulate them into effective natural language queries.

Our experiments demonstrate that these generated queries consistently outperform traditional lexical baselines, direct use of Boolean syntax, and even CLEF participant runs that rely on additional inputs. Despite being open-domain, ChatGPT produces highly effective reformulations, and fine-tuned open models like Alpaca approach its performance when trained on domain-specific data. Additionally, combining generated queries with their original Boolean forms further boosts ranking effectiveness, revealing complementary strengths.

In summary, we present a practical and effective approach to screening prioritisation by transforming structured Boolean queries (available at the time of screening) into neural-compatible inputs. Our findings show that Boolean queries, when reformulated using instruction-based models, can match or even surpass the effectiveness of using final review titles for screening prioritisation.

7.3 Summary of Findings

This chapter investigated neural approaches to screening prioritisation in systematic reviews, focusing on two widely used but underexplored sources of input queries: systematic review titles and Boolean queries. In the first part, we evaluated neural rankers using review titles as queries, including both zero-shot and fine-tuned encoder-based models. In the second part, we addressed the limitations of relying on finalised review titles—often unavailable at the outset—and instead explored using Boolean queries, which are typically constructed early in the review process. Across both settings, we conducted extensive experiments on CLEF TAR datasets and the Seed Collection.

To make Boolean queries compatible with neural ranking models, we proposed transforming them into natural language using instruction-following large language models. We implemented and evaluated this pipeline using both proprietary (ChatGPT) and open-source (Alpaca) LLMs. The resulting queries were used to train fine-tuned BioBERT-based cross-encoders for document ranking. We further investigated the impact of rank fusion, multiple generations, and model configuration choices through ablation studies.

Our findings show that fine-tuned neural rankers significantly outperform lexical baselines and previous CLEF shared task submissions. Review titles yield strong performance, but only when finalised versions are available; working titles are substantially less effective. In contrast, Boolean queries, when reformulated into natural language using LLMs, provide a practical and highly effective

alternative. ChatGPT-generated queries offer the best results overall, but Alpaca can achieve competitive performance with domain-specific fine-tuning. Combining Boolean and generated queries via fusion consistently improves ranking, and multi-generation strategies further enhance robustness and effectiveness.

While screening prioritisation aims to surface relevant documents earlier in the ranking, it does not eliminate the need for manual screening of the full candidate pool. In the next chapter, we investigate whether large language models can go a step further—by automatically making inclusion and exclusion decisions in a zero-shot setting. This explores the potential of LLMs to support or partially automate the screening process itself, beyond prioritisation.

Chapter 8

Automatic Screening Using Large Language Models

While we focused on prioritising documents for manual review in previous chapter, this chapter explores a more ambitious goal: automating the screening process itself using large language models (LLMs). Specifically, we investigate whether modern generative LLMs can be used to make inclusion or exclusion decisions directly, potentially reducing the human effort required during the systematic review screening stage.

We focus on a zero-shot setting, where LLMs are prompted to follow simple inclusion instructions without any task-specific fine-tuning. This approach is attractive for its practicality and generalisability: it eliminates the need for annotated training data, which is often expensive and time-consuming to obtain in systematic reviews, and enables immediate deployment across diverse review topics without retraining. By relying solely on model prompting and token-level likelihoods, this method takes advantage of the instruction-following capabilities of modern LLMs while avoiding the overhead of supervised learning. Our method formulates the classification decision by prompting the model with an inclusion/exclusion instruction and computing the likelihood of specific target tokens (e.g., *yes* vs. *no*) to determine inclusion. This decision-making process is lightweight and model-agnostic, requiring no gradient updates or modification to the underlying LLM.

We examine two versions of this method: an *uncalibrated* classifier, which selects the more probable label token, and a *calibrated* variant, which introduces a threshold hyperparameter θ based on the difference in token likelihoods. This threshold can be estimated using a small set of labelled documents or derived from previous reviews.

8.1 Method of Automatic Document Screening Using LLMs

Our framework for automatic document screening uses generative LLMs to determine whether a candidate document should be included in a systematic review. As shown in Figure 8.1, we model this task as a binary classification function $I(d,t) : D \times T \rightarrow \{0,1\}$, where $d \in D$ is a document and $t \in T$ is

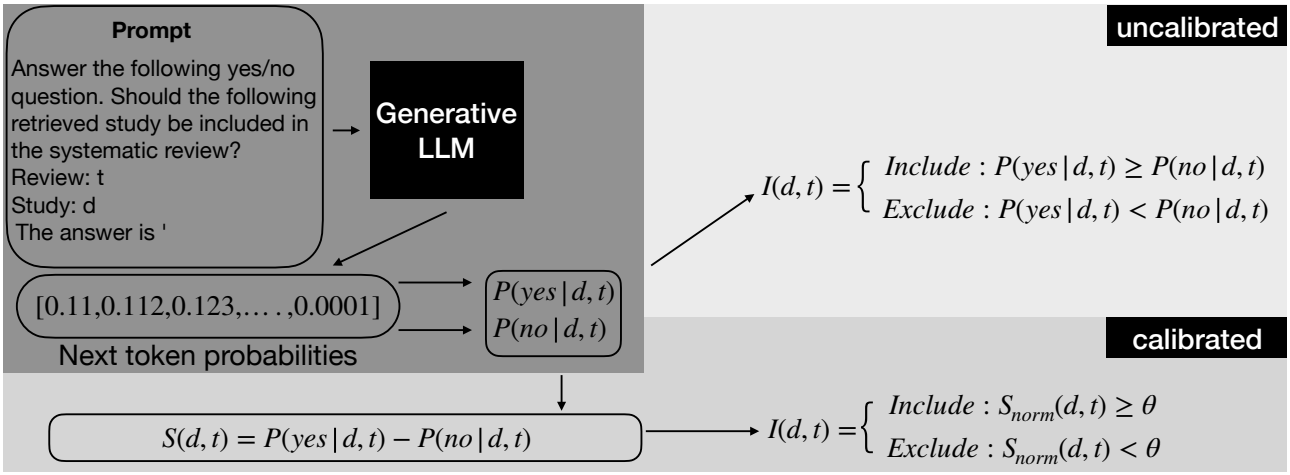


Figure 8.1: Our framework for automatic document screening using generative LLMs. $P(\texttt{yes}|d,t)$ ($P(\texttt{no}|d,t)$) is the likelihood of the `yes` (`no`) token in the next token probability list, and θ is the decision boundary(threshold) used by the calibrated setting.

a review topic. A value of $I(d,t) = 1$ indicates inclusion, and 0 indicates exclusion. The classification decision is computed based on the output probabilities returned by the LLM when prompted with information about the topic t and document d.

We investigate two variants of this decision function: an *uncalibrated* version, which uses direct token likelihood comparison, and a *calibrated* version, which introduces a learned threshold to control the inclusion decision.

8.1.1 Uncalibrated Screening

The uncalibrated approach assumes that the LLM can directly estimate the inclusion decision from a simple prompt. Given a prompt constructed from the topic and the document content, we compute the conditional probabilities $P(\texttt{yes}|d,t)$ and $P(\texttt{no}|d,t)$, corresponding to whether the document should be included.

The inclusion decision is based on whichever token has higher probability:

$$I(d,t) = \begin{cases} 1, & \text{if } P(\texttt{yes}|d,t) \geq P(\texttt{no}|d,t) \\ 0, & \text{otherwise.} \end{cases}$$

To make inference deterministic and efficient, we do not generate free-text outputs. Instead, we extract only the next-token probabilities from the model and use them to make the binary decision.

8.1.2 Calibrated Screening

While the uncalibrated method makes decisions based on a simple comparison of token likelihoods, it does not account for differences in model behaviour. Some LLMs may be biased toward inclusion—frequently predicting `yes`—while others may be overly conservative and favour `no`. These tendencies can lead to systematic errors, especially in imbalanced settings like systematic review

screening where recall is critical. To address this, we introduce a calibrated variant of the classifier that learns a decision threshold based on likelihood differences and normalisation.

We begin by defining a confidence score $S(d,t)$ as the difference between the likelihoods of the two label tokens:

$$S(d,t) = \begin{cases} P(\texttt{yes}|d,t) - P(\texttt{no}|d,t), & \text{if } P(\texttt{yes}|d,t) \geq P(\texttt{no}|d,t) \\ 0, & \text{otherwise.} \end{cases}$$

Because these raw scores vary across topics and document sets, we apply min-max normalisation within each topic to rescale scores between 0 and 1:

$$S_{\text{norm}}(d,t) = \frac{S(d,t) - \min\limits_{d_i \in D} S(d_i,t)}{\max\limits_{d_i \in D} S(d_i,t) - \min\limits_{d_i \in D} S(d_i,t)}$$

Next, we learn a threshold θ from training data that ensures a minimum desired recall level (e.g., 95%). A document is classified as included if its normalised score exceeds this threshold:

$$I(d,t) = \begin{cases} 1, & \text{if } S_{\text{norm}}(d,t) \geq \theta \\ 0, & \text{otherwise.} \end{cases}$$

This calibrated approach has two key advantages. First, it explicitly optimises for high recall, which is essential in systematic reviews to ensure comprehensive coverage of relevant studies. Second, it adjusts for systematic biases in different LLMs (whether they lean toward over-inclusion or under-inclusion) allowing for more reliable and tunable decision boundaries across topics and models.

8.1.3 Ensembling Screening Methods

To improve classification stability and reduce model-specific bias, we investigate an ensemble strategy that combines outputs from multiple screening methods. Specifically, we fuse the predictions from the two strongest zero-shot LLMs and a BERT-based classifier using CombSUM [124], which aggregates scores from each model to make a joint decision.

The motivation behind ensembling is to improve screening accuracy by aggregating decision signals from multiple models. Rather than relying on a single model's judgment, which may be brittle or suboptimal on certain topics. In the context of systematic reviews, correctness is important—errors in inclusion or exclusion can compromise the review's validity. By fusing scores from multiple models, we aim to reinforce correct predictions and suppress inconsistent ones. Although this approach may increase computational cost, the added expense is negligible compared to the cost of manual expert screening, making it a practical and effective strategy for improving overall decision quality.

Uncalibrated Ensemble. For uncalibrated models, we sum the token likelihoods from each method. The final inclusion decision is made by comparing the aggregated scores for `yes` and `no` tokens:

$$I(d,t) = \begin{cases} 1, & \text{if } \sum\limits_{m \in \text{Methods}} P_m(\texttt{yes}|d,t) \geq \sum\limits_{m \in \text{Methods}} P_m(\texttt{no}|d,t) \\ 0, & \text{otherwise.} \end{cases}$$

Calibrated Ensemble. For calibrated models, we first compute the normalised confidence score S_{norm} from each method. These scores are then summed and compared against a learned threshold θ:

$$I(d,t) = \begin{cases} 1, & \text{if } \sum_{m \in \text{Methods}} S_{\text{norm},m}(d,t) \geq \theta \\ 0, & \text{otherwise.} \end{cases}$$

This approach allows the system to incorporate diverse decision signals and reduce sensitivity to any single model's output distribution. In scenarios where models exhibit varying recall or precision tendencies, the ensemble provides a more balanced decision mechanism.

8.2 Experimental Setup

8.2.1 Considered LLMs

We evaluate a diverse set of zero-shot generative LLMs available at the time of conducting this research. These models vary in architecture, training scale, and fine-tuning approach. Our goal is to compare their effectiveness in automatic systematic review screening under consistent conditions.

Model Families: The considered models belong to several prominent open-source families:

- **LlaMa and LlaMa2:** Decoder-only transformers trained on 1.4T–2T tokens [244, 245], with LlaMa2 incorporating improvements in stability and instruction tuning through techniques like RLHF and Ghost Attention.
- **Alpaca:** A 7B LlaMa-derived model fine-tuned using the self-instruct method with prompts from text-davinci-003 [239, 272].
- **Guanaco:** LlaMa variants fine-tuned via QLoRA on OpenAssistant data (OASST1), offering memory efficiency while retaining strong benchmark performance [123].
- **Falcon:** Falcon-7B-Instruct is trained on 1.5T tokens from RefinedWeb and further tuned for assistant-like behaviour [175].

Selected Models: We evaluate eight models, summarised in Table 8.1. We adopt shortened names for clarity, with instruction-tuned variants suffixed by *-ins*. To ensure fairness, we standardise the maximum input length to 2048 tokens for all models. For models with originally trained lower or higher context limits (e.g., Alpaca: 512, LlaMa2: 4096), truncation or padding is applied accordingly.

Commercial Model Exclusion: Due to cost constraints, we exclude commercial models such as ChatGPT. Estimated usage costs range from USD $4,000 (GPT-3.5-turbo) to USD $80,000 (GPT-4), making open-source alternatives significantly more viable for large-scale evaluation.

Prompts used for each model are detailed in Table 8.2.

Table 8.1: Evaluated LLMs, their short names, and prompt length limits (tokens).

Short Name	Original Model Name	Prompt Length Limit
LlaMa-7b	LlaMa-7B [1]	2048
LlaMa2-7b	meta-llama/Llama-2-7b-hf	4096
LlaMa2-13b	meta-llama/Llama-2-13b-hf	4096
Alpaca-7b-ins	tatsu-lab/alpaca-7b-wdiff	512
Guanaco-7b-ins	timdettmers/guanaco-7b	2048
Falcon-7b-ins	tiiuae/falcon-7b-instruct	2048
LlaMa2-7b-ins	meta-llama/Llama-2-7b-chat-hf	4096
LlaMa2-13b-ins	meta-llama/Llama-2-13b-chat-hf	4096

Table 8.2: Input types and prompts designed for each model. Italicised text indicates placeholders replaced with actual content.

Model	Prompt
Alpaca	Below is an instruction that describes a task, paired with an input that provides further context. Write response that appropriately completes the request. ### Instruction: Answer 'yes' or 'no' to Judge if the following retrieved study should be included by the systematic review? ### Input: Review: *review_title* Study: *candidate_document* ### Response:
All Other Models	Answer 'yes' or 'no' to Judge if the following retrieved study should be included by the systematic review? Review: *review_title* Study: *candidate_document* The answer is '

8.2.2 Dataset and Evaluation

We evaluate our approach on the CLEF TAR datasets and the Seed Collection, both of which were introduced earlier in the thesis.

From CLEF TAR [113, 114, 115], we use all 128 topics: 88 Diagnostic Test Accuracy (DTA) and 40 Intervention topics released between 2017 and 2019. These datasets contain approximately 600,000 relevance judgments. We ignore the standard train-test splits and evaluate zero-shot performance on all topics. Each topic includes a review title, protocol, and binary inclusion labels. From the Seed Collection [258], we use 38 of the 39 topics (excluding topic 18, which has no relevant retrieved documents). The dataset includes over 50,000 candidate documents and around 2,300 relevance judgments. Evaluation is restricted to retrieved candidate sets; relevant documents outside this pool are excluded.

For evaluation, we adopt standard set-based metrics: precision, recall, and F-3, which emphasises recall over precision. We also report balanced accuracy (B-AC), well-suited for highly imbalanced settings like systematic review screening. It is defined as:

$$\text{B-AC} = \frac{1}{2}\left(\frac{\text{TP}}{\text{TP}+\text{FN}} + \frac{\text{TN}}{\text{TN}+\text{FP}}\right)$$

We also report the *success rate*—the proportion of topics achieving a minimum recall of 0.95. This threshold is commonly used in systematic review automation [24, 27, 56], as systems falling below

this level are typically considered inadequate.

Finally, we quantify efficiency using Work Saved over Sampling at a fixed recall level (WSS) [45]:

8.2.3 Baseline Methods

We compare the zero-shot LLMs against the BioBERT-based reranker introduced in Chapter 7. BioBERT is a BERT-based model pre-trained on large-scale biomedical corpora, making it well-suited for health-related tasks, including systematic review screening.

While the original BioBERT reranker was used for screening prioritisation (therefore ranking), we adapt it here for binary classification. Specifically, we use classification head with a sigmoid activation is used to produce an inclusion probability for the document. In the uncalibrated setting, we apply a fixed threshold of 0.5: a document is included if the model output is ≥ 0.5, and excluded otherwise.

8.2.4 Threshold Setting

For the calibrated setting, the decision threshold θ must be estimated. We consider two strategies:

1. **Extrapolation from Other Topics:** We perform a leave-one-out procedure across all topics in the dataset. For each held-out topic, we compute θ as the median normalised score of documents that meet the desired recall (e.g., 0.95) across the remaining topics. This threshold is then applied to the held-out topic. Importantly, this procedure uses cross-validation only to determine θ—the LLMs remain zero-shot and are not fine-tuned. While this approach assumes access to relevance labels for calibration, it avoids the computational cost of training the models themselves.

2. **Calibration with Seed Studies:** We apply the uncalibrated LLM to a set of seed studies (documents typically identified before screening begins.) If any seed study receives a score below the initial threshold, θ is lowered to the minimum seed score, ensuring all seed studies are classified as included. This aims to safeguard recall. We test both 0.95 and 1.0 as target recall levels, in line with common practice in systematic reviews.

8.3 Main Results

We structure our analysis with respect to four dimensions: (1) Impact of model architecture and size, (2) Impact of instruction-based fine-tuning, (3) Effect of calibration, (4) Effect of model ensembling; finally, we also compare our approach with current state-of-the-art fine-tuned LLMs for automated screening.

Impact of Model Architecture and Size

Table 8.3 and Table 8.4 demostrate uncalibrated screening performance across different LLM architectures and model sizes.

Table 8.3: Comparison of uncalibrated results for DTA reviews (including datasets: CLEF-2017, CLEF-2018 CLEF-2019-dta) between baseline method and generative large language models. Statistical significance, determined by a Student's two-tailed paired t-test with Bonferroni correction ($p < 0.05$), between the top-performing method *LlaMa2-7b-ins* and others is marked by *.

	Model	P	R	B-AC	F-3	Success	WSS
CLEF-2017	BioBERT	0.06	0.95*	0.61*	0.30	0.74*	0.26*
	LlaMa-7b	0.04*	0.92*	0.48*	0.24*	0.46*	0.03*
	LlaMa2-7b	0.07	0.50*	0.60*	0.23*	0.02*	0.70*
	LlaMa2-13b	0.04*	1.00*	0.50*	0.25*	0.98*	0.00*
	Falcon-7b-ins	0.05*	0.92*	0.52*	0.25*	0.44	0.12*
	Alpaca-7b-ins	0.04*	0.92*	0.51*	0.25*	0.38	0.11*
	LlaMa2-7b-ins	0.08	0.87	**0.72**	**0.35**	0.26	0.56
	LlaMa2-13b-ins	**0.19***	0.41*	0.66*	0.31	0.04*	**0.91***
	Guanaco-7b-ins	0.04*	**1.00***	0.50*	0.25*	**1.00***	0.00*
CLEF-2018	BioBERT	0.06	0.97*	0.59*	0.9	0.87*	0.19*
	LlaMa-7b	0.05*	0.92*	0.48*	0.25*	0.33	0.04*
	LlaMa2-7b	0.07	0.49*	0.59*	0.22*	0.03*	0.69*
	LlaMa2-13b	0.05*	1.00*	0.50*	0.26	**1.00***	0.00*
	Falcon-7b-ins	0.05*	0.92	0.51*	0.25*	0.40	0.11*
	Alpaca-7b-ins	0.05*	0.91	0.51*	0.25*	0.30	0.11*
	LlaMa2-7b-ins	0.09	0.88	**0.75**	**0.37**	0.27	0.59
	LlaMa2-13b-ins	**0.26***	0.36*	0.66*	0.30	0.00*	**0.94***
	Guanaco-7b-ins	0.05*	**1.00***	0.50*	0.26	**1.00***	0.00*
CLEF-2019-dta	BioBERT	0.07	0.99	0.58	0.30	0.88	0.18*
	LlaMa-7b	0.07	0.93	0.48*	0.27	0.25	0.03*
	LlaMa2-7b	0.08	0.48*	0.58*	0.23	0.00*	0.68
	LlaMa2-13b	0.07	1.00	0.50*	0.28	**1.00**	0.00*
	Falcon-7b-ins	0.07	0.95	0.54*	0.29	0.50	0.12*
	Alpaca-7b-ins	0.07	0.91	0.52*	0.28	0.25	0.12*
	LlaMa2-7b-ins	0.09	0.92	**0.71**	**0.35**	0.62	0.49
	LlaMa2-13b-ins	**0.19**	0.49*	0.69	0.32	0.00*	**0.87***
	Guanaco-7b-ins	0.07	**1.00**	0.50*	0.28	**1.00**	0.00*

For **model architecture**, we compare four models: *Falcon-7b-ins*, *Alpaca-7b-ins*, *LlaMa2-7b-ins* and *Guanaco-2-7b-ins* — all of which have the same number of parameters. The results indicate that *LlaMa2-7b-ins* is the most effective for the task, outperforming the others across all evaluation metrics except recall and success rate. Specifically, this model obtained a high WSS while incurring only a marginal drop in recall: a significant loss was observed only on CLEF-2017. Concerning success rate, *LlaMa2-7b-ins* exhibited comparable performance to its counterparts, showing no statistically significant differences.

For **model size**, we consider two variants of the *LlaMa2-ins* architecture: one with 7 billion

Table 8.4: Comparison of uncalibrated results for Intervension reviews (including datasets: CLEF-2019-Int and Seed Collection) between baseline method and generative large language models. Statistical significance, determined by a Student's two-tailed paired t-test with Bonferroni correction ($p < 0.05$), between the top-performing method *LlaMa2-7b-ins* and others is marked by *.

	Model	P	R	B-AC	F-3	Success	WSS
CLEF-2019-Int	BioBERT	0.10	0.98*	0.58*	0.32	0.90*	0.16*
	LlaMa-7b	0.05*	0.86	0.47*	0.26	0.30	0.08*
	LlaMa2-7b	0.08	0.30*	0.55*	0.18*	0.05*	0.80*
	LlaMa2-13b	0.05	1.00*	0.50*	0.29	0.97*	0.00*
	Falcon-7b-ins	0.05	0.91	0.50*	0.27	0.57	0.09*
	Alpaca-7b-ins	0.05	0.87	0.49*	0.27	0.30	0.12*
	LlaMa2-7b-ins	0.08	0.90	**0.70**	**0.35**	0.42	0.48
	LlaMa2-13b-ins	**0.17***	0.45*	0.67	0.33	0.05*	**0.87***
	Guanaco-7b-ins	0.05	**1.00***	0.50*	0.29	**1.00***	0.00*
Seed Collection	BioBERT	0.04	0.93	0.54*	0.24	0.77*	0.16*
	LlaMa-7b	0.04	0.89	0.48*	0.21	0.56	0.07*
	LlaMa2-7b	0.04	0.29*	0.53*	0.15*	0.03*	0.78*
	LlaMa2-13b	0.04	**1.00***	0.50*	0.23	**1.00***	0.00*
	Falcon-7b-ins	0.04	0.93	0.50*	0.22	0.69	0.07*
	Alpaca-7b-ins	0.04	0.90	0.50*	0.22	0.49	0.10*
	LlaMa2-7b-ins	0.05	0.90	0.66	0.27	0.54	0.40
	LlaMa2-13b-ins	**0.13***	0.48*	**0.67**	**0.28**	0.05*	**0.85***
	Guanaco-7b-ins	0.04	**1.00***	0.50*	0.23	**1.00***	0.00*

parameters (*LlaMa2-7b-ins*) and another with 13 billion parameters (*LlaMa2-13b-ins*). Our findings suggest a trade-off between recall and WSS. Specifically, the 7-billion parameter variant obtains significantly higher recall, but this comes at the expense of reduced savings, evidenced by significantly lower WSS. Regarding B-AC, *LlaMa2-7b-ins* generally outperforms its larger counterpart across multiple datasets, except for the Seed Collection. Statistically significant differences in B-AC were only noted for CLEF-2017 and CLEF-2018.

Effect of Instruction-Based Fine-Tuning

Again at Table 8.3 and Table 8.4, we present comparisons between instruction-fine-tuned models and their base counterparts: *LlaMa2-7b-ins* vs. *LlaMa2-7b*, *LlaMa2-13b-ins* vs. *LlaMa2-13b*, and *Alpaca-7b-ins* vs. *LlaMa-7b*. Across all cases, fine-tuning yields significant improvements in B-AC. However, the impact on other metrics varies. For *LlaMa-7b* and *LlaMa2-13b*, fine-tuning improves WSS but reduces recall. In contrast, *LlaMa2-7b-ins* achieves higher recall, success rate, and F3, albeit with lower WSS. The F3 improvement is not statistically significant for CLEF-2019-dta. We also evaluated *Guanaco-7b-ins*, a QLoRA fine-tuned variant. Although it shows higher B-AC than *LlaMa-7b*, it assigns all candidate documents as relevant, making it unsuitable for systematic review screening in practice.

Overall, instruction-based fine-tuning generally enhances screening accuracy, though the specific

Table 8.5: Comparison between the Calibrated (Cal) and Uncalibrated (Unc) approaches using the BioBERT model, LlaMa2-7b-ins mode, the LlaMa2-13b-ins model and the Ensemble of the three models for DTA reviews (including datasets: CLEF-2017, CLEF-2018 CLEF-2019-dta). The calibrated method's number or character in the bracket () denotes the pre-set target recall (0.95 & 1) or using seed documents (S). Statistical significance for each generative model across different datasets is assessed using a Student's two-tailed paired t-test with a Bonferroni correction ($p < 0.05$) with respect to the uncalibrated approach, denoted by *. The highest evaluated scores for *each dataset* are bolded.

	Model	Setting	P	R	B-AC	F-3	Success	WSS
CLEF-2017	BioBERT	Unc	0.06	0.95	0.61	0.30	0.74	0.26
		Cal(0.95)	0.06	0.92	0.64	0.31	0.50*	0.34*
		Cal(1)	0.06	0.97	0.60	0.29	0.82	0.23
	Llama2-7b-ins	Unc	0.08	0.87	0.72	0.35	0.26	0.56
		Cal(0.95)	0.06*	0.92*	0.69*	0.32	0.52	0.44
		Cal(1)	0.05*	**0.99***	0.60*	0.28	0.96	0.20
	Llama2-13b-ins	Unc	0.19	0.41	0.66	0.31	0.04	0.91
		Cal(0.95)	0.06*	0.93	0.59*	0.28	0.50*	0.25*
		Cal(1)	0.05*	0.98	0.53*	0.26	0.88*	0.08*
	Ensemb	Unc	**0.31**	0.13	0.56	0.13	0.00	**0.98**
		Cal(0.95)	0.08	0.93*	**0.72**	**0.35***	0.52*	0.50*
		Cal(1)	0.06	0.97*	0.63	0.30	**0.90***	0.29*
CLEF-2018	BioBERT	Unc	0.06	0.97	0.59	0.29	0.87	0.19
		Cal(0.95)	0.07	0.91*	0.63	0.30	0.57*	0.33*
		Cal(1)	0.06	0.97	0.59	0.29	0.87	0.21
	Llama2-7b-ins	Unc	0.09	0.88	0.75	0.37	0.27	0.59
		Cal(0.95)	0.08*	0.94*	0.71*	0.35*	0.50	0.46
		Cal(1)	0.06*	**0.99***	0.62*	0.30	**1.00**	0.24
	Llama2-13b-ins	Unc	0.26	0.36	0.66	0.30	0.00	0.94
		Cal(0.95)	0.06	0.94*	0.59*	0.29	0.47*	0.22*
		Cal(1)	0.05	0.97	0.53*	0.27	0.80*	0.08*
	Ensemb	Unc	**0.35**	0.12	0.54	0.12	0.00	**0.95**
		Cal(0.95)	0.09*	0.94*	**0.75**	**0.38***	0.50*	0.54*
		Cal(1)	0.06	0.99*	0.64	0.32*	0.93*	0.28*
CLEF-2019-dta	BioBERT	Unc	0.07	0.99	0.58	0.30	0.88	0.18
		Cal(0.95)	0.08	0.89	0.59	0.26	0.50	0.27
		Cal(1)	0.08	0.91	0.59	0.27	0.62	0.25
	Llama2-7b-ins	Unc	0.09	0.92	0.71	**0.35**	0.62	0.49
		Cal(0.95)	0.10*	0.91*	0.71*	0.34	0.50	0.50
		Cal(1)	0.08*	0.97*	0.66	0.32	0.75	0.34
	Llama2-13b-ins	Unc	0.19	0.49	0.69	0.32	0.00	0.87
		Cal(0.95)	0.08	0.95	0.56	0.29	0.50*	0.16*
		Cal(1)	0.07	0.99	0.51	0.28	0.88*	0.03*
	Ensemb	Unc	**0.31**	0.21	0.59	0.19	0.00	**0.96**
		Cal(0.95)	0.10	0.91	**0.73**	0.34*	0.50*	0.52*
		Cal(1)	0.09*	**0.99**	0.64	0.32*	**1.00***	0.28*

trade-offs (between recall and screening efficiency) depend on the underlying architecture. QLoRA-based fine-tuning, however, does not appear effective for this task.

Table 8.6: Comparison between the Calibrated (Cal) and Uncalibrated (Unc) approaches using the BioBERT model, LlaMa2-7b-ins model, the LlaMa2-13b-ins model and the Ensemble of the three models for Intervension reviews (including datasets: CLEF-2019-Int and Seed Collection). The calibrated method's number or character in the bracket () denotes the pre-set target recall (0.95 & 1) or using seed documents (S). Statistical significance for each generative model across different datasets is assessed using a Student's two-tailed paired t-test with a Bonferroni correction ($p < 0.05$) with respect to the uncalibrated approach, denoted by *. The highest evaluated scores for *each dataset* are bolded.

	Model	Setting	P	R	B-AC	F-3	Success	WSS
CLEF-2019-Int	BioBERT	Unc	0.10	0.98	0.58	0.32	**0.90**	0.16
		Cal(0.95)	0.10	0.87*	0.59	0.29	0.50*	0.31*
		Cal(1)	0.10	0.90*	0.59	0.30	0.62*	0.27*
	Llama2-7b-ins	Unc	0.08	0.90	0.70	0.35	0.42	0.48
		Cal(0.95)	0.08*	0.91*	0.67*	0.34	0.50	0.42
		Cal(1)	0.07*	0.93*	0.64*	0.33	0.65	0.34
	Llama2-13b-ins	Unc	0.17	0.45	0.67	0.33	0.05	0.87
		Cal(0.95)	0.07*	0.90	0.58	0.30	0.50*	0.25*
		Cal(1)	0.06*	0.94	0.55	0.29	0.62*	0.16*
	Ensemble	Unc	**0.35**	0.23	0.58	0.22	0.05	**0.92**
		Cal(0.95)	0.09*	0.93*	**0.70**	**0.37***	0.50*	0.45*
		Cal(1)	0.08*	**0.96***	0.67	0.35*	0.68*	0.35*
Seed Collection	BioBERT	Unc	0.04	0.93	0.54	0.24	0.77	0.16
		Cal(0.95)	0.05	0.80*	0.55	0.22	0.50*	0.29*
		Cal(1)	0.05	0.83	0.55	0.23	0.53*	0.26
		Cal (S)	0.04	0.93	0.54	0.23	0.76	0.15
	Llama2-7b-ins	Unc	0.05	0.90	0.66	0.27	0.54	0.40
		Cal(0.95)	0.05*	0.90*	0.66	0.28*	0.51	0.41
		Cal(1)	0.05*	0.92*	0.65	0.27*	0.56	0.38
		Cal (S)	0.05	0.97*	0.6*	0.26	0.77*	0.22*
	Llama2-13b-ins	Unc	0.13	0.48	0.67	0.28	0.05	0.85
		Cal(0.95)	0.06*	0.87	0.64*	0.27*	0.51*	0.39*
		Cal(1)	0.05*	0.93	0.59*	0.26	0.59*	0.25*
		Cal (S)	0.06*	0.87*	0.63	0.29	0.54*	0.38*
	Ensemble	Unc	**0.16**	0.18	0.52	0.14	0.00	**0.86**
		Cal(0.95)	0.07*	0.86*	**0.71**	0.31*	0.49*	0.53*
		Cal(1)	0.07*	0.88*	0.70	**0.30***	0.56*	0.49*
		Cal (S)	0.04*	**1.00***	0.55	0.25*	**0.97***	0.10*

Benefits of Calibration for Recall Control

Table 8.5 and Table 8.6 present the results obtained when applying calibrated settings with predefined recall targets.

We find that calibration effectively enables models to meet their recall targets, making them more suitable for practical implementation in automatic document screening. Under the extrapolation-from-collection calibration strategy, approximately 50% of topics meet the recall target of 0.95, as indicated by the success rate. This improves further—ranging from 0.56 to 1.00—when the calibration target is raised to 1.0. We also compare the performance of three calibrated models: *BioBERT*, *LlaMa2-7b-ins*, and *LlaMa2-13b-ins*. Overall, *LlaMa2-7b-ins* achieves significantly better B-AC and WSS compared to the others. In terms of recall and success rate, all models show comparable performance; *LlaMa2-*

Table 8.7: Comparison of Fine-tuned baseline to our method; Statistical significance, determined by a Student's two-tailed paired t-test with Bonferroni correction ($p < 0.05$), between Uncalibrated *Bio-SIEVE* method and others is marked by *.

Model	Setting	P	R	B-AC	F-3	Success	WSS
Bio-SIEVE	Original/Calibrated	**0.232**	0.576	0.727	**0.429**	0.111	0.858
	Calibrated(Recall=0.95)	0.102*	0.877*	0.683	0.348	0.481*	0.471*
	Calibrated (Recall=1)	0.088*	0.945*	0.666	0.339	0.704*	0.369*
LlaMa2-7b-ins	Uncalibrated	0.078*	0.920*	0.725	0.359	0.333	0.513*
	Calibrated (Recall=0.95)	0.068*	0.935*	0.685	0.333	0.481*	0.421*
	Calibrated(Recall=1)	0.059*	**0.990***	0.621*	0.311	**1.000***	0.241*
Ensemble	Uncalibrated	0.400*	0.204*	0.594*	0.199*	0.037	**0.972***
	Calibrated (Recall=0.95)	0.095*	0.937*	**0.729**	0.373	0.519*	0.500*
	Calibrated (Recall=1)	0.068*	0.981*	0.630*	0.322	0.889*	0.266*

7b-ins performs the same or better in 60% of cases for success rate, and in 40% of cases for average recall.

Calibration using seed documents is evaluated only on the Seed Collection, as seed studies are not available in CLEF TAR. Under this method, *LlaMa2-7b-ins* again shows superior recall: 70% of topics achieve perfect recall, compared to 50% using the extrapolation approach. While calibration with seed documents generally improves recall, *LlaMa2-13b-ins* demonstrates more variability—possibly due to inconsistency in the quality and quantity of seed documents across topics.

Effectiveness of Model Ensembling

Table 8.5 and Table 8.6 also presents the Ensemble results, obtained by ensambling *LlaMa2-7b-ins*, *LlaMa2-13b-ins* and the *BioBERT* baseline. The Ensemble strategy yields consistently higher B-AC and WSS, when calibrated. Moreover, when pitted against individual generative LLMs calibrated with the same threshold recall, the Ensemble method obtains higher WSS, precision, and F3. Exceptions are observed in CLEF-2018 and Seed Collection, where the Ensemble strategy registers lower success rates. Interestingly, the Ensemble's performance dips in recall when not calibrated. This decline may be attributed to the model's aggressive document exclusion strategy, as evidenced by its consistently high WSS across datasets. Overall, our findings indicate that a calibrated Ensemble approach generally outperforms single generative LLMs.

Comparison with Fine-Tuned LLMs

Although this study aimed to investigate the effectiveness of zero-shot generative LLMs in systematic review document screening, we are also interested in comparing our method to the state-of-the-art fine-tuned model. For this comparison, we consider the Bio-SIEVE approach, a fine-tuned model for

systematic review document screening, and compare it with our best methods in Table 8.7.[2]

We also apply our calibration method to Bio-SIEVE. Notably, *LlaMa2-7b-ins* achieves comparable balanced accuracy (B-AC) to Bio-SIEVE, and our ensemble method slightly outperforms it, though the differences are not statistically significant.

A more critical observation is Bio-SIEVE's low recall and success rate when used without calibration. This is problematic in the context of systematic review screening, where high recall is essential to avoid missing relevant studies. While calibration improves Bio-SIEVE's recall, it still underperforms compared to our zero-shot models under the same calibration setting. These results suggest that although fine-tuning can improve general classification performance, it must be paired with careful calibration to be practically effective for systematic reviews.

8.4 Summary of Findings

We conducted a comprehensive evaluation of zero-shot generative LLMs for systematic review document screening and introduced a calibration method to adjust model outputs for improved reliability. In addition, we explored an ensemble approach that combines the strongest zero-shot LLMs with a BioBERT baseline.

Our results highlight the critical role of output calibration in applying generative LLMs to this task. Calibration enables the system to consistently meet pre-defined recall targets, which is essential for maintaining the quality and completeness of systematic reviews. It also provides flexibility to adapt the model to the specific recall requirements of different reviews. Notably, our calibrated ensemble approach outperforms the current state-of-the-art fine-tuned model, Bio-SIEVE [195].

We also demonstrated that instruction-based fine-tuning is beneficial in enhancing LLM effectiveness for screening, whereas QLoRA-based tuning did not yield useful results in this context.

Overall, our findings suggest that zero-shot LLM-based methods can support automatic document screening in systematic reviews, offering substantial reductions in manual effort without the need for expensive fine-tuning (either computationally or in terms of annotation cost). The consistent high recall achieved across diverse review topics indicates that such methods are approaching the level of maturity required for practical adoption in systematic review workflows.

[2]Comparison is however not straightforward as Bio-SIEVE used most of the datasets we consider here for fine-tuning; we then evaluate effectiveness using the only 27 topics from CLEF-TAR that were not used to fine-tune Bio-SIEVE.

Chapter 9

Conclusion

This thesis consideres three research directions: (1) Investigate the effectiveness of seed studies for automated systematic review screening, as discussed in Part I, (2) Explor AI techniques to automate or semi-automate the creation of systematic reivew Boolean query , as discussed in Part II, (3) Exam AI techniques to speed up the systematic review screening process (including screening prioritisation and screen automation), as detailed in Part III.

This chapter summarises the findings from the three research directions and highlights respective future research opportunities and implications. Additionally, it presents a broader outlook on the automation of medical systematic reviews, discussing the challenges and potential impacts of adopting such technologies in practice.

9.1 Exploiting Seed Studies

Seed studies are commonly used during the early stages of a systematic review to help guide topic formulation. They can support processes such as PICO extraction [1], the design of Boolean queries, and the definition of inclusion and exclusion criteria. Beyond their original use, seed studies can also play a crucial role in accelerating the review process when integrated with AI-based tools. For instance, they can be used as exemplar queries to improve the reranking of studies retrieved via Boolean queries, thereby enhancing the precision of initial search results [213]. This section provides a detailed overview of our investigations into how seed studies can be effectively exploited throughout the review workflow. We also discuss implications for future research, including the integration of seed-based retrieval with large language models and iterative query refinement techniques.

[1] PICO stands for Population, Intervention, Comparator, and Outcome. These components are used to frame clinical research questions in a structured format. Extracting PICO elements from seed studies involves identifying how each of these components is represented in the study (e.g., who the patients are, what intervention was applied, what it was compared against, and what outcomes were measured), which can inform the scope of the review and the design of search queries.

9.1.1 Summary of Findings

Reproducing Seed-Driven Document Reranking (SDR). SDR is an approach that uses seed studies as exemplars to rerank retrieved documents, with the goal of improving the relevance of results in systematic reviews. We conducted a comprehensive evaluation of SDR across multiple CLEF TAR datasets and found that the original claims generally hold: using the SDR method improves the effectiveness of document ranking, and further fusion with AES—an embedding-based ranking approach—leads to even greater improvements. In the original SDR implementation, one key idea is the use of a Bag of Clinical Words (BOC) representation, where both seed studies and retrieved documents are represented using only clinical terms extracted from UMLS. Our experiments confirmed that employing this BOC representation, along with appropriate term weighting, can enhance the overall effectiveness of SDR. However, the fused SDR+AES method showed limited effectiveness on the CLEF TAR 2019 (intervention) topics. We hypothesise that this is due to topic-type differences: CLEF TAR 2017 and 2018 included diagnostic accuracy topics, while 2019 focused on intervention studies, which may exhibit different language and evidence structures. In addition to reproducing the single-seed setup used in prior work, we explored a variant called multi-SDR, which integrates multiple seed studies during reranking. Although multi-SDR showed higher average effectiveness compared to single-SDR, its performance dropped significantly when compared to an oracle-selected single seed study. This finding suggests that identifying the "best" seed study within a candidate pool is a critical direction for future research.

Test Collection with Real Seed Studies. While many prior works—including the original SDR study and our reproduction—refer to the use of "seed studies," these are often simulated by drawing from the included studies of a systematic review due to limitations in available datasets. In other words, the so-called seed studies are selected retrospectively from the final set of included articles, rather than identified prospectively as in real-world workflows. This poses a methodological problem: using included studies as seeds gives the model an unrealistic advantage, as it assumes prior knowledge of which studies are relevant. As a result, such simulations tend to overestimate performance, particularly in tasks like query formulation, ranking, or screening, where initial uncertainty is a defining characteristic of real review settings. To address this gap, we developed and released a realistic test collection containing *real* seed studies, drawn from the actual documents available to reviewers at the start of the process. This resource enables more faithful evaluation of seed-based systematic review automation methods under conditions that better mirror practical use. We have evaluated the seed collection using two tasks: Automatic Boolean query formulation [213, 214] and SDR method.

For *automatic Boolean query formulation*, we followed previous work by Scells et al. [214] for the concept-based approach and Scells et al. [213] for the objective-based method, and reproduced both using real seed studies. In our experiments, both methods showed that using real seed studies consistently outperforms the use of pseudo seed studies in the Boolean query formulation task. This finding highlights that real seed studies—likely the same studies used in the initial design of the systematic review's search strategy—enable a more realistic and meaningful evaluation of query formulation methods, as they better reflect how such methods would be used in practice. For *SDR*

method, empirical results using this collection revealed that relying on pseudo seeds can significantly overestimate retrieval performance. By contrast, *leveraging real seed studies* demonstrates tangible benefits for systematic review automation, particularly when the goal is to reduce screening workload without sacrificing recall.

This resource has also been adopted in both our own research—such as LLM-based Boolean query automation (Chapter 5), Screening Prioritisation (Chapter 7) LLM-based classification (Chapter 8)—as well as by other researchers [33, 90, 152, 217]. These studies use the resource to evaluate and refine seed-driven techniques, ultimately contributing to more robust and practically useful methods for automated evidence retrieval.

9.1.2 Future Research Directions

Active Learning with Real Seed Studies. Active learning has shown promise for systematic review automation [48, 250], but most prior work relies on artificially generated or pseudo seed sets that do not reflect actual review workflows. Incorporating *real* seed studies into active learning pipelines would enable models to learn from genuine examples identified at the start of the review process, rather than from retrospectively selected included studies. In this approach, an active learning system would iteratively query domain experts or information specialists to label uncertain documents, using their feedback to refine both the retrieval model and the working set of seed studies. This iterative loop could substantially improve review efficiency by continuously adapting search strategies based on authentic expert input. A particularly promising extension would explore *hybrid approaches* that combine real seed studies with pseudo seed studies (i.e., already-screened included studies). Our analysis has shown that real seed studies differ meaningfully from included studies in their characteristics and coverage. A hybrid method could leverage the broader terminology and diverse perspectives of real seed studies during initial retrieval, while incorporating signals from included studies identified during screening to refine ranking of remaining documents. This combination may yield more effective pipelines for real-world systematic review automation, as it captures both the exploratory value of authentic seed studies and the targeted relevance signals from confirmed inclusions.

Seed Study Selection for single-SDR. While multi-SDR has generally improved performance compared to single-SDR, our experiments also showed that an effectively chosen *single* seed study can occasionally match or exceed multi-SDR outcomes. Future work should therefore develop new methods for *seed study selection* that automatically identify the most beneficial subset (or even a single best seed) for a particular review topic. Approaches may include leveraging machine learning to estimate each seed's marginal contribution to retrieval performance or employing lightweight topic modelling to cluster seeds based on thematic coverage. This adaptive selection could yield a more robust multi-SDR framework that avoids unnecessary redundancy and maximises retrieval gains.

the effectiveness depends on topic characteristics. Mixture approaches could potentially leverage the broader terminology coverage of real seed studies while benefiting from the refined scope of included studies. Such hybrid methods might adaptively weight or select from both sources based on

topic-specific features, potentially achieving better performance than either source in isolation.

Overall, our exploration of seed studies underscores their significant potential in streamlining the retrieval process for systematic reviews. The strategies, test collections, and methods presented in this thesis can form the basis for more advanced, robust, and realistic applications of seed-driven techniques in future work.

9.2 Enhancing Query Formulation

Formulating high-quality search queries is central to the success of systematic review creation. Traditionally, Boolean queries have relied on the expertise of information specialists to balance recall and precision. Recent advances in large language models and pretrained BERT-based retrieval models, offer new ways to automate or augment this process. This section summarises our major findings on enhancing query formulation and then discusses future avenues to build on these insights.

9.2.1 Summary of Findings

LLM-based Boolean Query Generation. We explored the use of LLMs for generating zero-shot Boolean queries, finding that they can achieve notably higher precision compared to many existing automatic methods. However, a consistent trade-off was observed: these LLM-derived queries sometimes compromised recall, a critical metric in systematic reviews where missing relevant studies can lead to incomplete or biased conclusions. Guided prompts that provided more specific instructions to the LLM helped mitigate these shortcomings, suggesting the importance of carefully designed query-generation prompts. However, guided prompt works only if the best seed study can be selected in a posthoc fashion: this finding further strengthens the importance of seed study selection in an effective systematic review automation pipeline. Additionally, a core challenge in applying LLMs to query formulation is *reproducibility*. Even with the same input prompt, LLMs can produce varying queries due to their probabilistic nature. This variability poses practical challenges in systematic reviews, where transparency and repeatability are essential for trustworthiness. Without robust controls or constraints, these generative models risk undermining key principles of evidence-based methodologies.

Medical Subject Headings (MeSH) Term Suggestion. Beyond free-text query generation, we introduce and evaluate both lexical- and BERT-based approaches for suggesting MeSH terms. Our results show that BERT-based methods generally outperform lexical approaches in identifying relevant MeSH terms. Notably, BERT-based suggestion methods also outperform ATM, the algorithm originally deployed in PubMed for automatic MeSH term assignment [1].

9.2.2 Future Research Directions

Reinforcement Learning (RL) for Query Formulation. Recent work has shown that reasoning-oriented LLMs trained with reinforcement learning (RL) can significantly enhance output quality across various tasks [58]. However, this paradigm has not yet been applied to the specific task of

Boolean query formulation for systematic reviews. Existing RL-based approaches typically focus on optimising final outputs using high-level rewards, but offer limited control or interpretability over the model's internal reasoning process.

An open question is whether an RL-based model can be trained to formulate Boolean queries by using feedback from the set of included studies (e.g., as a reward signal), *without* requiring explicit supervision over the generation process or intermediate reasoning steps. Such an approach could better align the model's behaviour with the goals of systematic review automation, potentially leading to more transparent, effective, and adaptable query construction.

Boolean Formulation with MeSH Editing. We have developed and evaluated several MeSH term suggestion methods that achieve promising accuracy, but still require manual refinement to ensure correctness when used in large-scale Boolean queries. Meanwhile, Boolean queries generated by LLMs often suffer from incorrect or hallucinated MeSH terms, as specified in Chapter 5, a known limitation of LLMs [103]. As a result, automatically formulated Boolean queries may incorporate erroneous or suboptimal controlled vocabulary. Inspired by recent advances in posthoc response editing [75], future work could explore an *integrated* pipeline that couples automated Boolean query generation with real-time MeSH term validation and correction. In this setup, a dedicated submodule could work as a tool to dynamically identify and revise inaccurate MeSH terms during query construction. Such an approach would reduce the manual burden on information specialists and significantly improve the reliability and quality of automatically generated queries.

Human-AI-Collaborative Query Formulation. While fully automated approaches promise efficiency gains, systematic review query formulation may benefit most from a *human-in-the-loop* paradigm that leverages the complementary strengths of both information specialists and AI systems. Current automated methods somtimes struggle to capture domain-specific nuances, while purely manual approaches remain time-intensive and may miss optimisation opportunities that machine learning models could identify. A collaborative framework could operate through several mechanisms:

- *Interactive Query Refinement:* LLMs could propose initial Boolean queries that information specialists iteratively refine through a conversational interface, with the model learning from expert feedback to suggest progressively improved formulations. This approach would preserve human expertise in the loop while reducing the cognitive burden of constructing complex queries from scratch.

- *Explanation and Justification:* AI systems could provide transparent reasoning for their query suggestions, highlighting which terms target specific inclusion criteria or explain the rationale behind particular Boolean operators. This interpretability would enable specialists to quickly validate or correct automated suggestions, fostering trust and facilitating knowledge transfer.

- *Active Learning for Query Components:* Rather than generating complete queries, AI systems could suggest individual query components (search terms, MeSH headings, Boolean combinations) ranked by predicted utility, allowing experts to selectively incorporate elements they judge most appropriate while maintaining control over the final query structure.

Such collaborative systems would address key limitations of fully automated approaches—namely, lack of domain adaptability and reproducibility concerns—while substantially reducing the manual effort required in traditional query formulation. Moreover, interaction logs from these systems could generate valuable training data for future model improvements, creating a virtuous cycle of human-AI collaboration.

Hybrid Automatic Approaches: Combining Traditional and LLM-based Methods. While LLM-generated queries can excel in precision, they sometimes struggle with recall. Conversely, traditional lexical or structured approaches (e.g., Boolean expansions) can maintain broad coverage but risk retrieving large quantities of irrelevant documents. A promising direction involves *hybrid methods* that fuse the strengths of both worlds:

- *Semantic and Lexical Fusion:* Combine lexical indicators (e.g., BM25, keyword expansions) with deep semantic signals (e.g., BERT-based embeddings), potentially improving retrieval across a wide range of topics.

- *Dynamic Re-ranking:* Use LLM-generated queries to refine a first-pass retrieval from a lexical approach, or vice versa, based on iterative feedback or known relevant seeds.

Effective hybrid systems might yield a more balanced performance profile, improving both recall and precision without excessively increasing reviewer workload.

Overall, our work on query formulation demostrates the utility of advanced NLP techniques and the need to balance innovation with the reproducibility requirements of systematic review methodology. In combination with active learning, advanced seed-driven retrieval, and other approaches, LLM-based query generation and semantic enrichment hold substantial promise for shaping the next generation of automated systematic review platforms.

9.3 Optimising Screening

Screening is a critical step in systematic reviews, as it determines which documents will be included for further assessment and which can be safely discarded. It is also typically the most time-consuming phase in the systematic review process, often requiring the manual assessment of thousands of documents. Therefore, improvements in screening efficiency and effectiveness can substantially reduce the time and cost associated with large-scale evidence synthesis. This section discusses our findings on optimising screening methods using state-of-the-art neural rankers and large language models (LLMs), followed by opportunities for future research.

9.3.1 Summary of Findings

Effectiveness of Fine-Tuned Neural Rankers for Screening Prioritisation. Screening prioritisation refers to the task of ranking documents so that the most relevant ones are reviewed earlier in the screening process. By identifying eligible studies earlier, this approach can accelerate downstream

review stages—such as full-text screening, data extraction, and analysis—even if the total screening workload remains unchanged. These downstream tasks are often carried out by separate teams of researchers, so early identification of relevant studies can help parallelise efforts and improve overall review efficiency. As a result, screening prioritisation contributes to reducing the total time required to complete a systematic review, leading to more timely publication of systematic review findings. In our experiments, we find that neural BERT-based rankers fine-tuned on relatively small labeled datasets demonstrated substantial improvements over zero-shot rankers. In many cases, their effectiveness approached that of established iterative screening methods. These findings underscore the practicality of integrating neural models into the screening process, as fine-tuning can be accomplished with limited annotation effort.

Boolean Queries vs. LLM-Generated Queries for Screening Prioritisation. Much prior work on screening prioritisation—including in our own baseline experiments detailed in the previous point—relied on the assumption that a final review title could be used to rank documents. However, this setup is inherently *post hoc*, since such titles are only available after the review has been completed. In realistic screening workflows, reviewers typically begin with manually constructed Boolean queries to retrieve candidate documents. To model this early-stage scenario more faithfully, we evaluated the use of Boolean queries as inputs for screening prioritisation. Directly using Boolean queries as inputs to neural rankers, however, proved largely ineffective. This is likely due to several factors: their complex and rigid logical structure, unnatural syntax, and—crucially—their length, which often exceeds the 512-token input limit of BERT-based rankers. As a result, important parts of the query may be truncated or poorly represented in the model's input encoding.

To overcome these limitations, we explored whether LLMs could translate Boolean expressions into natural language queries more compatible with neural rankers. The LLM-generated queries consistently outperformed both the original Boolean queries and the working titles used in prior simulations. Notably, their performance approached that of the *final review titles*, despite being based solely on inputs available at the beginning of the screening process. Additionally, combining rankings from Boolean and LLM-generated queries via a simple rank-fusion method led to further gains, suggesting the two input types offer complementary information. However, we also note that this approach introduces new challenges: LLM-generated queries can vary significantly across generations due to prompt sensitivity and inherent randomness, leading to inconsistent performance. Addressing this variability will be essential for practical deployment. Future work may explore ensemble methods or selective query fusion strategies that aggregate multiple LLM-generated variants to improve robustness in screening workflows.

Calibration for LLM-Driven Automated Screening. We investigated the use of LLMs to directly assess whether a retrieved document should be included or excluded—entirely bypassing human reviewers. This zero-shot classification approach has the potential to significantly reduce the labour involved in screening. However, we found that it often fails to meet the high recall requirements expected in systematic reviews, where missing even a small number of relevant studies can undermine the validity of the review.

To address this, we explored simple forms of calibration, such as adjusting decision thresholds or using model confidence scores, which can substantially improve recall. While these strategies bring automated screening closer to practical utility, they also introduce new sources of complexity: thresholds must be carefully tuned, and performance can vary across domains. Thus, although LLM-driven screening is promising, it remains imperfect and should be approached with caution in high-stakes review settings. Importantly, this work considers a relatively simple, pointwise classification approach—where documents are judged independently and in isolation. It provides a foundation for more sophisticated future directions, including agentic LLMs that engage in multi-step decision-making, interactive query refinement, or justification-based screening. As model capabilities advance, we expect this line of work to evolve into richer, more dynamic screening pipelines that integrate planning, context-awareness, and real-time human feedback.

9.3.2 Future Research Directions

Low-Latency Iterative Screening. Traditional iterative screening methods can be time-consuming or computationally demanding, making them difficult to deploy in real-world workflows where expert assessors review studies in near real time. Future work should focus on *low-latency* iterative approaches that can:

- Rapidly retrain or update the ranking model after each batch of expert feedback.
- Efficiently incorporate new relevance assessments without requiring a complete reprocessing of the dataset.

Techniques like online learning or incremental model updates could help ensure that human experts need not wait extended periods for updated screening ranks, thus facilitating more fluid collaboration between human and machine.

Query Performance Prediction. Multiple queries (e.g., from different LLM prompts) can yield diverse results, both in recall and precision. Developing task-specific *query performance predictors* for systematic reviews would:

- Help identify high-quality queries with minimal user intervention.
- Guide adaptive ensemble methods by weighting or selecting the most promising query strategies.

Although query performance prediction has been studied in general information retrieval [12], it remains under-explored in systematic review contexts, where missing a relevant study has high costs.

End-to-End Training. Many current methods treat query formulation and document ranking as discrete steps, often relying on final review titles or extensive domain-specific fine-tuning. A promising research direction is *end-to-end training*:

- *Integrated Pipeline:* Instead of separating query generation from the ranking step, we could train both components simultaneously, so the model learns to generate queries that directly optimise downstream screening performance.

- *Avoiding Final Titles:* Research in systematic review automation should rely less on finalised systematic review titles, which are often only available *post hoc*, and more on automatically generated or partially specified queries that can still guide effective retrieval.

An end-to-end approach could potentially remove the dependency on manual or post-completion inputs, paving the way for a more autonomous and continuous screening workflow.

Automatic Screening Using Reasoning Models. While existing LLM-based methods can achieve acceptable recall with calibration (e.g., recall satisfaction thresholds), they often suffer from low precision and lack a transparent reasoning pipeline for inclusion/exclusion decisions. Future work could consider:

- *RL-Driven Reasoning:* Train reinforcement learning models to optimise a "reasoning" step by assigning rewards based on agreement with final inclusion/exclusion decisions drawn from actual systematic review data.

- *Structured Decision Pipelines:* Integrate domain knowledge or hierarchical rules (e.g., PICO elements) into the reasoning process, offering more interpretable justifications for automated decisions.

A robust RL-based reasoning model might significantly enhance precision by explicitly modelling how complex criteria are applied, thereby making automatic screening both more accurate and more transparent.

Overall, the exploration of neural rankers, query generation, and ensemble methods in this thesis illustrates the viable paths toward more efficient, accurate screening processes for systematic reviews. By focusing on low-latency updates, better query performance prediction, and fully integrated training pipelines, future research can further reduce the manual burden on expert reviewers while maintaining the high level of recall vital for evidence-based decision-making.

9.4 Broader Outlook

The three research directions explored in this thesis—exploiting seed studies, advancing query formulation, and optimising screening—demonstrate the promise of more realistic, adaptable, and effective systematic review automation. Our findings highlight:

- The necessity of realistic resources and evaluation frameworks (e.g., test collections) that account for the complexities of real-world review creation.

- The potential for Large Language Models and other neural methods to generate high-performance queries or ranking approaches, given appropriate control and reproducibility measures.

- The value of ensemble and calibration strategies to ensure essential recall targets while taking advantage of state-of-the-art retrieval methods.

As systematic reviews grow in volume and complexity, the demand for faster, more reliable automation will only increase. This thesis offers a robust foundation for future research on low-latency active learning, more sophisticated MeSH-based enhancements, multi-SDR methods without oracle selection, and fully integrated end-to-end pipelines. By continuing to refine these AI-driven tools and workflows, researchers can significantly reduce the time and cost of conducting systematic reviews, ultimately improving the timeliness and quality of health and policy-related decision-making.

9.5 Discussion and Limitations

This section discusses the broader implications and limitations encountered during my PhD process, reflecting upon the evolving landscape of IR research community and its impacts on the conducted studies in this thesis.

Firstly, the trajectory of my PhD highlights significant shifts within the IR research community. At the beginning of my PhD in 2021, the dominant focus was on encoder-based models, such as BERT, which works at the best after tuning with numerous samples for generating embeddings, ranking documents, and performing classification tasks [61]. These encoder-based models consistently outperformed earlier methods, establishing a new state-of-the-art standard. Although decoder-based models, such as GPT-2, existed at that time, their application within IR was less prevalent [185, 186]. This was primarily due to their architectural design, which is optimised for generative tasks rather than the discriminative tasks central to IR applications. Consequently, encoder-based models remained the preferred choice for IR tasks during that time.

However, the release of ChatGPT in late 2022 marked a significant turning point, rapidly shifting research interest toward decoder-only models capable of instruction-following, generating coherent and contextually relevant outputs, and functioning effectively as assistant agents or direct-answer generators. Consequently, this shift had a substantial influence on my systematic review automation research. Many of the earlier methods developed in this thesis are heavily rooted in encoder-based (BERT-based) approaches, such as those works focusing on MeSH Term suggestion and screening prioritisation. In contrast, methods developed later in my PhD increasingly leveraged the capabilities of decoder-only models, such as those that uses LLMs to formulate Boolean query directly, or directly replacing human labours for screening. Notably, our SIGIR paper [264] was among the first to demonstrate the potential of ChatGPT in automating Boolean query generation for systematic reviews.

This shift inherently introduces limitations into the conducted research. The methods developed and evaluated earlier in the PhD may not fully leverage the latest advancements in generative models, potentially limiting their effectiveness and generalizability in contemporary applications. Moreover, rapid advancements in generative AI technology suggest that techniques described later in the thesis might quickly be superseded by more advanced approaches. As a demonstration of this, our extensive 2025 evaluation of LLMs for generating systematic review Boolean queries was not a direct reproduction of our original 2023 work [264], but of a 2024 reproduction of that work by Staudinger et al. [230]—underscoring how rapidly methods in this area evolve and require continual re-evaluation.

Additionally, another limitation of this thesis stems from the datasets and benchmarks used. Systematic review datasets typically evolve slowly, making it challenging to assess the cutting-edge effectiveness of newly developed methods. Furthermore, systematic review processes rely heavily on domain-specific expertise, posing a challenge to fully capture the nuances and complexities within automated systems, especially when evaluating across multiple domains or contexts.

Lastly, the inherent variability and non-determinism observed in generative models, particularly decoder-based LLMs, pose practical challenges. Reproducibility and consistent effectiveness remain significant concerns, as demonstrated by variability in query generation effectiveness across repeated experiments with the same models and prompts.

In summary, the evolution of the IR community during the PhD process from encoder-based to generative decoder-based models has shaped the research approaches and introduced specific limitations. Addressing these limitations requires ongoing research, continual refinement of evaluation methods, and adapting to the rapidly evolving landscape of IR and generative AI.

Bibliography

[1] Pubmed tutorial: Automatic term mapping. `https://www.nlm.nih.gov/bsd/disted/pubmedtutorial/020_040.html`. Accessed: 2020-02-01.

[2] Mustafa Abualsaud, Nimesh Ghelani, Haotian Zhang, Mark D Smucker, Gordon V Cormack, and Maura R Grossman. A system for efficient high-recall retrieval. In *Proceedings of the 41st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1317–1320, 2018.

[3] Lisa Affengruber, Miriam M van der Maten, Isa Spiero, Barbara Nussbaumer-Streit, Mersiha Mahmić-Kaknjo, Moriah E Ellen, Käthe Goossen, Lucia Kantorova, Lotty Hooft, Nicoletta Riva, et al. An exploration of available methods and tools to improve the efficiency of systematic review production: a scoping review. *BMC Medical Research Methodology*, 24(1):210, 2024.

[4] Maristella Agosti, Giorgio Maria Di Nunzio, and Stefano Marchesin. An analysis of query reformulation techniques for precision medicine. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 973–976, 2019.

[5] Maristella Agosti, Giorgio Maria Di Nunzio, Stefano Marchesin, et al. A post-analysis of query reformulation methods for clinical trials retrieval. In *SEBD*, pages 152–159, 2020.

[6] Amal Alharbi and Mark Stevenson. Ranking abstracts to identify relevant evidence for systematic reviews: The university of sheffield's approach to CLEF eHealth 2017 task 2. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, 2017.

[7] Amal Alharbi and Mark Stevenson. A dataset of systematic review updates. In *Proceedings of the 42nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1257–1260, New York, NY, USA, 2019. Association for Computing Machinery.

[8] Amal Alharbi and Mark Stevenson. Ranking studies for systematic reviews using query adaptation: University of sheffield's approach to clef ehealth 2019 task 2 working notes for clef

2019. In *Working Notes of CLEF 2019-Conference and Labs of the Evaluation Forum*, volume 2380. CEUR Workshop Proceedings, 2019.

[9] Amal Alharbi and Mark Stevenson. Refining boolean queries to identify relevant studies for systematic review updates. *Journal of the American Medical Informatics Association*, 27(11): 1658–1666, 2020.

[10] Amal Alharbi, William Briggs, and Mark Stevenson. Retrieving and ranking studies for systematic reviews: University of Sheffield's approach to CLEF eHealth 2018 Task 2. In *CEUR Workshop Proceedings: Working Notes of CLEF 2018: Conference and Labs of the Evaluation Forum*, volume 2125. CEUR Workshop Proceedings, 2018.

[11] Antonios Anagnostou, Athanasios Lagopoulos, Grigorios Tsoumakas, and Ioannis P Vlahavas. Combining inter-review learning-to-rank and intra-review incremental training for title and abstract screening in systematic reviews. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, 2017.

[12] Negar Arabzadeh, Maryam Khodabakhsh, and Ebrahim Bagheri. Bert-qpp: contextualized pre-trained transformers for query performance prediction. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 2857–2861, 2021.

[13] Edoardo Aromataris and Dagmara Riitano. Systematic reviews: constructing a search strategy and searching for evidence. *AJN The American Journal of Nursing*, 114(5):49–56, 2014.

[14] Alan R Aronson. Effective mapping of biomedical text to the UMLS Metathesaurus: The MetaMap program. In *Proceedings of the AMIA Symposium*, page 17. American Medical Informatics Association, 2001.

[15] Sungmin Aum and Seon Choe. srbert: automatic article classification model for systematic review using bert. *Systematic reviews*, 10(1):1–8, 2021.

[16] Maisie Badami, Boualem Benatallah, and Marcos Baez. Systematic literature review search query refinement pipeline: Incremental enrichment and adaptation. In *International Conference on Advanced Information Systems Engineering*, pages 129–146. Springer, 2022.

[17] Krisztian Balog. *Entity-oriented search*. Springer, 2018.

[18] Elaine Barnett-Page and James Thomas. Methods for the synthesis of qualitative research: a critical review. *BMC medical research methodology*, 9:1–11, 2009.

[19] Tanja Bekhuis, Dina Demner-Fushman, and Rebecca S Crowley. Comparative effectiveness research designs: an analysis of terms and coverage in medical subject headings (mesh) and emtree. *Journal of the Medical Library Association: JMLA*, 101(2):92, 2013.

[20] Olivier Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270, 2004.

[21] Rohit Borah, Andrew W Brown, Patrice L Capers, and Kathryn A Kaiser. Analysis of the time and workers needed to conduct systematic reviews of medical interventions using data from the prospero registry. *BMJ open*, 7(2):e012545, 2017.

[22] Florian Boudin, Jian-Yun Nie, and Martin Dawes. Clinical information retrieval using document and pico structure. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 822–830. Association for Computational Linguistics, 2010.

[23] Wichor M Bramer, Dean Giustini, Gerdien B de Jonge, Leslie Holland, and Tanja Bekhuis. De-duplication of database search results for systematic reviews in EndNote. *Journal of the Medical Library Association*, 104(3):240, 2016.

[24] Wichor M Bramer, Melissa L Rethlefsen, Jos Kleijnen, and Oscar H Franco. Optimal database combinations for literature searches in systematic reviews: a prospective exploratory study. *Systematic reviews*, 6:1–12, 2017.

[25] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, and Amanda Askell. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[26] Krystal Bullers, Allison M Howard, Ardis Hanson, William D Kearns, John J Orriola, Randall L Polo, and Kristen A Sakmar. It takes longer than you think: Librarian time spent on systematic review tasks. *Journal of the Medical Library Association*, 106(2):198, 2018.

[27] Max W Callaghan and Finn Müller-Hansen. Statistical stopping criteria for automated screening in systematic reviews. *Systematic Reviews*, 9(1):1–14, 2020.

[28] Kathi Canese and Sarah Weis. Pubmed: the bibliographic database. *The NCBI handbook*, 2(1), 2013.

[29] Gabriele Capannini, Claudio Lucchese, Franco Maria Nardini, Salvatore Orlando, Raffaele Perego, and Nicola Tonellotto. Quality versus efficiency in document scoring with learning-to-rank models. *Information Processing & Management*, 52(6):1161–1177, 2016.

[30] Andres Carvallo, Denis Parra, Hans Lobel, and Alvaro Soto. Automatic document screening of medical literature using word and text embeddings in an active learning setting. *Scientometrics*, 125:3047–3084, 2020.

[31] Andres Carvallo, Denis Parra, Gabriel Rada, Daniel Perez, Juan Ignacio Vasquez, and Camilo Vergara. Neural language models for text classification in evidence-based medicine. *arXiv preprint arXiv:2012.00584*, 2020.

[32] Kevin EK Chai, Robin LJ Lines, Daniel F Gucciardi, and Leo Ng. Research screener: a machine learning tool to semi-automate abstract screening for systematic reviews. *Systematic reviews*, 10:1–13, 2021.

[33] Gary CK Chan, Estrid He, Janni Leung, and Karin Verspoor. A comprehensive systematic review dataset is a rich resource for training and evaluation of ai systems for title and abstract screening. *Research Synthesis Methods*, pages 1–15, 2025.

[34] Jacqueline Chandler, Miranda Cumpston, Tianjing Li, Matthew J Page, and Vivian A Welch. *Cochrane Handbook for Systematic Reviews of Interventions*. John Wiley & Sons, 2019.

[35] Darlene Chapman. Health-related databases. *Journal of the Canadian Academy of Child and Adolescent Psychiatry*, 18(2):148, 2009.

[36] Jiayi Chen, Su Chen, Yang Song, Hongyu Liu, Yueyao Wang, Qinmin Hu, Liang He, and Yan Yang. ECNU at 2017 eHealth task 2: Technologically assisted reviews in empirical medicine. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, 2017.

[37] Miew Keen Choong, Filippo Galgani, Adam G Dunn, and Guy Tsafnat. Automatic evidence retrieval for systematic reviews. *Journal of medical Internet research*, 16(10):e223, 2014.

[38] Justin Clark. Systematic reviewing. In Gail M. Williams Suhail A. R. Doi, editor, *Methods of Clinical Epidemiology*. Springer, 2013.

[39] Justin Clark, Paul Glasziou, Chris Del Mar, Alexandra Bannach-Brown, Paulina Stehlik, and Anna Mae Scott. A full systematic review was completed in 2 weeks using automation tools: a case study. *Journal of Chronic Diseases*, 121:81–90, May 2020. ISSN 0895-4356. doi: 10.1016/j.jclinepi.2020.01.008.

[40] Justin Clark, Paul Glasziou, Chris Del Mar, Alexandra Bannach-Brown, Paulina Stehlik, and Anna Mae Scott. A full systematic review was completed in 2 weeks using automation tools: a case study. *Journal of clinical epidemiology*, 121:81–90, 2020.

[41] Justin Michael Clark, Sharon Sanders, Matthew Carter, David Honeyman, Gina Cleo, Yvonne Auld, Debbie Booth, Patrick Condron, Christine Dalais, Sarah Bateup, et al. Improving the translation of search strategies using the Polyglot Search Translator: A randomized controlled trial. *Journal of the Medical Library Association*, 108(2):195, 2020.

[42] Vincent Claveau. Neural text generation for query expansion in information retrieval. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 202–209, ESSENDON VIC Australia, December 2021. ACM. ISBN 978-1-4503-9115-3. doi: 10.1145/3486622.3493957.

[43] Aaron M Cohen and Neil R Smalheiser. Uic/ohsu clef 2018 task 2 diagnostic test accuracy ranking using publication type cluster similarity measures. In *CEUR Workshop Proceedings*, volume 2125, 2018.

[44] Aaron M Cohen, Kyle Ambert, and Marian McDonagh. A prospective evaluation of an automated classification system to support evidence-based medicine and systematic review. In *AMIA annual symposium proceedings*, volume 2010, page 121. American Medical Informatics Association, 2010.

[45] A.M. Cohen, W.R. Hersh, K. Peterson, and P.Y. Yen. Reducing workload in systematic review preparation using automated citation classification. *JAMIA*, 13(2):206–219, 2006.

[46] Francesco Colace, Massimo De Santo, Luca Greco, and Paolo Napoletano. Improving text retrieval accuracy by using a minimal relevance feedback. In *International Joint Conference on Knowledge Discovery, Knowledge Engineering, and Knowledge Management*, pages 126–140. Springer, 2011.

[47] Gordon V Cormack and Maura R Grossman. Evaluation of machine-learning protocols for technology-assisted review in electronic discovery. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 153–162, 2014.

[48] Gordon V Cormack and Maura R Grossman. Autonomy and reliability of continuous active learning for technology-assisted review. *arXiv preprint arXiv:1504.06868*, 2015.

[49] Gordon V Cormack and Maura R Grossman. Scalability of continuous active learning for reliable high-recall text classification. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 1039–1048, 2016.

[50] Gordon V Cormack and Maura R Grossman. Technology-assisted review in empirical medicine: Waterloo participation in CLEF eHealth 2017. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, 2017.

[51] Gordon V Cormack and Maura R Grossman. Technology-assisted review in empirical medicine: Waterloo participation in CLEF eHealth 2018. In *CEUR Workshop Proceedings: Working Notes of CLEF 2018: Conference and Labs of the Evaluation Forum*, 2018.

[52] Gordon V Cormack and Maura R Grossman. Systems and methods for conducting a highly autonomous technology-assisted review classification, March 12 2019. US Patent 10,229,117.

[53] 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.

[54] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. Ms marco: Benchmarking ranking models in the large-data regime. In *proceedings of the 44th International*

ACM SIGIR conference on research and development in information retrieval, pages 1566–1576, 2021.

[55] W Bruce Croft. Combining approaches to information retrieval. In *Advances in Information Retrieval*, pages 1–36. Springer, 2002.

[56] Ellen T Crumley, Natasha Wiebe, Kristie Cramer, Terry P Klassen, and Lisa Hartling. Which resources should be used to identify rct/ccts for systematic reviews: a systematic review. *BMC Medical Research Methodology*, 5:1–13, 2005.

[57] Jonathan J Deeks, Patrick M Bossuyt, Mariska M Leeflang, and Yemisi Takwoingi. *Cochrane handbook for systematic reviews of diagnostic test accuracy*. John Wiley & Sons, 2022.

[58] DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu,

Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. Deepseek-v3 technical report, 2025. URL `https://arxiv.org/abs/2412.19437`.

[59] Dina Demner-Fushman and Jimmy Lin. Answering clinical questions with knowledge-based and statistical techniques. *Computational Linguistics*, 33(1):63–103, 2007.

[60] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*, 2023.

[61] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[62] Giorgio Maria Di Nunzio. A study on a stopping strategy for systematic reviews based on a distributed effort approach. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 112–123. Springer, 2020.

[63] Giorgio Maria Di Nunzio, Federica Beghini, Federica Vezzani, and Geneviève Henrot. An interactive two-dimensional approach to query aspects rewriting in systematic reviews. IMS unipd at CLEF eHealth task 2. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, 2017.

[64] Giorgio Maria Di Nunzio, Giacomo Ciuffreda, and Federica Vezzani. Interactive sampling for systematic reviews. IMS unipd at CLEF 2018 eHealth task 2. In *CEUR Workshop Proceedings: Working Notes of CLEF 2018: Conference and Labs of the Evaluation Forum*, 2018.

[65] Mary Dixon-Woods, Shona Agarwal, David Jones, Bridget Young, and Alex Sutton. Synthesising qualitative and quantitative evidence: a review of possible methods. *Journal of health services research & policy*, 10(1):45–53, 2005.

[66] Mette Brandt Eriksen and Tove Faber Frandsen. The impact of patient, intervention, comparison, outcome (pico) as a search strategy tool on literature search quality: a systematic review. *Journal of the Medical Library Association: JMLA*, 106(4):420, 2018.

[67] ROGERS FB. Medical subject headings. *Bulletin of the Medical Library Association*, 51: 114–116, 1963.

[68] Rafael Ferreira, Mariana Leite, David Semedo, and Joao Magalhaes. Open-domain conversational search assistants: the transformer is all you need. *Information Retrieval Journal*, 25(2): 123–148, 2022.

[69] Yong Zhi Foo, Rose E O'Dea, Julia Koricheva, Shinichi Nakagawa, and Malgorzata Lagisz. A practical guide to question formation, systematic searching and study screening for literature reviews in ecology and evolution. *Methods in Ecology and Evolution*, 12(9):1705–1720, 2021.

[70] Edward A Fox and Joseph A Shaw. Combination of multiple searches. *NIST special publication SP*, 243, 1994.

[71] Tove Faber Frandsen, Mette Brandt Eriksen, David Mortan Grøne Hammer, Janne Buck Christensen, and Johan Albert Wallin. Using embase as a supplement to pubmed in cochrane reviews differed across fields. *Journal of Clinical Epidemiology*, 133:24–31, 2021.

[72] Maik Fröbe, Andrew Parry, Harrisen Scells, Shuai Wang, Shengyao Zhuang, Guido Zuccon, Martin Potthast, and Matthias Hagen. Corpus subsampling: Estimating the effectiveness of neural retrieval models on large corpora. In Claudia Hauff, Craig Macdonald, Dietmar Jannach, Gabriella Kazai, Franco Maria Nardini, Fabio Pinelli, Fabrizio Silvestri, and Nicola Tonellotto, editors, *Advances in Information Retrieval*, pages 453–471, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-88708-6.

[73] Luyu Gao and Jamie Callan. Condenser: a pre-training architecture for dense retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 981–993, 2021.

[74] Luyu Gao, Zhuyun Dai, and Jamie Callan. Rethink training of bert rerankers in multi-stage retrieval pipeline. In *European Conference on Information Retrieval*, pages 280–286. Springer, 2021.

[75] Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, et al. Rarr: Researching and revising what language models say, using language models. *arXiv preprint arXiv:2210.08726*, 2022.

[76] Luyu Gao, Xueguang Ma, Jimmy J. Lin, and Jamie Callan. Tevatron: An efficient and flexible toolkit for dense retrieval. *ArXiv*, abs/2203.05765, 2022.

[77] Chantelle Garritty, Gerald Gartlehner, Barbara Nussbaumer-Streit, Valerie J King, Candyce Hamel, Chris Kamel, Lisa Affengruber, and Adrienne Stevens. Cochrane rapid reviews methods group offers evidence-informed guidance to conduct rapid reviews. *Journal of clinical epidemiology*, 130:13–22, 2021.

[78] Lukas Gienapp, Harrisen Scells, Niklas Deckers, Janek Bevendorff, Shuai Wang, Johannes Kiesel, Shahbaz Syed, Maik Fröbe, Guido Zuccon, Benno Stein, Matthias Hagen, and Martin Potthast. Evaluating generative ad hoc information retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, pages 1916–1929, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400704314. doi: 10.1145/3626772.3657849. URL `https://doi.org/10.1145/3626772.3657849`.

[79] Roberto Gozalo-Brizuela and Eduardo C Garrido-Merchan. Chatgpt is not all you need. a state of the art review of large generative ai models. *arXiv preprint arXiv:2301.04655*, 2023.

[80] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin

Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager,

Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL `https://arxiv.org/abs/2407.21783`.

[81] Sally Green and J Higgins. Cochrane handbook for systematic reviews of interventions, 2005.

[82] T. Greenhalgh and R. Peacock. Effectiveness and efficiency of search methods in systematic reviews of complex evidence: Audit of primary sources. *Biomedical Journal*, 331(7524): 1064–1065, 2005.

[83] Ann T Gregory and A Robert Denniss. An introduction to writing narrative and systematic reviews—tasks, tips and traps for aspiring authors. *Heart, Lung and Circulation*, 27(7):893–898, 2018.

[84] Maura R Grossman and Gordon V Cormack. Technology-assisted review in e-discovery can be more effective and more efficient than exhaustive manual review. *Richmond Journal of Law & Technology*, 17(3):11, 2011.

[85] Maura R Grossman, Gordon V Cormack, and Adam Roegiest. Automatic and semi-automatic document selection for technology-assisted review. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 905–908, 2017.

[86] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing, 2020.

[87] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23, 2021.

[88] Nathalia Sernizon Guimarães, Andrêa JF Ferreira, Rita de Cássia Ribeiro Silva, Adelzon Assis de Paula, Cinthia Soares Lisboa, Laio Magno, Maria Yury Ichiara, and Maurício Lima Barreto. Deduplicating records in systematic reviews: there are free, accurate automated ways to do so. *Journal of Clinical Epidemiology*, 152:110–115, 2022.

[89] Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*, 2023.

[90] Fang Guo, Yun Luo, Linyi Yang, and Yue Zhang. Scimine: An efficient systematic prioritization model based on richer semantic information. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 205–215, 2023.

[91] Neal Haddaway, Matt Grainger, and Charles Gray. citationchaser: An R package and Shiny app for forward and backward citations chasing in academic searching, February 2021. URL `https://doi.org/10.5281/zenodo.4543513`.

[92] Neal R Haddaway and Martin J Westgate. Predicting the time needed for environmental systematic reviews and systematic maps. *Conservation Biology*, 33(2):434–443, 2019.

[93] Abdelhakim Hannousse and Salima Yahiouche. A semi-automatic document screening system for computer science systematic reviews. In *Mediterranean Conference on Pattern Recognition and Artificial Intelligence*, pages 201–215. Springer, 2022.

[94] Greg Harris, Anand Panangadan, and Viktor Prasanna. Interactive query refinement for boolean search. In *proceeding SemADoc Workshop*, 2014.

[95] Scells Harrisen and Zuccon Guido. Generating better queries for systematic reviews. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pages 475–484, New York, NY, USA, 2018. ACM.

[96] Elke Hausner, Siw Waffenschmidt, Thomas Kaiser, and Michael Simon. Routine development of objectively derived search strategies. *Systematic reviews*, 1(1):19, 2012.

[97] Elke Hausner, Charlotte Guddat, Tatjana Hermanns, Ulrike Lampert, and Siw Waffenschmidt. Development of search strategies for systematic reviews: Validation showed the noninferiority of the objective approach. *Journal of clinical epidemiology*, 68(2):191–199, 2015.

[98] Julian PT Higgins, James Thomas, Jacqueline Chandler, Miranda Cumpston, Tianjing Li, Matthew J Page, and Vivian A Welch. *Cochrane handbook for systematic reviews of interventions*. John Wiley & Sons, 2019.

[99] Matt Holland, Michelle Dutton, and Steve Glover. How it's done: search tools and techniques for major bibliographic databases. *Journal of Paramedic Practice*, 13(5):210–213, 2021.

[100] Noah Hollmann and Carsten Eickhoff. Ranking and feedback-based stopping for recall-centric document retrieval. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, 2017.

[101] Brian E Howard, Jason Phillips, Kyle Miller, Arpit Tandon, Deepak Mav, Mihir R Shah, Stephanie Holmgren, Katherine E Pelch, Vickie Walker, Andrew A Rooney, et al. Swift-review: a text-mining workbench for systematic review. *Systematic reviews*, 5:1–16, 2016.

[102] William Hsu, William Speier, and Ricky K Taira. Automated extraction of reported statistical analyses: Towards a logical representation of clinical trial literature. In *AMIA Annual Symposium Proceedings*, volume 2012, page 350, 2012.

[103] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, 2025.

[104] Veritas Heath Innovation. Covidence systematic review software. *Veritas Health Innovation Melbourne*, 2016.

[105] Rolf Jagerman, Honglei Zhuang, Zhen Qin, Xuanhui Wang, and Michael Bendersky. Query expansion by prompting large language models. *arXiv preprint arXiv:2305.03653*, 2023.

[106] Sampath Jayarathna, Atish Patra, and Frank Shipman. Unified relevance feedback for multi-application user interest modeling. In *Proceedings of the 15th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 129–138, 2015.

[107] Yunjie Ji, Yan Gong, Yiping Peng, Chao Ni, Peiyan Sun, Dongyu Pan, Baochang Ma, and Xiangang Li. Exploring chatgpt's ability to rank content: A preliminary study on consistency with human preferences. *arXiv preprint arXiv:2303.07610*, 2023.

[108] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL `https://arxiv.org/abs/2310.06825`.

[109] Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts, 2024. URL `https://arxiv.org/abs/2401.04088`.

[110] Ma Jie, Liu Ying, ZHONG Lai-ping, ZHANG Chen-ping, and ZHANG Zhi-yuan. Comparison between jadad scale and cochrane collaboration's tool for assessing risk of bias on the quality and risk of bias evaluation in randomized controlled trials. *China Journal of Oral & Maxillofacial Surgery*, 10(5), 2012.

[111] Jimmy, Guido Zuccon, Bevan Koopman, and Gianluca Demartini. Health card retrieval for consumer health search: An empirical investigation of methods. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, pages 2405–2408, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450369763. doi: 10.1145/3357384.3358128. URL `https://doi.org/10.1145/3357384.3358128`.

[112] Higgins JPTTJ, J Chandler, M Cumpston, T Li, MJ Page, and VA Welch. Cochrane handbook for systematic reviews of interventions version 6.4 (updated august 2023). cochrane; 2023, 2024.

[113] Evangelos Kanoulas, Dan Li, Leif Azzopardi, and Rene Spijker. CLEF 2017 technologically assisted reviews in empirical medicine overview. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, 2017.

[114] Evangelos Kanoulas, Rene Spijker, Dan Li, and Leif Azzopardi. Clef 2018 technology assisted reviews in empirical medicine overview. In *CLEF 2018 Evaluation Labs and Workshop: Online Working Notes, CEUR-WS*, 2018.

[115] Evangelos Kanoulas, Dan Li, Leif Azzopardi, and Rene Spijker. Clef 2019 technology assisted reviews in empirical medicine overview. In *CEUR Workshop Proceedings*, volume 2380, 2019.

[116] Rianne Kaptein, Jaap Kamps, and Djoerd Hiemstra. The impact of positive, negative and topical relevance feedback. Technical report, AMSTERDAM UNIV (NETHERLANDS), 2008.

[117] Sarvnaz Karimi, Justin Zobel, Stefan Pohl, and Falk Scholer. The challenge of high recall in biomedical systematic search. In *Proceedings of the 3rd International Workshop on Data and Text Mining in Bioinformatics*, pages 89–92, 2009.

[118] Sarvnaz Karimi, Stefan Pohl, Falk Scholer, Lawrence Cavedon, and Justin Zobel. Boolean versus ranked querying for biomedical systematic reviews. *BMC medical informatics and decision making*, 10(1):1–20, 2010.

[119] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick 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.

[120] Liz Kellermeyer, Ben Harnke, and Shandra Knight. Covidence and rayyan. *Journal of the Medical Library Association: JMLA*, 106(4):580, 2018.

[121] Keyvan Khosrovian, Dietmar Pfahl, and Vahid Garousi. Gensim 2.0: A customizable process simulation model for software process evaluation. In *International conference on software process*, pages 294–306. Springer, 2008.

[122] Peter Kokol and Helena Blažun Vošner. Discrepancies among scopus, web of science, and pubmed coverage of funding information in medical journal articles. *Journal of the Medical Library Association: JMLA*, 106(1):81, 2018.

[123] Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi-Rui Tam, Keith Stevens, Abdullah Barhoum, Nguyen Minh Duc, Oliver Stanley, Richárd Nagyfi, et al. Openassistant conversations–democratizing large language model alignment. *arXiv preprint arXiv:2304.07327*, 2023.

[124] A Khudyak Kozorovitsky and Oren Kurland. From"identical"to"similar": Fusing retrieved lists based on inter-document similarities. *Journal of Artificial Intelligence Research*, 41:267–296, 2011.

[125] Wojciech Kusa, Oscar E Mendoza, Matthias Samwald, Petr Knoth, and Allan Hanbury. Csmed: bridging the dataset gap in automated citation screening for systematic literature reviews. *Advances in Neural Information Processing Systems*, 36:23468–23484, 2023.

[126] Athanasios Lagopoulos, Antonios Anagnostou, Adamantios Minas, and Grigorios Tsoumakas. Learning-to-rank and relevance feedback for literature appraisal in empirical medicine. In *CEUR Workshop Proceedings: Working Notes of CLEF 2018: Conference and Labs of the Evaluation Forum*, pages 52–63. Springer, 2018.

[127] Victor Lavrenko and W Bruce Croft. Relevance-based language models. In *ACM SIGIR Forum*, volume 51, pages 260–267. ACM New York, NY, USA, 2017.

[128] Grace E. Lee and Aixin Sun. Seed-driven document ranking for systematic reviews in evidence-based medicine. In *Proceedings of the 41st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 455–464, 2018.

[129] Grace Eunkyung Lee. A study of convolutional neural networks for clinical document classification in systematic reviews: Sysreview at CLEF eHealth 2017. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, 2017.

[130] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

[131] Sharon Levy, Michael Saxon, and William Yang Wang. Investigating memorization of conspiracy theories in text generation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4718–4729, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.416. URL https://aclanthology.org/2021.findings-acl.416.

[132] Dan Li and Evangelos Kanoulas. When to stop reviewing in technology-assisted reviews: Sampling from an adaptive distribution to estimate residual relevant documents. *ACM Transactions on Information Systems (TOIS)*, 38(4):1–36, 2020.

[133] Hang Li, Harrisen Scells, and Guido Zuccon. Systematic review automation tools for end-to-end query formulation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2141–2144, 2020.

[134] Hang Li, Ahmed 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*, SIGIR '22, pages 2154–2158, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450387323. doi: 10.1145/3477495.3531822. URL https://doi.org/10.1145/3477495.3531822.

[135] Hang Li, Shuai Wang, Shengyao Zhuang, Ahmed Mourad, Xueguang Ma, Jimmy Lin, and Guido Zuccon. To interpolate or not to interpolate: Prf, dense and sparse retrievers. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, pages 2495–2500, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450387323. doi: 10.1145/3477495.3531884. URL https://doi.org/10.1145/3477495.3531884.

[136] Hang Li, Shuai Wang, Shengyao Zhuang, Ahmed Mourad, Xueguang Ma, Jimmy Lin, and Guido Zuccon. To interpolate or not to interpolate: Prf, dense and sparse retrievers. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, pages 2495–2500, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450387323. doi: 10.1145/3477495.3531884. URL https://doi.org/10.1145/3477495.3531884.

[137] Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. A survey of pretrained language models based text generation. *arXiv preprint arXiv:2201.05273*, 2022.

[138] Valentin Liévin, Christoffer Egeberg Hother, and Ole Winther. Can large language models reason about medical questions?, January 2023.

[139] Binsheng Liu, Nick Craswell, Xiaolu Lu, Oren Kurland, and J Shane Culpepper. A comparative analysis of human and automatic query variants. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 47–50, 2019.

[140] Vivian Liu and Lydia B Chilton. Design Guidelines for Prompt Engineering Text-to-Image Generative Models. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, pages 1–23, New York, NY, USA, April 2022. Association for Computing Machinery. ISBN 978-1-4503-9157-3. doi: 10.1145/3491102.3501825.

[141] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[142] Hannah A Long, David P French, and Joanna M Brooks. Optimising the value of the critical appraisal skills programme (casp) tool for quality appraisal in qualitative evidence synthesis. *Research Methods in Medicine & Health Sciences*, 1(1):31–42, 2020.

[143] Edward Loper and Steven Bird. Nltk: The natural language toolkit. Association for Computational Linguistics, 2002.

[144] Theo Lorenc, Lambert Felix, Mark Petticrew, GJ Melendez-Torres, James Thomas, Sian Thomas, Alison O'Mara-Eves, and Michelle Richardson. Meta-analysis, complexity, and heterogeneity: a qualitative interview study of researchers' methodological values and practices. *Systematic Reviews*, 5:1–9, 2016.

[145] Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. BioGPT: Generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6):bbac409, November 2022. ISSN 1467-5463, 1477-4054. doi: 10.1093/bib/bbac409.

[146] Xueguang Ma, Kai Sun, Ronak Pradeep, and Jimmy Lin. A replication study of dense passage retriever. *arXiv preprint arXiv:2104.05740*, 2021.

[147] Sean MacAvaney, Adam Roegiest, Aldo Lipani, Andrew Parry, Björn Engelmann Engelmann, Christin Katharina Kreutz, Chuan Meng, Erlend Frayling, Eugene Yang, Ferdinand Schlatt, et al. Report on the collab-a-thon at ecir 2024. In *ACM SIGIR Forum*, volume 58, pages 1–11. Association for Computing Machinery (ACM), 2024.

[148] Andrew MacFarlane, Tony Russell-Rose, and Farhad Shokraneh. Search strategy formulation for systematic reviews: Issues, challenges and opportunities. *Intelligent Systems with Applications*, page 200091, 2022.

[149] Donna Maglott, Jim Ostell, Kim D Pruitt, and Tatiana Tatusova. Entrez gene: gene-centered information at ncbi. *Nucleic acids research*, 33(suppl_1):D54–D58, 2005.

[150] Richard Mallett, Jessica Hagen-Zanker, Rachel Slater, and Maren Duvendack. The benefits and challenges of using systematic reviews in international development research. *Journal of development effectiveness*, 4(3):445–455, 2012.

[151] Rokiah Mamikutty, Ameera Syafiqah Aly, and Jamaludin Marhazlinda. Databases selection in a systematic review of the association between anthropometric measurements and dental caries among children in asia. *Children*, 8(7):565, 2021.

[152] Xinyu Mao, Shengyao Zhuang, Bevan Koopman, and Guido Zuccon. Dense retrieval with continuous explicit feedback for systematic review screening prioritisation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2357–2362, 2024.

[153] Iain J Marshall and Byron C Wallace. Toward systematic review automation: a practical guide to using machine learning tools in research synthesis. *Systematic reviews*, 8:1–10, 2019.

[154] Iain J Marshall, Rachel Marshall, Byron C Wallace, Jon Brassey, and James Thomas. Rapid reviews may produce different results to systematic reviews: a meta-epidemiological study. *Journal of clinical epidemiology*, 109:30–41, 2019.

[155] D. Martinez, S. Karimi, L. Cavedon, and T. Baldwin. Facilitating biomedical systematic reviews using ranked text retrieval and classification. In *Proceedings of the 13th Australasian Document Computing Symposium*, 2008.

[156] Jessie McGowan and Margaret Sampson. Systematic reviews need systematic searchers (IRP). *Journal of the Medical Library Association*, 93(1):74, 2005.

[157] Liam McKeever, Van Nguyen, Sarah J Peterson, Sandra Gomez-Perez, and Carol Braunschweig. Demystifying the search button: a comprehensive pubmed search strategy for performing an exhaustive literature review. *Journal of parenteral and enteral nutrition*, 39(6):622–635, 2015.

[158] Adamantios Minas, Athanasios Lagopoulos, and Grigorios Tsoumakas. Aristotle university's approach to the technologically assisted reviews in empirical medicine task of the 2018 CLEF eHealth lab. In *CEUR Workshop Proceedings: Working Notes of CLEF 2018: Conference and Labs of the Evaluation Forum*, 2018.

[159] M. Miwa, J. Thomas, A. O'Mara-Eves, and S. Ananiadou. Reducing systematic review workload through certainty-based screening. *Journal of Biomedical Informatics*, 51:242–253, 2014.

[160] SPFGH Moen and Tapio Salakoski2 Sophia Ananiadou. Distributional semantics resources for biomedical text processing. *Proceedings of LBM*, pages 39–44, 2013.

[161] Alistair Moffat, Falk Scholer, Paul Thomas, and Peter Bailey. Pooled evaluation over query variations: Users are as diverse as systems. In *proceedings of the 24th ACM international on conference on information and knowledge management*, pages 1759–1762, 2015.

[162] Muhammad Naeem, Wilson Ozuem, Kerry Howell, and Silvia Ranfagni. A step-by-step process of thematic analysis to develop a conceptual model in qualitative research. *International journal of qualitative methods*, 22:16094069231205789, 2023.

[163] Shinichi Nakagawa, Yefeng Yang, Erin L Macartney, Rebecca Spake, and Malgorzata Lagisz. Quantitative evidence synthesis: a practical guide on meta-analysis, meta-regression, and publication bias tests for environmental sciences. *Environmental Evidence*, 12(1):8, 2023.

[164] Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. Multi-stage document ranking with bert. *arXiv preprint arXiv:1910.14424*, 2019.

[165] Christopher Norman, Mariska Leeflang, and Aurélie Névéol. Limsi@ clef ehealth 2018 task 2: Technology assisted reviews by stacking active and static learning. In *CLEF (Working Notes)*, 2018.

[166] Christopher Norman12, Mariska Leeflang, and Aurélie Névéol. Limsi@ clef ehealth 2017 task 2: Logistic regression for automatic article ranking. In *CEUR Workshop Proceedings: Working Notes of CLEF 2019: Conference and Labs of the Evaluation Forum*, 2017.

[167] Natalya F Noy, Nigam H Shah, Patricia L Whetzel, Benjamin Dai, Michael Dorf, Nicholas Griffith, Clement Jonquet, Daniel L Rubin, Margaret-Anne Storey, Christopher G Chute, et al. Bioportal: ontologies and integrated data resources at the click of a mouse. *Nucleic acids research*, 37(suppl_2):W170–W173, 2009.

[168] Barbara Nussbaumer-Streit, Moriah Ellen, Irma Klerings, Raluca Sfetcu, Nicoletta Riva, Mersiha Mahmić-Kaknjo, Georgios Poulentzas, P Martinez, Eduard Baladia, Liliya Eugenevna Ziganshina, et al. Resource use during systematic review production varies widely: a scoping review. *Journal of clinical epidemiology*, 139:287–296, 2021.

[169] Reham Omar, Omij Mangukiya, Panos Kalnis, and Essam Mansour. Chatgpt versus traditional question answering for knowledge graphs: Current status and future directions towards knowledge graph chatbots. *arXiv preprint arXiv:2302.06466*, 2023.

[170] Alison O'Mara-Eves, James Thomas, John McNaught, Makoto Miwa, and Sophia Ananiadou. Using text mining for study identification in systematic reviews: A systematic review of current approaches. *Systematic reviews*, 4(1):5, 2015.

[171] Jonas Oppenlaender. A Taxonomy of Prompt Modifiers for Text-To-Image Generation, July 2022.

[172] Ramith Padaki, Zhuyun Dai, and Jamie Callan. Rethinking query expansion for bert reranking. In *European conference on information retrieval*, pages 297–304. Springer, 2020.

[173] Matthew J Page, Joanne E McKenzie, Patrick M Bossuyt, Isabelle Boutron, Tammy C Hoffmann, Cynthia D Mulrow, Larissa Shamseer, Jennifer M Tetzlaff, Elie A Akl, Sue E Brennan, et al.

The prisma 2020 statement: an updated guideline for reporting systematic reviews. *bmj*, 372, 2021.

[174] Joao Palotti, Guido Zuccon, Johannes Bernhardt, Allan Hanbury, and Lorraine Goeuriot. Assessors agreement: A case study across assessor type, payment levels, query variations and relevance dimensions. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction: 7th International Conference of the CLEF Association, CLEF 2016, Évora, Portugal, September 5-8, 2016, Proceedings 7*, pages 40–53. Springer, 2016.

[175] Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. The refinedweb dataset for falcon llm: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*, 2023.

[176] Yifan Peng, Shankai Yan, and Zhiyong Lu. Transfer learning in biomedical natural language processing: an evaluation of bert and elmo on ten benchmarking datasets. *arXiv preprint arXiv:1906.05474*, 2019.

[177] Mateus Pereira, Elham Etemad, and Fernando Paulovich. Iterative learning to rank from explicit relevance feedback. In *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, pages 698–705, 2020.

[178] Mark Petticrew, Eva Rehfuess, Jane Noyes, Julian PT Higgins, Alain Mayhew, Tomas Pantoja, Ian Shemilt, and Amanda Sowden. Synthesizing evidence on complex interventions: how meta-analytical, qualitative, and mixed-method approaches can contribute. *Journal of clinical epidemiology*, 66(11):1230–1243, 2013.

[179] Ba' Pham, Jelena Jovanovic, Ebrahim Bagheri, Jesmin Antony, Huda Ashoor, Tam T Nguyen, Patricia Rios, Reid Robson, Sonia M Thomas, Jennifer Watt, et al. Text mining to support abstract screening for knowledge syntheses: a semi-automated workflow. *Systematic reviews*, 10(1):156, 2021.

[180] Jay M Ponte and W Bruce Croft. A language modeling approach to information retrieval. In *ACM SIGIR Forum*, volume 51, pages 202–208. ACM New York, NY, USA, 2017.

[181] Mohammadreza Pourreza and Faezeh Ensan. Towards semantic-driven boolean query formalization for biomedical systematic literature reviews. *International Journal of Medical Informatics*, page 104928, 2022.

[182] Animesh Prasad, Manpreet Kaur, and Min-Yen Kan. Neural ParsCit: A deep learning based reference string parser. *Journal on Digital Libraries*, 19:323–337, 2018.

[183] Piotr Przybyła, Austin J Brockmeier, Georgios Kontonatsios, Marie-Annick Le Pogam, John McNaught, Erik von Elm, Kay Nolan, and Sophia Ananiadou. Prioritising references for

systematic reviews with robotanalyst: a user study. *Research synthesis methods*, 9(3):470–488, 2018.

[184] Chen Qu, Liu Yang, Minghui Qiu, W Bruce Croft, Yongfeng Zhang, and Mohit Iyyer. Bert with history answer embedding for conversational question answering. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pages 1133–1136, 2019.

[185] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[186] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020.

[187] Harleen Kaur Rai, Aline Cavalcanti Barroso, Lauren Yates, Justine Schneider, and Martin Orrell. Involvement of people with dementia in the development of technology-based interventions: narrative synthesis review and best practice guidelines. *Journal of medical Internet research*, 22 (12):e17531, 2020.

[188] John Rathbone. *Automating systematic reviews.* PhD thesis, Bond University, 2017.

[189] John Rathbone, Matt Carter, Tammy Hoffmann, and Paul Glasziou. Better duplicate detection for systematic reviewers: Evaluation of Systematic Review Assistant-Deduplication Module. *Systematic reviews*, 4(1):6, 2015.

[190] David Rau, Hervé Déjean, Nadezhda Chirkova, Thibault Formal, Shuai Wang, Stéphane Clinchant, and Vassilina Nikoulina. Bergen: A benchmarking library for retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7640–7663, 2024.

[191] David Rau, Shuai Wang, Hervé Déjean, Stéphane Clinchant, and Jaap Kamps. Context embeddings for efficient answer generation in retrieval-augmented generation. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, pages 493–502, 2025.

[192] Scott Reeves, Ivan Koppel, Hugh Barr, Della Freeth, and Marilyn Hammick. Twelve tips for undertaking a systematic review. *Medical teacher*, 24(4), 2002.

[193] Radim Rehurek, Petr Sojka, et al. Gensim—statistical semantics in python. *Retrieved from genism. org*, 2011.

[194] Laria Reynolds and Kyle McDonell. Prompt Programming for Large Language Models: Beyond the Few-Shot Paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in*

Computing Systems, pages 1–7, Yokohama Japan, May 2021. ACM. ISBN 978-1-4503-8095-9. doi: 10.1145/3411763.3451760.

[195] Ambrose Robinson, William Thorne, Ben P Wu, Abdullah Pandor, Munira Essat, Mark Stevenson, and Xingyi Song. Bio-sieve: Exploring instruction tuning large language models for systematic review automation. *arXiv preprint arXiv:2308.06610*, 2023.

[196] JJ Rocchio and Gerard Salton. Information search optimization and interactive retrieval techniques. In *Proceedings of the November 30–December 1, 1965, fall joint computer conference, part I*, pages 293–305, 1965.

[197] Raul Rodriguez-Esteban and Ivan Iossifov. Figure mining for biomedical research. *Bioinformatics*, 25(16):2082–2084, 2009.

[198] Tony Russell-Rose and Philip Gooch. 2dSearch: A visual approach to search strategy formulation. In *Proceedings of the 1st Biennial Conference on Design of Experimental Search and Information Retrieval Systems*, 2018.

[199] Malik Sallam. Chatgpt utility in healthcare education, research, and practice: systematic review on the promising perspectives and valid concerns. In *Healthcare*, volume 11, page 887. MDPI, 2023.

[200] José Antonio Salvador-Oliván, Gonzalo Marco-Cuenca, and Rosario Arquero-Avilés. Errors in search strategies used in systematic reviews and their effects on information retrieval. *Journal of the Medical Library Association : JMLA*, 107(2):210–221, April 2019. ISSN 1536-5050. doi: 10.5195/jmla.2019.567.

[201] Claude Sammut and Geoffrey I. Webb, editors. *Leave-One-Out Cross-Validation*, pages 600–601. Springer US, Boston, MA, 2010. ISBN 978-0-387-30164-8. doi: 10.1007/978-0-387-30164-8_469. URL `https://doi.org/10.1007/978-0-387-30164-8_469`.

[202] M. Sampson and J. McGowan. Errors in search strategies were identified by type and frequency. *JCE*, 59(10):1057–e1, 2006.

[203] Eric Sayers. A general introduction to the e-utilities. *Entrez Programming Utilities Help [Internet]. Bethesda: National Center for Biotechnology Information*, 2010.

[204] H. Scells, G. Zuccon, B. Koopman, A. Deacon, L. Azzopardi, and S. Geva. Integrating the framing of clinical questions via PICO into the retrieval of medical literature for systematic reviews. In *Proceedings of the 26th International Conference on Information and Knowledge Management*, pages 2291–2294, 2017.

[205] H. Scells, G. Zuccon, B. Koopman, A. Deacon, S. Geva, and L. Azzopardi. A test collection for evaluating retrieval of studies for inclusion in systematic reviews. In *Proceedings of the 40th*

International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 1237–1240, 2017.

[206] H. Scells, D. Locke, and G. Zuccon. An information retrieval experiment framework for domain specific applications. In *Proceedings of the 41st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1281–1284, 2018.

[207] Harrisen Scells and Guido Zuccon. Searchrefiner: A query visualisation and understanding tool for systematic reviews. In *Proceedings of the 27th International Conference on Information and Knowledge Management*, pages 1939–1942, 2018.

[208] Harrisen Scells and Guido Zuccon. You can teach an old dog new tricks: Rank fusion applied to coordination level matching for ranking in systematic reviews. In *Proceedings of the 42nd European Conference on Information Retrieval*, pages 399–414, 2020.

[209] Harrisen Scells, Guido Zuccon, Anthony Deacon, and Bevan Koopman. Qut ielab at clef ehealth 2017 technology assisted reviews track: Initial experiments with learning to rank. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, volume 1866, pages Paper–98. CEUR Workshop Proceedings, 2017.

[210] Harrisen Scells, Guido Zuccon, Anthony Deacon, and Bevan Koopman. Qut ielab at clef ehealth 2017 technology assisted reviews track: initial experiments with learning to rank. In *Working Notes of CLEF 2017-Conference and Labs of the Evaluation Forum [CEUR Workshop Proceedings, Volume 1866]*, pages 1–6. Sun SITE Central Europe, 2017.

[211] Harrisen Scells, Guido Zuccon, and Bevan Koopman. Automatic boolean query refinement for systematic review literature search. In *The world wide web conference*, pages 1646–1656, 2019.

[212] Harrisen Scells, Guido Zuccon, and Bevan Koopman. Automatic boolean query refinement for systematic review literature search. In *The World Wide Web Conference*, WWW '19, pages 1646–1656, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450366748. doi: 10.1145/3308558.3313544. URL `https://doi.org/10.1145/3308558.3313544`.

[213] Harrisen Scells, Guido Zuccon, and Bevan Koopman. A computational approach for objectively derived systematic review search strategies. In *Proceedings of the 42nd European Conference on Information Retrieval*, pages 385–398, 2020.

[214] Harrisen Scells, Guido Zuccon, Bevan Koopman, and Justin Clark. Automatic boolean query formulation for systematic review literature search. In *Proceedings of the 29th World Wide Web Conference*, pages 1071–1081, 2020.

[215] Harrisen Scells, Guido Zuccon, Mohamed A. Sharaf, and Bevan Koopman. *Sampling Query Variations for Learning to Rank to Improve Automatic Boolean Query Generation in Systematic Reviews*, pages 3041–3048. Association for Computing Machinery, New York, NY, USA, 2020. ISBN 9781450370233. URL `https://doi.org/10.1145/3366423.3380075`.

[216] Harrisen Scells, Guido Zuccon, and Bevan Koopman. A comparison of automatic boolean query formulation for systematic reviews. *Information Retrieval Journal*, 24(1):3–28, 2021.

[217] Harrisen Scells, Connor Forbes, Justin Clark, Bevan Koopman, and Guido Zuccon. The impact of query refinement on systematic review literature search: A query log analysis. In *Proceedings of the 2022 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 34–42, 2022.

[218] Connie Schardt, Martha B Adams, Thomas Owens, Sheri Keitz, and Paul Fontelo. Utilization of the PICO framework to improve searching PubMed for clinical questions. *BMC medical informatics and decision making*, 7(1):16, 2007.

[219] Martin Schiavenato and Frances Chu. Pico: What it is and what it is not. *Nurse education in practice*, 56:103194, 2021.

[220] Philip Sedgwick. Meta-analyses: how to read a funnel plot. *Bmj*, 346, 2013.

[221] Philip Sedgwick. Meta-analyses: what is heterogeneity? *Bmj*, 350, 2015.

[222] Nour Shaheen, Ahmed Shaheen, Alaa Ramadan, Mahmoud Tarek Hefnawy, Abdelraouf Ramadan, Ismail A Ibrahim, Maged Elsayed Hassanein, Mohamed E Ashour, and Oliver Flouty. Appraising systematic reviews: a comprehensive guide to ensuring validity and reliability. *Frontiers in research metrics and analytics*, 8:1268045, 2023.

[223] Ian Shemilt, Nada Khan, Sophie Park, and James Thomas. Use of cost-effectiveness analysis to compare the efficiency of study identification methods in systematic reviews. *Systematic reviews*, 5(1):140, 2016.

[224] Michael Simon, Elke Hausner, Susan F Klaus, and Nancy E Dunton. Identifying nurse staffing research in Medline: Development and testing of empirically derived search strategies with the PubMed interface. *BMC medical research methodology*, 10(1):76, 2010.

[225] Gaurav Singh, Iain Marshall, James Thomas, and Byron Wallace. Identifying diagnostic test accuracy publications using a deep model. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, volume 1866, 2017.

[226] Jaspreet Singh and Lini Thomas. IIIT-H at CLEF eHealth 2017 task 2: Technologically assisted reviews in empirical medicine. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, 2017.

[227] Alison Sneyd and Mark Stevenson. Stopping criteria for technology assisted reviews based on counting processes. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2293–2297, 2021.

[228] Luca Soldaini and Nazli Goharian. Quickumls: a fast, unsupervised approach for medical concept extraction. In *MedIR workshop, sigir*, 2016.

[229] Moritz Staudinger, Wojciech Kusa, Florina Piroi, Aldo Lipani, and Allan Hanbury. A reproducibility and generalizability study of large language models for query generation. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, SIGIR-AP 2024, pages 186–196, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400707247. doi: 10.1145/3673791.3698432. URL https://doi.org/10.1145/3673791.3698432.

[230] Moritz Staudinger, Wojciech Kusa, Florina Piroi, Aldo Lipani, and Allan Hanbury. A reproducibility and generalizability study of large language models for query generation. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, pages 186–196, 2024.

[231] Carolyn RT Stoll, Sonya Izadi, Susan Fowler, Paige Green, Jerry Suls, and Graham A Colditz. The value of a second reviewer for study selection in systematic reviews. *Research synthesis methods*, 10(4):539–545, 2019.

[232] Rodney Summerscales, Shlomo Argamon, Jordan Hupert, and Alan Schwartz. Identifying treatments, groups, and outcomes in medical abstracts. In *Proceedings of the 6th Midwest Computational Linguistics Colloquium*, 2009.

[233] Rodney L Summerscales, Shlomo Argamon, Shangda Bai, Jordan Hupert, and Alan Schwartz. Automatic summarization of results from clinical trials. In *Proceedings of the 2011 IEEE International Conference on Bioinformatics and Biomedicine*, pages 372–377, 2011.

[234] Shuoqi Sun, Shengyao Zhuang, Shuai Wang, and Guido Zuccon. An investigation of prompt variations for zero-shot llm-based rankers. In *European Conference on Information Retrieval*, pages 185–201. Springer, 2025.

[235] Weiwei Sun, Lingyong Yan, Xinyu Ma, Pengjie Ren, Dawei Yin, and Zhaochun Ren. Is chatgpt good at search? investigating large language models as re-ranking agent. *arXiv preprint arXiv:2304.09542*, 2023.

[236] Eugene Syriani, Istvan David, and Gauransh Kumar. Assessing the ability of chatgpt to screen articles for systematic reviews. *arXiv preprint arXiv:2307.06464*, 07 2023.

[237] Yiming Tan, Dehai Min, Yu Li, Wenbo Li, Nan Hu, Yongrui Chen, and Guilin Qi. Evaluation of chatgpt as a question answering system for answering complex questions. *arXiv preprint arXiv:2303.07992*, 2023.

[238] Elizabeth Tanjong-Ghogomu, Peter Tugwell, and Vivian Welch. Evidence-based medicine and the cochrane collaboration. *Bulletin of the NYU hospital for joint diseases*, 67(2):198, 2009.

[239] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

[240] James Thomas and Angela Harden. Methods for the thematic synthesis of qualitative research in systematic reviews. *BMC medical research methodology*, 8(1):45, 2008.

[241] Kristian Thorlund, Stephen D Walter, Bradley C Johnston, Toshi A Furukawa, and Gordon H Guyatt. Pooling health-related quality of life outcomes in meta-analysis—a tutorial and review of methods for enhancing interpretability. *Research synthesis methods*, 2(3):188–203, 2011.

[242] Jessica To, Ernesto Panadero, and David Carless. A systematic review of the educational uses and effects of exemplars. *Assessment & Evaluation in Higher Education*, 47(8):1167–1182, 2022.

[243] Mercedes Torres Torres and Clive E Adams. Revmanhal: towards automatic text generation in systematic reviews. *Systematic Reviews*, 6(1), 2017.

[244] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[245] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[246] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. Zephyr: Direct distillation of lm alignment, 2023.

[247] Matt Vassar, Vadim Yerokhin, Philip Marcus Sinnett, Matthew Weiher, Halie Muckelrath, Branden Carr, Laura Varney, and Gregory Cook. Database selection in systematic reviews: an insight through clinical neurology. *Health Information & Libraries Journal*, 34(2):156–164, 2017.

[248] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[249] M Viswanathan, MT Ansari, ND Berkman, S Chang, L Hartling, M McPheeters, and JR Treadwell. Assessing the risk of bias of individual studies in systematic reviews of health care interventions. agency for healthcare research and quality methods guide for comparative effectiveness reviews. *AHRQ Methods for Effective Health Care*, 2012.

[250] Byron C Wallace, Thomas A Trikalinos, Joseph Lau, Carla Brodley, and Christopher H Schmid. Semi-automated screening of biomedical citations for systematic reviews. *BMC bioinformatics*, 11(1):55, 2010.

[251] Byron C Wallace, Kevin Small, Carla E Brodley, Joseph Lau, and Thomas A Trikalinos. Deploying an interactive machine learning system in an evidence-based practice center: Abstrackr. In *Proceedings of the 2nd ACM International Health Informatics Symposium*, pages 819–824, 2012.

[252] Junmei Wang, Min Pan, Tingting He, Xiang Huang, Xueyan Wang, and Xinhui Tu. A pseudo-relevance feedback framework combining relevance matching and semantic matching for information retrieval. *Information Processing & Management*, 57(6):102342, 2020.

[253] Shuai Wang and Guido Zuccon. Balanced topic aware sampling for effective dense retriever: A reproducibility study. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, pages 2542–2551, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450394086. doi: 10.1145/3539618.3591915. URL https://doi.org/10.1145/3539618.3591915.

[254] Shuai Wang, Hang Li, Harrisen Scells, Daniel Locke, and Guido Zuccon. Mesh term suggestion for systematic review literature search. In *Proceedings of the 25th Australasian Document Computing Symposium*, ADCS '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450395991. doi: 10.1145/3503516.3503530. URL https://doi.org/10.1145/3503516.3503530.

[255] Shuai Wang, Shengyao Zhuang, and Guido Zuccon. Bert-based dense retrievers require interpolation with bm25 for effective passage retrieval. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '21, pages 317–324, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450386111. doi: 10.1145/3471158.3472233. URL https://doi.org/10.1145/3471158.3472233.

[256] Shuai Wang, Hang Li, and Guido Zuccon. Mesh suggester: A library and system for mesh term suggestion for systematic review boolean query construction. *arXiv preprint arXiv:2212.09018*, 2022.

[257] Shuai Wang, Harrisen Scells, Justin Clark, Bevan Koopman, and Guido Zuccon. From little things big things grow: A collection with seed studies for medical systematic review literature search. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, pages 3176–3186, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450387323. doi: 10.1145/3477495.3531748. URL https://doi.org/10.1145/3477495.3531748.

[258] Shuai Wang, Harrisen Scells, Justin Clark, Bevan Koopman, and Guido Zuccon. From little things big things grow: A collection with seed studies for medical systematic review literature search. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3176–3186, 2022.

[259] Shuai Wang, Harrisen Scells, Bevan Koopman, and Guido Zuccon. Automated mesh term suggestion for effective query formulation in systematic reviews literature search. *Intelligent Systems with Applications*, page 200141, 2022.

[260] Shuai Wang, Harrisen Scells, Bevan Koopman, and Guido Zuccon. Neural rankers for effective screening prioritisation in medical systematic review literature search. In *Proceedings of the 26th Australasian Document Computing Symposium*, pages 1–10, 2022.

[261] Shuai Wang, Harrisen Scells, Ahmed Mourad, and Guido Zuccon. Seed-driven document ranking for systematic reviews: A reproducibility study. In *European Conference on Information Retrieval*, pages 686–700. Springer, 2022.

[262] Shuai Wang, Hang Li, and Guido Zuccon. Mesh suggester: A library and system for mesh term suggestion for systematic review boolean query construction. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, pages 1176–1179, 2023.

[263] Shuai Wang, Harrisen Scells, Bevan Koopman, Martin Potthast, and Guido Zuccon. Generating natural language queries for more effective systematic review screening prioritisation. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, SIGIR-AP '23, pages 73–83, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400704086. doi: 10.1145/3624918.3625322. URL https://doi.org/10.1145/3624918.3625322.

[264] Shuai Wang, Harrisen Scells, Bevan Koopman, and Guido Zuccon. Can chatgpt write a good boolean query for systematic review literature search? In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, pages 1426–1436, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450394086. doi: 10.1145/3539618.3591703. URL https://doi.org/10.1145/3539618.3591703.

[265] Shuai Wang, Ekaterina Khramtsova, Shengyao Zhuang, and Guido Zuccon. Feb4rag: Evaluating federated search in the context of retrieval augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, pages 763–773, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400704314. doi: 10.1145/3626772.3657853. URL https://doi.org/10.1145/3626772.3657853.

[266] Shuai Wang, Harrisen Scells, Shengyao Zhuang, Martin Potthast, Bevan Koopman, and Guido Zuccon. Zero-shot generative large language models for systematic review screening automation. In Nazli Goharian, Nicola Tonellotto, Yulan He, Aldo Lipani, Graham McDonald, Craig Macdonald, and Iadh Ounis, editors, *Advances in Information Retrieval*, pages 403–420, Cham, 2024. Springer Nature Switzerland. ISBN 978-3-031-56027-9.

[267] Shuai Wang, Shengyao Zhuang, and Guido Zuccon. Large language models based stemming for information retrieval: Promises, pitfalls and failures. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, pages 2492–2496, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400704314. doi: 10.1145/3626772.3657949. URL https://doi.org/10.1145/3626772.3657949.

[268] Shuai Wang, Harrisen Scells, Bevan Koopman, and Guido Zuccon. Autobool: Reinforcement-learned llm for effective automatic systematic reviews boolean query generation. In *openreview arxiv*, 2025. URL https://openreview.net/forum?id=D9NNghIgUR.

[269] Shuai Wang, Harrisen Scells, Bevan Koopman, and Guido Zuccon. Reassessing large language model boolean query generation for systematic reviews. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, pages 3296–3305, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400715921. doi: 10.1145/3726302.3730329. URL https://doi.org/10.1145/3726302.3730329.

[270] Shuai Wang, Shengyao Zhuang, Bevan Koopman, and Guido Zuccon. 2d matryoshka training for information retrieval. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, pages 3125–3134, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400715921. doi: 10.1145/3726302.3730330. URL https://doi.org/10.1145/3726302.3730330.

[271] Shuai Wang, Shengyao Zhuang, Bevan Koopman, and Guido Zuccon. Resllm: Large language models are strong resource selectors for federated search. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 1360–1364, 2025.

[272] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*, 2022.

[273] Jacob White. Pubmed 2.0. *Medical reference services quarterly*, 39(4):382–387, 2020.

[274] Huaying Wu, Tingting Wang, Jiayi Chen, Su Chen, Qinmin Hu, and Liang He. Ecnu at 2018 ehealth task 2: Technologically assisted reviews in empirical medicine. *Methods-a Companion to Methods in Enzymology*, 4(5):7, 2018.

[275] Eugene Yang, Sean MacAvaney, David D Lewis, and Ophir Frieder. Goldilocks: Just-right tuning of bert for technology-assisted review. In *European Conference on Information Retrieval*, pages 502–517. Springer, 2022.

[276] Zheng Yao, Shuai Wang, and Guido Zuccon. Pre-training vs. fine-tuning: A reproducibility study on dense retrieval knowledge acquisition. In *Proceedings of the 48th International ACM SIGIR*

Conference on Research and Development in Information Retrieval, SIGIR '25, pages 3276–3285, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400715921. doi: 10.1145/3726302.3730332. URL https://doi.org/10.1145/3726302.3730332.

[277] Zhe Yu and Tim Menzies. Data balancing for technologically assisted reviews: Undersampling or reweighting. In *CEUR Workshop Proceedings: Working Notes of CLEF 2017: Conference and Labs of the Evaluation Forum*, 2017.

[278] ChengXiang Zhai and Sean Massung. *Text data management and analysis: a practical introduction to information retrieval and text mining*. Morgan & Claypool, 2016.

[279] Haotian Zhang, Mustafa Abualsaud, Nimesh Ghelani, Mark D Smucker, Gordon V Cormack, and Maura R Grossman. Effective user interaction for high-recall retrieval: Less is more. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pages 187–196, 2018.

[280] Shengyao Zhuang, Hang Li, and Guido Zuccon. Deep query likelihood model for information retrieval. In *European Conference on Information Retrieval*, pages 463–470. Springer, 2021.

[281] Shengyao Zhuang, Shuai Wang, Bevan Koopman, and Guido Zuccon. Starbucks: Improved training for 2d matryoshka embeddings. *arXiv preprint arXiv:2410.13230*, 2024.

[282] Jie Zou, Dan Li, and Evangelos Kanoulas. Technology assisted reviews: Finding the last few relevant documents by asking Yes/No questions to reviewers. In *Proceedings of the 41st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 949–952, 2018.

[283] Guido Zuccon, Joao Palotti, and Allan Hanbury. Query variations and their effect on comparing information retrieval systems. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 691–700, 2016.