Agentic RAG for Personalized Learning: Design of an AI-Powered Learning Agent Using Open-Source Small Language Models

Shilpi Taneja, Siddhartha Sankar Biswas, Bhavya Alankar and Harleen Kaur
Department of Computer Science and Engineering, School of Engineering Sciences and Technology, Jamia Hamdard, New Delhi, India

shilpi.asija@gmail.com

https://doi.org/10.34190/ejel.23.4.4044

Abstract: This paper presents the design of a personalized learning agent powered by the Agentic RAG technique. The agent can interpret learners' queries and autonomously decide which tools should be used to generate the most suitable response. When the learner shares an Open Educational Resource (OER) they wish to learn from, the agent first breaks the content into smaller, manageable chunks. These chunks are then indexed sequentially to preserve the natural flow of the text. At the same time, chunks are also converted into vector embeddings that allow semantic retrieval. Depending on the learner's request, different tools are selected by the agent. For example, when the learner requests learning aids like summaries, quizzes, or flashcards, the agent invokes the corresponding tool. This tool passes the sequentially indexed chunks to a small language model to generate the output. For context-specific queries, another specialized tool that relies on vector indexing and retrieval-augmented generation (RAG), is invoked. Visual question answering is handled by a separate tool that leverages multimodal RAG using a multimodal small language model. This agentic setup improves the accuracy and relevance of responses generated by the agent. To test its agentic behaviour, we probed our agent with a diverse set of questions drawn from four different OERs. We thoroughly examined each response and tracked the tools that got invoked autonomously. We also compared the similarity of summaries produced by our agent against those generated by ChatGPT (GPT-4o) using BERT Score as the evaluation metric. Our findings indicate that the agent consistently selected the appropriate tools and the summaries generated by our agent showed close semantic similarity to those produced by GPT-4o, suggesting that the proposed approach can provide performance reasonably close to a state-of-the-art model. The agent being lightweight resides on learner's local machine and avoid dependence on cloud-based AI ensuring the privacy of learner's data. It is affordable as it entirely relies on open source frameworks and small models. As the agent provides personalized support to learners by answering their context-based queries and providing on-demand learning aids, it improves their engagement with the educational content. This research shows that designing agentic AI tools using open-source software to address diverse learning needs is technically and economically feasible as well as educationally valuable.

Keywords: AI in education, LlamaIndex, Agentic RAG, Small language model, Generative artificial intelligence, OER

1. Introduction

The field of personalized learning has seen a number of studies that investigate individuals' learning needs in order to improve learning outcomes through adaptive content delivery and feedback (Holmes, Bialik and Fadel, 2019; Watters, 2023). The recent advances in generative artificial intelligence are impacting education in many ways. Large language models (LLMs), for example, can interact with learners, provide answers to their educational queries with explanations and interactive feedback (Bozkurt, 2023). With the increasing availability of open-source LLMs and small language models (SLMs), it has become technically feasible and economical to design intelligent learning assistants that provide personalized educational support to learners.

Despite their brilliant capabilities, the downside of using LLMs for educational support is that they sometimes fail to provide domain-specific educational information. This is because their responses are based on their pre-training data rather than learning resources. A known drawback of LLMs is hallucinations, responses that appear plausible but are in fact fabricated and not factual. When LLMs do not have the necessary context needed to answer the learners' query, they often end up hallucinating and may mislead learners. Retrieval-Augmented Generation (RAG) mitigates this limitation to a large extent as it grounds the LLM's responses in domain-specific knowledge and improves completeness, accuracy and relevance of the responses (Lewis et al., 2020).

In practice, AI-powered personalized learning is largely dependent on cloud-based LLMs (Chimezie, 2024; Sajja et al., 2024; Slade, Hyk and Gurung, 2024) that have notable limitations, such as privacy concerns and cost. This paper demonstrates how open-source small language models combined with Agentic AI techniques can be used to build autonomous learning agents that preserve privacy and are affordable. These models can run on modest hardware; for instance, running a less than 7b parameter model with Ollama typically requires 8 to16 GB RAM and a modern CPU with at least 4 cores, making it feasible for standard personal computers (GPU Mart, 2025).

Cite this article: Taneja, S.et al. 2025. "Agentic RAG for Personalized Learning: Design of an AI-Powered Learning Agent Using Open-Source Small Language Models", *Electronic Journal of e-Learning* , 23(4), pp 69-80, https://doi.org/10.34190/ejel.23.4.4044

In our earlier work, we outlined the architecture of an AI-powered personal learning assistant using multi-modal Retrieval Augmented Generation (RAG) with small language models (Taneja et al., in press). The present study builds on that design by integrating an emerging approach 'Agentic RAG' in it (Singh et al., 2025). With this addition, the agent not only answer the learner's queries from an Open Educational Resource (OER) in natural language, but autonomously decide which tool is most relevant for the query. In educational context, this helps learners by answering their context-specific queries and providing on-demand learning aids such as summaries, flashcards, and quizzes.

To simplify the interaction between learner and agent, we implemented a chatbot interface. This was designed using Gradio, which is an open-source Python package used for quickly building AI applications (DeepLearning.AI, 2024b). The chatbot maintains the context of the conversations, which allows learners to ask follow-up questions based on the ongoing interaction.

The study is guided by the following primary research question: How can open, locally-deployable Agentic RAG-based AI systems be designed to match or surpass the pedagogical utility of cloud-based AI tools, while preserving learner privacy and minimizing cost? The study further explores the following sub-questions, in order to answer this main question:

Q1. How can Agentic RAG enhance the learning agents by answering context-specific educational queries and providing learning-aids such as summaries, flashcards, and quizzes?

Q2. Are there any design and deployment guidelines for creating affordable, multimodal and context-aware learning agents that run locally using open-source tools and how these learning agents compare with cloud-based AI tools?

The rest of the paper is organized as follows: Section 2 discusses related work that has already been done in this field and identifies the gaps in current research which can be filled by our work. Section 3 explains the system design of proposed agent in detail. Section 4 presents development methodology and provide practical guidelines on the implementation. Section 5 reports the analysis and results, while Section 6 summarizes key findings and discusses limitations of the current study. Finally, Section 7 points to the potential directions for future research.

2. Related Work

In this section, we review prior work in the fields of AI-powered personalized learning based on language models. We highlight the main contributions of these studies and state how our approach differs from them.

2.1 Cloud-Hosted LLM Based Learning Assistants

Several studies have proposed intelligent assistants built on cloud-hosted LLMs. For instance, an AI enabled intelligent assistant presented in a recent work provides many features similar to our proposed agent, such as summaries, quizzes, flashcards, and context-relevant responses (Sajja et al., 2024). However, their system uses GPT 3.5, which is hosted on the cloud, and hence it doesn't provide data confidentiality. Moreover, GPT 3.5 is a text-based model, so the assistant lacks multimodal capabilities. Other cloud-based learning assistants/intelligent tutors have been proposed for teaching Data Structures & Algorithms and Introductory Psychology courses using a RAG approach (Chimezie, 2024; Slade, Hyk and Gurung, 2024). While these works demonstrate the utility of cloud-based LLMs in education, they raise concerns of privacy, cost, and accessibility.

2.2 Open-Source SLM Based Learning Assistants

Recent studies have started exploring the efficacy of small language models with RAG for educational purpose. For instance, a recent work used is neural-chat-7b-v3, a fine-tuned version of Mistral-7B-v0.1 to provide learning support for computing education (Liu et al., 2024).The model they have used is text-based, therefore limiting its use to non-visual question answering.

Recent advances in multimodal open source small models such as gemma3, llava, mini-cpm-v present a promising opportunity which combines vision and language capabilities in learning assistants. Our proposed agent uses a multimodal small language model for image-based question answering in one of its tools. The Agentic RAG technique enhances the agent's ability to understand and respond to user queries with high contextual accuracy by invoking the dedicated tools for specific tasks (DeepLearning.AI, 2024a).

2.3 Contribution Beyond Existing Work

While prior studies have largely focused on cloud-based, text-only assistants, our work emphasizes local deployment of the proposed Learning Agent by using open-source small language models and a multimodal model. This ensures that sensitive user data remains on personal devices, thereby enhancing privacy, security and affordability. Moreover, unlike existing literature, we share detailed guidelines and instructions for implementing the proposed personal learning agent, which are not available to the best of our knowledge.

From a pedagogical point of view, our proposed agent enables learner-driven exploration of the educational resources, generates personalized explanations, quizzes, and other learning aids. It provides feedback in natural language, resulting in improved learner engagement and interest in the subject matter. These aspects align with the well-known ideas of personalized learning that encourage learners to take charge of their own learning and actively participate in the learning process (Reeve and Tseng, 2011) and that providing formative feedback can help enhance learning outcomes (Shute, 2007).

3. System Design

The design of the proposed AI-powered personal learning agent is based on an Agentic AI technique, namely Agentic Retrieval Augmented Generation. Agentic AI refers to artificial intelligence systems that can autonomously perform tasks, make decisions, and solve complex problems without constant human intervention (Pounds, 2024). Agentic RAG incorporates agentic behaviour into the RAG-based AI systems by adding intelligence that can autonomously analyze queries, select the most effective tools for data retrieval, and refine responses (DeepLearning.AI, 2024a), which allows for more accurate and comprehensive answers.

3.1 Technical Building Blocks

Our previous work shared the detailed technology stack required to implement the learning assistant (Taneja et al., in press). All the tools and resources contained in this stack are open source. We have used one additional tool, Gradio, for chatbot interface development for our agent. To summarize, we have used the following open-source tools/packages/frameworks for our design:

- **Small language model**, gemma 2:2b (Ollama, 2024a), for question answering and natural language interaction with the learner (Team et al., 2024).
- **Small Multimodal models,** Llava (Ollama, 2024b) or minicpm-v (Ollama, 2024c)for visual question answering.
- **Ollama** (Ollama, 2025) for hosting these small models on a local machine.
- **LlamaIndex** (LlamaIndex, 2025g), for Agentic RAG implementation.
- **Gradio** for chatbot interface and chat history implementation.

3.1.1 LlamaIndex components

From LlamaIndex, we have primarily used the following components: Index (Sequential, Vector Store, and Multimodal Vector Store), Query Engine, Response Synthesizer, and Router Query Engine. A brief description of these components is discussed below and is summarized in Figure 1.

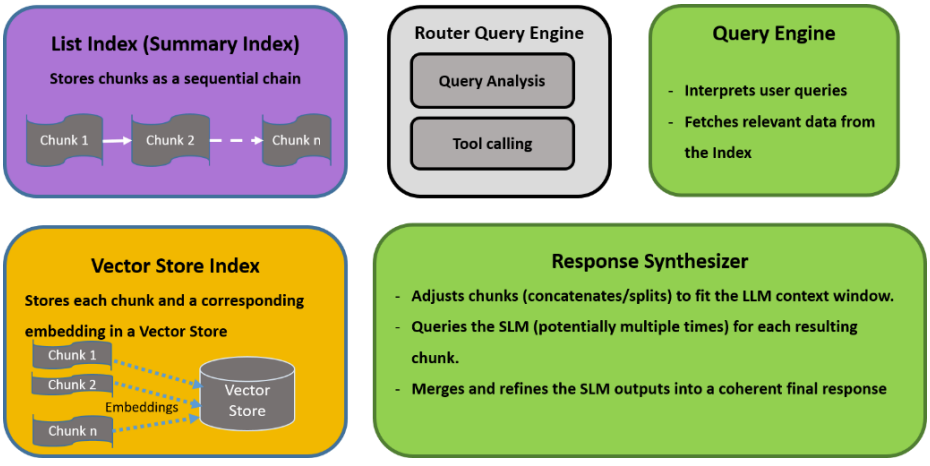

Figure 1: LlamaIndex Components for Agentic RAG

- **List Index / Summary Index** organizes chunks in a sequential list format, making it easy to manage and retrieve chunks (LlamaIndex, 2024e). In the LlamaIndex library, what was earlier referred to as a List Index is now called a Summary Index. For clarity, we continue to use the term List Index in this paper because it better reflects its sequential structure. In the proposed agent, List Index is used when a learner requests the summary of OER or other learning-aids, like flash cards and quizzes. Unlike the Vector Store Index which focuses on retrieving semantically similar chunks, List Index is helpful in cases when there is a need to access the entire content for a holistic view of the material**.**
- **Vector Store Index** converts text chunks into vector embeddings, which enables semantic search and supports RAG (LlamaIndex, 2024f). This mechanisms helps the agent to generate contextually relevant answers to learners' queries. This helps in enhancing the quality of interactions between the learner and the AI agent (Taneja et al., in press). **Multimodal Vector Store Index** is a special type of Vector Store Index that uses multimodal embeddings such as CLIP to represent both images and text. In this paper, we discuss the multimodal vector store index in one of the approaches for visual question answering.
- The **Query Engine** is the interface through which a user query is processed. It interprets the input query, retrieves relevant information from an Index, and then passes the user's query and retrieved data to the Response Synthesizer for generating a coherent response (LlamaIndex, 2024b). Thus, it acts as the coordinator that connects the user's query to the Index and the Response Synthesizer.
- The **Response Synthesizer** in LlamaIndex is a component responsible for transforming the retrieved data into a clear, human-readable final response (LlamaIndex, 2024d). It uses Large/Small Language Models to synthesize information, ensuring that the final output is contextually relevant, complete, and concise. There are three main response modes for the synthesizer that we have used in our implementation-Refine, Compact, and Tree Summarize (LlamaIndex, 2024c). In the **Refine** mode, the synthesizer sequentially processes the chunks, beginning with an initial response produced from the first chunk. It makes a separate SLM call per chunk. The previous answer and the next chunk are integrated with the original question to be used in the next query iteratively, until all chunks are used. In contrast, the **Compact** mode focuses on efficiency by reducing the SLM calls, concatenating as many chunks as possible into a single input prompt such that the total text fits within the context window. Our agent uses Compact mode for generating flashcards and quizzes and context-specific Question Answering. The **Tree Summarize** mode takes a hierarchical approach, where chunks are recursively summarized and merged. Summaries are generated at each stage, combined into larger summaries, and refined until a single, coherent response emerges. This method is used by our agent to create summaries.
- The **Router Query Engine** is a specialized Query Engine that interprets and analyzes users' queries and routes them to the appropriate tools by identifying which tool is best suited to handle the user query(Zhang et al., 2020). It allows for modular and efficient query processing by integrating multiple independent tools in a single application, which are specialized in their tasks.

3.1.2 Front-end chatbot interface

To develop a web-based front-end interface, we have used Gradio (DeepLearning.AI, 2024b), which is an open-source Python package to build the chatbot interfaces. This interface allows learners to upload any OERs that they wish to learn from. It also maintains chat history and allows learners to ask follow-up questions based on the previous chat. A screenshot of the chatbot interface of the learning agent is shown in Figure 2.

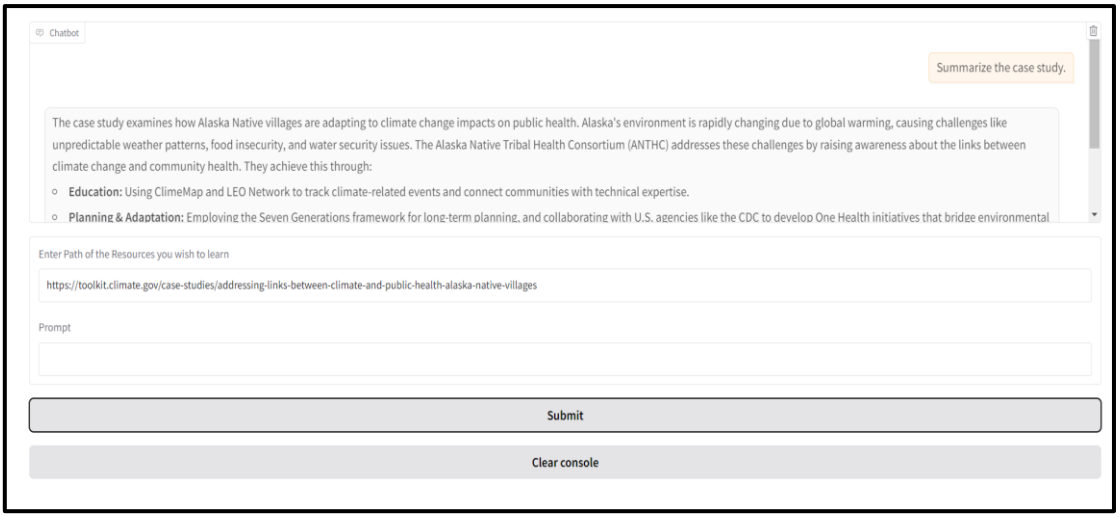

Figure 2: Chatbot Interface built using Gradio

3.2 Educational Rationale Behind Design Choices

The design of our learning agent is based on the educational theory and instructional design practices. We have included features such as flashcards, quizzes, and summaries because they are recognized to encourage active recall and improve comprehension (Dunlosky et al., 2013; Putnam, Sungkhasettee and Roediger, 2016). Moreover, multimodal interaction (text and images) makes the system more inclusive and engaging as learners can work with both textual and visual educational resources they typically encounter in classrooms and digital learning environments.

4. Development Methodology

The methodology involves initial data preparation (chunking and indexing) at the time of data loading by learners, followed by Agentic Retrieval-Augmented Generation process while question-answering and generating learning aids.

4.1 Data Preparation

The OER documents that the learner wishes to learn are chunked into smaller segments to facilitate processing. This approach, known as **Chunking**, is very important due to the context window limitations of language models, which can only process a finite number of tokens at a time. By creating smaller, more coherent chunks, the system ensures that each segment fits within the model's context window, thereby enabling more effective and accurate responses. After Chunking, the next step is **Indexing**, a method of organizing data in a way that makes searching and retrieving information faster and more efficient. LlamaIndex offers several types of indexes (LlamaIndex, 2024a) to be used in different use cases, as discussed in section 3.1.1. Figure 3 demonstrates the process of Chunking and Indexing.

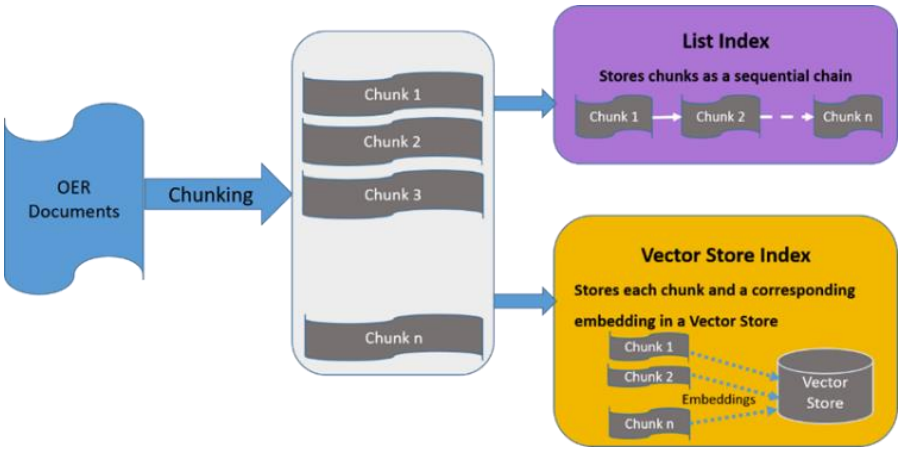

Figure 3: Indexing Techniques

4.2 Agentic RAG Process

During the learning time, the Agentic RAG approach is used by our agent to respond to learners' queries. Learner's query is passed to a Router Query Engine, which analyses it and, based on the intent of the query, an appropriate tool is invoked. There is an exclusive tool for each type of learner's request. Each tool has its dedicated query engine that works along with the Index and Response synthesizer to create the final response. This design is presented in Figure 4.

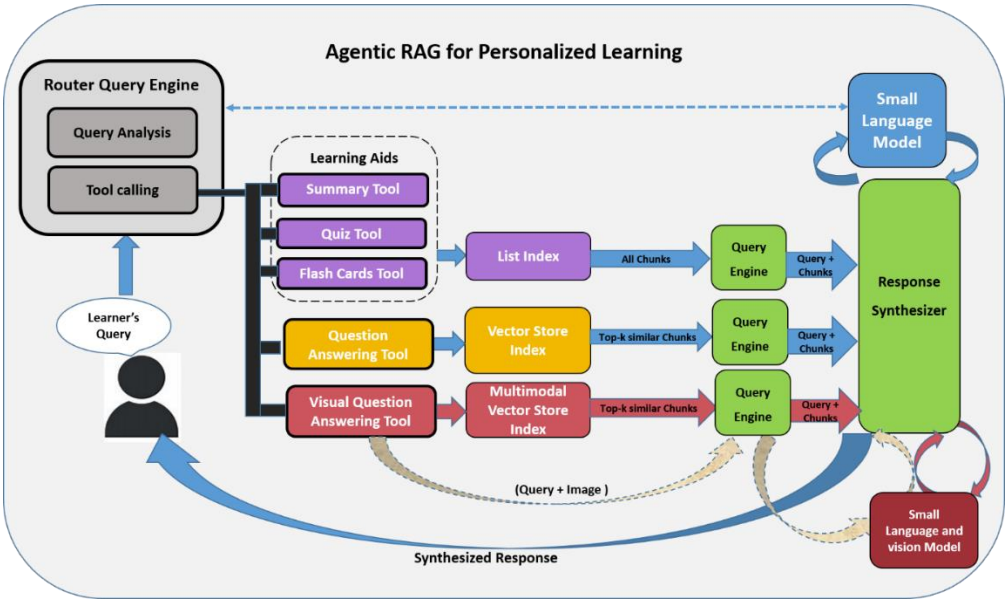

Figure 4: Agentic RAG Design

If the learner is interested in obtaining a summary of the OER, the Summary tool is invoked by the Router. When a learner asks for flashcards or a quiz, the agent are activates the corresponding tool. For content-specific questions, Question Answering tool is used while image-based questions are routed to the Visual Question Answering tool by the Router Query Engine. A brief description of these tools is given below:

- **Tools for Learning aids (Summary Tool, Flashcards Tool, Quiz Tool):** Whenever the learner requests the agent for learning aids, it uses one of these tools. These learning aids are recognized in educational practice for their role in helping learners understand, practice, and retain key ideas. Summary condense the main points of an educational resource so that learners can revisit important concepts without rereading the entire content. Flashcards convert complex topics into smaller units, making them easier to remember. Quizzes foster active engagement and help learners in strengthening their understanding of the topic (Dunlosky et al., 2013; Putnam, Sungkhasettee and Roediger, 2016). Collectively, these tools allow the agent to provide structured support to the learners across different stages of learning. The process that our Learning agent undertakes to generate these learning aids is demonstrated in Figure 5. The learner's query is passed to the Router Query Engine, which analyzes it and then calls the appropriate tool for learning by forwarding the query to the Query Engine. The Query Engine pulls the sequentially indexed chunks from the List Index. All learning-aid tools rely on the List Index and Response Synthesizer described in Section 3.1.1

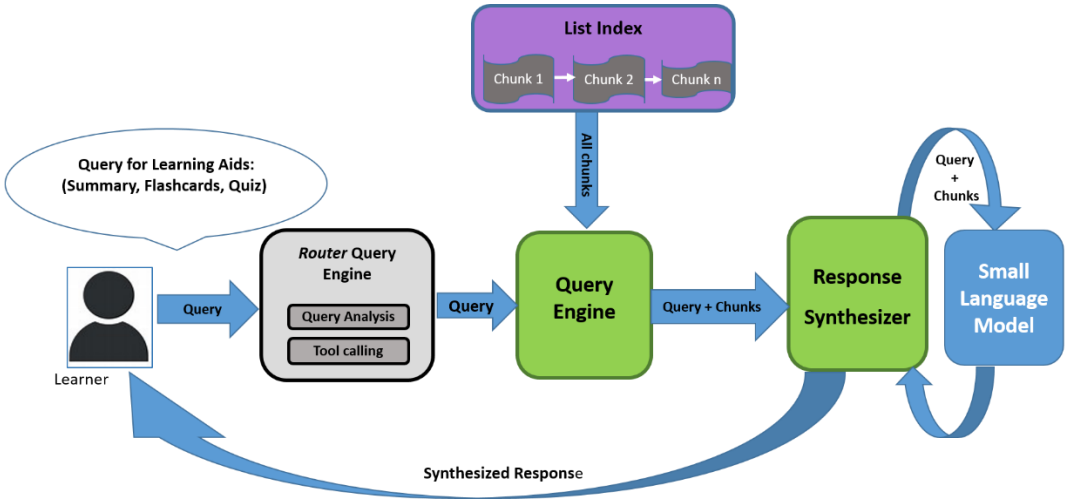

Figure 5: Tools for Learning Aids

- **Question Answering Tool (RAG Tool):** Figure 6 illustrates the RAG process used by the Question Answering Tool. Vector Store Index stores embeddings of all the chunks in a vector store. When a learner asks a query, it is passed to the Router Query Engine, which then chooses to invoke the Question-Answering Tool. The Router Query Engine forwards the query to the Query Engine. The top-k chunks that are most similar to the learner's query are retrieved by the Query Engine. These similar chunks, along with the Query, are passed on to the Response synthesizer, which adjusts the chunks and passes them to the SLM along with the query and returns the synthesized response to the learner.

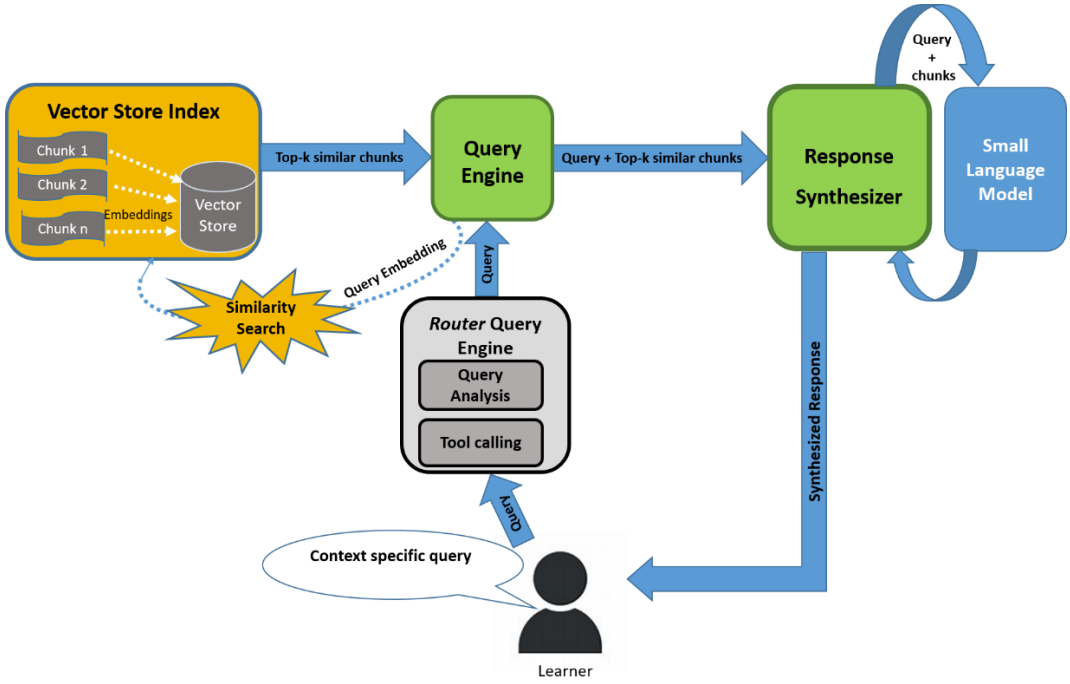

Figure 6: Question-Answering Tool (RAG-Tool)

- **Visual Question Answering Tool (Image Tool):** We experimented with two different methods for building the Visual question answering tool. In the first method, process is very similar to text-based question answering. We used a multimodal embedding model (CLIP) to create a multimodal vector index for images and text. When learner submitted a query, is was also embedded with the same model and most relevant chunks were retrieved from the vector space. These were then combined with the query and passed to a multimodal SLM such as Llava, which also uses CLIP embedding. Though this multimodal RAG approach functioned as expected, but the quality of responses were inferior as compared to our second approach. In the second approach, we developed a Custom Query Engine in place of the default one provided by LlamaIndex. Normally, creating a Query engine requires

building an index, but we bypassed this requirement by using the Custom Query Engine. Instead, we directly passed the learner's query and the associated image to a more efficient small multimodal model, minicpm-v. The model's output was wrapped into a LlamaIndex response object so that it could be integrated smoothly with rest of our system. This approach generated better image-based responses and aligned well into our broader agentic design.

5. Results and Discussion

For RAG-based question answering, i.e., to answer context-specific queries of the learner, our previously proposed assistant demonstrated satisfactory performance. We have reported the results of RAG evaluation concerning Faithfulness, Answer relevance, and Context Relevance as well as human evaluation in our previous work. Hence, we decided to go ahead with our findings of the previous study and chose the Gemma 2 model with 2b parameters, with a chunk-size of 512 tokens, and integrate it in the question answering tool of our agent.

To evaluate the quality of summaries generated by the proposed agent, we obtained 4 OERs (case study/Readings) from different subject areas from OER Commons and summarized them using the Summary tool of our agent. A small set of four OERs was chosen to cover a diverse range of content types and structures, ensuring a focused and manageable study while still allowing us to observe the agent's behaviour across varied content. We generated the reference summaries of the same OERs using ChatGPT (GPT4o) and compared both using BERTScore. BERTScore (Zhang et al., 2020) is a metric for evaluating quality of text generation. It computes the cosine similarity between token embeddings of reference and candidate sentences using pre-trained BERT representations. It evaluates Precision, Recall, and F1 score, where Precision focuses on how much of the candidate's content is relevant compared to the reference. Recall focuses on how much of the reference's content is captured by the candidate. F1 Score, the harmonic mean of Precision and Recall, balances the two and measures the overall quality of the match. Table 1 shows the results of our experiment.

Table 1: BERTScore Evaluation

Subject Area	OER Details	BERTScore
Environmental Science	Addressing Links Between Climate and Public Health in Alaska Native Villages (https://toolkit.climate.gov/case-studies/addressing-links-between-climate-and-public-health-alaska-native-villages)	Precision: 0.88 Recall: 0.89 F1: 0.88
Business Communication	Adjusting for Inflation (https://www.stlouisfed.org/publications/page-one-economics/2023/01/03/adjusting-for-inflation)	Precision: 0.90 Recall: 0.89 F1: 0.89
Law	Equity vs. Equality (https://oercommons.org/courseware/lesson/97984/overview)	Precision: 0.90 Recall: 0.91 F1: 0.90
Life Sciences	Anatomy and Physiology of the Respiratory System (https://oercommons.org/authoring/26964-1-1-anatomy-and-physiology-of-respiratory-system/view)	Precision: 0.85 Recall: 0.84 F1: 0.84

As seen in Table 1, the Bert Score, F1 Score are 0.84 or more for all the OERs summaries generated by GPT as reference and summaries generated by our agent as candidate. This clearly shows that the assistant's performance is quite reasonable and similar in terms of semantic similarity and relevance to that of the state-of-the-art model like GPT4o, for summarizing the content. From educational point of view, this means that learners can rely on the summaries generated by our agent to quickly review the main ideas of a resource before diving into details.

To test the autonomous (tool selection) behaviour of the agent, we challenged it with a diverse set of questions drawn from each of the four OERs. Table 2 lists a representative sample of twenty-five questions. We selected 25 queries to reflect common learning tasks such as summarization, retrieval, quiz generation, and flashcard creation. This number was appropriate for an exploratory study, allowing us to examine the system's behaviour across different tasks and content types without making the evaluation unwieldy. The set was designed so that each question naturally aligned with a different tool- Summary Tool (ST), RAG Tool (RT), Quiz Tool (QT), and

Flashcard Tool (FT). We have included visual question answering-based questions, which are expected to invoke Image Tool (IT), from the Environmental Sciences and Life Sciences OER, as these resources contained multiple images from which meaningful questions could be asked.

To assess the agent's responses, we applied a 3-point scoring system. Each response was evaluated both on whether the correct tool was activated (verified through analysis of logs) and the correctness of the generated response. Each answer is rewarded points based on the following scoring scheme:

- *2 points* → Response is Satisfactory & Actual Tools used are the same as the Expected Tools.
- *1 point* → Response is Partially Satisfactory (in terms of completion, correctness, or relevance) or mismatch between the expected and the Actual Tool
- *0 point* → Unsatisfactory Response.

As can be seen from Table 2, for most of the questions (24 out of 25) from our sample dataset, the Actual Tools used were the same as the Expected Tools. The only case where there is a mismatch was Q5, where we expected the agent to call the RAG tool, followed by the quiz tool to generate the response. The Agent autonomously decided to call the Quiz tool directly and generated a response that was still reasonable. This instance shows that while the agent occasionally bypasses a tool (RAG) in favour of another (Quiz), it may still produce a pedagogically useful output. This may be positive in some learning contexts, though further investigation is needed to assess how such deviations effect learning outcomes. In the case of the Image Tool (IT), the agent performed well in retrieving visual information (Q6 - Q8, Q24 - Q25). However, the accuracy of interpretation of images depends heavily on image clarity, domain-specific labels, and alignment between text and image context. The outputs of other tools (flashcards, quizzes) produced simpler and structured learning aids that may be helpful to learners and support them in an educational environment where clarity and alignment with source content are essential. Therefore, we saw that the proposed prototype of the agent was working satisfactorily with the OERs and questions on which it was assessed.

These findings show that the proposed agent is technically viable and pedagogically significant. It answers context-specific text-based questions as well as questions based on image understanding. This implies that learners can engage with the content in multiple ways like reading, question-answering, summarizing, practicing, and testing their knowledge. When the learning material is very technical or image-heavy, the agent may miss small but important details which is a matter of investigation in future research. By addressing these areas, the system could further increase learner engagement and provide them stronger support.

Table 2: Agentic Behaviour Analysis

Q. No.	Queries from OERs	Expected Tool	Score
	OER- Addressing Links Between Climate and Public Health in Alaska Native Villages (**Link**: https://toolkit.climate.gov/case-studies/addressing-links-between-climate-and-public-health-alaska-native-villages**)**		
1	Summarize the case study.	ST	2
2	What are the climate change impacts on public health in Alaska Native Villages?	RT	2
3	Make flashcards for this case study.	FT	2
4	Create a multiple-choice questions-based assessment from this case study	QT	2
5	Create a short quiz on health adaptation strategies.	RT+QT	1 (only QT called)
6	Analyze the image to identify the primary challenges and explain their significance. (**Image Link**: https://toolkit.climate.gov/image/1108)	IT	2
7	Describe the key visual elements in the provided image. What do they convey about climate change? (**Image Link**: https://toolkit.climate.gov/image/1110)	IT	2
8	Does the image show temperature trends or anomalies? (**Image Link**: https://toolkit.climate.gov/image/1108)	IT	2

Q. No.	Queries from OERs	Expected Tool	Score
	OER- Adjusting for Inflation (**Link**: https://www.stlouisfed.org/publications/page-one-economics/2023/01/03/adjusting-for-inflation)		
9	Give a brief overview of the document in three to four sentences	ST	2
10	Why is it important to compare real values rather than nominal values when analyzing economic data over time?	RT	2
11	Can you turn this document into a set of flashcards for learning?	FT	2
12	Create a comprehensive quiz from the entire content of this document	QT	2
13	Please provide a summary of key concepts mentioned in this article, and also answer from the text: What does adjusting for inflation involve, and what term do economists use to describe dollar amounts that have been adjusted for inflation?	ST+RT	2
	OER- Equity Vs. Equality (**Link**: https://oercommons.org/courseware/lesson/97984/overview)		
14	Please summarize the concepts explained in this text.	ST	2
15	How can misunderstanding the concepts of equality and equity impact societal institutions?	RT	2
16	Create flashcards to review the concepts covered in this material	FT	2
17	Prepare a practice quiz to reinforce the learning from this document.	QT	2
18	Provide a quick summary suitable for a presentation slide, and create a quiz having 5 multiple-choice questions for assessing learners.	ST+QT	2
	OER- Anatomy and Physiology of the Respiratory System (**Link**: https://oercommons.org/authoring/26964-1-1-anatomy-and-physiology-of-respiratory-system/view)		
19	Provide a bullet-point summary of the key points from the given article	ST	2
20	How does air follow from the nasal cavity to the larynx during inhalation?	RT	2
21	Generate flashcards to help study the main points of this text.	FT	2
22	Generate a short 5-question quiz covering the key points of this text	QT	2
23	Give me a 100-word summary of this text and generate flashcards	ST+FT	2
24	Describe the image. (**Image Link:** https://img.oercommons.org/oercommons/media/editor/153219/8cd649589ce741b39f269c72837e7910.jpg)	IT	2
25	What are the labelled parts of the respiratory system in this image? (**Image Link**: https://img.oercommons.org/oercommons/media/editor/153219/47628ec67eb441df8f9a195aaf0d69e5.jpg)	IT	2

6. Conclusion

We presented the detailed design, methodology, and technical guidelines for designing an Agentic RAG-based multimodal agent, which is based on open-source software and hence is likely highly cost-effective compared to cloud-based systems. It is lightweight as it is powered by small language models and can run on a learner's local machine, ensuring privacy without the need for any special computation-intensive resources. Based on learners' request, the agent can generate summaries, flashcards, and quizzes from the OERs they provide and also allows context-aware interactions with those resources.

Our work adopts Agentic RAG architecture and sets out practical design & deployment guidelines for affordable, locally operated educational AI agents. Compared with cloud-based systems, this approach offers benefits in privacy, cost, and contextual relevance. That said, the proposed agent has its own limitations. While hallucinations are reduced to a large extent by using RAG, some responses may still be irrelevant and fabricated by the SLM, which is a known drawback of language models. The quality of responses may also vary for highly

technical or image-intensive OERs, where fine details or complex diagrams are harder to interpret. Another concern (shared with other SLMs and LLMs) is the occasional possible use of offensive, insensitive or inappropriate language, which poses ethical and safety concerns.

Our research supports the broader use of AI in education and learning by showing that learning agents can be build without requiring costly computational resources. It presents the detailed technical guidelines to develop pedagogically valuable AI agents and deploy them on local machines, opening the opportunities for future research in the field of AI in education.

7. Recommendations for Future Research

Looking ahead, several areas invite further research and exploration in this field. On the technical side, incorporating guardrails could ensure that output remain safe and pedagogically appropriate. Stronger domain-specific grounding can be achieved by integrating structured resources such as glossaries, ontologies, or knowledge graphs for technical subjects or complex visual reasoning. Extending multimodal capabilities to include video, or enabling voice interaction, could make the system more versatile. Another promising avenue is optimizing the agent by using efficient techniques and lighter models to allow it to run on low-resource mobile devices like smartphones as well.

From educational standpoint, future research could study how agents can be aligned closely with learning theories and instructional design principles. For example, quiz questions can be mapped to different levels of Bloom's Taxonomy. Additionally, agents could provide adaptive feedback and recommendations to learners by scoring quiz responses. Adding gamification features such as badges, progress milestones, and challenges might increase motivation. Finally, multi-agent designs could be explored in instructional design, where different agents handle different stages of developing learning experiences. For example, following the ADDIE model one agent could analyze the existing content based on the learning requirements and share its results, observations and ideas. Based on this agent's analysis second agent could design and develop the learning design; finally third agent could review and evaluate the results of previous agents and iteratively interact with them for improvements. These directions offer opportunities for technical improvements as well as learning enhancements to advance personalized and ethical AI applications in education and learning.

AI Statement: The authors declare that no AI tools were used in the conceptualization, preparation, interpretation or conclusions in this paper.

Ethics Statement: This research did not involve human participants, animal subjects, or any material that requires ethical approval.

References

Bozkurt, A., 2023. Generative artificial intelligence (AI) powered conversational educational agents: The inevitable paradigm shift. *Asian Journal of Distance Education*, [online] 18(1). Available at: <https://www.asianjde.com/ojs/index.php/AsianJDE/article/view/718> [Accessed 23 July 2024].

Chimezie, M.O., 2024. LEVERAGING RETRIEVAL-AUGMENTED GENERATION IN LARGE LANGUAGE MODELS FOR EFFECTIVE LEARNING: A DATA STRUCTURES & ALGORITHMS LEARNING ASSISTANT.

DeepLearning.AI, 2024a. *Building Agentic RAG with Llamaindex*. [online] DeepLearning.AI - Learning Platform. Available at: <https://learn.deeplearning.ai/courses/building-agentic-rag-with-llamaindex/lesson/1/introduction> [Accessed 29 December 2024].

DeepLearning.AI, 2024b. *Building Generative AI Applications with Gradio*. [online] DeepLearning.AI - Learning Platform. Available at: <https://learn.deeplearning.ai/courses/huggingface-gradio/lesson/1/introduction> [Accessed 29 December 2024].

Dunlosky, J., Rawson, K.A., Marsh, E.J., Nathan, M.J. and Willingham, D.T., 2013. What Works, What Doesn't. *Scientific American Mind*, 24(4), pp.46–53. https://doi.org/10.1038/scientificamericanmind0913-46.

GPU Mart, 2025. *How to Run LLMs Locally with Ollama AI*. [online] GPU Servers Mart. Available at: <https://www.gpu-mart.com/blog/run-llms-with-ollama> [Accessed 13 June 2025].

Holmes, W., Bialik, M. and Fadel, C., 2019. Artificial Intelligence In Education.

Liu, S., Yu, Z., Huang, F., Bulbulia, Y., Bergen, A. and Liut, M., 2024. Can Small Language Models With Retrieval-Augmented Generation Replace Large Language Models When Learning Computer Science? In: *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1*, ITiCSE 2024. [online] New York, NY, USA: Association for Computing Machinery. pp.388–393. https://doi.org/10.1145/3649217.3653554.

LlamaIndex, 2024a. *Index Guide*. [online] Available at: <https://docs.llamaindex.ai/en/stable/module_guides/indexing/index_guide/> [Accessed 13 December 2024].

LlamaIndex, 2024b. *Query Engine*. [online] Available at: <https://docs.llamaindex.ai/en/stable/module_guides/deploying/query_engine/?utm_source=chatgpt.com> [Accessed 30 December 2024].
LlamaIndex, 2024c. *Response Modes*. [online] Available at: <https://docs.llamaindex.ai/en/stable/module_guides/deploying/query_engine/response_modes/> [Accessed 13 December 2024].
LlamaIndex, 2024d. *Response Synthesizer*. [online] Available at: <https://docs.llamaindex.ai/en/stable/module_guides/querying/response_synthesizers/> [Accessed 30 December 2024].
LlamaIndex, 2024e. *Summary*. [online] Available at: <https://docs.llamaindex.ai/en/stable/api_reference/indices/summary/?form=MG0AV3> [Accessed 13 December 2024].
LlamaIndex, 2024f. *Vector Store Index*. [online] Available at: <https://docs.llamaindex.ai/en/stable/module_guides/indexing/vector_store_index/> [Accessed 30 December 2024].
LlamaIndex, 2025g. *Build Knowledge Assistants over your Enterprise Data*. [online] Available at: <https://www.llamaindex.ai/framework> [Accessed 2 January 2025].
Ollama, 2024a. *gemma2:2b*. [online] Available at: <https://ollama.com/gemma2:2b> [Accessed 2 January 2025].
Ollama, 2024b. *llava*. [online] Available at: <https://ollama.com/llava> [Accessed 2 January 2025].
Ollama, 2024c. *minicpm-v*. [online] Available at: <https://ollama.com/minicpm-v> [Accessed 2 January 2025].
Ollama, 2025. *Ollama*. [online] Available at: <https://ollama.com> [Accessed 2 January 2025].
Pounds, E., 2024. What Is Agentic AI? *NVIDIA Blog*. Available at: <https://blogs.nvidia.com/blog/what-is-agentic-ai/> [Accessed 12 December 2024].
Putnam, A.L., Sungkhasettee, V.W. and Roediger, H.L., 2016. Optimizing Learning in College: Tips From Cognitive Psychology. *Perspectives on Psychological Science*, 11(5), pp.652–660. https://doi.org/10.1177/1745691616645770.
Reeve, J. and Tseng, C.-M., 2011. Agency as a fourth aspect of students' engagement during learning activities. *Contemporary Educational Psychology*, 36(4), pp.257–267. https://doi.org/10.1016/j.cedpsych.2011.05.002.
Sajja, R., Sermet, Y., Cikmaz, M., Cwiertny, D. and Demir, I., 2024. Artificial Intelligence-Enabled Intelligent Assistant for Personalized and Adaptive Learning in Higher Education. *Information*, 15(10), p.596. https://doi.org/10.3390/info15100596.
Shute, V.J., 2007. Focus on Formative Feedback.
Singh, A., Ehtesham, A., Kumar, S. and Khoei, T.T., 2025. *Agentic Retrieval-Augmented Generation: A Survey on Agentic RAG*. https://doi.org/10.48550/arXiv.2501.09136.
Slade, J.J., Hyk, A. and Gurung, R.A.R., 2024. Transforming Learning: Assessing the Efficacy of a Retrieval-Augmented Generation System as a Tutor for Introductory Psychology. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 68(1), pp.1827–1830. https://doi.org/10.1177/10711813241275509.
Taneja, S., Biswas, S.S., Alankar, B. and Kaur, H., in press. Architecture and prototype of a lightweight AI-powered personal learning assistant using open source small language model and multi-modal retrieval augmented generation. *International Journal of Technology Enhanced Learning*. https://doi.org/10.1504/IJTEL.2025.10069095.
Team, G., Riviere, M., Pathak, S., Sessa, P.G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., Shahriari, B., Ramé, A., Ferret, J., Liu, P., Tafti, P., Friesen, A., Casbon, M., Ramos, S., Kumar, R., Lan, C.L., Jerome, S., Tsitsulin, A., Vieillard, N., Stanczyk, P., Girgin, S., Momchev, N., Hoffman, M., Thakoor, S., Grill, J.-B., Neyshabur, B., Bachem, O., Walton, A., Severyn, A., Parrish, A., Ahmad, A., Hutchison, A., Abdagic, A., Carl, A., Shen, A., Brock, A., Coenen, A., Laforge, A., Paterson, A., Bastian, B., Piot, B., Wu, B., Royal, B., Chen, C., Kumar, C., Perry, C., Welty, C., Choquette-Choo, C.A., Sinopalnikov, D., Weinberger, D., Vijaykumar, D., Rogozińska, D., Herbison, D., Bandy, E., Wang, E., Noland, E., Moreira, E., Senter, E., Eltyshev, E., Visin, F., Rasskin, G., Wei, G., Cameron, G., Martins, G., Hashemi, H., Klimczak-Plucińska, H., Batra, H., Dhand, H., Nardini, I., Mein, J., Zhou, J., Svensson, J., Stanway, J., Chan, J., Zhou, J.P., Carrasqueira, J., Iljazi, J., Becker, J., Fernandez, J., Amersfoort, J. van, Gordon, J., Lipschultz, J., Newlan, J., Ji, J., Mohamed, K., Badola, K., Black, K., Millican, K., McDonell, K., Nguyen, K., Sodhia, K., Greene, K., Sjoesund, L.L., Usui, L., Sifre, L., Heuermann, L., Lago, L., McNealus, L., Soares, L.B., Kilpatrick, L., Dixon, L., Martins, L., Reid, M., Singh, M., Iverson, M., Görner, M., Velloso, M., Wirth, M., Davidow, M., Miller, M., Rahtz, M., Watson, M., Risdal, M., Kazemi, M., Moynihan, M., Zhang, M., Kahng, M., Park, M., Rahman, M., Khatwani, M., Dao, N., Bardoliwalla, N., Devanathan, N., Dumai, N., Chauhan, N., Wahltinez, O., Botarda, P., Barnes, P., Barham, P., Michel, P., Jin, P., Georgiev, P., Culliton, P., Kuppala, P., Comanescu, R., Merhej, R., Jana, R., Rokni, R.A., Agarwal, R., Mullins, R., Saadat, S., Carthy, S.M., Cogan, S., Perrin, S., Arnold, S.M.R., Krause, S., Dai, S., Garg, S., Sheth, S., Ronstrom, S., Chan, S., Jordan, T., Yu, T., Eccles, T., Hennigan, T., Kocisky, T., Doshi, T., Jain, V., Yadav, V., Meshram, V., Dharmadhikari, V., Barkley, W., Wei, W., Ye, W., Han, W., Kwon, W., Xu, X., Shen, Z., Gong, Z., Wei, Z., Cotruta, V., Kirk, P., Rao, A., Giang, M., Peran, L., Warkentin, T., Collins, E., Barral, J., Ghahramani, Z., Hadsell, R., Sculley, D., Banks, J., Dragan, A., Petrov, S., Vinyals, O., Dean, J., Hassabis, D., Kavukcuoglu, K., Farabet, C., Buchatskaya, E., Borgeaud, S., Fiedel, N., Joulin, A., Kenealy, K., Dadashi, R. and Andreev, A., 2024. *Gemma 2: Improving Open Language Models at a Practical Size*. https://doi.org/10.48550/arXiv.2408.00118.
Watters, A., 2023. *Teaching Machines: The History of Personalized Learning*. MIT Press.
Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q. and Artzi, Y., 2020. *BERTScore: Evaluating Text Generation with BERT*. https://doi.org/10.48550/arXiv.1904.09675.

