applied sciences

MDPI

Article

AI-Powered System to Facilitate Personalized Adaptive Learning in Digital Transformation

Yao Yao * and Horacio González–Vélez *

National College of Ireland, D01 N6P6 Dublin, Ireland
* Correspondence: yao.yao@ncirl.ie (Y.Y.); horacio@ncirl.ie (H.G.-V.)

Featured Application: The proposed framework can be integrated into learning platforms to enhance personalized adaptive learning. By leveraging knowledge-driven agents and RAG pipelines, this framework improves the accuracy and effectiveness of AI assistants, while expanding their capabilities through the incorporation of customized knowledge. Continuous updates to the knowledge base enable AI models to dynamically adapt to individual learners, delivering context-aware and precise responses tailored to their needs. This approach is particularly valuable for integrated interdisciplinary learning such as digital transformation, where multidisciplinary knowledge integration plays a crucial role in fostering deeper understanding and knowledge retention.

Abstract: As Large Language Models (LLMs) incorporate generative Artificial Intelligence (AI) and complex machine learning algorithms, they have proven to be highly effective in assisting human users with complex professional tasks through natural language interaction. However, in addition to their current capabilities, LLMs occasionally generate responses that contain factual inaccuracies, stemming from their dependence on the parametric knowledge they encapsulate. To avoid such inaccuracies, also known as hallucinations, people use domain-specific knowledge (expertise) to support LLMs in the corresponding task, but the necessary knowledge engineering process usually requires considerable manual effort from experts. In this paper, we developed an approach to leverage the collective strengths of multiple agents to automatically facilitate the knowledge engineering process and then use the learned knowledge and Retrieval Augmented Generation (RAG) pipelines to optimize the performance of LLMs in domain-specific tasks. Through this approach, we effectively build AI assistants based on particular customized knowledge to help students better carry out personalized adaptive learning in digital transformation. Our initial tests demonstrated that integrating a Knowledge Graph (KG) within a RAG framework significantly improved the quality of domain-specific outputs generated by the LLMs. The results also revealed performance fluctuations for LLMs across varying contexts, underscoring the critical need for domain-specific knowledge support to enhance AI-driven adaptive learning systems.

Keywords: large language models; personalized adaptive learning; retrieval augmented generation; multi-agent system; digital transformation

Academic Editor: Jose María Alvarez Rodríguez

Received: 4 April 2025
Revised: 24 April 2025
Accepted: 28 April 2025
Published: 30 April 2025

Citation: Yao, Y.; González–Vélez, H. AI-Powered System to Facilitate Personalized Adaptive Learning in Digital Transformation. *Appl. Sci.* **2025**, *15*, 4989. https://doi.org/10.3390/app15094989

1. Introduction

Through successful use cases in diverse domains, Large Language Models (LLMs) have been recognized as the Natural Language Processing (NLP) technology of choice. Language Models (LMs) leverage advanced deep learning architectures and exhibit unparalleled proficiency in understanding, generating, and manipulating human-like text,

fundamentally reshaping the landscape of artificial intelligence. The widespread adoption of LLMs is a testament to their versatility and efficacy. Despite their remarkable achievements, LLMs have also faced some challenges in their deployment. Issues such as bias, hallucination, and domain-based knowledge challenges for specific use cases have surfaced, prompting a reevaluation of their role in shaping the future of artificial intelligence [1]. Moreover, according to recent research by Uchida, S. [2], even advanced LLMs also have limitations in memorizing complete knowledge of ontology in domain-specific contexts. To extend the advantages and resolve the problems of LLMs, people use external knowledge to support their performance. For example, using the Retrieval Augmented Generation (RAG) method [3], we can improve the performance of many LLMs in specific tasks. RAG efficiently improves the output of LLMs based on the particular context, while sufficient knowledge support and an integrated knowledge base are required in addition to the deployment of LLMs.

Traditional approaches for constructing knowledge bases typically require significant human effort or computational resources to process relevant textual documents. These approaches utilize knowledge engineering techniques to extract and organize concepts for the construction of knowledge bases. However, these conventional methods exhibit two main deficiencies when it comes to building practical knowledge bases based on domain-specific expertise. First, identifying domain-specific knowledge and accurate contextual matching can pose significant challenges [4], and relevant knowledge can exceed the coverage of textual documents [5]. This requires input from domain experts in specific scenarios. Secondly, in many contexts, practicality demands that domain-specific knowledge be dynamically adjusted and updated according to specific task requirements [6]. This requires domain experts to continually identify knowledge within the knowledge base and promptly update the corresponding knowledge based on new circumstances. As can be seen from the above two points, the intensive and constant participation of corresponding experts is a key element in establishing a domain-specific knowledge base, which makes the interaction between experts and the knowledge engineering pipeline critical to the quality of knowledge. In this research, we propose using a multi-agent system supported by LLMs to help construct a domain-specific knowledge base and improve the quality of knowledge. Our approach provides an efficient way of updating and validating domain-specific knowledge through the interaction between expert users and the system. By optimizing support knowledge, we enhance the performance of LLMs in domain-specific tasks, enabling Artificial Intelligence (AI) assistants to provide more accurate, context-sensitive responses tailored to the needs of each learner. This approach strengthens AI-driven personalized learning applications, ensuring more effective and adaptive educational support.

2. Related Work

2.1. The Deployment of LLMs in Domain-Specific Applications

Pre-trained LLMs are neural networks trained using deep learning and natural language processing techniques. These models are typically trained on large amounts of textual data, such as previous language corpora, which may contain large amounts of textual data from websites, books, and other textual sources. Pre-trained LLMs trained on comprehensive data have demonstrated impressive capabilities to comprehend natural language for tasks related to structural prediction and reasoning [7].

However, based on previous applications in various domains, people have also found that LLMs are constrained by their training data in practical use [8]. The output LLMs produce is highly dependent on the training corpus. As a result, their inherent knowledge cannot cover topics not included in previous training data. If there are insufficient training

data or biased opinions on the corresponding topic, this usually affects the quality of the model output and even causes serious hallucinations of AI [9]. In addition, LLMs also need to be further improved in focusing on specific levels of concrete knowledge and a precise understanding of context [10]. To provide answers to ordinary or abstract questions, the impact of the above issues may not be obvious, but in special and highly professional applications, these mentioned problems actually limit the efficacy and practicality of LLMs and become an inevitable challenge for deploying LLMs.

To address this challenge, researchers have developed domain-specified LLMs to serve particular uses in a certain domain [11,12]. Domain-specified LLMs refer to language models that are fine-tuned or specialized for a particular domain or field of knowledge. Based on the pre-trained LLM, researchers use specific training datasets that include corresponding domain-specific knowledge to further train the model and make the model adapt to the understanding of particular domains. Many domain-specified LLMs have been developed in different domains, and we list some typical examples as follows:

- E-commerce: EcomGPT [13]
- Finance domain: BloombergGPT [14]
- Legal domain: LawGPT [15]
- Biomedical domain: BioGPT [16], BioMedLM [17]
- Geoscience: K2 [18], Oceangpt [19]

In the previous examples listed, domain-specified LLMs offer several advantages, including improved performance and accuracy in tasks relevant to the specific domain, as well as the ability to generate text that aligns more closely with the expectations and requirements of users within that domain. They are particularly useful in applications such as information retrieval, content generation, question answering, and language translation within specialized fields. However, domain-specified LLMs require additional fine-tuning work on relevant new training data, and this increases the initial cost of LLM deployment [20]. Moreover, the context and knowledge of domain-specified LLMs are embedded in the fine-tuned model, and this may limit the explainability of the model to users [21]. Consequently, the additional training cost makes it difficult for users to dynamically update their expertise under a given context. For scenarios that require intensive interaction with users and updating knowledge based on user needs, such as customized data searching, education, and consulting, people have increasingly started to accept RAG as a supplement and alternative to fine-tuning LLMs. Relevant examples can be seen in previous research on general conversation [22], education [23,24], research [25,26], and business consulting [27,28].

2.2. Structured Knowledge Base with RAG

As discussed above, fine-tuning LLMs works well with certain predefined tasks, but the complexity of training limits the extensibility and explainability of the models. For better and more adaptive performance in tasks that are highly dependent on the user context and require integrated knowledge from multiple domains, alternative methods such as RAG are sometimes recommended. RAG is a NLP approach that combines elements of both retrieval and generation models to improve performance in various language understanding and generation tasks. An RAG method can provide additional context to the LLM based on the prompt that has been passed to it [29]. This additional context can be retrieved from various sources, such as online datasets, vector databases, and the knowledge base. RAG provides the ability to use external and customized knowledge to improve the performance of LLMs without additional fine-tuning training. Based on previous research [30,31], it can increase accuracy and avoid hallucination problems based on the knowledge obtained. However, the quality of the output generated is highly dependent on the quality of the

retrieved knowledge. Poor retrieval results can lead to suboptimal generation performance. In addition, if the retrieval knowledge is biased or limited in scope, the generation model may produce biased or incomplete responses. This makes the efficiency of RAG highly dependent on the corresponding knowledge engineering process and the knowledge base behind it. To improve the performance of LLMs and make the RAG loop run efficiently, we need to enhance the efficiency and accuracy of the knowledge retrieval and engineering process in RAG methods by providing structured, semantically rich, and context-aware knowledge.

Knowledge Graphs (KGs) are an efficient method of representing knowledge and have been widely used to support AI models in many different domains [32]. A KG formally represents semantic models by describing the relevant entities and their relationships. KGs may make use of ontologies as a schema layer. In doing this, they can represent the complicated structure of knowledge with hierarchical concepts [33]. This feature makes KGs an ideal platform to contain and present a user's domain-specific knowledge in various scenarios.

Furthermore, graphic representation of the ontologies also gives KGs a unique advantage in supporting the reasoning inference of AI models. The potential fusion of KGs with cutting-edge AI techniques holds promise for fostering diverse and efficient applications within the data analysis and management domains. Last but not least, KGs give us a good platform to integrate and explain new knowledge graphically, in a structured manner. With this advantage, we could easily integrate customized knowledge or context into the concepts in a given knowledge base or elaborate an explanation of the particular topic to users. More recently, we have seen new applications, such as GraphRAG [34], that use structured and graphic knowledge to support RAG in applications based on LLMs, improving the performance of LLMs in many sophisticated use cases. However, there have also been many pieces of research that reported the use of LLMs in the corresponding knowledge engineering process [35,36]. In the research presented in this paper, we tried to integrate both types of approach into one framework and leverage the advantages of each side in knowledge-driven applications.

2.3. Knowledge-Driven AI Agents for Helping Personalized Adaptive Learning

Personalized adaptive learning refers to an educational approach that uses technology and data-driven insights to tailor learning experiences to the unique needs, preferences, and abilities of individual learners [37]. Unlike traditional "one-size-fits-all" methodologies, adaptive personalized learning adjusts dynamically to the pace, level of knowledge, and learning style of the student, ensuring that instruction is relevant and effective [38]. The goal of personalized adaptive learning is to optimize educational outcomes by recognizing and accommodating the diversity of learners, thus fostering deeper understanding and long-term retention.

This approach needs to integrate advanced technologies such as artificial intelligence (AI), machine learning (ML), and real-time analytics to assess a learner's progress and adjust the content or schedules accordingly. Throughout the entire process, providing students with personalized real-time arrangements is the key to the successful implementation of personalized adaptive learning, which requires that the AI system has accurate and relevant knowledge support, and is able to flexibly handle differences between different students in context. In previous research by [39], a successful example was demonstrated in which large language models (LLMs) were utilized as chatbots to deliver personalized adaptive learning features in a practical setting. In the research of Cho et al. [40], the researchers used a transformer-based model that achieved an Area Under the Curve (AUC) of 0.865 in knowledge tracing. Furthermore, in the research in [41], the importance of incorporating

specific knowledge from both students and teachers into the design of the learning process was emphasized, highlighting its potential to improve adaptability in learning. In addition, some of the latest research [42,43] has also shown us the great potential of robots and multi-agent in human–machine interaction and examples of effectively promoting learning. More quantitative results were reported by [44], where the research demonstrated that an AI-based tutor significantly improved post-test scores for underperforming math students, with those in the lowest pretest quartile gaining an average of 8.2 points, compared to 3.5 points in the control group, an improvement of over 130%. Another research study [45] used an AI model to generate personalized feedback, and the given model outperformed the baselines, resulting in a 45% improvement in student learning gains.

To integrate AI models and user-specific knowledge into the learning process, knowledge-driven agents could be a useful tool for the corresponding implementation. AI agents (empowered by LLMs) can play a crucial role in the implementation of personalized adaptive learning by using intensive human–machine interaction to tailor educational experiences to individual needs. With personalized domain-specific knowledge support, they can analyze learner behavior, preferences, and progress to provide customized content and support, fostering more effective and engaging learning outcomes. In addition, by adapting in real time and incorporating feedback from both students and teachers, AI agents improve adaptability, improve learning efficiency, and make education more accessible and inclusive. The latest explorations and attempts to combine agents with personalized adaptive learning have yielded some very successful cases, with especially promising results in adapting to customized knowledge to providing personalized services [46–48]. In our approach, particular AI agents retrieve the extra relevant knowledge (i.e., from domain-specific knowledge bases) based on the user's requests and use this specific knowledge to improve the performance of LLMs in each task. With this, the system can not only generate personalized responses based on context but also ensure the accuracy of the contents by providing domain-specific and context-related knowledge.

2.4. The Impact of Our Approach on Personalized Adaptive Learning in Digital Transformation

The digital transformation in education is not only a response to technological advancements but also a necessity to address the diverse needs of students from various backgrounds. Traditional one-size-fits-all education models often fail to accommodate learners in digital transformation contexts with different cultural contexts, learning styles, paces, and abilities [49,50]. Personalized adaptive learning fits the essential requests in digital transformation education well and can increase the adaptability of studies by providing customized learning processes and contents [51]. Using technology, educators can create more inclusive learning environments that provide equitable access to quality education for all students, including those who are underserved or face unique challenges. Adaptive technologies can support learners regarding interdisciplinary knowledge domains, language barriers, or varying levels of skills, ensuring that everyone can progress at their own pace [52]. In addition, digital transformation equips students with the skills necessary for success in the modern workforce, such as digital literacy, critical thinking, and problem solving. It also prepares them to adapt to continuous changes in technology and promotes lifelong learning. This customized learning method not only enhances academic outcomes, but also empowers learners, building confidence and resilience in a rapidly evolving world.

In fact, the value of adaptive learning is seen as multidimensional, being evaluated not just through academic achievement but also through its adaptability, student perception, and support for diverse learning needs. Researchers interpret and measure the value of adaptive learning through various lenses, including learning outcomes, user perception, engagement, and longitudinal performance data. For example, Sun et al. [53] conducted an

empirical study using the LearnSmart platform, finding that students perceived greater value and effectiveness in adaptive systems compared to traditional ones. These perceptions were measured through online surveys that analyzed the effectiveness of self-reported learning. A broader systematic review by Martin et al. [54] synthesized adaptive learning research from 2009 to 2018, identifying key themes such as instructional strategy integration, learner analytics, and scaffolding as vital factors in the evaluation of adaptive learning. More recent studies such as [55] used longitudinal data to analyze the impact of adaptive learning on student engagement and performance in online education. The results supported the efficacy of adaptive learning systems, showing improved retention and personalized timing as the main contributors to better learning outcomes. In this study, we focus on the use of AI to provide personalized learning suggestions and integration of customized strategies to help with the implementation of adaptive learning. In terms of metrics for evaluating the quality of the output of the AI model, we refer to various previous methods for evaluating adaptive learning, which will be discussed in Section 4.

To embrace the digital transformation and implement adaptive personalized learning today, we need the support of new techniques and approaches, to develop a suitable learning system for users. As discussed above, traditional static knowledge structures and rigid, formulaic learning models no longer meet the personalized learning needs of today's digital era. In the realm of knowledge engineering and AI, the traditional approach to knowledge extraction is being reshaped by the advent of LLMs. These cutting-edge artificial intelligence systems have demonstrated exceptional proficiency in understanding, generating, and structuring large amounts of textual data. However, knowledge integration in sophisticated and professional domains, such as high-level education, requires constant updating of the corresponding knowledge under a specific context, as well as understanding of the new data with domain-specific knowledge. This ability is still a challenge for LLMs and can caused a number of relevant issues, such as hallucinations, data credibility, and bias in LLM applications [8,56].

In this research, our goal was to combine the cutting-edge techniques discussed to forge an AI-based system that can effectively help students with personalized adaptive learning adapt to the digital transformation, and to mitigate the issues mentioned above in the use of LLM. By extracting the context and knowledge from the user's input, profile files, and previous conversations, the system can interactively construct a customized knowledge model under the domain-specific context and use this knowledge to better improve the performance of AI models. This exploration aims to contribute to the refinement of knowledge engineering and enrich the application of LLMs, paving the way for a more advanced and effective deployment of AI models in practical learning applications.

3. Materials and Methods

The approach used in this study was to develop an AI-based framework and use the framework to help personalize adaptive learning. The developed framework includes a multi-agent system and a customized knowledge base to provide domain-specific knowledge. In the agents of the system, we embedded LLMs to help the system better process text-based data and interact with users for the delivery of customized knowledge engineering and learning services. With the RAG pipelines implemented by agents, the LLMs can provide better recommendations and responses based on the context and user's request.

In our approach, there is an initial KG that works as a local knowledge base to contain the customized knowledge model that is defined by expert users or extracted from the given training documents. The knowledge model is made up of multiple relevant concepts and each concept has its unique description, attributes, and keywords. Expert users can define the content of the concepts directly or select the corresponding prompt templates to retrieve

the content from the designated datasets and then review them on the KG. After validation with the users, newly updated content will be included in the KG concepts. The knowledge of the KG remains dynamically updated and can be applied to the corresponding RAG method to improve the performance of LLMs. The whole pipeline can be divided into two steps: the corresponding multi-agent's knowledge engineering, and RAG-based response generation. Figure 1 shows an outline of the proposed pipeline for our framework.

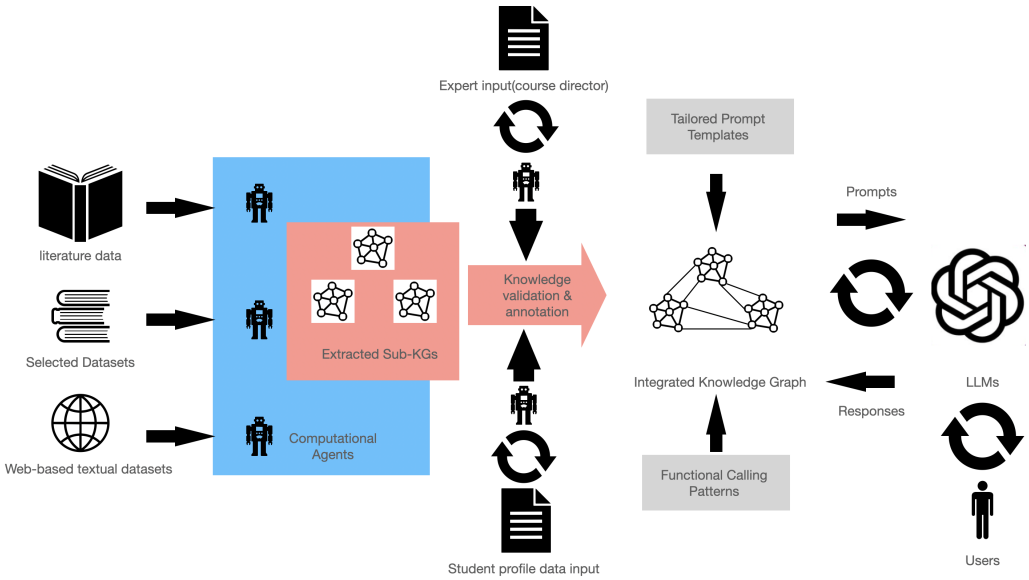


Figure 1. The outline of the framework with multi-agent and RAG pipelines.

3.1. Knowledge Engineering Facilitated by Context Awareness Agents Empowered with LLMs

Efficient extraction of relevant knowledge based on the given context is one of the main goals our approach aims to achieve. In this section, we focus on how to use multi-agent to better extract the context of a given task process and use it to serve knowledge engineering in a system.

The context of given tasks comes from the original user requests. The system extracts the keywords (meaningful nouns and verbs) from the text input and then uses agents to explain the semantic meaning of these words. Agents request diverse Language Models (LMs) that output the ontologically grounded representations of meaning based on the given natural language words and then search the matchable items in the semantic lexicon of the local knowledge base. To save computational resources, the agents are assigned a corresponding language model based on their task. In the research of Oruganti et al. [57], the authors discussed a sophisticated method of using agents and LLMs to process given textual seeds. In our case, we use pre-trained LMs to elaborate the semantic meaning of keywords in the user input and extract the relevant concepts to represent the context. The process is implemented with several agents, and each agent works with a given extracted keyword. The agent (a) checks the concepts (c) in knowledge and decides the relevance (R) of the concepts based on the semantic distance (S, calculated by the indexing model) between the given keyword ($K_{context}$) from the user input and the list of keywords (K_{ci}) of the concept. For each concept, this may include multiple (n) description labels.

$$R_{a,c} = \frac{\sum_{i=1}^{n} S(K_{context} - K_{c,i})}{n} \quad (1)$$

Based on the semantic distance, the agents will select the most relevant concepts and return them to the system. The threshold for selecting the most relevant concepts is given by the predefined knowledge in the knowledge base. The complete context of the given

request is a combination of the returns of all the agents and is represented as a list of concepts that are predefined in the semantic lexicon. The system will add the complete context as a prefix to the original user request. Users can also refer to textual documents in their request. To process a large document, the system will divide the data into different chunks and integrate the summary of the different chunks at the end. In general, the system analyzes the inherent context of the user input based on the given knowledge and identifies the related concepts of the knowledge base in this step. The purpose of this step is to collect the knowledge necessary to process the given data in the subsequent steps. Figure 2 illustrates an general example of this pipeline. This step is directly triggered by the user request. The initial input is the original user input (requests or attached text-based data), and the output is the revised prompts, which include the original user input and a reference to related knowledge.

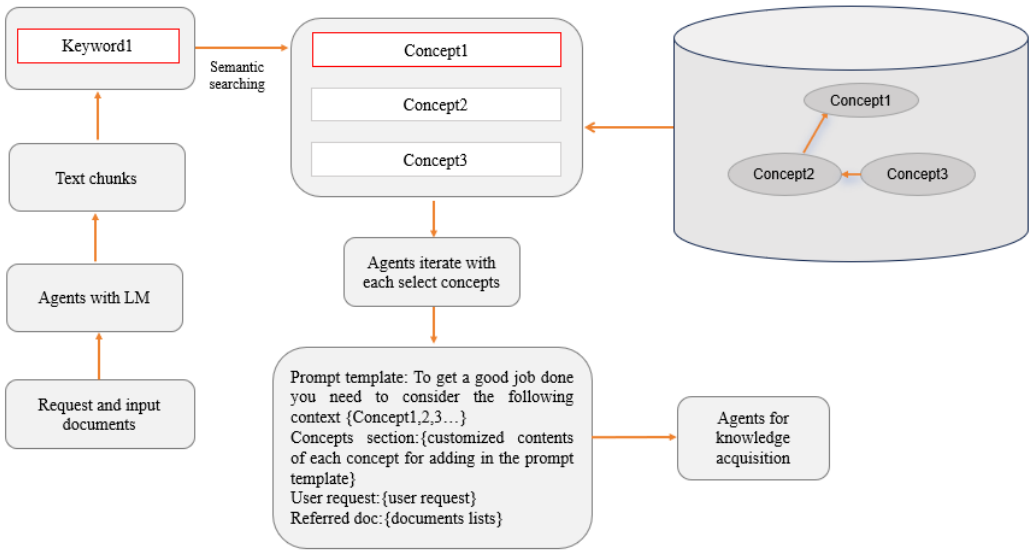


Figure 2. Context identification with multi-agent.

After the system has retrieved the context from user inputs, multi-agent process the context and the user input separately in different roles. The role of the involved agents can be customized based on predefined context patterns, which are groups of concepts in the combination sets. For example, it may define a particular pattern in the context of the student profile. In this case, the pattern needs to include relevant concepts such as the student's name, gender, registered courses, background, etc. Figure 3 illustrates a typical example of this process. The precondition for this step is that the prompt templates are chosen on the basis of a reference to context concepts. The expected output of this process is to synthesize a new prompt based on the combination of related prompt templates.

In the given example, there are three different roles: knowledge retrieval, self-validation, and format control. Each role corresponds to a particular agent that focuses on a specific question. For example, for the knowledge retrieval role, the agent aims to solve the question "what kind of extra knowledge is required to answer the users' request?" To answer this question, the agent will search through the concepts in the given context and integrate the related information into the question prompt with the help of an LLM. The prompt that the agent used in the process was derived from the original question of the role and could be extended with the relevant updated information. In this step, the agent requests that LLMs give suggestions or instructions for further search using the updated prompt, and iterate the loop after the prompt update until no new information is found. The output of one agent could be the input of other agents in a different role. For example,

the agent with the role of self-validation takes the output of knowledge retrieval and checks the integrity and quality of the extracted knowledge.

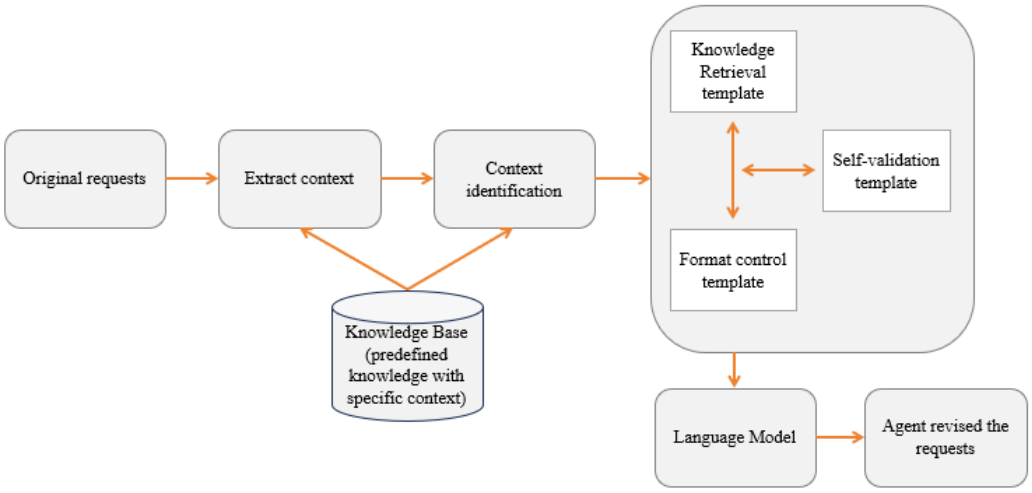


Figure 3. The collaboration of different agents in knowledge acquisition.

Finally, the agent with format control integrates all retrieval information based on the given format. Through this interaction, agents collaborate to search and build knowledge as the user requests. Take as an example curriculum development, assuming that the user inputs a sample document of the curriculum in machine learning to our framework. The request from the user is "taking this document as a sample of making the curriculum in machine learning" and the given textual document. First, the system asks the LM to extract the keywords and search the related concepts in the knowledge base based on keywords. In this case, the related concepts in the knowledge base are document preparation, document samples, and machine learning. The agent for knowledge retrieval asks the LM to check if data are available to present for each of these concepts. For the concept of a document sample, there is an attribute called samples, and the description of this attribute is used to store the given textual data as samples and to add the co-occurrence concepts as the context of the sample. The LMs fit the current context to this attribute description, and then the agent creates a new sample entry with the given data here and adds the rest of the concept as the context of the sample.

The extracted knowledge is represented in JSON format and sent back to the system. In this step, the program checks if there are any violations or conflicts with the interaction and integration rules that are defined in the concepts of previous knowledge. If there are no conflicts, the new knowledge is accepted and stored in the knowledge base, and the system updates the semantic lexicon accordingly. Otherwise, the new knowledge is reported to the expert user with the detected problems, waiting for further annotation by the expert users during knowledge validation. With the participation of experts, we apply active learning to the corresponding knowledge model to constantly optimize the model based on their expertise. The constraints of related concepts ensure the integrity and quality of the knowledge extracted. These rules are constantly reviewed and optimized by experts in active learning loops. Figure 4 shows an overview of active learning in our framework. The preconditions for invoking knowledge validation include the user's direct request or a system request from active agents. When active agents find unexpected knowledge conflicts, they request that experts validate the relevant knowledge model. The output of knowledge validation is the final feedback from experts and the knowledge update. After knowledge acquisition, the system builds specific knowledge models that

are correlated with a particular context, and this customized knowledge is used to improve the performance of LLMs in a similar context.

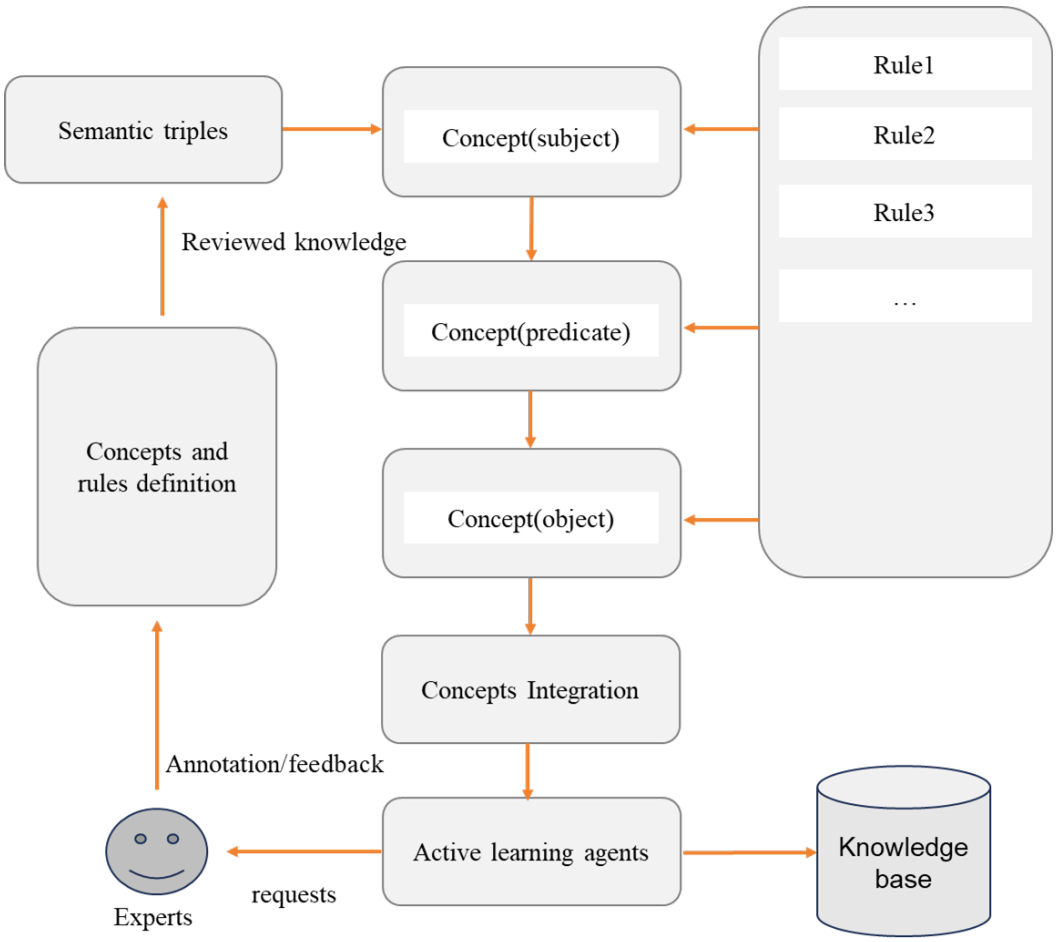


Figure 4. Active learning for knowledge validation and optimization.

In this paper, we primarily describe the development of the overall framework, while also emphasizing the crucial role of the related human–machine interaction modules. For the work of using active learning in knowledge validation, we suggest reading previous papers which described similar approaches, with more details of interactive modules such as explanatory dialogue [58] and knowledge-rich task planning [59]. Our goal here was to continuously improve the efficiency of the human–machine interaction within the framework.

3.2. RAG with Context-Based Knowledge Retrieval

In our framework, on the one hand, agents constantly extract external knowledge from users and try to integrate the new knowledge into the knowledge base. On the other hand, the agent also accesses the knowledge base and uses the RAG pipeline to facilitate the performance of LLMs by providing domain-specific knowledge support.

As discussed above, all knowledge engineering processes of the framework are based on the interaction between agents and users. The user's expertise is requested by agents to validate new knowledge when there is any violation of the constraint rules detected during the knowledge engineering. Each user is considered an expert in the corresponding aspects according to their profile and is requested to participate in the relevant knowledge validation processes. The constraint rules are also retrieved from the previous conversation between the users and the agents. Therefore, the professionalism of users regarding the

given knowledge is important for efficient knowledge updating and integration. For users involved within interdisciplinary topics, we could use validated domain-specific knowledge in the knowledge base to better select or explain these topics during knowledge engineering. At any time, the user can choose to update the relevant documents in the system, or type the text-based description instead to update the corresponding knowledge in the system. In addition, the user can also choose a previous conversation as context and input the context information into the system for the knowledge update. After that, the agents use LLMs to process user inputs and extract the relevant content to extend the concepts in the knowledge base.

The knowledge learned in the knowledge engineering processes will later be used to better serve user requests in domain-specific tasks with RAG pipelines. The system adopts the same method to extract the context of user requests in the RAG pipeline, but assigns agents to different roles based on the different context patterns. Take the example of curriculum development, after knowledge retrieval, the system assigns the role of document preparation to the next agent, and this agent searches for the curriculum samples based on the given topic and adds the retrieved samples to the prompt to optimize the response of the LLMs. For this role, the LLMs in the agent depend on the knowledge given and the included examples to perform the in-context learning and optimize the response with domain-specific knowledge. In the report by [60], the performance of LLMs based on in-context learning was comparable to fine-tuning, which usually requires a higher cost and more data to optimize the model itself. As discussed, the agent for knowledge retrieval will check if the input data include any samples for in-context learning during the knowledge-acquisition tasks. If there are related samples, the agent extracts the samples and annotates them as a sample attribute. The pseudocode presented in Algorithm 1 gives more details on the implementation.

Algorithm 1 Example pseudocode in RAG pipeline

Input: Retrieved knowledge X, Parameter synthesized prompt θ
Output: Processed results R
for each $x \in X$ **do**
    Update agent $y = f(x, \theta)$
    Running agent $y.RAGloop()$
    if $y.states == completed$ **then**
        Store new knowledge if there is any
        $r = y.output()$
        Release y
    else
        while Check $y.states$ **do**
            Invoke corresponding functions F or agents Z
            $r = y.output(F/Z)$
        end while
        Release y
    end if
    Store r into R
end for
for each $r \in R$ **do**
    Integrate θ with r
    Update (θ)
end for
return R

In this step, the agent retrieves the content of the sample attribute from the related knowledge concept in the reverse way and provides it to LLMs in the RAG loop. Figure 5 provides the workflow of the agent working with RAG to improve LLMs in a given task.

In summary, the whole RAG process in our framework can be concluded in the following four steps:

- Contextual information matching: The first step in the process is to capture and encode the user's context. Contextual information may include explicit user inputs (for example, search requests), implicit signals (e.g., user profiles), and external environmental factors (e.g., location or time). This contextual data are transformed into a few concept keywords using an encoder, such as in a pre-trained language model. These keywords represent the user's intent and needs in a compact, semantic format.
- Knowledge retrieval from the knowledge base: The concept keywords are used to search for relevant knowledge from the knowledge base. These sources may include the following:
  - KGs: Structured representations of domain-specific knowledge, such as product attributes, expert reviews, or user-generated content.
  - Document repositories: Collections of relevant textual data, such as articles, manuals, or FAQs. The system employs similarity-based ranking techniques (cosine similarity) to fetch the most relevant knowledge nodes. Figure 5 shows an example of the use of customized knowledge to recommend better instruction examples in curriculum development. In this example, the system can find the predefined prompt template from the knowledge base based on the user's requests and use that prompt to request that the LLMs give more specific responses.
- Context-aware prompt and contextual adaptation: The retrieved information is added to the given prompt template along with the original user request for synthesis of the input prompt for LLMs. This step ensures that its responses are informed by the most up-to-date and relevant knowledge.
- Personalized recommendation generation: The LLMs generate responses in a natural language format, enriched with context-specific explanations. The user can define their favorite styles in the relevant prompt templates to personalize the format of the output. Using RAG, the system can achieve better contextual awareness, accuracy, and user satisfaction.

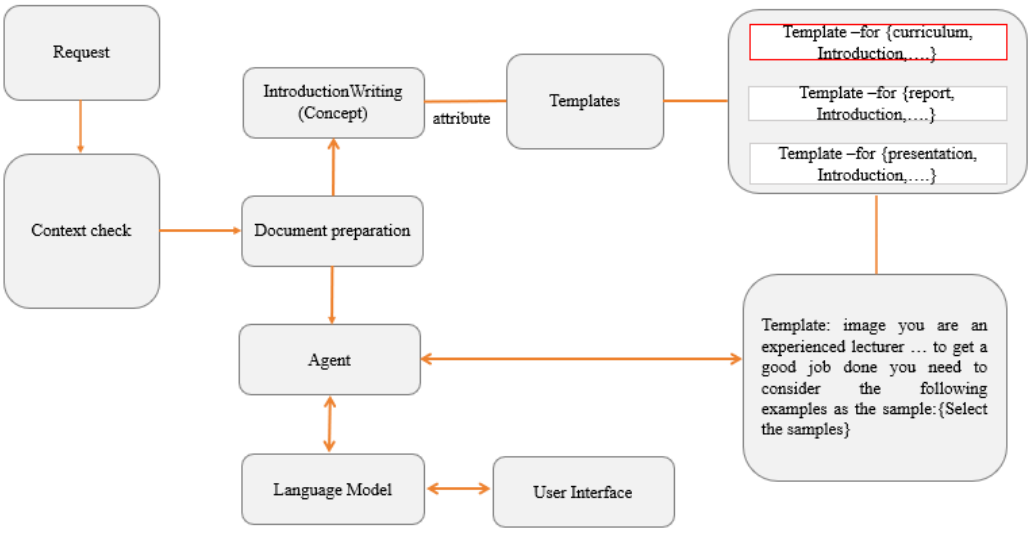


Figure 5. RAG workflow with customized knowledge support.

All RAG workflows are based on the user's request and the knowledge support from the knowledge base. The output of the RAG processes is the extended prompts, which

include all additional knowledge and context. This additional knowledge can help the LLMs provide more accurate and context-aware responses during tasks.

4. Results and Discussion

The ultimate objective of this research is to leverage AI-powered agents for personalized adaptive learning. To accommodate diverse user backgrounds and deliver context-aware personalized content, it is essential to develop and use domain-specific knowledge that improves the capabilities of the AI model. This study aims to optimize the responses of embedded LLMs in AI agents by integrating relevant contextual and domain-specific knowledge. To assess the effectiveness of our approach, we investigated how improvements in our knowledge support impacted the performance of LLMs across various domain-specific contexts.

In this paper, we tested the performance of different LLMs in different domains and examined the potential of using our knowledge support to improve the performance of LLMs in these domains. The results can be divided into two parts. In the first part, we benchmark-tested a few open-source LLMs with datasets from diverse domains. After this, we used the selected LLM (Gemma:7b) to redo the test with the support of the relevant knowledge that our framework extracted from the dataset. To gain a more comprehensive view of the LLM working in different domains, we tested a few of the latest open-source LLMs, including Gemma 2b [61], Gemma 7b [61], Llama3 [62], and Mistral [63] at the beginning of our test. All experiments discussed in this paper used the given datasets that were selected from MMLU (Massive Multitask Language Understanding) [64]. MMLU is a new benchmark created to assess the knowledge gained during pre-training by evaluating models solely in zero-shot and few-shot scenarios. Spanning 57 different subjects, it encompasses a wide range of disciplines, including STEM (Science, Technology, Engineering, and Mathematics), the humanities, social sciences, and beyond. This benchmark offers a comprehensive array of topics in diverse domains, ensuring a thorough and rigorous evaluation. In our experiments, the initial knowledge in the knowledge base was automatically retrieved by agents from the given text documents which included the corresponding domain knowledge, and we manually designed the necessary rules and constraints to check the retrieved knowledge.

In addition to testing the AI models and knowledge models, we also tested dynamic updates to the knowledge base. This update is supposed to be based on the interaction between the AI system and the users. As discussed above, customized knowledge is important to support AIs in adapting to the user's various contexts. In the test, we focused on proving the impact of customized knowledge to improve the responses of AI agents in given practical scenarios. As a first step, the results in this paper in particular demonstrate the impact of user-customized knowledge and the RAG loop to improve the responses of AI agents in a given scenario. In future tests, our objective is to assess the impact of continuous human–machine interaction to optimize complex knowledge and to test the responses of AI agents from a more comprehensive point of view. We recognize the importance of continuous optimization and will integrate ongoing updates with diverse evaluation metrics and participation in future tests.

4.1. Test on the Performance of LLMs with Knowledge Support

In this section, we focus on the test to evaluate the performance of LLMs with knowledge support. To measure and compare the performance of the LLMs under different conditions, we selected questions related to different domains and compared the performance of pre-trained LLMs in answering these questions. The different questions were randomly selected from the MMLU dataset to check the response of pre-trained LLMs

in diverse domains, without any extra knowledge support. We tested 4 different LLMs (Gemma2b, Gemma7b, Llama3, and Mistral) with questions from 9 different groups. For each group, questions were followed with topics from the same domain, and we calculated the average performance data of the LLMs. The performance of the LLMs was measured using the F1 score [65]. Figure 6 shows the conclusions of these experiments. In the figure below, the results clearly demonstrate that the different topics exerted complex and varied influences on the performance of the different LLMs. In contrast, Gemma:7b and Mistral had a comparatively more stable and better average performance on the testing sample data. More details of the experiment results can be found in Table A1 in Appendix A. To simplify the experiment and measurement, we decided to use Gemma:7b as the only default LLM in the subsequent testing experiments to test the impact of knowledge support in different domains, and this choice was based on a consideration of the representativeness and stability of the model.

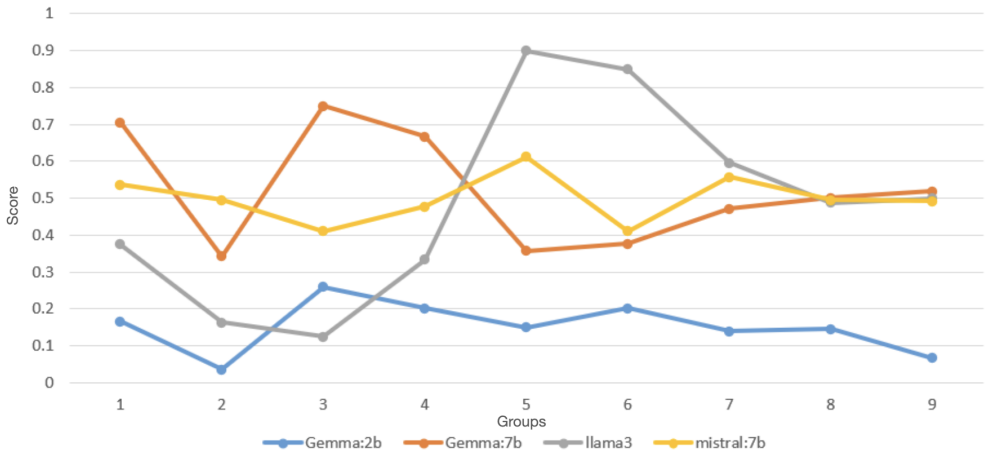


Figure 6. Comparison of the performance of the LLMs based on different domain topics (X axis for groups, and Y axis for F1 score).

To gain a broader view of the impact of the knowledge domain, we tested with LLM Gemma:7b on datasets from 31 different domains. These tests aimed to examine the impact of more diverse topics on the performance of the pre-trained LLM. In the latter tests, we calculated performance metrics including accuracy, precision, recall, and F1 score. Figure 7 gives an overview of the results of these tests, and the details of the subjects tested can be found in Table A2 in Appendix A.

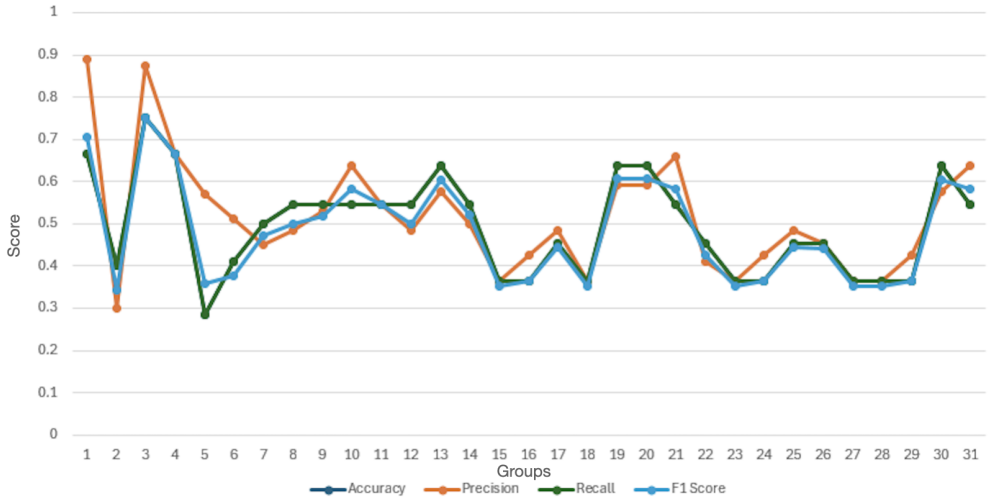


Figure 7. The performance of Gemma:7b across different domains (X axis for groups, and Y axis for scores).

Throughout all these tests, we observed that different topics imposed an strong influence on the performance of the LLMs, and the difference in model performance was mainly caused by the understanding of diverse domain-specific knowledge. Missing such related knowledge in the general model seriously limited the performance of the model in specific tasks, giving space for using expertise to improve the model.

Improvement with Domain-Specific Knowledge Support

To further prove our assumption based on the impact of domain knowledge, we extend our experiments with knowledge support from the framework discussed in Section 3. Except for benchmarking the performance of the different LLMs in diverse domains, we tested the performance of the given LLM (Gemma:7b) in different domains with customized knowledge support and compared the difference with a model without extra knowledge support in the same scenarios. The purpose of these tests was to examine the potential of our proposed approach and to identify the characteristics of the framework. In these tests, we used the default LLM Gemma:7b for all agents to process prompts during the tasks. Meanwhile, the agents in the framework also used the "paraphrase-MiniLML6-v2" sentence transformer model [66] from the hugging-face library as the indexing model for semantic search for RAG. To test the impact of adding relevant domain-specific knowledge, we first established corresponding knowledge models in the knowledge base. These knowledge models were automatically created by the agents that extracted the knowledge from the given text-based data related to the domain-specific knowledge. In these tests, the knowledge was extracted from the files given by the users, so we tested our framework with the predefined knowledge that was prepared based on the test scenarios. The dataset for this test was 11 groups of questions from different domains and topics covered by various subjects in STEM. Given the considerable variation in the scale of datasets across domains in MMLU, we decided to select datasets of comparable scale in this experiment (instead of random selection) to better compare the improvement in LLM efficiency on different domains after horizontal knowledge enhancement. We also ensured a balanced representation of domains from the natural and social sciences, to facilitate meaningful comparisons.

First, we requested the framework to analyze the related datasets and extract knowledge concepts from the data. The description of these concepts was automatically extracted from documents and indexed by the sentence transformer model. The embeddings index was saved with the knowledge as labels in the knowledge base. After the knowledge

had been prepared, we provided the questions and added the retrieved knowledge of the corresponding domains to improve the LLM and check the response (K_support). The same questions were also given to agents with prompt template support (prompt) or only working with LLM (baseline). For the prompt template support group, we provided the prompt templates that were designed based on the Chain of Thought (CoT) technique to improve the performance of the LLMs. All results were recorded and analyzed as F1 scores to measure the performance of the LLM. In addition, we also recorded the computation time the LLM spent in completing the given tasks, as another way to measure the performance. Figure 8 shows the conclusions for the time cost of LLMs (in seconds) in all the tests in various domains.

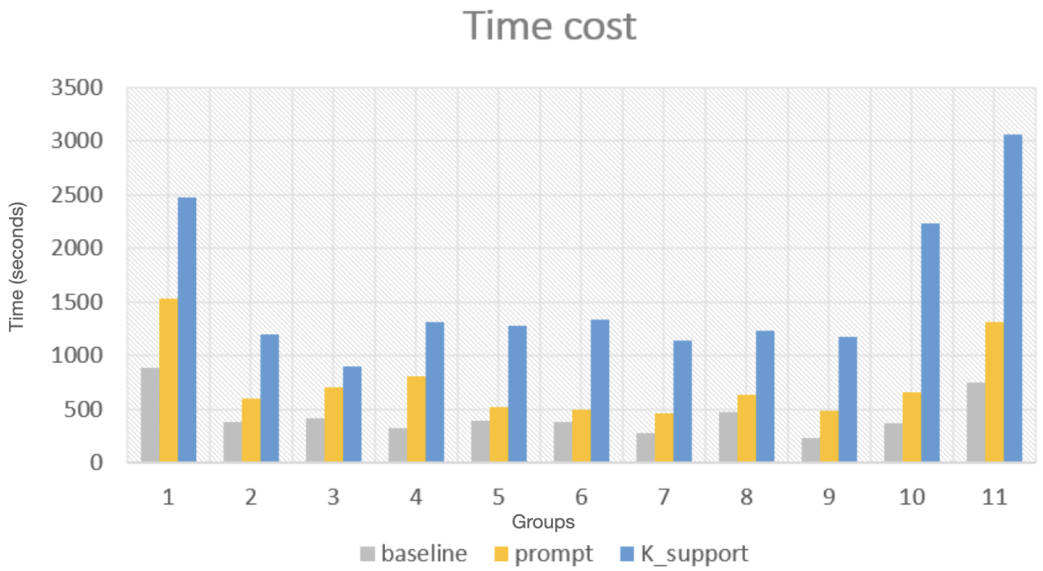


Figure 8. The time cost of Gemma:7b in various domains (X axis for groups, and Y axis for time cost).

Figure 9 shows the conclusions for the performance of the LLMs (F1 score).

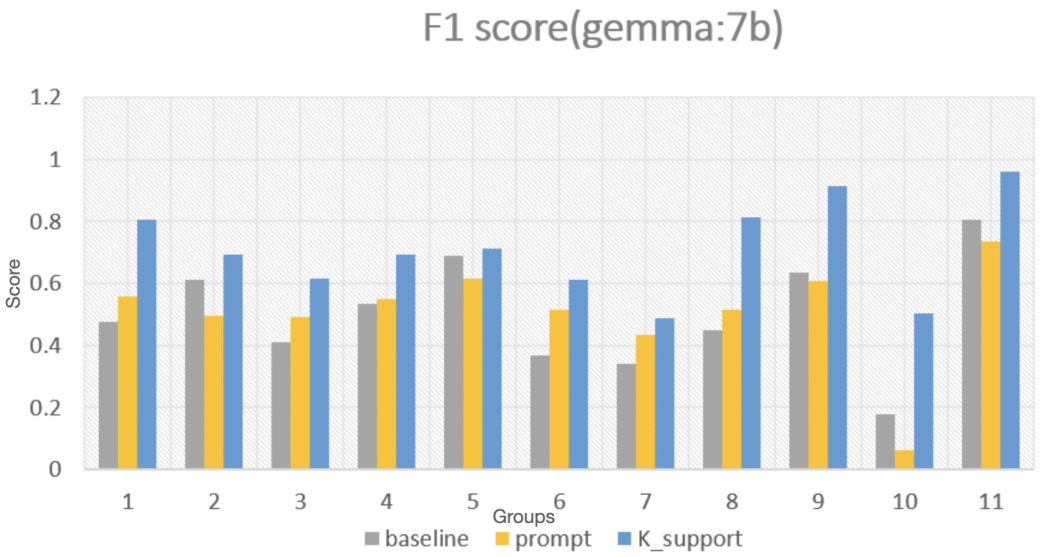


Figure 9. The performance of Gemma:7b in various domains (X axis for groups, and Y axis for F1 score).

From the figures above, we can see that the given knowledge support effectively improved the performance of the LLMs, although extra knowledge retrieval and semantic searching also cost more time in the tasks. Compared to tests with CoT prompt support

only, the effect of knowledge support was significant in almost all domain tasks. To provide a more concrete comparison, we list the actual improvements for the knowledge support tests compared to the test without any knowledge support with different domain topics (11 domains), and the details can be seen in Table 1. The "performance" represents the F1 score of the knowledge support tests. The "distance" and "improvement", respectively, refer to absolute and relative differences compared to the test without any knowledge support.

Table 1. The impact of knowledge support across different domains (F1 scores)

Domains	Performance	Distance	Improvement %
Biology	0.81	0.36	44%
Clinical knowledge	0.80	0.33	40%
Machine learning	0.50	0.32	64%
Management	0.91	0.27	30%
Global facts	0.61	0.24	39%
Anatomy	0.62	0.21	33%
Medicine	0.69	0.16	22%
Marketing	0.96	0.15	16%
Chemistry	0.48	0.14	30%
Business ethics	0.69	0.08	12%
History	0.71	0.02	3%

For these results, we found that the impact of knowledge support was also different in diverse domain topics. Some specific scientific topics such as biology and machine learning seemed to have a better chance of being improved by providing the necessary expertise than others. To explain this difference, we assume that knowledge support can help domains include more specific terms with a better chance. In the opposite case, the general knowledge in a pre-trained model can already make the performance quite good for some popular topics. Figure 10 shows an intuitive comparison of the LLM performance in the different domains.

In the experiments described above, we observed that the tested LLMs generally achieved higher average performance scores in domains that aligned with widely discussed or trending topics in the media and on the Internet, such as marketing and clinical knowledge. In contrast, for more specialized domains that require deeper domain-specific expertise, such as chemistry and machine learning, external knowledge support contributed to more substantial performance improvements, although the overall performance in these areas remained comparatively low. These observations suggest that the distribution of domain-specific knowledge in training data may be uneven and that this imbalance may have been reflected in the variable performance of the model in different domains.

Through a series of comparative experiments, we observed that the incorporation of additional knowledge significantly enhanced the overall performance of the LLMs, with this improvement evident in various domains. However, the specific impact of this knowledge varied and was influenced by a range of contextual factors, as well as by the underlying architecture of the model. Given the limited transparency regarding both the structure of the LLM and the composition of its training data, this study focused on a preliminary investigation of the general effects of external domain knowledge on model performance, without attempting a detailed analysis of domain-specific differences. To further investigate the hypothesis of an uneven knowledge distribution, future research should incorporate a deeper understanding of the model architecture and training mechanisms. In future work, we hope to design targeted experiments that can more precisely elucidate how different

types of domain knowledge impact LLM performance and how this knowledge can be leveraged more effectively to optimize model outcomes.

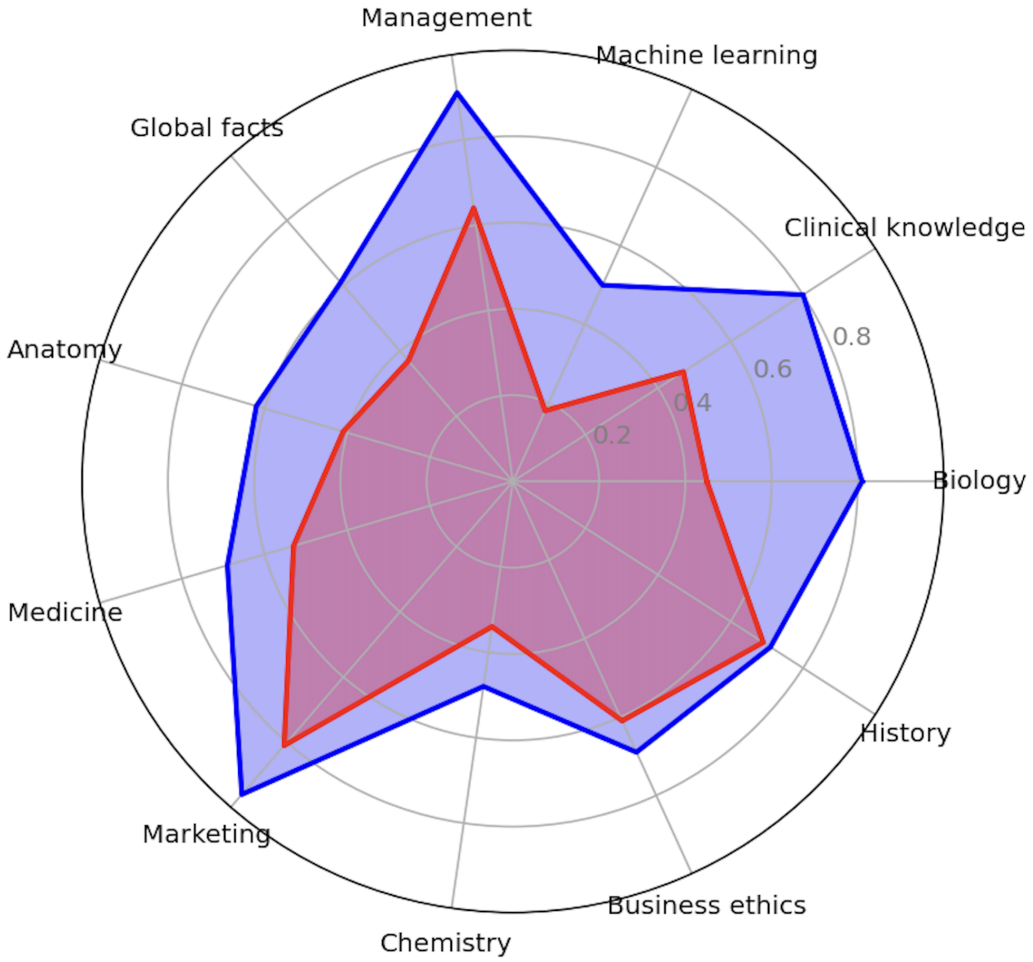


Figure 10. Comparison of LLM performance in various domains. Blue curve: The performance of the LLM with knowledge support; Red curve: the performance of the LLM without knowledge support (based on F1 score).

Finally, we checked the results of knowledge retrieval during the tasks and found that many of the inaccurate answers were actually caused by a lack of a match of the knowledge content with the context. This issue should be able to be mitigated by introducing a more precise indexing model and more elaborated prompt templates. On the other hand, we will also consider integrating more effective ontology matching models to improve the semantic searching process in our future work. In fact, the problem discussed above means that the framework still has some space to improve its efficiency in optimizing the knowledge-engineering process. This fact also encourages us to implement an active learning process with experts to improve our knowledge in future work.

4.2. Test with Customized Knowledge Update

The discussed AI system aims to help personalized adaptive learning in a digital transformation context. This goal requires an excellent learning process, with comprehensive knowledge integration in multiple domains. Taking into account the diversity of student backgrounds and the many cross-domain challenges in learning, a personalized adaptive learning approach will be widely adopted in the system to improve learning efficiency and reduce cross-regional and cross-domain challenges. This means that the knowledge we use needs to be customized with each user and be dynamically updated based on the requests

of users. In this section, we will discuss the results related to the customized knowledge updates and their importance.

To assess the significance and effectiveness of personalized knowledge updates in practical applications, we performed an experimental evaluation using a real-world scenario of study plan creation. In this study, the system processed 100 anonymized resumes and generated customized study plan recommendations tailored to the corresponding students based on the content of each resume. The recommendations were designed to provide preparatory guidance for a specific course. To evaluate the quality of these recommendations, we established five key assessment criteria based on expert opinions, which were used to measure user satisfaction and assign performance scores. The anonymized resume data were downloaded from the Kaggle Resume Dataset, which was obtained by scraping individual resume examples from the www.livecareer.com, accessed on 29 April 2025 [67].

In our test, we used the system as an AI assistant to help new students with suitable study strategies, recommended based on their profiles. The experiment procedure included the following steps:

- Data Preparation: Anonymized resume data available online were downloaded and used to simulate student registration processes to initiate the program.
- Personal Knowledge Extraction: The resume and personal information of each student were entered into the system to provide a basic dataset for personalized recommendations.
- Profile and Customized Knowledge Update: The system was tasked with retrieving the most suitable user profile knowledge and relevant domain-specific knowledge (i.e., description of the course module) from the knowledge base. This data were used to prepare a customized learning plan tailored to the individual's background and interests. All domain-specific knowledge was extracted from relevant literature documents and stored in the system knowledge base. The user could ask the system to extend this customized knowledge by updating more files or inputs at any time.
- Knowledge Integration and Recommendation: Based on domain-specific knowledge and the student profile, the system was requested to generate advice and compile a report detailing possible suggestions for the student.
- Interactive Guidance: Finally, the system interacted with the students to address specific questions and provide contextual guidance based on the generated learning plan.

The default LLM used in this experiment was Llama 3.1 (8b) [68] and the user request was as follows: "Based on the applicant's resume and profile, you suggest to the applicant how to prepare for the given course [cloud computing]." To evaluate the quality of the LLM responses (suggestion), we listed five rules to check if the content satisfied each important aspect based on the learning outcome specification of the given course. The five rules are listed below:

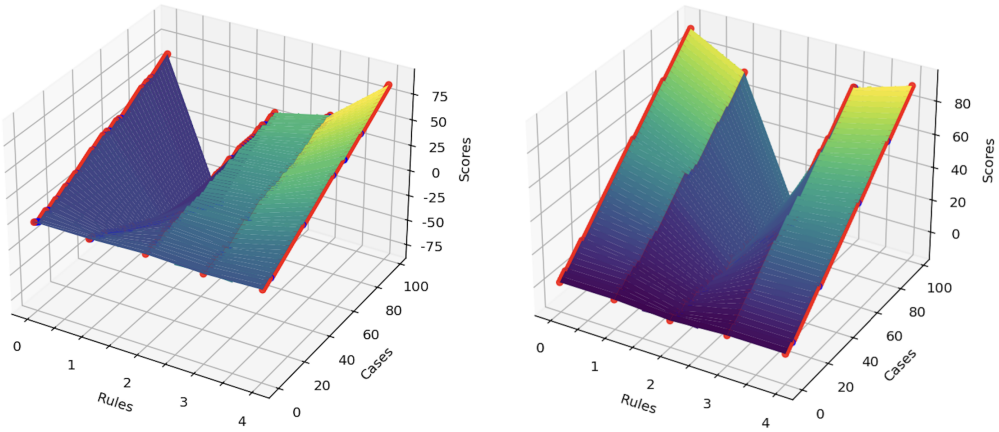
- Rule0: The suggestion should include a recommended reading list.
  - Importance: A reading list provides tangible, actionable resources to start learning, and bridges the gap between wanting to study and knowing where to begin. As the curriculum designers suggested, a good reading list is the primary part of a pre-study guide, preventing the guide from remaining vague or overly theoretical.
  - Related aspects:
    - Aligns resources with the course's depth and complexity.
    - Sets expectations about content difficulty.
- Rule1: The suggestion should explain the prerequisites for studying the course.
  - Importance: Prerequisites prevent students from jumping into material they are not ready for. Such a suggestion protects learners from frustration by making sure they

are adequately prepared. The instructors hope to use this information to better guide students in self-assessment and to be fully prepared before starting the course.
  - Related aspects:
    * Guide students to fill knowledge gaps first if needed.
    * Supports scaffolded learning, where new knowledge builds on existing understanding.
- Rule2: The suggestion should be of proper length.
  - Importance: Length affects clarity and usability. If the suggestion is too short, it might be incomplete or vague. Otherwise, it could overwhelm the reader or bury key points.
  - Related aspects:
    * Efficient communication.
    * Easy to digest.
    * Focused on essentials, without filler.
- Rule3: The suggestion should include some custom advice for the applicants.
  - Importance: Generic advice may not resonate or be useful for all learners. To provide adaptive learning, the system needs to be able to understand the personal characteristics of each student and provide them with customized advice.
  - Related aspects:
    * Makes learners feel seen and supported.
    * Can address things like learning style, time management, or career goals based on the student's context.
    * Make learning suggestions that are appropriate for the student's context.
- Rule4: The suggestion should consider the particular information on the personal profile of the given applicants.
  - Importance: This ensures that personalization in adaptive learning is based on the particular profiles of the students. This rule aims to examine the knowledge integration between domain-specific knowledge and personal profile knowledge. All features need to be extracted from the correct real user profile and seamlessly adapted to the background of the chosen courses.
  - Related aspects:
    * Personalized feature extraction
    * Knowledge integration

Based on the opinions of relevant experts, we identified the rules above to examine the output of the AI model. These rules address three key concerns about the quality of the output. Rules 0 and 1 assess whether the output is informative, Rule 2 ensures that it is accessible and user-friendly, and Rules 3 and 4 assess the essential functionalities needed to support adaptive learning in practical tasks.

After the LLM generates the suggestions, we check if the suggestions satisfy each of the above rules. If the suggestion follows the particular evaluation rule, the corresponding score will be +1, otherwise -1. With this, we can score the given suggestion for the five different aspects. In our experiments, we initially ran the test using only the personal profile as a context to request suggestions from the LLM, and then we repeated the same use cases again. The second time, the system allowed the agents to retrieve more customized knowledge from the designated knowledge base and integrated this with the user profile to support the LLM in generating the responses. The knowledge in the knowledge base was customized with the user's previous input information, and can be dynamically reviewed and updated by users. In the latter cases, we simulated the situation in which the system

dynamically retrieves the relevant updated customized knowledge based on the task context. Figure 11 shows a comparison between the use cases with or without a customized knowledge update. The two sets of tests each included the insignificant 100 use cases, and we summarize all the comparison results in Figures 11 and 12 below. The three axes X, Y, and Z represent the corresponding evaluation rules, the order of use cases, and the instant score for a given rule.

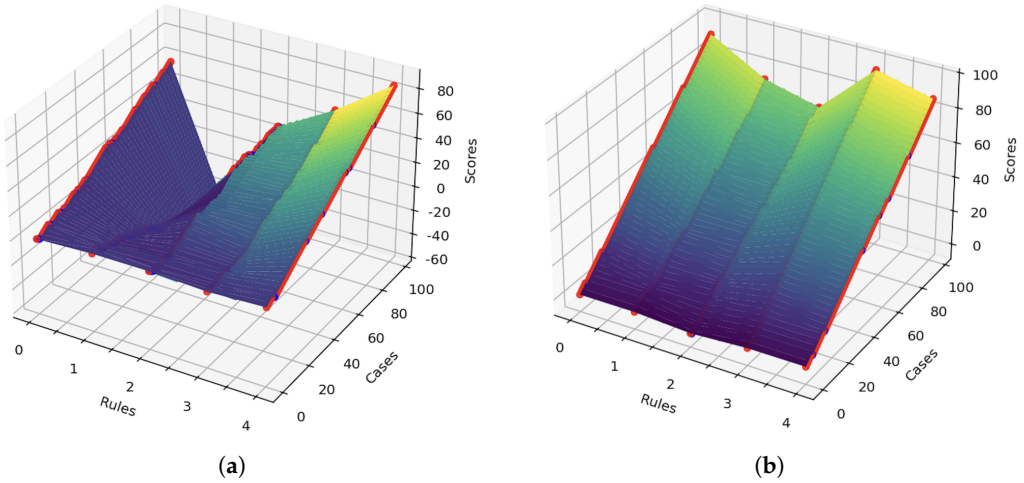


(**a**) (**b**)

Figure 11. Comparison for the impact of customized knowledge updates (**a**) The result of use cases without customized knowledge support (**b**) The result of use cases with customized knowledge support.

As can be seen in the above figure, compared with use cases without customized knowledge updates, use cases with knowledge updates generally met the evaluation rules better and ultimately achieved better overall scores on most rules. One exception here was for rule 2-"The suggestion should be of proper length." Based on that rule, use cases with customized knowledge updates seemed not to have an advantage. The reason for this was that the evaluation rule statement is too subjective and lacks specific knowledge and concrete examples to support the concept of "proper length". The scoring of experimental results was also based on the LLM's judgment according to the given rule statement, but when the LLM does not understand the connotation of specific concepts such as "proper length", the evaluation results will appear arbitrary. Fortunately, in our system, this type of problem can also be efficiently solved by updating specific knowledge. To fix this particular problem, we updated a manual course preparation suggestion as a sample and asked the LLM to consider the given sample when it gave the evaluation based on the rules. After updating the knowledge, we repeated the evaluation of the same suggestions, and generated the new results shown in Figure 12.

In the new re-evaluation results, we can see that the scores of all use cases were improved because of the custom and concrete sample. Moreover, the score of rule 2 on the latter set improved significantly and the discussed bias was effectively fixed. This made the evaluation results closer to user requirements and more reasonable, while maintaining the necessary consistency. This demonstrated the potential of suitable customized knowledge updates to solve the LLM performance limitation problem. In our system, agents constantly interact with users and according to user feedback, and input to update the knowledge base to improve the performance of LLMs during tasks.



(**a**) (**b**)

Figure 12. Re−evaluating the comparison of the impact of customized knowledge support (**a**) The result of use cases without customized knowledge support (**b**) The result of use cases with customized knowledge support.

5. Conclusions

In this research, we proposed a novel approach using multi-agent to improve the domain-specific knowledge engineering process and then applied this customized knowledge to facilitate the performance of LLMs in domain-specific tasks. Through the initial benchmark tests in different scenarios, we proved the potential of this approach and identified the essential problems that we aim to solve through our framework. The importance of the relevant customized knowledge (expertise) and the necessity of user participation in the loop were proved and demonstrated by testing with the experiments. Despite the surprising ability of the LLMs in semantic processing and inference, the many limitations and fluctuations with different contexts revealed in the tests tell us that the performance of LLMs could still be improved by providing the necessary knowledge inputs from human users to construct sophisticated knowledge models for practical tasks in a given domain.

However, building knowledge models automatically based on a given domain is a challenge and requires constant dynamic updates and optimization based on user feedback. Fortunately, with the application of multi-agent, LLMs can help us simplify the whole loop from knowledge extraction and integration to the deployment of LLMs in specific tasks in a collective manner. The experiments discussed in this paper showed promise and good feasibility in this direction.

In future studies, we will continue to improve the approach discussed in the paper based on the experience from our results and try to address the problems discovered. In addition, we will apply this platform within various practical scenarios to test the overall impact and use more comprehensive metrics to evaluate the recommendations from different perspectives. The main plans that will be implemented in future research and development are listed below.

First, we will continue to improve prompt templates and agents to facilitate knowledge integration and task applications. Second, the role of users in the knowledge validation pipeline and the interactive persona simulation should be highlighted. To better evaluate the improvement in responses, we will also introduce new datasets and scenarios into our tests, with the integration of diverse interactive modules (i.e., specific dialogues and prompt templates). User feedback through interaction will help the system constantly improve the quality of knowledge during the task, and the corresponding impacts will be demonstrated and evaluated in future tests. In our future work, we will continue

to improve the knowledge engineering process and the user interface to facilitate the approach discussed in this paper. In addition, in the future, the development of an efficient and expandable local knowledge base is expected to allow interdisciplinary tasks at a larger scale. At this stage, enhancing the knowledge maintenance and retrieval process requires a more robust knowledge base system, to enable the framework to scale effectively. By integrating an open-source vector database such as VectorDB, we aim to improve the local knowledge base to support efficient semantic searching and knowledge integration. Last but not least, data protection and privacy are among the primary concerns for our future work when deploying the framework for practical use cases. Implementing policies of the General Data Protection Regulation (GDPR) within this framework will be a critical focus of our ongoing research.

Author Contributions: Conceptualization, Y.Y. and H.G.-V.; methodology, Y.Y.; software, Y.Y.; validation, Y.Y. and H.G.-V.; formal analysis, Y.Y.; investigation, Y.Y.; resources, Y.Y.; data curation, Y.Y.; writing—original draft preparation, Y.Y.; writing—review and editing, Y.Y. and H.G.-V.; visualization, Y.Y.; supervision, H.G.-V.; project administration, H.G.-V.; funding acquisition, H.G.-V. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by European Commission DIGITAL under Grant No.:101084013.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable

Data Availability Statement: Original result data in .csv format is available on request.

Acknowledgments: This work was developed under the auspices of the "DIGITAL4Business Masters Programme focused on the practical application of Advanced Digital Skills within European Companies," a project funded from Dec/2022 to Nov/2026 by the European Commission DIGITAL https://digital4business.eu/ under Grant No.: 101084013.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. The List of Tested Subjects in the Experiments

The objective of this experiment was to evaluate how the performance of different large language models (LLMs) varied across subject domains, due to differences in domain-specific knowledge and expertise. The experiment was structured in two parts:

In the first part, we assessed the performance of four representative open-source LLMs across nine distinct domains. The results, presented in Table A1, indicate that each model exhibited varying levels of performance depending on the domain, reflecting differences in their underlying knowledge representations.

In the second part, we expanded the evaluation to 31 domains using the Gemma-7B model, in order to further investigate the sensitivity of a single LLM to domain-specific content. The results, shown in Table A2, demonstrated that even within a single model, performance could fluctuate significantly depending on the domain, underscoring the importance of domain knowledge in LLM behavior.

Appendix A.1. List of Tested Subjects with Different LLMs Across Different Domains

Table A1. Performance of different LLMs across different domains.

Subject	Gemma:7b	Mistral:7b	Llama3	Gemma:2b
Politics	0.71	0.53	0.38	0.17
Chemistry	0.34	0.49	0.17	0.05
Astronomy	0.75	0.42	0.12	0.28
History	0.69	0.48	0.34	0.2
Computer security	0.36	0.62	0.89	0.15
Global facts	0.37	0.41	0.86	0.21
Clinical knowledge	0.47	0.56	0.59	0.14
Geography	0.50	0.50	0.49	0.15
Medicine	0.53	0.49	0.5	0.08

Appendix A.2. List of Tested Subjects with Gemma:7b Across Different Domains

Table A2. The performance of Gemma:7b across different domains.

Subject	F1 Score
Politics	0.71
Chemistry	0.34
Astronomy	0.75
History	0.69
Computer security	0.36
Global facts	0.37
Clinical knowledge	0.47
Geography	0.50
Medicine	0.53
Microeconomics	0.57
Moral disputes	0.55
Algebra	0.51
Business ethics	0.61
Miscellaneous	0.52
Philosophy	0.34
Psychology	0.35
Biology	0.45
Statistics	0.34
International law	0.62
Moral disputes	0.62
Human aging	0.59
Anatomy	0.41
Electrical engineering	0.35
Logical fallacies	0.36
Mathematics	0.47
Human sexuality	0.46
Virology	0.35
Accounting	0.36
Nutrition	0.37
Moral scenarios	0.61
Religions	0.58

References

1. Xing, M.; Zhang, R.; Xue, H.; Chen, Q.; Yang, F.; Xiao, Z. Understanding the weakness of large language model agents within a complex android environment. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Barcelona, Spain, 25–29 August 2024; pp. 6061–6072.
2. Uchida, S. Using early LLMs for corpus linguistics: Examining ChatGPT's potential and limitations. *Appl. Corpus Linguist.* **2024**, *4*, 100089.
3. Fan, W.; Ding, Y.; Ning, L.; Wang, S.; Li, H.; Yin, D.; Chua, T.S.; Li, Q. A survey on rag meeting llms: Towards retrieval-augmented large language models. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Barcelona, Spain, 25–29 August 2024; pp. 6491–6501.
4. Wang, H.; Li, Y.F. Large Language Model Empowered by Domain-Specific Knowledge Base for Industrial Equipment Operation and Maintenance. In Proceedings of the 2023 5th International Conference on System Reliability and Safety Engineering (SRSE), Beijing, China, 20–23 October 2023; IEEE: Piscataway, NJ, USA,2023; pp. 474–479.
5. Wu, J.; Yang, S.; Zhan, R.; Yuan, Y.; Chao, L.S.; Wong, D.F. A survey on LLM-generated text detection: Necessity, methods, and future directions. *Comput. Linguist.* **2025**, *51*, 275–338.
6. Hofer, M.; Obraczka, D.; Saeedi, A.; Köpcke, H.; Rahm, E. Construction of Knowledge Graphs: Current State and Challenges. *Information* **2024**, *15*, 509.
7. Yildirim, I.; Paul, L. From task structures to world models: What do LLMs know? *Trends Cogn. Sci.* **2024**, *28*, 404–415.
8. Yang, J.; Jin, H.; Tang, R.; Han, X.; Feng, Q.; Jiang, H.; Zhong, S.; Yin, B.; Hu, X. Harnessing the power of llms in practice: A survey on chatgpt and beyond. *ACM Trans. Knowl. Discov. Data* **2024**, *18*, 1–32.
9. Deng, J.; Zubair, A.; Park, Y.J. Limitations of large language models in medical applications. *Postgrad. Med J.* **2023**, *99*, 1298–1299.
10. Asher, N.; Bhar, S.; Chaturvedi, A.; Hunter, J.; Paul, S. Limits for Learning with Language Models. In Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023), Toronto, ON, Canada, 13–14 July 2023.
11. Pal, S.; Bhattacharya, M.; Lee, S.S.; Chakraborty, C. A domain-specific next-generation large language model (LLM) or ChatGPT is required for biomedical engineering and research. *Ann. Biomed. Eng.* **2024**, *52*, 451–454.
12. Chen, Z.; Lin, M.; Wang, Z.; Zang, M.; Bai, Y. PreparedLLM: Effective pre-pretraining framework for domain-specific large language models. *Big Earth Data* **2024**, *8*, 649–672.
13. Li, Y.; Ma, S.; Wang, X.; Huang, S.; Jiang, C.; Zheng, H.T.; Xie, P.; Huang, F.; Jiang, Y. EcomGPT: Instruction-tuning Large Language Model with Chain-of-Task Tasks for E-commerce. *arXiv* **2023**, arXiv:2308.06966.
14. Wu, S.; Irsoy, O.; Lu, S.; Dabravolski, V.; Dredze, M.; Gehrmann, S.; Kambadur, P.; Rosenberg, D.; Mann, G. Bloomberggpt: A large language model for finance. *arXiv* **2023**, arXiv:2303.17564.
15. Nguyen, H.T. A Brief Report on LawGPT 1.0: A Virtual Legal Assistant Based on GPT-3. *arXiv* **2023**, arXiv:2302.05729.
16. Luo, R.; Sun, L.; Xia, Y.; Qin, T.; Zhang, S.; Poon, H.; Liu, T.Y. BioGPT: Generative pre-trained transformer for biomedical text generation and mining. *Briefings Bioinform.* **2022**, *23*, bbac409.
17. Venigalla, A.; Frankle, J.; Carbin, M. Biomedlm: A domain-specific large language model for biomedical text. *MosaicML* **2022**, *23*, 2.
18. Deng, C.; Zhang, T.; He, Z.; Chen, Q.; Shi, Y.; Zhou, L.; Fu, L.; Zhang, W.; Wang, X.; Zhou, C.; et al. Learning A Foundation Language Model for Geoscience Knowledge Understanding and Utilization. *arXiv* **2023**, arXiv:2306.05064.
19. Bi, Z.; Zhang, N.; Xue, Y.; Ou, Y.; Ji, D.; Zheng, G.; Chen, H. Oceangpt: A large language model for ocean science tasks. *arXiv* **2023**, arXiv:2310.02031.
20. Soman, S.; HG, R. Observations on LLMs for telecom domain: Capabilities and limitations. In Proceedings of the Third International Conference on AI-ML Systems, Bangalore, India, 25–28 October 2023; pp. 1–5.
21. Pan, S.; Luo, L.; Wang, Y.; Chen, C.; Wang, J.; Wu, X. Unifying large language models and knowledge graphs: A roadmap. *IEEE Trans. Knowl. Data Eng.* **2024**, *36*, 3580–3599.
22. Siriwardhana, S.; Weerasekera, R.; Wen, E.; Kaluarachchi, T.; Rana, R.; Nanayakkara, S. Improving the domain adaptation of retrieval augmented generation (RAG) models for open domain question answering. *Trans. Assoc. Comput. Linguist.* **2023**, *11*, 1–17.
23. Amarnath, N.S.; Nagarajan, R. An Intelligent Retrieval Augmented Generation Chatbot for Contextually-Aware Conversations to Guide High School Students. In Proceedings of the 2024 4th International Conference on Sustainable Expert Systems (ICSES), Kaski, Nepal, 15–17 October 2024; IEEE: Piscataway, NJ, USA, 2024, pp. 1393–1398.
24. Ko, H.T.; Liu, Y.K.; Tsai, Y.C.; Suen, S. Enhancing Python Learning Through Retrieval-Augmented Generation: A Theoretical and Applied Innovation in Generative AI Education. In Proceedings of the International Conference on Innovative Technologies and Learning, Tartu, Estonia, 14–16 August 2024; Springer: Cham, Switzerland, 2024; pp. 164–173.
25. Long, C.; Liu, Y.; Ouyang, C.; Yu, Y. Bailicai: A Domain-Optimized Retrieval-Augmented Generation Framework for Medical Applications. *arXiv* **2024**, arXiv:2407.21055.

26. Li, Y.; Zhao, J.; Li, M.; Dang, Y.; Yu, E.; Li, J.; Sun, Z.; Hussein, U.; Wen, J.; Abdelhameed, A.M.; et al. RefAI: A GPT-powered retrieval-augmented generative tool for biomedical literature recommendation and summarization. *J. Am. Med. Inform. Assoc.* **2024**, *31*, 2030–2039.
27. Xu, Z.; Cruz, M.J.; Guevara, M.; Wang, T.; Deshpande, M.; Wang, X.; Li, Z. Retrieval-augmented generation with knowledge graphs for customer service question answering. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, Washington, DC, USA, 14–18 July 2024; pp. 2905–2909.
28. Halgin, F.; Mohammedand, A.T. Intelligent patent processing: Leveraging retrieval-augmented generation for enhanced consultant services. In Proceedings of the International Symposium on Digital Transformation, Växjö, Sweden, 11–13 September 2024.
29. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.t.; Rocktäschel, T.; et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 9459–9474.
30. Perković, G.; Drobnjak, A.; Botički, I. Hallucinations in LLMs: Understanding and Addressing Challenges. In Proceedings of the 2024 47th MIPRO ICT and Electronics Convention (MIPRO), Opatija, Croatia, 20–24 May 2024; IEEE: Piscataway, NJ, USA; pp. 2084–2088.
31. Ayala, O.; Bechard, P. Reducing hallucination in structured outputs via Retrieval-Augmented Generation. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track), Mexico City, Mexico, 16–21 June 2024; pp. 228–238.
32. Peng, C.; Xia, F.; Naseriparsa, M.; Osborne, F. Knowledge graphs: Opportunities and challenges. *Artif. Intell. Rev.* **2023**, *56*, 13071–13102.
33. Song, Y.; Li, W.; Dai, G.; Shang, X. Advancements in Complex Knowledge Graph Question Answering: A Survey. *Electronics* **2023**, *12*, 4395.
34. Edge, D.; Trinh, H.; Cheng, N.; Bradley, J.; Chao, A.; Mody, A.; Truitt, S.; Larson, J. From local to global: A graph rag approach to query-focused summarization. *arXiv* **2024**, arXiv:2404.16130.
35. Meyer, L.P.; Stadler, C.; Frey, J.; Radtke, N.; Junghanns, K.; Meissner, R.; Dziwis, G.; Bulert, K.; Martin, M. Llm-assisted knowledge graph engineering: Experiments with chatgpt. In Proceedings of the Working Conference on Artificial Intelligence Development for a Resilient and Sustainable Tomorrow, Leipzig, Germany, 20–29 June 2023; Springer Fachmedien Wiesbaden: Wiesbaden, Germany, 2023; pp. 103–115.
36. Zhu, Y.; Wang, X.; Chen, J.; Qiao, S.; Ou, Y.; Yao, Y.; Deng, S.; Chen, H.; Zhang, N. Llms for knowledge graph construction and reasoning: Recent capabilities and future opportunities. *World Wide Web* **2024**, *27*, 58.
37. Peng, H.; Ma, S.; Spector, J.M. Personalized adaptive learning: An emerging pedagogical approach enabled by a smart learning environment. *Smart Learn. Environ.* **2019**, *6*, 1–14.
38. Fariani, R.I.; Junus, K.; Santoso, H.B. A systematic literature review on personalised learning in the higher education context. *Technol. Knowl. Learn.* **2023**, *28*, 449–476.
39. Chen, D.L.; Aaltonen, K.; Lampela, H.; Kujala, J. The Design and Implementation of an Educational Chatbot with Personalized Adaptive Learning Features for Project Management Training. *Technol. Knowl. Learn.* **2024**, 1–26. https://doi.org/10.1007/s10758-024-09807-5
40. Shin, D.; Shim, Y.; Yu, H.; Lee, S.; Kim, B.; Choi, Y. Saint+: Integrating temporal features for ednet correctness prediction. In Proceedings of the LAK21: 11th International Learning Analytics and Knowledge Conference, Irvine, CA, USA, 12–16 April 2021; pp. 490–496.
41. Mejeh, M.; Sarbach, L. Co-design: From Understanding to Prototyping an Adaptive Learning Technology to Enhance Self-regulated Learning. *Technol. Knowl. Learn.* **2024**, 1–26. https://doi.org/10.1007/s10758-024-09788-5.
42. Lozano, E.A.; Sánchez-Torres, C.E.; López-Nava, I.H.; Favela, J. An open framework for nonverbal communication in human-robot interaction. In Proceedings of the International Conference on Ubiquitous Computing and Ambient Intelligence, Riviera Maya, Mexico, 28–29 November 2023; Springer: Cham, Switzerland, 2023, pp. 21–32.
43. Favela, J.; Cruz-Sandoval, D.; Rocha, M.M.d.; Muchaluat-Saade, D.C. Social Robots for Healthcare and Education in Latin America. *Commun. ACM* **2024**, *67*, 70–71.
44. Ruan, S.; Nie, A.; Steenbergen, W.; He, J.; Zhang, J.; Guo, M.; Liu, Y.; Dang Nguyen, K.; Wang, C.Y.; Ying, R.; et al. Reinforcement learning tutor better supported lower performers in a math task. *Mach. Learn.* **2024**, *113*, 3023–3048.
45. Kulshreshtha, D.; Shayan, M.; Belfer, R.; Reddy, S.; Serban, I.V.; Kochmar, E. Few-shot question generation for personalized feedback in intelligent tutoring systems. In *PAIS 2022*; IOS Press: Amsterdam, The Netherlands, 2022; pp. 17–30.
46. El Fazazi, H.; Elgarej, M.; Qbadou, M.; Mansouri, K. Design of an adaptive e-learning system based on multi-agent approach and reinforcement learning. *Eng. Technol. Appl. Sci. Res.* **2021**, *11*, 6637–6644.
47. Yekollu, R.K.; Bhimraj Ghuge, T.; Sunil Biradar, S.; Haldikar, S.V.; Farook Mohideen Abdul Kader, O. AI-driven personalized learning paths: Enhancing education through adaptive systems. In Proceedings of the International Conference on Smart data intelligence, Trichy, India, 2–3 February 2024; Springer: Singapore; pp. 507–517.

48. Amin, S.; Uddin, M.I.; Alarood, A.A.; Mashwani, W.K.; Alzahrani, A.O.; Alzahrani, H.A. An adaptable and personalized framework for top-N course recommendations in online learning. *Sci. Rep.* **2024**, *14*, 10382.
49. Bilyalova, A.; Salimova, D.; Zelenina, T. Digital transformation in education. In Proceedings of the Integrated Science in Digital Age: ICIS 2019, Munich, Germany, 15–18 December 2019; Springer: Cham, Switzerland, 2020; pp. 265–276.
50. Mukul, E.; Büyüközkan, G. Digital transformation in education: A systematic review of education 4.0. *Technol. Forecast. Soc. Chang.* **2023**, *194*, 122664.
51. Tomás, B.S. Extrinsic and Intrinsic Personalization in the Digital Transformation of Education. *J. Ethics High. Educ.* **2024**, *5*, 1–34.
52. Cantú-Ortiz, F.J.; Galeano Sánchez, N.; Garrido, L.; Terashima-Marin, H.; Brena, R.F. An artificial intelligence educational strategy for the digital transformation. *Int. J. Interact. Des. Manuf.* **2020**, *14*, 1195–1209.
53. Sun, Q.; Abdourazakou, Y.; Norman, T.J. LearnSmart, adaptive teaching, and student learning effectiveness: An empirical investigation. *J. Educ. Bus.* **2017**, *92*, 36–43.
54. Martin, F.; Chen, Y.; Moore, R.L.; Westine, C.D. Systematic review of adaptive learning research designs, context, strategies, and technologies from 2009 to 2018. *Educ. Technol. Res. Dev.* **2020**, *68*, 1903–1929.
55. Xiaoyu, Z.; Tobias, T. Exploring the efficacy of adaptive learning technologies in online education: A longitudinal analysis of student engagement and performance. *Int. J. Sci. Eng. Appl.* **2023**, *12*, 28–31.
56. Hagendorff, T. Deception abilities emerged in large language models. *Proc. Natl. Acad. Sci. USA* **2024**, *121*, e2317967121.
57. Oruganti, S.; Nirenburg, S.; English, J.; McShane, M. Automating Knowledge Acquisition for Content-Centric Cognitive Agents Using LLMs. In Proceedings of the AAAI Symposium Series, Arlington, VA, USA, 25–27 October 2023; Volume 2, pp. 379–385.
58. Yao, Y., González-Vélez, H.; Croitoru, M. Explanatory Dialogues with Active Learning for Rule-based Expertise. In Proceedings of the RuleML+ RR (Companion), Bucharest, Romania, 16–18 September 2024.
59. Shekhar, S.; Favier, A.; Alami, R.; Croitoru, M. A Knowledge Rich Task Planning Framework for Human-Robot Collaboration. In Proceedings of the International Conference on Innovative Techniques and Applications of Artificial Intelligence, Cambridge, UK, 12–14 December 2023; Springer: Cham, Switzerland, 2023; pp. 259–265.
60. Dong, Q.; Li, L.; Dai, D.; Zheng, C.; Ma, J.; Li, R.; Xia, H.; Xu, J.; Wu, Z.; Chang, B.; et al. A survey on in-context learning. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, Miami, FL, USA, 12–16 November 2024; pp. 1107–1128.
61. Team, G.; Mesnard, T.; Hardin, C.; Dadashi, R.; Bhupatiraju, S.; Pathak, S.; Sifre, L.; Rivière, M.; Kale, M.S.; Love, J.; et al. Gemma: Open models based on gemini research and technology. *arXiv* **2024**, arXiv:2403.08295.
62. Meta, A. Llama 3. 2024. Available online: https://llama.meta.com/llama3/ (accessed on 1 November 2024).
63. Jiang, A.Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D.S.; Casas, D.d.l.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. Mistral 7B. *arXiv* **2023**, arXiv:2310.06825.
64. Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; Steinhardt, J. Measuring massive multitask language understanding. *arXiv* **2020**, arXiv:2009.03300.
65. Goutte, C.; Gaussier, E. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In Proceedings of the European Conference on Information Retrieval, Santiago de Compostela, Spain, 21–23 March 2005; Springer: Berlin/Heidelberg, Germany, 2005; pp. 345–359.
66. Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics; Stroudsburg, PA, USA, 2019.
67. Bhawal, S. Resume Dataset. Data Set. 2021. Available online: https://www.kaggle.com/datasets/snehaanbhawal/resume-dataset (accessed on 29 April 2025).
68. Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. The llama 3 herd of models. *arXiv* **2024** arXiv:2407.21783.