Vrije Universiteit Amsterdam
Universiteit van Amsterdam

Master Thesis

FastSLR: An LLM-Powered Tool for Systematic Literature Review Automation

Author: Qihua Han (UvA: 15321487, VU: 2796806)

1st supervisor: Adam Belloum
daily supervisor: Zhiheng Yang
2nd reader: Yixian Shen

A thesis submitted in fulfillment of the requirements for the joint UvA-VU Master of Science degree in Computer Science

April 17, 2025

"I like criticism. It makes you strong," by LeBron James.

Abstract

Systematic Literature Reviews (SLRs) are a cornerstone of evidence-based research, providing a comprehensive and structured approach to synthesizing existing knowledge. However, traditional SLRs are time-consuming, labor-intensive, and prone to human error, often taking months or even years to complete. To address these challenges, this thesis introduces FastSLR , an innovative system designed to automate and accelerate the SLR process using Large Language Models (LLMs). FastSLR automates all stages of SLRs, including literature search, screening, data extraction, and synthesis, while incorporating features such as multi-database support, configurable models, full-text analysis, and an interactive RAG-based chatbot. Experimental evaluation demonstrates that FastSLR can process 100 papers, screen them for relevance, synthesize answers to research questions, and generate results in just 41 minutes and 28 seconds, achieving an accuracy of 0.92, precision of 0.9268, and recall rate of 0.9744 in literature screening. These results highlight the system's potential to significantly reduce the time and effort required for SLRs while maintaining high accuracy and reliability. Despite its success, FastSLR faces limitations, such as the lack of multi-modal model integration and limited academic database compatibility, which present opportunities for future improvement. By leveraging cutting-edge AI technologies, FastSLR represents a significant step forward in transforming the way SLRs are conducted, empowering researchers to navigate the ever-growing body of academic literature efficiently and effectively.

Contents

List of Figures

LIST OF FIGURES

List of Tables

LIST OF TABLES

1

Introduction

A systematic literature review (SLR) is a comprehensive, structured, and reproducible process of identifying, evaluating, and synthesizing all available research relevant to a particular research question, topic area, or phenomenon of interest (6). It goes beyond a simple narrative review by employing a predefined protocol to ensure that the review is unbiased, transparent, and reliable, which is usually composed of research question formulation, literature searching, literature screening, and knowledge synthesizing (7).

SLRs are crucial for researchers. For one, it provides a comprehensive overview of the existing knowledge in a particular field. By synthesizing a large body of research, researchers can identify the state-of-the-art, current advancements, trends, gaps, and contradictions in the literature. This helps in avoiding duplication of research efforts and allows for the identification of new research questions that have not been adequately explored. Moreover, SLRs enhance the credibility of research. The use of a systematic and transparent approach ensures that the review is based on the best available evidence, which can be replicated by other researchers.

Despite the importance, conducting an SLR is time-consuming and labor-intensive. Researchers need to search multiple academic databases to obtain comprehensive literature. Screening literature requires careful reading and understanding of articles to determine whether it is relevant to the research question. Synthesizing data requires careful analysis of article content to extract and compare valuable information. Each step requires a significant amount of time and effort. For researchers who want to quickly grasp current research progress, limitations, and potential improvements, this is a great challenge.

With the rapid development of deep learning, especially the breakthrough of Transformer-based (2) Large Language Models (LLMs), the execution process of SLRs may undergo great changes. LLMs that have been pre-trained on a large corpus have strong long-context semantic understanding and text generation capabilities (8), which are exactly what effective and efficient SLRs need. For example, LLMs can quickly read and understand articles to determine relevance. They can also accurately extract valuable information related to research questions from relevant articles. Additionally, they can comprehensively generate texts or answer questions raised by users by integrating the obtained information and knowledge.

Currently, there is already some research that tries to automate part or even the entire SLR process. For instance, LitLLM (3) and a Retrieval-Augmented Generation (RAG)-based framework (9) use LLMs to automate all SLR stages. LLAssist (10) is a tool used to determine whether an article is relevant to a user's research question. Susnjak et al. introduce LLM fine-tuning in the synthesis stage of SLRs to eliminate hallucinations (11). However, these studies or tools still have problems such as weak model performance, lack of domain-specific knowledge, poor compatibility with different academic databases, lack of full-text analysis, poor interpretability, and high cost of calling model APIs.

In order to fully utilize LLMs to accelerate SLRs and address the shortcomings of existing research, I designed and implemented FastSLR. It dcovers all stages of SLR, helping researchers search and filter literature from multiple databases, extract data from literature according to questions and conduct knowledge synthesis to generate accurate answers. It can also answer researchers' questions in an interactive chatbot. To address the problems of existing tools mentioned above, I integrated many features into FastSLR: multi-database support, configurable models, recording of filtering reasons, full-text analysis, RAG, user-friendly interface, containerized deployment, etc.

The structure of the thesis is as follows: Chapter 2 introduces some relevant concepts and background information. A comprehensive review of existing research is presented in Chapter 3. Chapter 4 introduces the design and implementation of FastSLR and explains every detail from top to bottom. In Chapter 5, the experimental results of evaluating FastSLR are presented. Chapter 6 analyzes and discusses the results and proposes some limitations and future improvement directions. Finally, Chapter 7 concludes the thesis.

2

Background

2.1 Systematic Literature Review

2.1.1 The Concept of SLR and its Importance

A Systematic Literature Review (SLR) is a highly organized, comprehensive, and methodologically rigorous approach in academic research. It is a systematic process designed to identify, evaluate, and synthesize all the research materials that are relevant to a specific research topic. In contrast to traditional literature reviews, which often lack a standardized and explicit methodology, an SLR adheres to a carefully predefined protocol. Strictly following the protocol can make the conducting of SLR more standardized, rigorous, and reproducible.

The importance of SLR is multi-faceted. In an era during which the volume of academic research is expanding exponentially across all disciplines, the SLR serves as a valuable compass for researchers. It provides them with a clear, unbiased, and comprehensive overview of a particular field. By synthesizing a large amount of literature, it enables researchers to identify the current state of knowledge. For example, when researchers explored cutting-edge fields like using LLMs for program repair, they first conducted an SLR (12) on existing research, so as to have a clear understanding of the research progress of using LLMs for program repair at present, obtain new research ideas and avoid conducting repetitive research.

SLR also plays a key role in highlighting the gaps in existing research, which is especially important in cutting-edge research areas like multi-modal models. By critically evaluating the available literature, researchers can identify areas that have been under-explored or areas where there is a lack of consensus. These gaps can possibly spark new research ideas. For instance, researchers conducted an SLR on the multi-modal models in the field of machine learning (13). Thus, they understood the current applications and challenges of multi-modal models and pointed out future research directions. This is very beneficial for every researcher in the same field.

Furthermore, SLR has a significant impact on policy-making processes. Policymakers in various sectors, such as healthcare, education, and environmental protection, rely on evidence-based research to make informed decisions. A well-conducted SLR can provide them with a comprehensive understanding of the relevant research landscape. For instance, when the COVID-19 virus was prevalent, it posed a great threat to the life and health of people all over the world. Some researchers studied the risk factors of COVID-19 (14), thus enabling policy-makers to have a more accurate understanding of the current situation and risk of the virus, and be more cautious and more evidence-based when making decisions.

2.1.2 The Stages of SLR

How to efficiently conduct SLR is very important for researchers. Therefore, some researchers have summarized a set of SLR workflows to help other researchers better carry out SLR (6). In general, an SLR can be divided into two phases: planning and conducting. But sometimes there is a third phase: reporting. Figure 2.1 describes the three phases of an SLR, and each phase is composed of several stages.

1. Planning: In the planning phase, researchers first define keywords and their synonyms according to the research topics. Then, define meaningful research questions that are in line with the topic. Next, researchers should choose which academic databases to use according to their research fields. Since the number of retrieved literature may be relatively large, they need to formulate appropriate inclusion and exclusion criteria to filter retrieved papers. The quality of academic literature is uneven, so a quality assessment checklist needs to be formulated to improve research efficiency and quality. Finally, researchers need to create a data extraction form to prepare for extracting useful information from articles.

2. Conducting: When it comes to conducting SLRs, researchers need to search literature in various academic databases to obtain comprehensive studies. Then, they will conduct literature screening to filter the retrieved literature according to the predefined inclusion and exclusion criteria. Next, researchers should read and analyze each selected paper and record retrieved information and knowledge in the data extract form. Finally, they can synthesize the extracted information to formulate their own thoughts and research ideas.

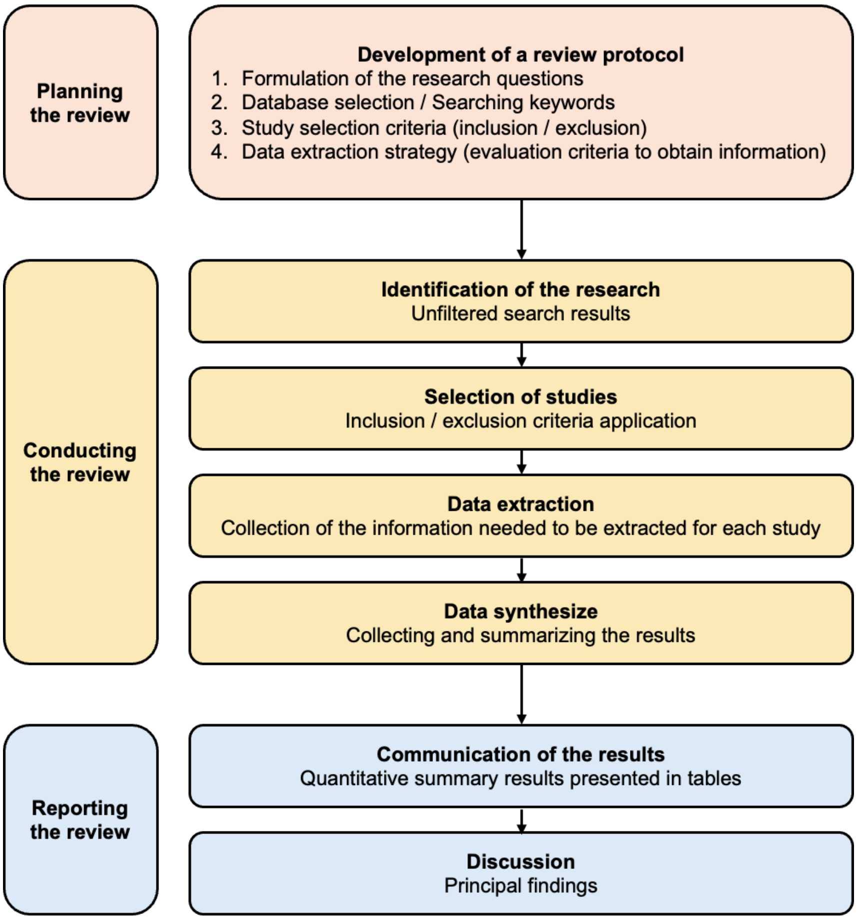

Planning the review
Development of a review protocol
1. Formulation of the research questions
2. Database selection / Searching keywords
3. Study selection criteria (inclusion / exclusion)
4. Data extraction strategy (evaluation criteria to obtain information)
Conducting the review
Identification of the research
Unfiltered search results
Selection of studies
Inclusion / exclusion criteria application
Data extraction
Collection of the information needed to be extracted for each study
Data synthesize
Collecting and summarizing the results
Reporting the review
Communication of the results
Quantitative summary results presented in tables
Discussion
Principal findings

Figure 2.1: Stages of an SLR, adapted and redrawn from (1).

3. Reporting: Sometimes researchers need to turn their thoughts and obtained knowledge into words. Especially when they are writing a literature review, a reporting phase is added to the SLR workflow. In the reporting phase, researchers use concise and professional language to record, synthesize, and summarize the information obtained in the previous stages so as to help themselves and other researchers master the current research status more efficiently.

2.1.3 Common Academic Databases

An academic database is a comprehensive collection of scholarly literature, research papers, journal articles, conference proceedings, and other academic resources. It serves as a centralized repository, allowing researchers, scholars, and students to access a vast amount of high-quality information relevant to their fields of study. These databases are curated and maintained by professional organizations, publishers, or academic institutions, ensuring the reliability and credibility of the content. There are four common digital academic libraries that are used by millions of researchers daily: IEEE Xplore, ACM Digital Library, ScienceDirect, and Arxiv. Table 2.1 describes the different between these four databases.

Criteria	**IEEE Xplore**	**ACM Digital Library**	**ScienceDirect**	**arXiv**
Focus	Engineering, Technology	Computer Science	Broad Sciences	Preprints
Access	Subscription/Pay	Subscription/Pay	Subscription	Free
Search	Advanced, Full-text	Advanced, Full-text	Advanced, Full-text	Basic/Advanced
Cite Tools	BibTeX, RIS, EndNote	BibTeX, RIS, EndNote	BibTeX, RIS, EndNote	Manual
Peer Review	Yes	Yes	Yes	No
Pub Types	Journals, Conf	Journals, Conf	Journals, Books	Preprints, Postprints

Table 2.1: Comparison of academic databases: IEEE Xplore, ACM Digital Library, ScienceDirect, and arXiv.

1. IEEE Xplore(15): IEEE Xplore is a leading database in the fields of electrical engineering, computer science, and related technology areas. It contains a vast collection of peer-reviewed papers from IEEE (Institute of Electrical and Electronics Engineers) and its partner societies. The database includes journals, conference proceedings, standards, and educational courses. With its advanced search features, researchers can easily find the latest research on topics such as artificial intelligence, power systems, and telecommunications.

2. ACM Digital Library(16): ACM Digital Library is the go-to resource for computer science and information technology research. It offers access to the full-text of ACM-published journals, magazines, conference proceedings, and books. The ACM Digital Library is known for its comprehensive coverage of computer-related topics, from algorithms and programming languages to human-computer interaction and cybersecurity. It also provides features like citation tracking, which helps researchers follow the impact of their work and explore related studies.

3. ScienceDirect(17): ScienceDirect is one of the largest multidisciplinary academic databases. It offers access to more than 25 million research articles from over 2,500 peer-reviewed journals and 34,000 e-books across various scientific, technical, and medical disciplines. ScienceDirect's intuitive interface and powerful search tools enable users to quickly locate relevant information. It also offers features such as article recommendations based on user search history and saved content.

4. Arxiv(18): Arxiv is an open-access repository that mainly focuses on physics, mathematics, computer science, quantitative biology, quantitative finance, and statistics. It allows researchers to pre-print their papers, making them available to the scientific community before formal peer-review. This enables faster dissemination of research findings and promotes collaboration. Arxiv has been a crucial resource for researchers who want to stay updated on the latest developments in these fields, as well as for those seeking to share their work at an early stage.

2.1.4 Challenges in Traditional SLR

Although SLRs are important in many aspects, the traditional SLR process is very complex, time-consuming, and labor-intensive. Researchers need to formulate research questions according to research goals, select databases suitable for the research field, and specify inclusion/exclusion criteria. They need to perform searches in multiple databases and read the abstracts or even the full texts of each paper to determine whether the literature is related to the research topic. The selected papers also need to be carefully read to extract key information and synthesize it into researchers' own ideas. Therefore, according to the study (19), conducting a traditional SLR takes more than 15 months and there is a great possibility of facing the risk that the retrieved literature becomes outdated.

In addition, due to limited human energy and inevitable errors, it is difficult to guarantee the comprehensiveness of the selected literature, and the quality may not be the best. This will lead to a decline in the quality of SLR, and the grasp of the current research status and possible limitations in the research field may not be accurate.

In general, traditional human-dependent SLRs have the disadvantages of being too time-consuming, consuming a lot of energy, and not necessarily being accurate and comprehensive in results.

2.2 Large Language Models

2.2.1 Transformer Architecture

Since 2022, large language models (LLMs) have been a hot research topic in the field of artificial intelligence. Various large language models and multi-modal models have emerged in an endless stream. Every day, a large number of researchers are engaged in the research of LLMs. New research results are continuously published on Arxiv. Great progress has been made in the architecture, application, training, inference, and deployment of models.

Now, almost all LLMs are based on the Transformer architecture (2), which is proposed by a Google research team in 2017. Figure 2.2 illustrates the architecture of Transformer. Unlike traditional NLP models such as RNN and LSTM, Transformer adopts an encoder-decoder structure and uses a multi-head self-attention mechanism to capture text semantic information, uses position encoding to endow tokens with position information, and uses a masking mechanism to train the decoder in a auto-regression manner. Multi-head self-attention mechanism is the core of the Transformer architecture. Although some variants of Transformer models such as the GPT series use a decoder-only architecture, the multi-head self-attention mechanism always exists.

The multi-head self-attention mechanism in the Transformer architecture is used to capture complex semantic relationships within text. At its core, it uses a Query-Key-Value (QKV) mechanism.

Mathematically, given an input sequence of vectors $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]$, where $\mathbf{x}_i \in \mathbb{R}^d$, the Q, K, and V matrices are linearly projected from $\mathbf{X}$. We have $\mathbf{Q} = \mathbf{X}\mathbf{W}^Q$, $\mathbf{K} = \mathbf{X}\mathbf{W}^K$, and $\mathbf{V} = \mathbf{X}\mathbf{W}^V$, where $\mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^V \in \mathbb{R}^{d \times d_k}$ are learnable weight matrices.

The attention score between a query vector $\mathbf{q}_i$ and key vector $\mathbf{k}_j$ is calculated as:

$$\text{Attention}(\mathbf{q}_i, \mathbf{k}_j) = \frac{\exp(\frac{\mathbf{q}_i^T \mathbf{k}_j}{\sqrt{d_k}})}{\sum_{j=1}^{n} \exp(\frac{\mathbf{q}_i^T \mathbf{k}_j}{\sqrt{d_k}})}$$

The output of the attention mechanism for a query $\mathbf{q}_i$ is then a weighted sum of value vectors:

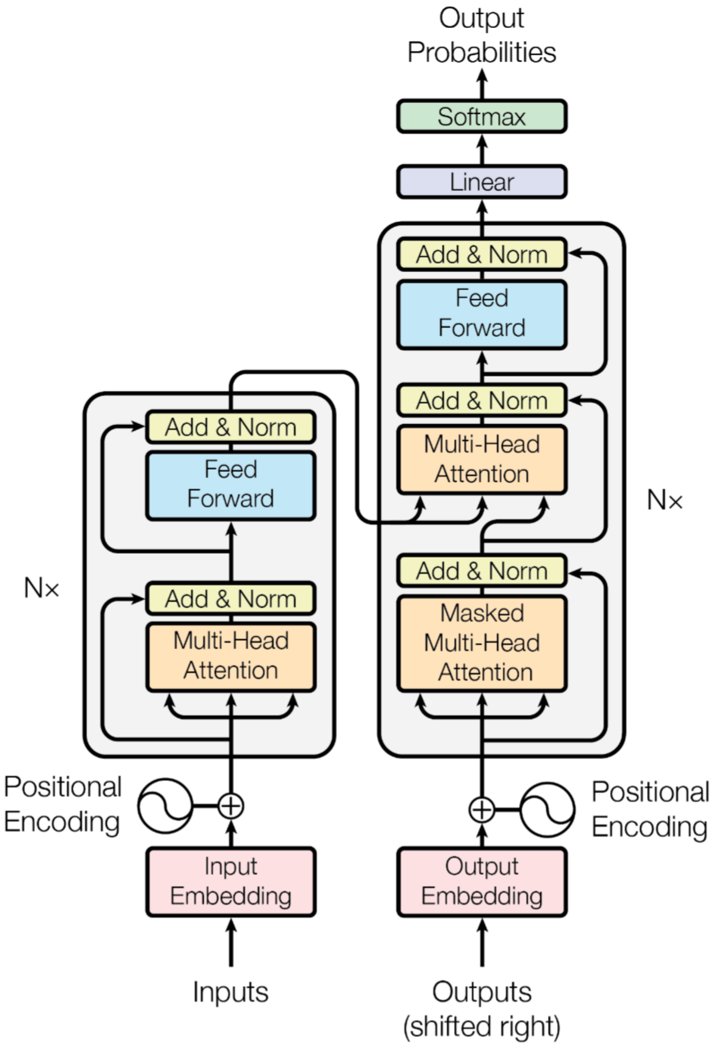


Figure 2.2: Transformer architecture, adapted from (2).

$$\text{Attention}(\mathbf{q}_i, \mathbf{K}, \mathbf{V}) = \sum_{j=1}^{n} \text{Attention}(\mathbf{q}_i, \mathbf{k}_j)\mathbf{v}_j$$

In the multi-head attention, this attention calculation is repeated h times (where h is the number of heads) with different sets of weight matrices ($\mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V$ for $i = 1, \cdots, h$). The results from all heads are then concatenated and linearly projected again to get the final output of the multi-head attention mechanism. This allows the model to attend to different aspects of the input simultaneously, enhancing its ability to capture diverse semantic information.

2.2.2 LLM Applications in NLP

LLMs have emerged as a revolutionary force in the field of Natural Language Processing (NLP), bringing about significant advancements and transformations. Their capabilities, stemming from extensive training on vast amounts of text data, have enabled a wide array of applications that were previously challenging or even impossible to achieve with traditional NLP techniques.

One of the primary applications of LLMs in NLP is in chatbots and virtual assistants. These models can understand the nuances of human language queries and generate responses that closely mimic human language patterns. For instance, companies like OpenAI and Anthropic have released some popular chatbots application: ChatGPT (20) and Claude (21). These assistants can handle a wide range of user requests, from answering simple factual questions to engaging in more complex conversations, providing 24/7 support to users without the need for continuous human intervention.

LLMs also play a crucial role in content generation. They can be used to generate various types of text, including articles, blog posts, and even creative writing pieces like stories and poetry. For example, Semantic Scholar (22) is using LLMs to generate a brief summary for each paper's abstract, making users grasp the main idea and contribution of each paper without reading through the possible long abstract.

In addition, sentiment analysis, an important NLP task for understanding the sentiment or opinion expressed in a piece of text, has also been enhanced by LLMs. They can analyze context and linguistic subtleties better than many previous models. For example, in social media monitoring, companies can use LLMs to analyze customer feedback on their products or services. By determining whether the sentiment is positive, negative, or neutral, businesses can make more informed decisions about product improvements or customer service enhancements.

2.2.3 Common Large Language Models

In the burgeoning realm of Large Language Models (LLMs), several models have emerged as top players, each bringing unique capabilities and advancements. This section provides an overview of some of the most influential LLMs, namely GPT (20), Claude (21), Qwen (23), and DeepSeek (24).

1. GPT: OpenAI's GPT series models have always been leading the LLM's market. In 2022, the powerful capabilities of ChatGPT 3.5 attracted people's attention to LLMs and also marked the starting point for the rapid progress of LLMs. Now, the version of GPT has reached 4.5, and its semantic analysis and text generation capabilities have made another breakthrough.

2. Claude: Anthropic's Claude series models have powerful capabilities in coding and programming. It is good at analyzing the complex logic of programs, writing code, and debugging. It helps developers greatly reduce development time and changes the workflow in developing projects.

3. Qwen: Developed by Alibaba, Qwen has always been a leader in global open source large language models (LLMs) and has made an indispensable contribution to the rapid development of LLMs. Many excellent models choose Qwen as the base or perform model distillation on Qwen to achieve better performance on models with fewer parameters. Now, the Qwen series has evolved to the powerful language model Qwen 2.5-Max and multimodal model Qwen2.5-VL. These help many scientific researchers reduce model training costs and quickly verify their ideas.

4. DeepSeek: The DeepSeek model brought a great shock to the global LLM market in early 2025. By introducing reinforcement learning, DeepSeek R1 achieved performance comparable to top-level models at an extremely low cost and changed the paradigm of model training.

2.2.4 Limitations of LLM in Academic Contexts

LLMs have shown great potential in various natural language processing tasks, but when it comes to automating SLRs, they have several limitations:

1. Lack of domain-specific understanding: SLR often involves specific academic fields and requires a deep understanding of the relevant domain knowledge. LLMs may not have an in-depth and accurate understanding of specialized concepts, theories, and

research methods in specific fields. They might misinterpret or misunderstand the meaning of certain terms, leading to incorrect categorization or analysis of literature.

2. Need to handle long context: Academic papers are often composed of a large amount of text. When faced with long contexts, LLMs may have difficulty understanding or may experience forgetting. Therefore, long texts in academic scenarios are a potential challenge for LLMs.

3. Lack of transparency and interpretability: The decision-making process of LLMs are often complex and difficult to understand. It is hard to trace how they arrive at a particular conclusion or recommendation in the context of SLR. This lack of transparency makes it challenging for researchers to trust and validate the results, especially in academic research where transparency and reproducibility are highly valued.

2.3 Retrieval Augmented Generation

2.3.1 The Concept of RAG and its Importance

LLMs are remarkable in their ability to generate text based on the patterns they've learned from vast amounts of data during training. However, they have limitations, such as not always having the most up-to-date or specific knowledge. RAG (25) addresses these shortcomings by retrieving relevant external information from a knowledge base or a corpus of documents at the time of text generation.

Instead of relying solely on the pre-trained weights of the language model, RAG uses an information retrieval system to search for relevant passages. These retrieved passages are then fed into the language model as additional context. This enables the LLM to generate more accurate, fact-based, and context-rich responses. For example, when answering a question about a recent scientific discovery, RAG can retrieve the latest research papers and use the information within them to craft a response, while a traditional LLM might rely only on the knowledge it had during training, which could be outdated.

In the context of real-world applications, RAG is of great significance. First, it enhances the reliability of LLM-generated content. By incorporating up-to-date and relevant information from external sources, the generated text is more likely to be factually correct. This is crucial in fields such as healthcare, law, and finance, where accuracy is non-negotiable.

Second, RAG improves the interpretability of the generated output. Since the information used for generation comes from identifiable external sources, it becomes easier to trace back and verify the origin of the facts presented in the generated text. This transparency is essential for building trust in AI-generated content, especially in professional and high-stake scenarios.

Finally, RAG has the potential to reduce the amount of data required for training LLMs. Instead of trying to cram all possible knowledge into the model during training, the model can rely on retrieving information as needed. This not only makes the training process more efficient but also allows for more targeted and cost-effective development of language models.

2.3.2 The Stages of RAG

RAG's operation can be broken down into several key stages, each playing a crucial role in enabling more informed and accurate text generation. The process of RAG is shown in figure 2.3.

1. Query formulation: The process commences with the formation of a query. This query can be a user-inputted question, a statement for which additional information is sought, or a task-related instruction. For example, in a customer service chatbot scenario, a user might ask, "What are the new features of your latest product?" The system then formulates this natural-language input into a query that can be processed by the subsequent components of the RAG system. The query needs to be refined to ensure it captures the essence of the information required, often involving techniques like tokenization and part-of-speech tagging to better represent the semantics of the input.

2. Information retrieval: Once the query is formed, it is passed to the information retrieval module. This module is responsible for sifting through a large knowledge base, which could be a collection of documents, a database, or even an index of web pages. The retrieval system uses various algorithms, such as TF-IDF (Term Frequency-Inverse Document Frequency) (26) or more advanced neural-based retrieval models like Dense Passage Retrieval (DPR) (27). These algorithms calculate the relevance of each document or passage in the knowledge base to the query.

3. Generation with augmented context: Finally, the retrieved context are combined with the original query and fed into the LLMs. The language model then uses this augmented context to generate the final output. It leverages its pre-trained

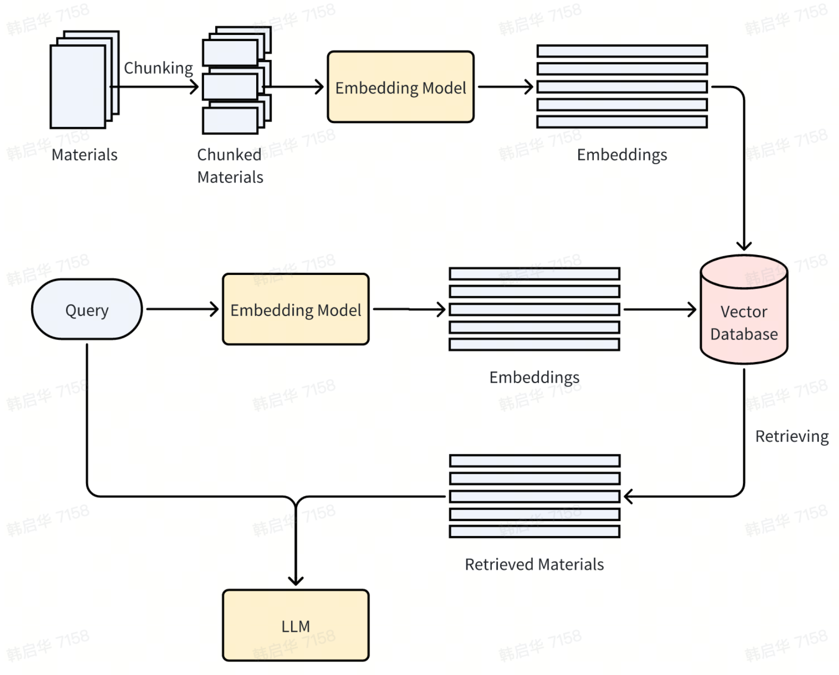


Figure 2.3: The process of RAG.

knowledge along with the newly provided information to produce a response that is more accurate, detailed, and context-aware.

3

Related Work

Due to the fact that LLMs are naturally good at text understanding tasks, some researchers have already conducted preliminary explorations on automating SLRs using large models. Some studies aim to automate the entire SLR process end-to-end, while others focus on one or several stages in the SLR process, such as literature screening or synthesizing.

3.1 Comprehensive Tools for end-to-end SLR Automation

As mentioned above, the process of SLR is long and complex. It is both time-consuming and labor-intensive. In order to simplify the process and accelerate the execution of SLRs, some end-to-end SLR automation tools have been developed. Table 3.1 compares the following three tools.

Tool	Features	Model Used	Evaluated	Open Sourced	Commercialized
LitLLM (3)	Uses RAG to mitigate hallucination	GPT-3.5 & GPT-4	Yes	Yes	No
Study (28)	Uses full-text synthesis	Not specified	Yes	Yes	No
Study (9)	Uses multimodal models and RAG	Not specified	No	No	No

Table 3.1: Comparison of comprehensive tools.

LitLLM (3) is an open-sourced LLM-based end-to-end SLR automation tool with a user-friendly interface. By leveraging retrieval-augmented generation (RAG), the authors aimed to eliminate the model hallucination and enhance the credibility of model output. It adopts a modular design, uses LLMs to generate search query based on user's research idea, searches relevant literature using Semantic Scholar's APIs, reranks the retrieved papers

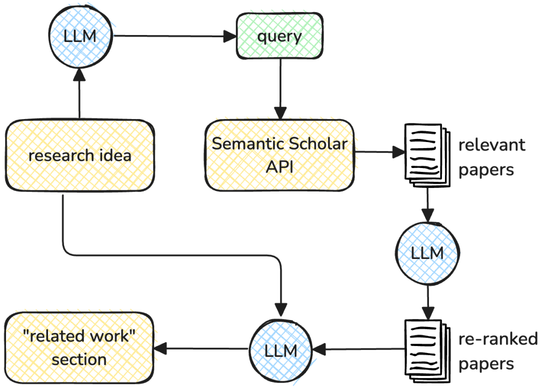


Figure 3.1: The modular pipeline in LitLLM, adapted and redrawn from (3).

based on their abstracts, and finally generates a "related work" section from the reranked papers. Figure 3.1 shows the pipeline in LitLLM.

Another study (28) proposed a multi-AI agent system for SLR automation. It uses four AI agents: planner, literature identification, data extraction, and data compilation to streamline and automate the entire SLR process. Compared to LitLLM, it uses full texts instead of solely abstracts when synthesizing, which makes it be able to perceive more semantic details.

Similar to LitLLM, Han et al. also applied RAG to the LLM-based SLR automation process (9). The author emphasized that RAG can help LLMs acquire domain knowledge, so as to better conduct literature synthesis and text generation.

3.2 Methods and Tools for Literature Screening

Literature screening is a key stage in the process of an SLR, because it determines the dataset on which the subsequent stages are carried out (29). At this stage, researchers need to read, understand, and analyze all articles retrieved from the databases to determine whether they are related to their own research questions based on some predefined inclusion and exclusion criteria (7). Obviously, this is a stage with a huge amount of work, and its automation is necessary. Next, five tools related to literature screening are introduced. Their comparison is shown in Table 3.2.

Alice [Roth2024ALISEAA] is a literature screening engine which solves the low screening accuracy problem by introducing full-text analysis. Figure 3.2 depicts the workflow in ALISE: The retrieved papers are converted into texts and split into chunks before being fed into the LLM, and then the LLM determines whether each paper is relevant

Tool	Features	Full-Text	Evaluated	Open Sourced
ALISE (4)	Records explanations for inclusion	Yes	Yes	No
Study (30)	Uses multiple AI agents and consensus scheme	No	Yes	No
LLAssist (10)	Supports JSON and CSV outputs	No	Yes	Yes
Study (31)	Combines BRF and LLMs	No	Yes	Yes
Set-Wise (5)	Proposes a set-wise ranking approach	Yes	Yes	Yes

Table 3.2: Comparison of literature screening tools.

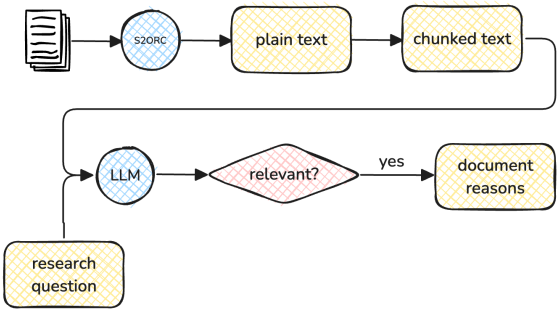


Figure 3.2: The workflow in ALISE, adapted and redrawn from (4).

to research questions and records the explanation to improve the transparency. The researchers claimed that ALISE achieves an average true negative rate of 61.87% and a median of 74.38%, indicating great effectiveness.

The study (30) proposed a structured pipeline for literature screening, which uses a n-consensus mechanism to increase the credibility. To decide if a paper is relevant, multiple LLMs first give their own judgments and then make a final decision through voting. The experiment results demonstrate that the n-consensus mechanism can achieve a 98% recall rate.

LLAssist (10) is another literature screening tool. Its characteristic is to accept formatted CSV input and generate formatted JSON and CSV output to facilitate downstream tasks. Processing datasets of 115 articles requires about 30 minutes and 2,576 articles about 10 hours, significantly reducing manual time.

The study (31) uses a hybrid method combing the traditional machine learning classifier and LLM refinement to conduct the literature screening. In the initial classification, a balanced random forest classifier is used to pre-classify abstracts into "relevant" or "irrelevant". Then the uncertain classifications are reviewed by LLMs in a few-shot manner. The evaluation proves that this hybrid solution can reduce screening time for 6 to 10 times.

Ranking retrieved articles according to relevance is an important step in literature screening. Traditional methods of article ranking include point-wise, pair-wise, and list-wise. The paper (5) proposed a new literature ranking method: the set-wise approach. Table 3.3 displays the difference between these ranking approaches. The evaluation results demonstrate that the set-wise approach is more scalable and effective than the other three methods. It significantly reduces the token consumption while maintaining great performance and robustness.

Method	**Effectiveness**	**Efficiency**	**Robustness**
Point-Wise	Low	High	Unknown
Pair-Wise	High	Low	Unknown
List-Wise	Balanced	Balanced	Struggles
Set-Wise	High	High	Robust

Table 3.3: Comparison of ranking methods, summarized from (5).

3.3 Tools and Techniques for Synthesizing Literature Insights

Literature synthesis is an important stage in a systematic literature review (SLR). At this stage, the screened articles are carefully read and analyzed to extract useful information. Finally, the information extracted from these articles is organized to address research problems, summarize the research status, gaps and trends, and transform it into new viewpoints. LLMs have strong text understanding and generation capabilities, which makes them very suitable for automating literature synthesis. However, problems such as lack of domain information and weak factual fidelity also need to be solved. At present, there are already some studies that attempt to use LLMs to solve the problems of literature synthesis. Below, two of them will be introduced.

Susnjak et al. (11) automated literature synthesis by fine-tuning LLMs with domain-specific knowledge. Language models are usually pre-trained on large-scale corpora but lack domain-specific knowledge. By fine-tuning on domain-specific datasets, LLMs can be endowed with domain knowledge, thereby reducing model hallucinations and enhancing factual fidelity. The fine-tuning framework proposed in this paper is composed of four stages:

1. Q&A pairs generation: LLMs are utilized to generate summaries from the selected literature and then generate a large number of Q&A pairs from these summaries.

2. Token insertion: Some special markers are inserted into the fine-tuning datasets to let the LLM distinguish the knowledge obtained from fine-tuning and pre-training.

3. Q&A pairs permutation: For each Q&A pair, some different questions are with the same meaning are generated by the LLM.

4. Fine-tuning: Parameter efficient fine-tuning (PEFT) (32) is applied in the fine-tuning stage to increase the efficiency.

The author conducted experiments on the previously published SLR using two PEFT methods, LoRA (33) and NEFTune (34) respectively. The results show that NEFTune can achieve an accuracy of 89.2% in generating consistent responses.

Another study, SciDaSynth (35), proposed a structured extraction and synthesis pipeline. It leverages RAG to address the problem of domain-specific information missing and multimodal information extraction. After users upload the articles, the system extracts text, tables, and figures separately, converts them into vector embeddings, and stores the embeddings in vector stores. When users ask questions about these articles, the system will perform a RAG workflow to retrieve related data from the vector stores, and then feed it with the questions together into LLMs to get a response. One of the innovations in this study is that the system marks low-relevant and missing data as highlighted, so that humans can check the data's credibility by retrieving back to the original papers.

3.4 Limitations of Current Research

Although there are some advancements in using LLMs to automate the entire or part of the SLR process, there are still some challenges and limitations left to be addressed.

1. Compatibility with different databases: Obtaining sufficient and comprehensive literature is very important for performing a perfect SLR. And the acquisition of literature mainly relies on academic databases. There are many well-known academic databases, such as IEEE Xplore, ACM Digital Library, etc. They focus on different fields and have included a large number of literature in their respective fields. The research discussed above mainly accesses literature from a single database or from academic search engines like Semantic Scholar, which can only obtain the full text of articles with open licenses; otherwise, it can only obtain abstract information. This makes the literature obtained when performing an SLR once relatively limited,

and important progress may be missing. Therefore, it is necessary to design an SLR automation system that supports as many databases as possible.

2. Full-text analysis: Current SLR automation tools predominantly utilize title-abstract screening to minimize computational load, yet this approach often fails to capture critical methodological details and technical nuances embedded in full texts. Although abstracts often convey the core viewpoints and concepts of papers, they lack design details and experimental evaluation standards, so it is difficult to cover detailed information. To address these limitations, advanced systems like ALISE (4) employ full-text analysis through PDF parsing tools combined with long-context language models. However, such implementations face challenges in balancing parsing accuracy, multimodal compatibility, and computational efficiency - issues partially addressed by emerging multi-agent frameworks (28), though their specific model architectures remain undisclosed.

3. Lack of domain-specific knowledge: Current LLMs employed in academic research are typically pre-trained on multi-domain corpora, enabling broad interdisciplinary understanding. However, this generalized training often fails to develop specialized expertise in specific research domains. When conducting domain-specific SLRs, models may lack critical discipline-specific knowledge bases, potentially leading to hallucinatory outputs in unfamiliar areas - generating plausible but factually incorrect content through speculative inference. To address these limitations, Han et al. ((9)) introduce RAG to enhance factual accuracy and reduce hallucinations. Wang et al. ((35)) suggest that domain adaptation through fine-tuning could potentially establish specialized knowledge frameworks within LLMs. While these approaches demonstrate potential in establishing domain-specific capabilities and reducing hallucinations, achieving complete hallucination elimination remains an ongoing research challenge.

4. Prompt quality and inference speed: Effective prompt engineering plays a critical role in guiding LLM output generation for systematic literature reviews. As demonstrated in (4), precisely formulated prompts establish domain-specific screening criteria, define output structures for literature synthesis, and directly influence model accuracy through contextual instruction embedding. However, these models' computational demands pose significant challenges: their multi-billion parameter architectures require substantial inference time, especially when processing large token

volumes during full-text analysis. This limitation becomes particularly pronounced in resource-constrained environments, as noted in (36), where inadequate GPU support leads to prohibitive latency during multi-stage SLR automation processes.

5. Cost of APIs and deployment: Large language models serve as the foundational engine for SLR automation, with implementation approaches divided into commercial API integration (e.g., ChatGPT, Gemini) and self-hosted deployment of open-source models. While API solutions offer optimized inference performance and maintenance support through enterprise-grade infrastructure (37), their operational costs scale with token volume and model complexity - exemplified by GPT-4o's $2.50/$10.00 per million input/output tokens - compounded by geographic access limitations in certain regions. Conversely, local/cloud deployment provides enhanced data governance and customization capabilities but requires substantial capital investment in GPU clusters and cloud computing resources (4). Both paradigms face critical challenges: API rate limits degrade processing efficiency (30, 36), while hardware provisioning costs and regulatory constraints create persistent financial and operational barriers (36).

4

Design and Implementation

4.1 System Architecture

SLRs are systematic and complex engineering projects that involve stages such as literature search, screening, and synthesis. They are time-consuming and labor-intensive. To automate SLRs using LLMs, I designed and implemented FastSLR. It has features such as support for multiple databases, configurable models, full-text analysis, containerized deployment, and a user-friendly interface. It has made great contributions to achieving efficient and effective SLRs. Next, the system architecture will be introduced as a whole first, and then the design details will be entered.

Figure 4.1 depicts the designed system architecture. It is composed of four layers: access layer, configure layer, service layer, and infrastructure layer. Here are the introduction to these layers:

1. Access layer: Access layer is the entrance for users to use the system. In general, users can use the system in two ways: through web pages or command lines. When using a web page, users first fill in the research topic, selected academic database, number of papers, the LLM to be used and its API Key, and upload research questions in CSV format. After clicking "Run Assistant", SLR starts to execute, and the results will be displayed in tabular form on the web page. In addition, when using a web page, after SLR execution is completed, users can also interactively ask questions related to SLR results through a chatbot, and the system will output answers in a streaming manner. When using the command line, users need to upload configuration information in JSON format, and then run the system through specified commands. The SLR results will be generated in a specified directory in CSV format.

2. Configure layer: Configure layer contains the configuration information set by users, mainly including three parts: "config.json", "questions.json", and chatbot prompts. The "config.json" file stores configuration information such as the research topic, the explanation of the research topic, academic databases and advanced search information, LLM and its API key. The "questions.json" file stores the user's research questions and the format schema of the final generated SLR results. Chatbot prompts are prompts for users to ask about SLR results through an interactive chatbot.

3. Service layer: Service layer is the core of the system for performing SLRs. Generally, performing an SLR by the system can be divided into five stages: metadata download, relevance analysis, filtering, full-text download, and information synthesis. The system supports four academic databases: ACM Digital Library, IEEE Xplore, arXiv, and ScienceDirect. In addition, in order to support the interactive chatbot on web pages, retrieval-enhanced generation (RAG) is required.

4. Infrastructure layer: Infrastructure layer is the foundation for the reliable operation of the system. To support the containerized deployment of the system, Docker (38) is essential. Streamlit (39) provides a beautiful front-end page for easy integration of chatbots. To obtain the metadata information and full text of literature, database APIs, BeautifulSoup (40) for web crawling, and PyMuPDF (41) for parsing PDF files are needed. LLMs such as GPT (20), Qwen (23), and DeepSeek (24) provide the core capabilities of semantic understanding and text generation. To implement RAG, the all-MiniLM-L6-v2 model is used for generating embeddings, and FAISS is used as a vector database.

Figure 4.2 depicts the different processing flow for the four academic databases. This difference is caused by whether the database provides an API and whether it is possible to obtain the abstract of the papers.

4.2 Design Details

4.2.1 User Interface and Containerization

To enhance the user experience, an elegant user interface is essential. Streamlit (39) provides out-of-the-box Python APIs that can generate beautiful interfaces with simple code. I have made a reasonable design for FastSLR, using the left side of the page as

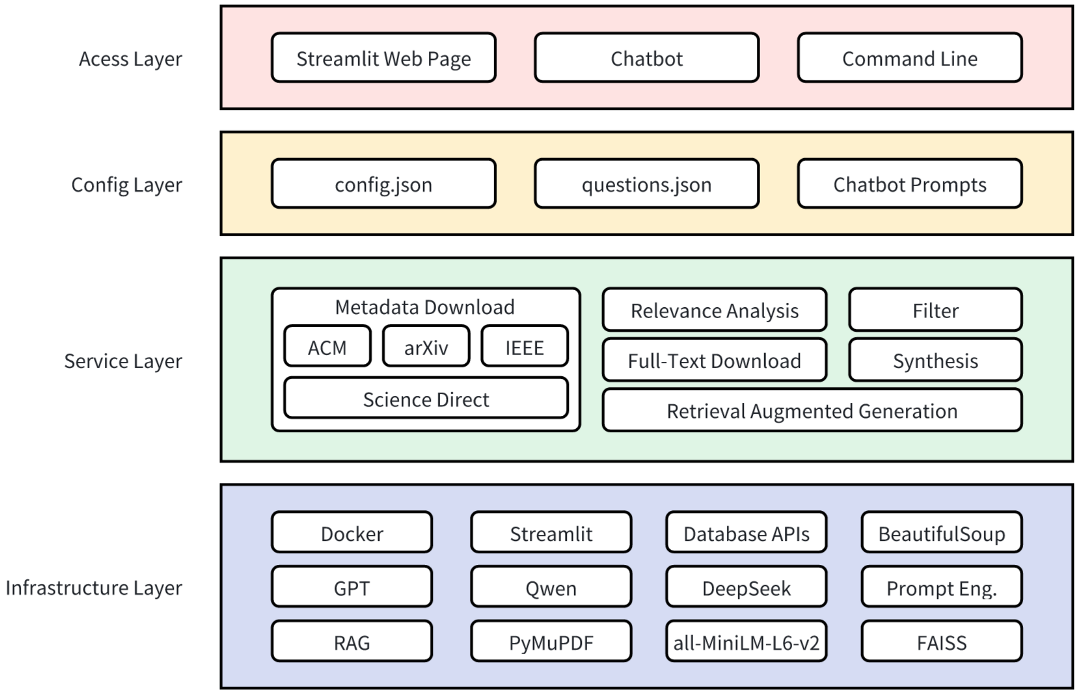


Figure 4.1: System architecture.

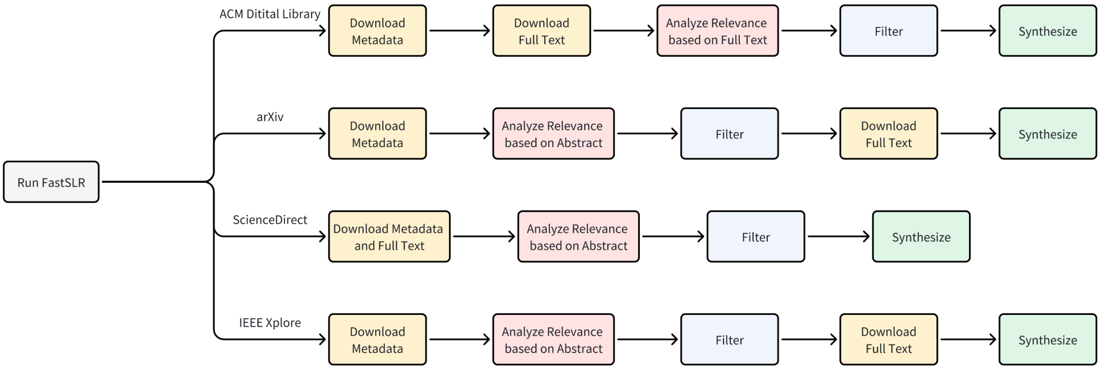
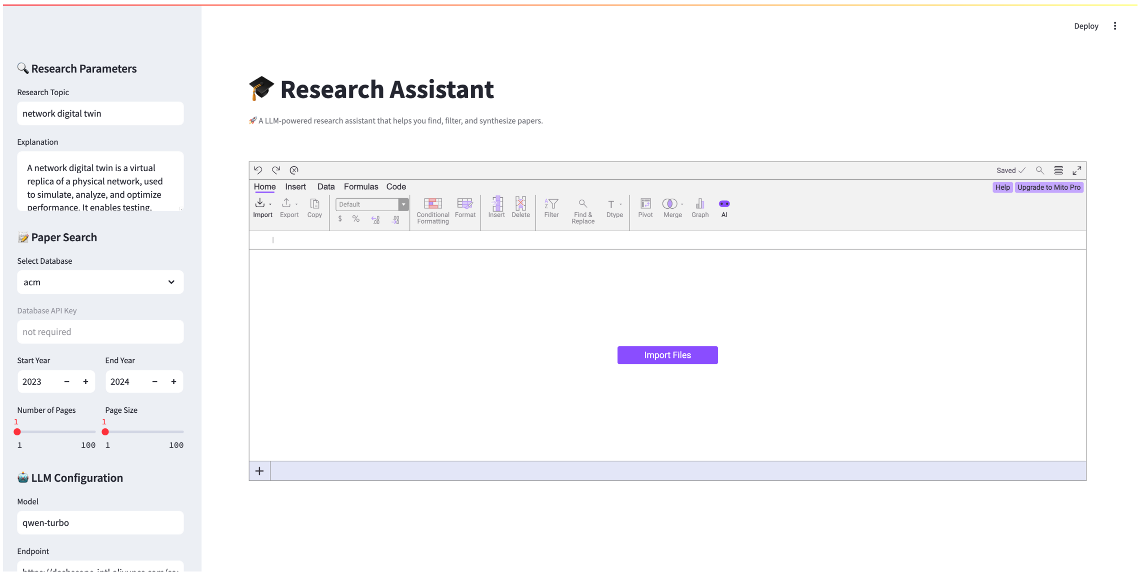


Figure 4.2: Processing flow for different databases.



Research Parameters

Research Topic: network digital twin

Explanation: A network digital twin is a virtual replica of a physical network, used to simulate, analyze, and optimize performance. It enables testing.

Paper Search

Select Database: acm

Database API Key: not required

Start Year: 2023

End Year: 2024

Number of Pages: 1 (1–100)

Page Size: 1 (1–100)

LLM Configuration

Model: qwen-turbo

Endpoint

Research Assistant

A LLM-powered research assistant that helps you find, filter, and synthesize papers.

Figure 4.3: User interface.

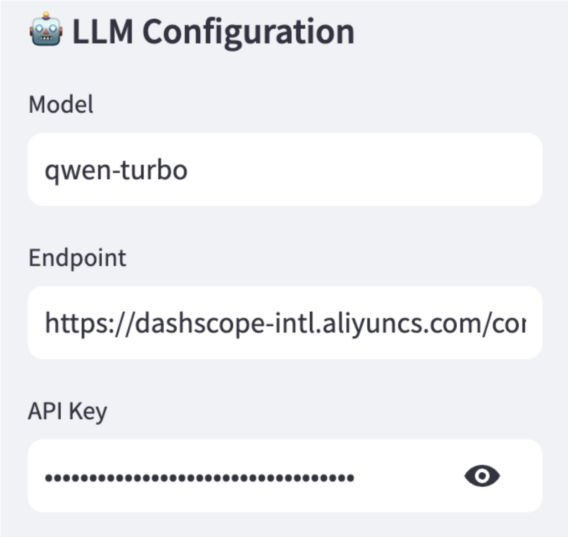
the configuration area and the right side as the display area, enabling users to quickly get started. Figure 4.3 shows the user interface.

In addition, containerization can lower the threshold for users, increase deployment speed, and avoid the troubles caused by downloading source code and missing dependent libraries. Therefore, I have written a Dockerfile and pushed the container image to Dockerhub for users' convenient use.

The Dockerhub address of FastSLR: https://hub.docker.com/repository/docker/hanqihua/llm-assistant/general.

4.2.2 Metadata Download

In the metadata download stage, literature search needs to be performed in different academic databases according to research topics, publication years of literature, and the number of literature to obtain metadata such as article titles, abstracts, authors, and publication times. Since ACM Digital Library and arXiv do not have open APIs, while IEEE Xplore and ScienceDirect have APIs that can be called, different search strategies need to be formulated for different databases. In addition, the number of literature searches supported by different databases is also different. For example, ACM Digital Library supports 1 to 100, and arXiv supports 50 or 100, etc. Therefore, it is also necessary to enable users to select the correct quantity on the front end page and further perform parameter verification on the back end.



🤖 LLM Configuration

Model

qwen-turbo

Endpoint

https://dashscope-intl.aliyuncs.com/co

API Key

•••••••••••••••••••••••••••••••••

Figure 4.4: Configurable LLMs.

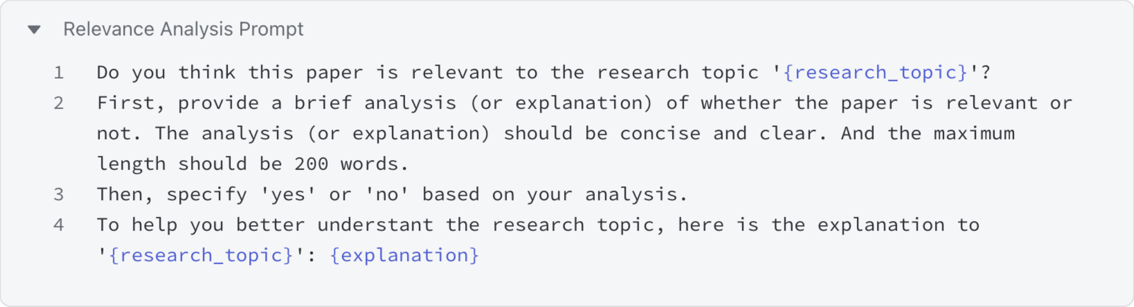
4.2.3 Configurable Models

Nowadays, there are many popular LLMs, such as the GPT series, the Qwen series, DeepSeek series, etc. Different models have different calling prices and supported regions. Moreover, with the continuous research and development investment of various technology companies, LLMs are constantly being updated and iterated, and their capabilities are gradually improving. Therefore, making the model used by the system configurable can not only meet the needs of different users but also automatically support the latest models to enhance the system's capabilities. The model configuration information mainly includes model name, API address, API Key, etc. Figure 4.4 is the LLM configuration presented on the Web page.

4.2.4 Relevance Analysis

In the relevance analysis stage, the system needs to determine whether the abstract or full text of each article is relevant to the user's research topic. I made two unique designs in this stage to ensure the accuracy and transparency of LLM analysis.

1. Users need to provide an explanation of the research topic: In addition to providing the research topic, users also need to provide a paragraph to explain the research topic. This is to prevent LLMs from misunderstanding the user's research intention. For example, if the research topic provided by the user is "Network Digital Twins" and it is not specified whether it is a communication network or a neural



▼ Relevance Analysis Prompt

```
Do you think this paper is relevant to the research topic '{research_topic}'?
First, provide a brief analysis (or explanation) of whether the paper is relevant or not. The analysis (or explanation) should be concise and clear. And the maximum length should be 200 words.
Then, specify 'yes' or 'no' based on your analysis.
To help you better understant the research topic, here is the explanation to '{research_topic}': {explanation}
```

Figure 4.5: Prompt for relevance analysis.

network, this creates ambiguity. When LLMs perform classification, they may analyze incorrectly and produce incorrect results. Therefore, after the user provides the explanatory information of the research topic, the LLM can accurately judge the relevance according to the explanatory information.

2. LLMs need to output explanations for judging relevance: The interpretability of LLMs are poor. In order to improve the transparency of LLMs in judging whether an article is relevant, prompts are used to require LLMs to first output their analysis of relevance and then output the result of whether it is relevant. This can enable users to know the basis and analysis process of LLMs for judging relevance.

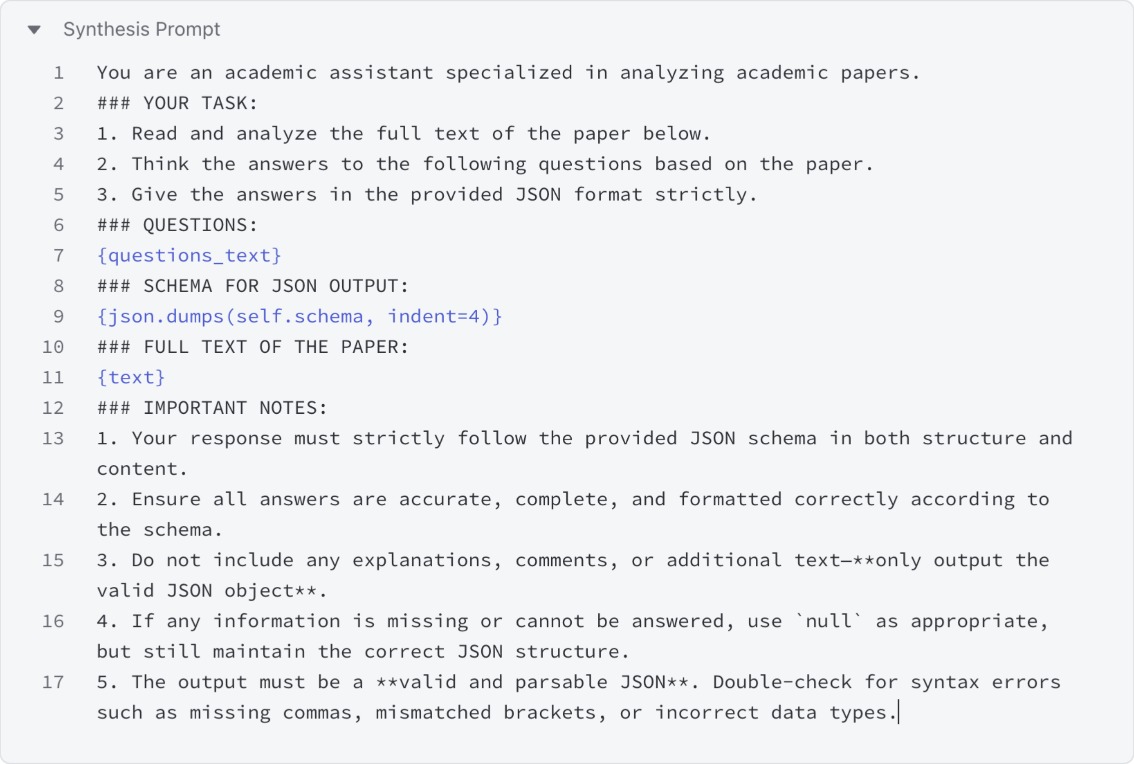
The prompt for relevance analysis is in figure 4.5, which integrates the two designs and intends to generate a more accurate and transparent output.

4.2.5 Full-text Analysis

The abstract of a paper is a summary and generalization of the article's content. However, a large number of design details are in the full text and cannot be reflected in the abstract. Therefore, analyzing the full text using LLMs are beneficial for an accurate and comprehensive SLR. To achieve full-text analysis, I use Python's PyMuPDF library to extract the text from the paper's PDF and combine it with the research questions and result schema provided by the user into the prompt. The prompt for synthesis stage is in figure 4.6 and 4.7 is an example of user-provided questions and schema.

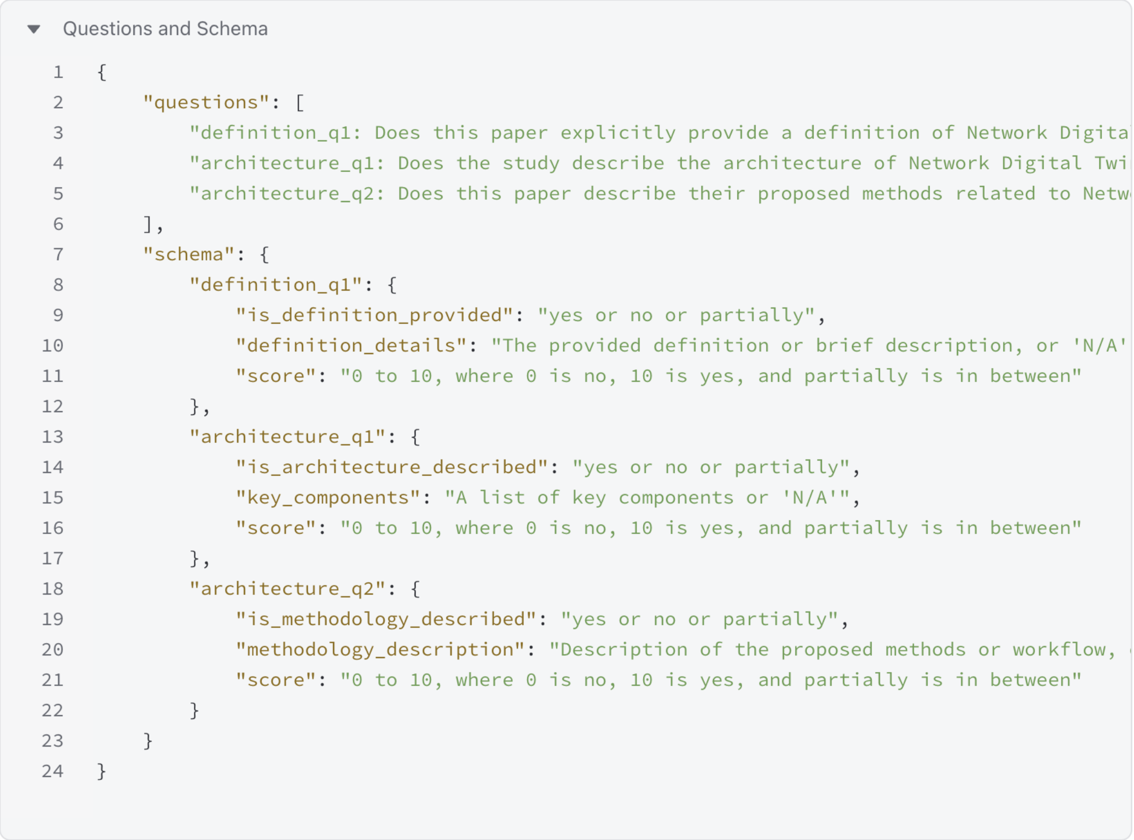
4.2.6 RAG-based Chatbot

When an SLR execution is completed, users may still have questions about the results of the SLR or the content of the papers. At this time, an interactive chatbot can become



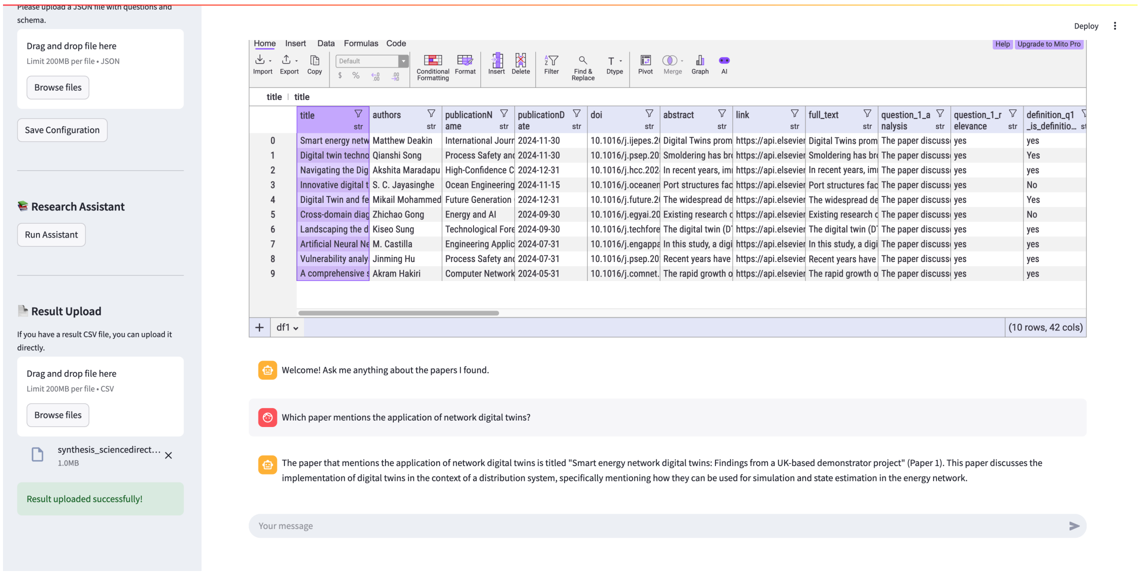
```
You are an academic assistant specialized in analyzing academic papers.
### YOUR TASK:
1. Read and analyze the full text of the paper below.
2. Think the answers to the following questions based on the paper.
3. Give the answers in the provided JSON format strictly.
### QUESTIONS:
{questions_text}
### SCHEMA FOR JSON OUTPUT:
{json.dumps(self.schema, indent=4)}
### FULL TEXT OF THE PAPER:
{text}
### IMPORTANT NOTES:
1. Your response must strictly follow the provided JSON schema in both structure and content.
2. Ensure all answers are accurate, complete, and formatted correctly according to the schema.
3. Do not include any explanations, comments, or additional text—**only output the valid JSON object**.
4. If any information is missing or cannot be answered, use `null` as appropriate, but still maintain the correct JSON structure.
5. The output must be a **valid and parsable JSON**. Double-check for syntax errors such as missing commas, mismatched brackets, or incorrect data types.
```

Figure 4.6: Prompt for synthesis.



```
{
    "questions": [
        "definition_q1: Does this paper explicitly provide a definition of Network Digita
        "architecture_q1: Does the study describe the architecture of Network Digital Twi
        "architecture_q2: Does this paper describe their proposed methods related to Netw
    ],
    "schema": {
        "definition_q1": {
            "is_definition_provided": "yes or no or partially",
            "definition_details": "The provided definition or brief description, or 'N/A'
            "score": "0 to 10, where 0 is no, 10 is yes, and partially is in between"
        },
        "architecture_q1": {
            "is_architecture_described": "yes or no or partially",
            "key_components": "A list of key components or 'N/A'",
            "score": "0 to 10, where 0 is no, 10 is yes, and partially is in between"
        },
        "architecture_q2": {
            "is_methodology_described": "yes or no or partially",
            "methodology_description": "Description of the proposed methods or workflow,
            "score": "0 to 10, where 0 is no, 10 is yes, and partially is in between"
        }
    }
}
```

Figure 4.7: An example of user-provided questions and schema.



Please upload a JSON file with questions and schema.

Drag and drop file here
Limit 200MB per file • JSON

📚 Research Assistant

📄 Result Upload

If you have a result CSV file, you can upload it directly.

Drag and drop file here
Limit 200MB per file • CSV

synthesis_sciencedirect…
1.0MB

Result uploaded successfully!

	title	authors	publicationName	publicationDate	doi	abstract	link	full_text	question_1_analysis	question_1_relevance	definition_q1_is_definitio…
	str	str	str	str	str	str	str	str	str	str	st
0	Smart energy netw	Matthew Deakin	International Journ	2024-11-30	10.1016/j.ijepes.2	Digital Twins prom	https://api.elsevier	Digital Twins prom	The paper discuss	yes	yes
1	Digital twin techno	Qianshi Song	Process Safety and	2024-11-30	10.1016/j.psep.20	Smoldering has br	https://api.elsevier	Smoldering has br	The paper discuss	yes	Yes
2	Navigating the Dig	Akshita Maradapu	High-Confidence C	2024-12-31	10.1016/j.hcc.202	In recent years, im	https://api.elsevier	In recent years, im	The paper discuss	yes	yes
3	Innovative digital t	S. C. Jayasinghe	Ocean Engineering	2024-11-15	10.1016/j.oceanen	Port structures fac	https://api.elsevier	Port structures fac	The paper discuss	yes	No
4	Digital Twin and fe	Mikail Mohammed	Future Generation	2024-12-31	10.1016/j.future.2	The widespread de	https://api.elsevier	The widespread de	The paper discuss	yes	Yes
5	Cross-domain diag	Zhichao Gong	Energy and AI	2024-09-30	10.1016/j.egyai.20	Existing research c	https://api.elsevier	Existing research c	The paper discuss	yes	No
6	Landscaping the d	Kiseo Sung	Technological Fore	2024-09-30	10.1016/j.techfore	The digital twin (D	https://api.elsevier	The digital twin (D	The paper discuss	yes	yes
7	Artificial Neural Ne	M. Castilla	Engineering Applic	2024-07-31	10.1016/j.engappa	In this study, a digi	https://api.elsevier	In this study, a digi	The paper discuss	yes	yes
8	Vulnerability analy	Jinming Hu	Process Safety and	2024-07-31	10.1016/j.psep.20	Recent years have	https://api.elsevier	Recent years have	The paper discuss	yes	yes
9	A comprehensive s	Akram Hakiri	Computer Network	2024-05-31	10.1016/j.comnet.	The rapid growth o	https://api.elsevier	The rapid growth o	The paper discuss	yes	yes

(10 rows, 42 cols)

Welcome! Ask me anything about the papers I found.

Which paper mentions the application of network digital twins?

The paper that mentions the application of network digital twins is titled "Smart energy network digital twins: Findings from a UK-based demonstrator project" (Paper 1). This paper discusses the implementation of digital twins in the context of a distribution system, specifically mentioning how they can be used for simulation and state estimation in the energy network.

Figure 4.8: RAG-based chatbot.

an assistant for users to understand the papers. Since LLMs themselves do not have the knowledge of the current SLR, it is necessary to use RAG to help LLMs obtain the knowledge. Therefore, I designed and implemented a chatbot based on RAG. First, the full text of all papers is segmented and tokenized. Then, the embedding model is used to convert tokens into embeddings and store them in the vector database FAISS. When a user asks a question, relevant content is first retrieved from FAISS, and then combined with the user's question and given to the LLM to generate a reply. On the web page, I implemented the streaming output of LLM answers to improve user experience. Figure 4.8 displays the usage of chatbot on the web page.

5

Evaluation

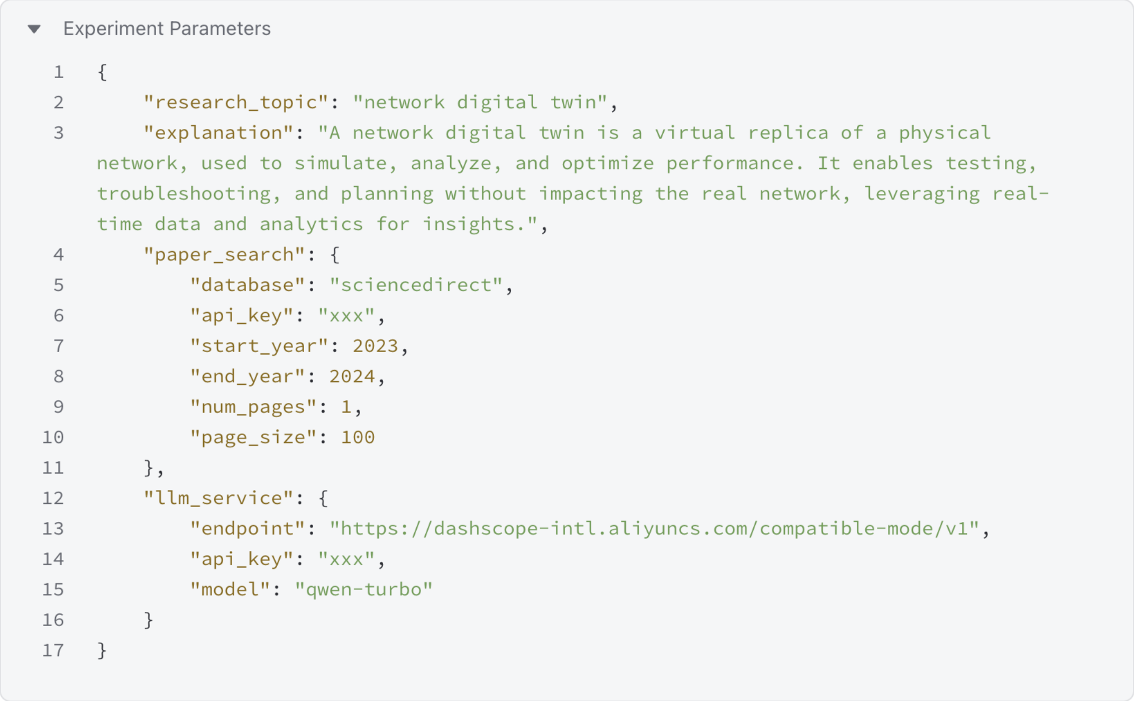
5.1 Time to Conduct an SLR

An important goal of automating SLRs using LLMs are to reduce time consumption and improve efficiency. Therefore, I conducted an experiment to test the time consumed by performing an SLR using FastSLR. The research topic I chose is "Network Digital Twins" and the detailed parameters are shown in figure 5.1. For synthesizing stage, I selected ten questions and each question has two or three sub-questions.

After launching FastSLR, it automatically searched and downloaded the metadata and full texts of 100 papers, analyzed the relevance to "Network Digital Twins" and filtered out 82 relevant papers. Finally, it answered questions one by one and generated replies. The whole process only took 41 minutes and 28 seconds. This significantly reduces the time required for humans to complete the same amount of work.

5.2 Literature Screening Accuracy

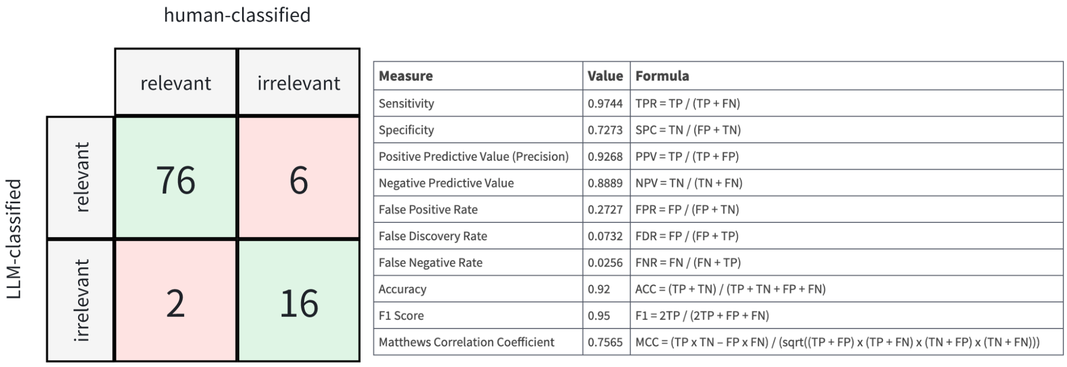
FastSLR can significantly improve the efficiency of executing SLRs, but its effectiveness still needs to be further verified. In the above experiment, FastSLR screened out 82 papers related to the research topic from the 100 papers it searched. To verify the accuracy of FastSLR in screening literature, I manually screened these 100 papers and compared the results with those of FastSLR. The confusion matrix and some measures are in figure 5.2. The results show that in this SLR, FastSLR achieved an accuracy of 0.92, a precision of 0.9268, and a recall rate of 0.9744. This indicates that FastSLR can screen out literature related to the research topic accurately and comprehensively.



▼ Experiment Parameters

```
{
    "research_topic": "network digital twin",
    "explanation": "A network digital twin is a virtual replica of a physical network, used to simulate, analyze, and optimize performance. It enables testing, troubleshooting, and planning without impacting the real network, leveraging real-time data and analytics for insights.",
    "paper_search": {
        "database": "sciencedirect",
        "api_key": "xxx",
        "start_year": 2023,
        "end_year": 2024,
        "num_pages": 1,
        "page_size": 100
    },
    "llm_service": {
        "endpoint": "https://dashscope-intl.aliyuncs.com/compatible-mode/v1",
        "api_key": "xxx",
        "model": "qwen-turbo"
    }
}
```

Figure 5.1: Experiment parameters.



		human-classified	
		relevant	irrelevant
LLM-classified	relevant	76	6
	irrelevant	2	16

Measure	Value	Formula
Sensitivity	0.9744	TPR = TP / (TP + FN)
Specificity	0.7273	SPC = TN / (FP + TN)
Positive Predictive Value (Precision)	0.9268	PPV = TP / (TP + FP)
Negative Predictive Value	0.8889	NPV = TN / (TN + FN)
False Positive Rate	0.2727	FPR = FP / (FP + TN)
False Discovery Rate	0.0732	FDR = FP / (FP + TP)
False Negative Rate	0.0256	FNR = FN / (FN + TP)
Accuracy	0.92	ACC = (TP + TN) / (TP + TN + FP + FN)
F1 Score	0.95	F1 = 2TP / (2TP + FP + FN)
Matthews Correlation Coefficient	0.7565	MCC = (TP x TN – FP x FN) / (sqrt((TP + FP) x (TP + FN) x (TN + FP) x (TN + FN)))

Figure 5.2: Confusion matrix for LLM's classification of literature relevance.

6

Discussion

6.1 Summary of Findings

The development and implementation of FastSLR represent a significant step forward in automating Systematic Literature Reviews (SLRs) using Large Language Models (LLMs). The experimental evaluation demonstrates that FastSLR can significantly reduce the time required to conduct an SLR while maintaining high accuracy in literature screening. For instance, in the experiment on "Network Digital Twins", FastSLR conducted an entire SLR on 100 papers in just about 41 minutes. This is a remarkable improvement over traditional SLR methods, which typically take months to complete.

Furthermore, the system achieved an accuracy of 0.92, a precision of 0.9268, and a recall rate of 0.9744 in screening relevant literature. These metrics indicate that FastSLR is not only efficient but also effective in identifying and synthesizing pertinent research materials. The integration of features such as multi-database support, configurable models, full-text analysis, and Retrieval-Augmented Generation (RAG) has contributed to this success.

6.2 Implications of the Results

The findings of this research have several important implications for both academia and industry. First, FastSLR addresses many of the challenges associated with traditional SLRs, such as their time-consuming nature, labor-intensive processes, and potential for human error. By automating key stages of the SLR workflow—ranging from literature search and screening to data extraction and synthesis—FastSLR enables researchers to focus on higher-level tasks, such as interpreting results and generating new research ideas.

Second, the use of LLMs and RAG in FastSLR highlights the transformative potential of artificial intelligence in academic research. These technologies allow for more accurate and context-aware analyses, which are crucial for fields where up-to-date and reliable information is paramount, such as healthcare, engineering, and computer science. Additionally, the transparency introduced by requiring LLMs to provide explanations for their relevance judgments enhances trust in AI-generated outputs, addressing one of the major criticisms of LLMs in academic contexts.

Finally, the modular and user-friendly design of FastSLR makes it accessible to researchers with varying levels of technical expertise. Features such as containerized deployment and an intuitive web interface lower the barrier to entry, enabling broader adoption across different research communities.

6.3 Limitations

Despite its promising performance, there are two limitations need to be considered.

1. **Lack of Multi-Modal Model Integration:** FastSLR currently does not incorporate multi-modal models, which could enhance its ability to process and analyze non-textual data such as charts, graphs, and images commonly found in academic papers. Including multi-modal capabilities would enable the system to extract valuable insights from visual content, thereby improving the comprehensiveness of literature analysis.

2. **Limited Academic Database Support:** While FastSLR supports several prominent academic databases, there is a need to expand compatibility to include additional databases. Broader database integration would ensure that researchers can access an even wider range of scholarly resources, further enhancing the system's utility across diverse research fields.

7

Conclusion

FastSLR represents a significant advancement in using Large Language Models (LLMs) to automate Systematic Literature Reviews (SLRs). By addressing key limitations of traditional SLRs—such as their time-consuming nature, labor-intensive processes, and potential for human error—FastSLR enables researchers to conduct comprehensive reviews efficiently and effectively.

The system's ability to conduct an SLR of 100 papers in about 41 minutes highlights its transformative potential. With an accuracy of 0.92, precision of 0.9268, and recall rate of 0.9744, FastSLR demonstrates both efficiency and reliability in automating SLRs. Its innovative features, including multi-database support, configurable models, full-text analysis, and RAG-based chatbot functionality, further enhance its usability and adaptability.

Despite these achievements, challenges remain. The lack of multi-modal model integration limits FastSLR's ability to analyze non-textual data, while expanding database compatibility could improve access to scholarly resources. Future work should focus on the processing of multi-modal information and support for more databases.

In conclusion, FastSLR has demonstrated its potential to revolutionize SLRs, empowering researchers to navigate the growing body of academic literature with unprecedented speed and precision. As AI technologies continue to evolve, systems like FastSLR will play an increasingly vital role in advancing scientific discovery.

References

[1] Christos Papakostas, Christos Troussas, Akrivi Krouska, and Cleo Sgouropoulou. **Exploration of Augmented Reality in Spatial Abilities Training: A Systematic Literature Review for the Last Decade**. *Informatics Educ.*, **20**:107–130, 2021. iii, 5

[2] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. **Attention is All you Need**. In *Neural Information Processing Systems*, 2017. iii, 1, 8, 9

[3] Shubham Agarwal, Issam Hadj Laradji, Laurent Charlin, and Christopher Pal. **LitLLM: A Toolkit for Scientific Literature Review**. *ArXiv*, **abs/2402.01788**, 2024. iii, 2, 15, 16

[4] Hendrik Roth and Carsten Lanquillon. **ALISE: An Automated Literature Screening Engine for Research**. In *International Conference on Agents and Artificial Intelligence*, 2024. iii, 17, 20, 21

[5] Shengyao Zhuang, Honglei Zhuang, Bevan Koopman, and G. Zuccon. **A Setwise Approach for Effective and Highly Efficient Zero-shot Ranking with Large Language Models**. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023. v, 17, 18

[6] Debajyoti Pati and Lesa N. Lorusso. **How to Write a Systematic Review of the Literature**. *HERD: Health Environments Research & Design Journal*, **11**:15 – 30, 2018. 1, 4

[7] Angela Carrera-Rivera, William Ochoa-Agurto, Felix Larrinaga, and Ganix Lasa. **How-to conduct a systematic literature review: A quick guide for computer science research**. *MethodsX*, **9**, 2022. 1, 16

[8] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Ma teusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. **Language Models are Few-Shot Learners**. *ArXiv*, **abs/2005.14165**, 2020. 2

[9] Binglan Han, Teo Susnjak, and Anuradha Mathrani. **Automating Systematic Literature Reviews with Retrieval-Augmented Generation: A Comprehensive Overview**. *Applied Sciences*, 2024. 2, 15, 16, 20

[10] Christoforus Yoga Haryanto. **LLAssist: Simple Tools for Automating Literature Review Using Large Language Models**. *ArXiv*, **abs/2407.13993**, 2024. 2, 17

[11] Teo Susnjak, Peter Hwang, Napoleon H. Reyes, Andre L. C. Barczak, Timothy R. McIntosh, and Surangika Ranathunga. **Automating Research Synthesis with Domain-Specific Large Language Model Fine-Tuning**. *ArXiv*, **abs/2404.08680**, 2024. 2, 18

[12] Quanjun Zhang, Chunrong Fang, Yang Xie, Yuxiang Ma, Weisong Sun, Yun Yang, and Zhenyu Chen. **A Systematic Literature Review on Large Language Models for Automated Program Repair**. *ArXiv*, **abs/2405.01466**, 2024. 3

[13] Arnab Barua, Mobyen Uddin Ahmed, and Shahina Begum. **A Systematic Literature Review on Multimodal Machine Learning: Applications, Challenges, Gaps and Future Directions**. *IEEE Access*, **11**:14804–14831, 2023. 4

[14] Zhaohai Zheng, Fang Peng, Buyun Xu, Jingjing Zhao, Huahua Liu, Jiahao Peng, Qingsong Li, Chiqiu Jiang, Yan Zhou, Shuqing Liu, Chunji Ye, Peng Zhang, Yangbo Xing, Hangyuan Guo, and Weiliang Tang. **Risk factors of critical & mortal COVID-19 cases: A systematic literature review and meta-analysis**. *The Journal of Infection*, **81**:e16 – e25, 2020. 4

[15] **IEEE Xplore**. https://ieeexplore.ieee.org/Xplore/home.jsp. 6

[16] **ACM Digital Library**. https://dl.acm.org/. 7

[17] **ScienceDirect**. https://www.sciencedirect.com/. 7

[18] **arXiv**. https://arxiv.org/. 7

[19] Simpy Amit Mahuli, Arpita Rai, Amit Vasant Mahuli, and Ansul Kumar. **Application ChatGPT in conducting systematic reviews and meta-analyses**. *British Dental Journal*, **235**:90 – 92, 2023. 7

[20] **ChatGPT**. https://chatgpt.com/. 10, 11, 24

[21] **Claude**. https://claude.ai/. 10, 11

[22] **Semantic Scholar**. https://www.semanticscholar.org/. 10

[23] **Qwen**. https://chat.qwen.ai/. 11, 24

[24] **DeepSeek**. https://www.deepseek.com/. 11, 24

[25] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. **Retrieval-augmented generation for knowledge-intensive nlp tasks**. *Advances in neural information processing systems*, **33**:9459–9474, 2020. 12

[26] Hans Christian, Mikhael Pramodana Agus, and Derwin Suhartono. **Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF)**. *ComTech: Computer, Mathematics and Engineering Applications*, **7**(4):285–294, 2016. 13

[27] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. **Dense Passage Retrieval for Open-Domain Question Answering.** In *EMNLP (1)*, pages 6769–6781, 2020. 13

[28] Malik Abdul Sami, Zeeshan Rasheed, Kai-Kristian Kemell, Muhammad Waseem, Terhi Kilamo, Mika Saari, Anh Nguyen-Duc, Kari Systä, and

Pekka Abrahamsson. **System for systematic literature review using multiple AI agents: Concept and an empirical evaluation**. *ArXiv*, **abs/2403.08399**, 2024. 15, 16, 20

[29] Siw Waffenschmidt, Marco Knelangen, Wiebke Sieben, Stefanie Bühn, and Dawid Pieper. **Single screening versus conventional double screening for study selection in systematic reviews: a methodological systematic review**. *BMC Medical Research Methodology*, **19**, 2019. 16

[30] Lucas Joos, Daniel A. Keim, and M. T. Fischer. **Cutting Through the Clutter: The Potential of LLMs for Efficient Filtration in Systematic Literature Reviews**. *ArXiv*, **abs/2407.10652**, 2024. 17, 21

[31] Paul Herbst and Henning Baars. **Accelerating literature screening for systematic literature reviews with Large Language Models - development, application, and first evaluation of a solution**. In *Lernen, Wissen, Daten, Analysen*, 2023. 17

[32] Jeremy Howard and Sebastian Ruder. **Universal Language Model Fine-tuning for Text Classification**. In *Annual Meeting of the Association for Computational Linguistics*, 2018. 19

[33] J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. **LoRA: Low-Rank Adaptation of Large Language Models**. *ArXiv*, **abs/2106.09685**, 2021. 19

[34] Neel Jain, Ping yeh Chiang, Yuxin Wen, John Kirchenbauer, Hong-Min Chu, Gowthami Somepalli, Brian Bartoldson, Bhavya Kailkhura, Avi Schwarzschild, Aniruddha Saha, Micah Goldblum, Jonas Geiping, and Tom Goldstein. **NEFTune: Noisy Embeddings Improve Instruction Finetuning**. *ArXiv*, **abs/2310.05914**, 2023. 19

[35] Xingbo Wang, Samantha Lee Huey, Rui Sheng, Saurabh Mehta, and Fei Wang. **SciDaSynth: Interactive Structured Knowledge Extraction and Synthesis from Scientific Literature with Large Language Model**. *ArXiv*, **abs/2404.13765**, 2024. 19, 20

[36] Pablo Castillo-Segura, Carlos Alario-Hoyos, Carlos Delgado Kloos, and Carmen Fernández Panadero. **Leveraging the Potential of Generative**

AI to Accelerate Systematic Literature Reviews: An Example in the Area of Educational Technology. *2023 World Engineering Education Forum - Global Engineering Deans Council (WEEF-GEDC)*, pages 1–8, 2023. 21

[37] **OpenAI pricing**. https://openai.com/api/pricing/. 21

[38] **Docker**. https://www.docker.com/. 24

[39] **Streamlit**. https://streamlit.io/. 24

[40] **BeautifulSoup**. https://pypi.org/project/beautifulsoup4/. 24

[41] **PyMuPDF**. https://pypi.org/project/PyMuPDF/. 24