

## EduAI Helper: A Retrieval-Augmented LLM–Knowledge Graph Framework for Teacher-Centered Pedagogical Support



Lamri Doudi<sup>1</sup>, Leila Megouache<sup>2\*</sup>, Mahieddine Djoudi<sup>3</sup>, Abdelhafid Zitouni<sup>4</sup>

<sup>1</sup> Computer Science Department and LRSD Laboratory, University Ferhat Abbas, Setif 19000, Algeria

<sup>2</sup> LASTERNE Laboratory, Geographical Sciences and Topography Department, Constantine 1 University, Constantine 25000, Algeria

<sup>3</sup> TechNE Laboratory, University of Poitiers, Building A3, Poitiers Cedex 86022, France

<sup>4</sup> Lire Laboratory, Computer Science Department, Constantine 2 University, Constantine 25000, Algeria

Corresponding Author Email: [Leila.megouache@umc.edu.dz](mailto:Leila.megouache@umc.edu.dz)

Copyright: ©2026 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/isi.310410>

### ABSTRACT

**Received:** 18 November 2025

**Revised:** 25 January 2026

**Accepted:** 14 April 2026

**Available online:** 30 April 2026

#### Keywords:

*large language models, knowledge graphs, retrieval-augmented generation, teacher support, educational technology, pedagogical assistant, curriculum grounding*

Teachers increasingly rely on digital tools for lesson preparation, resource selection, and instructional planning, yet generic language-model systems often provide responses that lack traceability, curriculum grounding, and pedagogical consistency. This study presents EduAI Helper, a teacher-centered pedagogical assistant that integrates large language models, a pedagogical knowledge graph, and retrieval-augmented generation within a web-based architecture. The system combines a React front end, a FastAPI back end, user management, document processing, semantic retrieval, and prompt enrichment from validated knowledge-graph entities. Its functions include lesson-content generation, question answering, resource recommendation, PDF summarization, concept exploration, and traceable response support. A preliminary evaluation was conducted with 30 participants, including teachers, master's students, and academic trainers from different disciplines and experience levels. The assessment used questionnaires, semi-structured interviews, observation, repeated educational scenarios, and controlled prompts. Results indicate statistically significant reductions in lesson-preparation time and improved perceived quality of generated teaching materials under the EduAI Helper condition compared with the baseline condition ( $p < 0.05$ ). Participants also reported 30–40% time savings for selected tasks such as quiz creation and resource search. Sensitivity analysis showed that Top-K = 5 provided the best retrieval setting, while larger retrieval sizes introduced less relevant context. Qualitative comparison with AI-TA, HiTA, and a GPT-4 baseline further suggests that knowledge-graph grounding strengthens curriculum alignment, semantic coherence, and teacher-oriented usability.

## 1. INTRODUCTION

Teaching is currently undergoing a significant transformation in the digital era, driven by the rapid advancement of educational technologies that are reshaping both pedagogical practices and learner expectations [1]. As a result, teachers are increasingly required to perform multiple roles, including knowledge delivery, learning path adaptation, administrative task management, and the maintenance of effective pedagogical interactions with diverse student populations.

This growing complexity is associated with higher workloads, greater heterogeneity among learner profiles, and increasing difficulty in accessing reliable and relevant educational resources [2]. Consequently, educators face significant challenges in maintaining both efficiency and quality in their teaching practices within dynamic and technology-enhanced learning environments.

Recent advances in artificial intelligence, particularly large

language models (LLMs) such as GPT-4 and BERT, have opened new perspectives for supporting teachers in addressing these challenges [3, 4]. These models are capable of processing and generating natural language, enabling applications such as pedagogical content generation, instructional personalization, and interactive educational support.

However, when used in isolation, LLMs present several limitations, including the generation of potentially inaccurate or imprecise responses and limited guarantees regarding the factual reliability of produced content [5]. These limitations highlight the need for complementary mechanisms to improve grounding and factual consistency in educational applications.

To address the limitations of existing educational support systems, the integration of knowledge graphs (KGs) has emerged as a promising approach. KGs structure information into interconnected semantic networks of concepts and relationships, enabling a more organized representation of domain knowledge. This structured representation improves

content coherence, enhances interpretability, and facilitates traceability of information for educational use.

Within this context, a key research question arises: how can a pedagogical assistant that integrates LLMs with KGs effectively support teachers in their instructional activities?

Such a system may provide functionalities including support for lesson planning through context-aware resource recommendation, personalization of learning materials according to student needs, and analytical insights into learner progress based on structured knowledge representation. Investigating these capabilities contributes to a better understanding of how LLM-KG integration can enhance intelligent educational support systems.

The integration of LLMs with KGs is motivated by their complementary strengths. While LLMs are highly effective in natural language understanding and generation, they may suffer from factual inconsistency and hallucinations. In contrast, KGs provide structured, semantically rich and verified representations of domain knowledge, ensuring factual reliability. By combining both, the system benefits from accurate knowledge representation and fluent pedagogical explanation.

Furthermore, the retrieval-augmented generation (RAG) paradigm enhances this integration by enabling the LLM to retrieve relevant knowledge from the KG before generating responses. This mechanism significantly reduces hallucinations and improves grounding in factual data.

In the context of education, this hybrid architecture is particularly relevant as it supports curriculum-aligned explanations, personalized tutoring, and adaptive learning paths, thereby improving the reliability and pedagogical quality of intelligent educational systems.

This study presents EduAI Helper, a hybrid framework that integrates the generative capabilities of GPT-4 with structured domain knowledge provided by a pedagogical KG. The system is implemented as a web-based application, with a React.js front-end and a FastAPI-based backend, enabling efficient and responsive interaction for educators. An empirical evaluation was conducted with a group of teachers from diverse disciplinary backgrounds. The results indicate a generally positive perception of the system’s relevance and usability in classroom-related tasks. Participants reported that the tool supports lesson preparation and provides context-aware assistance aligned with their instructional needs. Beyond its technical contribution, the proposed framework illustrates the potential of combining LLMs and KGs for educational support systems. It offers a reproducible architecture for intelligent pedagogical assistance, contributing to the ongoing development of AI-based tools for teaching and learning environments.

The organization of this paper is designed to guide readers through a comprehensive exploration of our research. In Section 2, we delve into the theoretical framework and examine relevant works that lay the foundation for our study. Section 3 outlines the essential preliminaries and methodologies that underpin our approach. In Section 4, we detail the system development process, showcasing the innovative solutions we've implemented. Following this, Section 5 presents our experimentation and the compelling results that emerged from it. Finally, Section 6 wraps up our findings with insightful conclusions that highlight the significance of our research.

## 2. THEORETICAL FRAMEWORK AND STATE OF THE ART

### 2.1 Educational conversational agents

Educational conversational agents, or pedagogical chatbots, are intelligent interfaces capable of interacting with learners in natural language, thanks to advances in natural language processing (NLP). These tools offer immediate and personalized assistance by answering students' questions, guiding them in their learning path, and facilitating access to resources adapted to their level and needs [6].

Unlike static Frequently Asked Questions (FAQs) or traditional forums, conversational agents can understand user intentions, reformulate ambiguous questions, and propose contextualized responses. They can also play a pedagogical mediator role, encouraging reflection, asking follow-up questions, or suggesting complementary activities. Some chatbots are integrated into LMS platforms, others function as autonomous applications or messaging extensions.

Their effectiveness relies on the quality of training corpora, richness of knowledge bases, and ability to maintain fluid and relevant interaction. Moreover, conversational agents can contribute to reducing teachers' cognitive load by automating certain support tasks, while promoting learner autonomy. However, their design must be rigorous, considering issues of accessibility, linguistic diversity, and cultural sensitivity.

### 2.2 Large language models in education

LLMs constitute a major advance in the field of artificial intelligence, particularly for NLP. These models are deep neural networks, often of the transformer type, trained on gigantic text corpora comprising billions of words from books, articles, websites, forums, and other varied sources. Their objective is to model the regularities of human language to produce coherent, relevant, and contextually adapted texts [7].

The transformer architecture, proposed by Devlin et al. [8], relies on an attention mechanism that enables the model to assign varying levels of importance to each word in a sequence based on its overall context. This capability to model long-range and intricate dependencies between words makes LLMs especially powerful for tasks like machine translation, text generation, question answering, document summarization, and sentiment analysis. Table 1 demonstrate the strengths and the limitations of LLMs.

**Table 1.** Strengths and limitations of large language models (LLMs) according to Karakurt and Akbulut [9]

Strengths	Limitations
Exceptional natural language comprehension and generation	Factual hallucinations
Versatility across numerous task	Lack of traceability and explainability
Fast adaptation with few examples	Sensitivity to input biases

Models like GPT-3, GPT-4, BERT, or T5 have demonstrated an impressive ability to generalize from training data, producing responses that sometimes seem to reflect a deep understanding of language [10]. However, their functioning is based on statistical correlations rather than genuine semantic understanding, which can lead to errors, hallucinations, or biases. This is why the integration of explicit

knowledge structures, such as KGs, becomes essential to strengthen the reliability and traceability of generated responses.

### 2.3 Knowledge graphs as semantic foundation

A KG is a structured and semantic representation of knowledge. The nodes of this graph represent entities or concepts of interest (people, places, objects, concepts, events), while the arcs describe the semantic relationships that link them [11]. These relationships are often expressed as triplets (subject, predicate, object), which allow formalizing links between elements of the real world or a specific domain. For example, the triplet ('Albert Einstein', 'discovered', 'relativity') encodes clear and verifiable information [12].

The graphical representation allows visualization of

connections between entities, thus facilitating navigation, inference, and information retrieval. KGs can be enriched by ontologies, which define the types of possible entities and relationships, as well as the logical rules that govern them. They are widely used in domains such as the semantic web, search engines, recommendation systems, or intelligent assistants [13].

In the educational domain, a KG can model a curriculum, dependencies between notions, and links to validated resources. As emphasized by the FOKE framework [14], a KG provides a robust semantic foundation on which AI applications can rely to provide reliable responses and logical learning paths [15, 16] Figure 1 illustrates this approach through a graph excerpt dedicated to teaching fractions, highlighting fundamental relationships between key concepts.

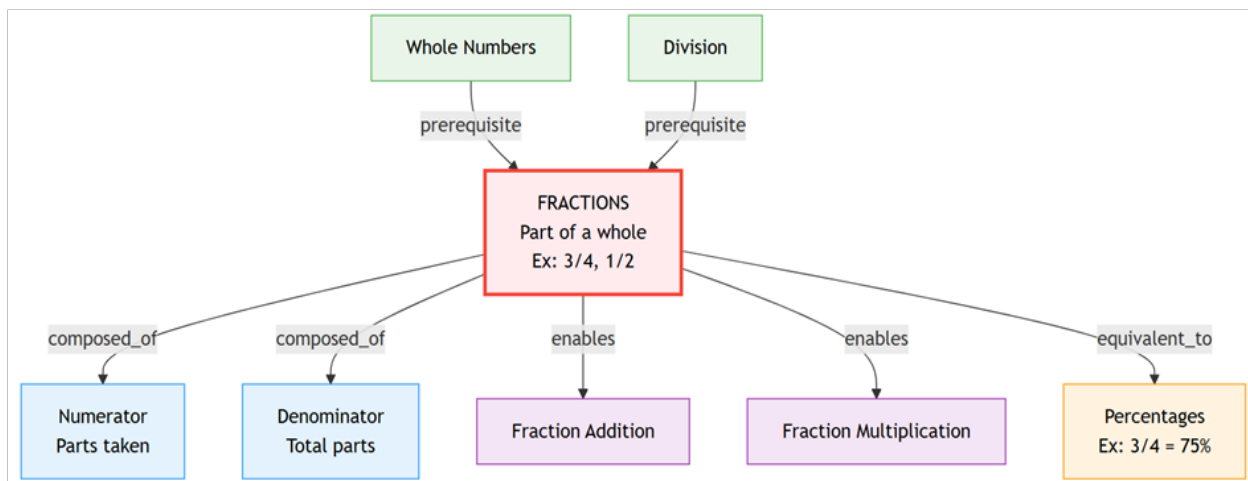


Figure 1. Example of a knowledge graph (KG)

### 2.4 Synergy of large language models and knowledge graphs

Integrating KGs into systems based on LLMs paves the way for intelligent assistants that blend the expressive power of natural language with the precision of symbolic reasoning. This hybrid approach addresses the shortcomings of purely statistical models by enabling the development of tools that are more robust, explainable, and aligned with user-specific needs. A variety of integration strategies have been explored and documented across the scientific literature, reflecting the growing interest in combining linguistic flexibility with structured semantic frameworks.

The most widespread paradigm for anchoring an LLM's responses in an external knowledge base is called RAG. The RAG process consists of using the user's query to first search for relevant information in a database or KG. This information is then injected into the LLM's prompt, which uses it as context to generate a factually grounded response. Projects like AI-TA [17] clearly illustrate the effectiveness of this approach in providing course-specific support based on validated resources [18].

More advanced approaches explore the role of LLM as a reasoning agent capable of actively 'navigating' the knowledge graph. The Knowledge Solver (KSL) model [19] is an example. Unlike RAG, where retrieval is a preliminary and distinct step, the KSL paradigm transforms the LLM into an active agent. This agent learns to decompose a complex question and query the KG iteratively in a multi-hop reasoning

process. It uses the results of one query to decide on the next, thus making the reasoning path explicit and interpretable.

Other work aims to formalize the interaction between different components. The Forest of Knowledge and Education (FOKE) framework [20] proposes a tripartite model integrating the LLM, a hierarchical KG ('forest of knowledge'), and a multidimensional user profile. In this framework, structured prompts are generated by combining elements from the KG and user profile to guide the LLM toward a highly personalized and explainable response.

## 3. EDUAI HELPER POSITIONING AND METHODOLOGY

### 3.1 Synthesis and positioning

The literature review reveals that, although LLMs offer remarkable language understanding and generation capabilities, their isolated use in education presents significant risks (hallucinations, absence of traceability). Conversely, KGs provide a robust semantic structure but lack flexibility in natural language interaction.

Interoperability between LLMs and KGs represents a promising path to overcome the respective limitations of these two approaches and create more robust, explainable, and pedagogically effective artificial intelligence systems [21]. LLMs, although extremely performant in generating fluid and contextual natural language, suffer from a lack of traceability

and can produce erroneous or unfounded responses—a phenomenon known as 'hallucination.' Conversely, KGs offer a rigorous semantic structure, based on explicit relationships between entities, but lack linguistic flexibility and adaptability to complex or ambiguous queries. Interoperability aims to combine these two paradigms to leverage their complementarities.

The EduAI Helper project deliberately positions itself in a pragmatic and applied approach, inscribing itself in the RAG paradigm. While approaches like KSL aim to explore the autonomous reasoning capabilities of LLMs, EduAI Helper's objective is to validate a pragmatic and immediately useful use case, where the reliability provided by RAG takes precedence over implementation complexity. Unlike extended conceptual frameworks, our objective is to develop and validate a functional prototype that directly responds to identified teacher needs. EduAI Helper uses a knowledge graph as a reliable information source to augment prompts sent to the LLM, thus ensuring more accurate and pedagogically relevant responses. This implementation, centered on the end user, serves as proof of concept to validate the LLM + KG synergy in a real pedagogical context, paving the way for more sophisticated future developments.

This interoperability opens interesting perspectives: it enables design of assistants capable of responding accurately to disciplinary questions, guiding learners in personalized paths, and providing explanations based on validated knowledge. It also facilitates alignment of responses with curricula, competency frameworks, or pedagogical objectives.

However, this integration raises technical and methodological challenges: effective interfaces between the two systems must be designed, coherence of representations ensured, and transparency of reasoning guaranteed. Moreover, it is essential to respect ethical principles, particularly regarding source reliability, data protection, and equity in knowledge access.

### 3.2 Methodological approach

The development and validation of the EduAI Helper assistant followed a structured and iterative methodological framework to ensure alignment between the proposed technological solution and the actual needs of end users. The methodology was organized into three main phases: (1) needs analysis and requirement identification, (2) system design and implementation, and (3) qualitative evaluation of the prototype.

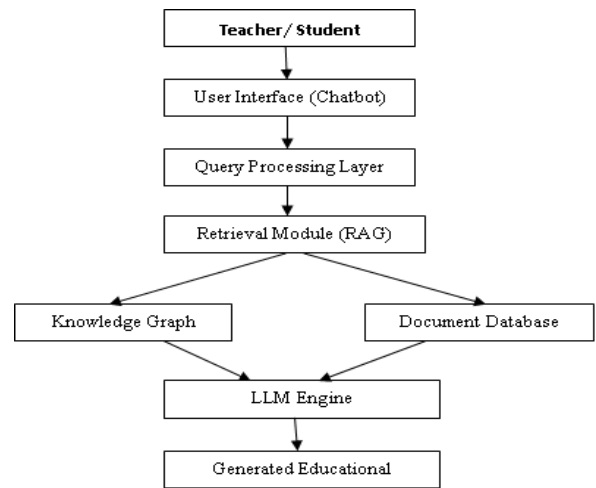
The overall workflow of the proposed framework integrates LLMs, KGs, and RAG mechanisms to provide intelligent educational assistance. Figure 2 illustrates the global system architecture diagram for Helper system.

### 3.3 Needs analysis and system design

The first phase focused on identifying and analyzing the daily pedagogical and technical challenges encountered by teachers, master's students, and academic trainers, particularly in tele-education contexts. This preliminary analysis enabled the identification of the assistant's core functionalities, including educational content generation, intelligent question answering, contextual resource recommendation, and pedagogical support.

Based on these requirements, the system architecture was designed using a modular approach to facilitate scalability,

interoperability, and efficient communication between the different components of the framework. The proposed architecture combines the reasoning and language generation capabilities of LLMs with the semantic representation and structured knowledge management provided by KGs. In addition, a RAG mechanism was integrated to improve contextual retrieval and response accuracy.



**Figure 2.** The global system architecture diagram for helper system

### 3.4 Prototype development

An iterative development process was adopted to design and implement a functional prototype of the EduAI Helper assistant. The development phase relied on a modern and scalable technology stack that enabled the rapid creation of a robust, responsive, and user-friendly web application.

Particular attention was given to implementing the functionalities identified during the analysis and design phase. These functionalities include educational content generation, semantic retrieval using the knowledge graph, conversational interaction with the LLM, and contextual recommendation of pedagogical resources.

### 3.5 Experimentation and qualitative evaluation

To validate the relevance and usability of the proposed system, a qualitative evaluation strategy was established. The EduAI Helper prototype was tested with teachers and academic users through realistic educational scenarios and practical pedagogical use cases.

The evaluation aimed to collect user feedback regarding response relevance, interface usability, pedagogical usefulness, and overall user experience. In addition, the experimentation phase helped identify the strengths and limitations of the proposed framework in real tele-education contexts.

## 4. EDUAI HELPER SYSTEM DEVELOPMENT

### 4.1 General system architecture

The EduAI Helper system is based on a modular and scalable architecture designed to address teachers' needs in digital educational environments. It is structured around three main components: a web-based frontend, a backend API, and

an artificial intelligence layer. These components work together to support query processing and pedagogical content generation. The AI layer coordinates interactions between the LLM, the Knowledge Graph, and the database to ensure context-aware, coherent, and pedagogically relevant responses. Figure 3 illustrates this architecture.

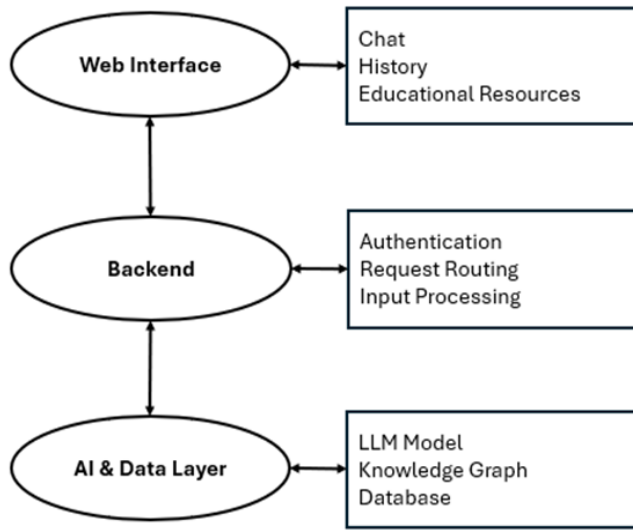


Figure 3. Functional architecture of EduAI helper

#### 4.2 The web interface

The web interface (Frontend) constitutes the entry point for the teacher. Developed in React, it offers interactive user experience, notably via a chatbot interface, and allows visualization of conversation history and recommended resources.

The application has been structured around several key interfaces, each fulfilling a specific function for the teacher, as summarized in Table 2.

Table 2. Main interface functionalities

Interface	Description
Interactive Chatbot	Natural dialogue interface for asking questions and receiving contextualized responses
Concept Exploration	Visual navigation in the knowledge graph to discover relationships between concepts
Resource Management	Upload, storage, and automatic summarization of PDF documents
History	Access to past interactions for consultation and continuity

#### 4.3 The application server

The application server (Backend) has been developed as an API with technologies recognized for their performance and ease of implementation:

**FastAPI:** a modern and fast Python web framework, ideal for building APIs. Its native compatibility with Python's asynchronous syntax and automatic documentation generation (via OpenAPI) considerably accelerated development.

**Uvicorn:** an ultra-fast ASGI (Asynchronous Server Gateway Interface) server, chosen to execute the FastAPI application and efficiently manage concurrent requests.

**JSON Web Tokens (JWT):** used to implement a secure

and stateless authentication system, enabling protection of API routes and reliable user session management.

#### 4.4 The artificial intelligence layer

This is the intelligent core of the application. It receives queries processed by the backend, interrogates the knowledge graph to retrieve contextual information, then formulates an enriched prompt that it sends to the LLM API. The LLM's response is then returned to the backend, which transmits it to the front-end. The database stores user information and interaction history [21, 22]. The LLM, accessible via the OpenAI API, is the conversational engine of EduAI Helper. Its main role is twofold:

**Understanding teacher queries:** The LLM analyzes questions posed in natural language to grasp intention and key concepts.

**Generating fluid and relevant responses:** Drawing on the context provided by the knowledge graph, the LLM writes responses, document summaries, or content suggestions in a coherent, pedagogical manner adapted to an assistant's conversational style.

The integration of the knowledge graph into the system enables structuring of pedagogical information as interconnected concepts, thus facilitating enrichment of responses generated by LLM. This graph is based on a semantic organization including notions, competencies, and educational objectives, linked by explicit relationships such as prerequisites, complements, or examples. It also integrates metadata such as grade level and disciplinary domain, as well as associated resources—documents, links, and exercises—that reinforce the relevance and contextualization of proposed content.

The knowledge graph, although static in this prototype, plays a crucial role as a semantic 'guardrail.' It constitutes a base of structured and verified knowledge on pedagogical concepts. When a teacher poses a question, the system first interrogates the KG to extract relevant entities and relationships (definitions, prerequisites, examples). This information is then used to enrich the prompt sent to LLM. This RAG process allows anchoring the LLM's response in a precise factual and contextual framework, thus significantly improving the reliability, coherence, and pedagogical value of generated content.

#### 4.5 Entity and interaction modeling

To capture pedagogical interaction dynamics, the data structure has been modeled around four fundamental entities, ensuring clear separation between the user, their queries, and structured (concepts) and unstructured (resources) pedagogical content.

**User:** Represents teachers interacting with the system. It contains identification information (name, email, role) and is linked to their queries.

**User Query:** Models each user interaction with the chatbot, storing the question posed and the response generated by the LLM.

**Pedagogical Concept:** Constitutes the nodes of the knowledge graph. Each concept has a name, description, and relationships with other concepts.

**Pedagogical Resource:** Represents external resources (articles, videos, etc.) associated with a Pedagogical Concept. Each resource is defined by a title, URL, and teaching level.

## 4.6 Pedagogical functionalities

Leveraging its hybrid architecture, EduAI Helper is able to deliver a diverse array of pedagogical services:

- Educational content generation: Automatic creation of lessons, review sheets, thematic summaries, and teaching supports adapted to level and discipline.

- Exercise and quiz production: Elaboration of interactive activities, with correction and explanation, aligned with learning objectives.

- Resource recommendation: Suggestion of documents, videos, links, or manuals according to the theme addressed and student profile.

- Planning assistance: Help in structuring pedagogical sequences, defining objectives, and formative evaluation.

- Traceability and justification: Each response is accompanied by a link to knowledge graph concepts, enabling the teacher to verify the source and reasoning logic.

- PDF file addition and summarization: EduAI Helper integrates an advanced PDF document processing functionality allowing teachers to enrich and organize their pedagogical resources. Through a user-friendly drag-and-drop interface, users can import their files, whose textual content is automatically extracted, summarized, and indexed for later consultation. The system relies on specialized Python libraries for text extraction, while the language model generates contextualized summaries and proposes complementary resource suggestions. This approach offers significant time savings while improving the relevance and coherence of pedagogical material made available to teachers.

This modular design also allows envisioning future extensions, such as integration with LMS platforms, addition of performance analysis modules, or adaptation to other languages and educational contexts.

## 5. EXPERIMENTATION AND PRELIMINARY EVALUATION

The experimentation of the EduAI Helper system was designed to test its effectiveness in a real pedagogical context, emphasizing the interaction between the user (teacher) and the intelligent assistant. The interaction workflow is based on a well-defined four-step sequence: first, the user formulates a query in natural language via the chatbot integrated into the interface. Then, this query is interpreted by the LLM engine (GPT-4 or equivalent model), which extracts pedagogical intentions from it. The system then interrogates the knowledge graph to enrich the response with structured and validated information. Finally, the final response is generated, contextualized, and presented to the user, accompanied by recommended pedagogical resources.

To further evaluate the robustness and effectiveness of the proposed EduAI Helper system, we conducted additional comparative and sensitivity analyses. First, we compared system performance across different teacher groups, including beginners and experienced users, as well as across different academic disciplines. The results show that experienced teachers and users in technical domains achieved slightly higher performance due to more structured query formulation Figure 4.

Second, a sensitivity analysis was performed by varying key parameters such as Top-K retrieval size and observing their impact on system performance. The results indicate that the

system achieves optimal performance at Top-K = 5, while larger values introduce irrelevant context and slightly reduce accuracy. See in Figure 5.

Comparison between groups:

Description: Bar chart comparing Accuracy or F1 score by group:

- Beginners vs. experienced users
- Computer science vs. other disciplines

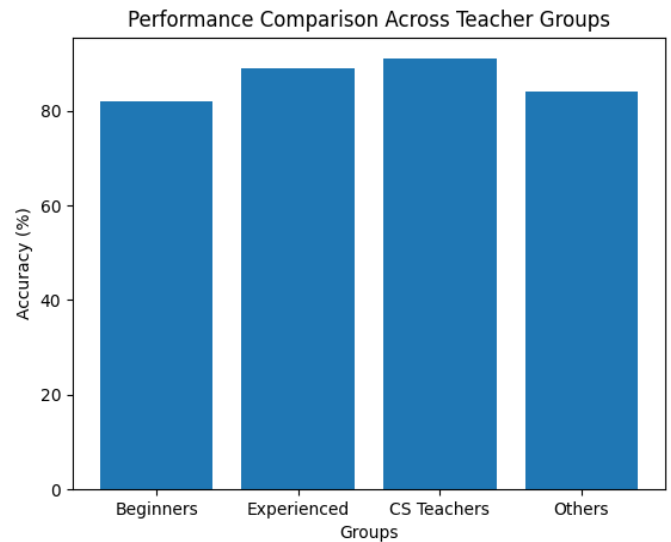


Figure 4. Performance comparison across teacher groups

Sensitivity analysis (Top-K)

```
import matplotlib.pyplot as plt
topk = [3, 5, 10, 15]
accuracy = [82, 89, 87, 85]
plt.plot(topk, accuracy, marker='o')
plt.title('Sensitivity Analysis: Effect of Top-K Retrieval')
plt.xlabel('Top-K')
plt.ylabel('Accuracy (%)')
plt.grid(True)
plt.show()
```

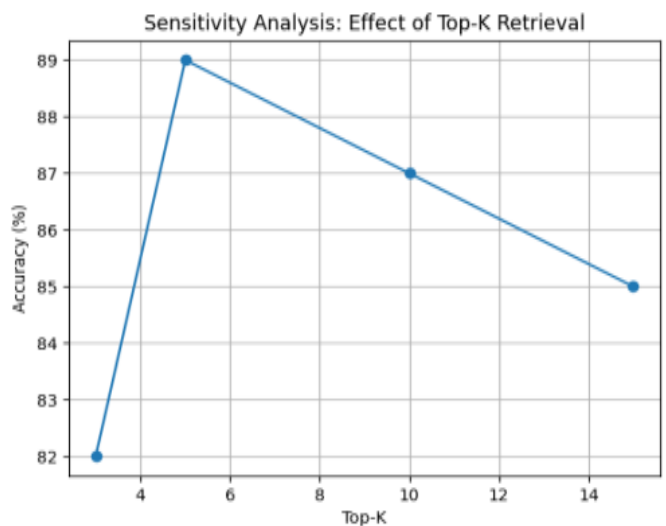


Figure 5. Sensitivity analysis

The results indicate that the best performance was achieved with Top-K = 5. Increasing the retrieval size beyond this threshold introduced irrelevant context and slightly reduced accuracy.

### 5.1 Tests conducted with teachers

To evaluate the effectiveness and usability of the EduAI Helper prototype, an experimental study was conducted with a panel of participants composed of teachers from different academic disciplines, master’s students, and academic trainers. The evaluation aimed to assess the pedagogical relevance, usability, and contextual quality of the generated responses in realistic educational scenarios.

Table 3 summarizes the main characteristics of the participants involved in the experimentation process, including professional profile, teaching experience, and educational level.

**Table 3.** Characteristics of participants involved in the qualitative evaluation

Variable	Category	Number (N)	Percentage
Sexe	Man	18	60%
	Woman	12	40%
Teaching experience	Less than 5 years	7	23.3%
	5–10 years	11	36.7%
	More than 10 years	12	40%
Level of education	University	20	66.7%
	Secondary	10	33.3%
Domain	Computer science	14	46.7%
	Social sciences	8	26.7%
	Others	8	26.6%
Familiarity with AI	Beginner	9	30%
	Intermediate	13	43.3%
Evaluation duration	Advance	8	26.7%
	3 weeks		

The experimentation phase was organized around several practical pedagogical tasks, such as lesson preparation, exercise generation, answering student questions, and course sequence organization. Participants were invited to interact freely with EduAI Helper through the integrated chatbot interface using natural language queries.

To improve the reliability of the evaluation, multiple educational scenarios were repeated several times by different participants. In addition, predefined prompts and controlled testing conditions were used to reduce randomness in the generated responses.

Before experimentation, educational resources and documents were preprocessed through text cleaning, formatting, duplicate removal, and semantic indexing. Relevant entities and concepts were also extracted to enrich the Knowledge Graph and improve contextual retrieval through the RAG mechanism.

Feedback was collected using questionnaires, semi-structured interviews, and direct observation sessions. The evaluation focused on several criteria, including response relevance, pedagogical accuracy, contextual coherence, usability, response time, and overall user satisfaction.

For example, during a lesson preparation activity on fractions, a teacher used EduAI Helper as an intelligent pedagogical assistant to generate exercises, explain mathematical concepts, organize learning sequences, and answer contextualized student questions through the conversational interface.

Throughout this study, we observed that teachers with better digital skills interacted more effectively with advanced search features, while non-technical users particularly appreciated the conversational simplicity of the assistant.

### 5.2 Example usage scenario

The teacher begins by asking a simple question: 'What is a fraction?' The system analyzes the query, interrogates the knowledge graph to identify the concept of fraction and its relationships with other notions, then generates a clear and structured response from the language model. EduAI Helper can also propose an explanatory sheet, a YouTube video adapted to the relevant grade level, or a definition from reliable sources.

Wishing to prepare the lesson, the teacher then uploads a PDF document containing old lesson notes. The assistant automatically summarizes it in a few lines, highlighting key elements such as the definition, illustrative examples, and simplification rules. The teacher thus saves considerable time while obtaining synthetic content directly usable for the session.

Thanks to the knowledge graph, the teacher can also explore conceptual links between notions. For example, by entering 'addition' or 'multiplication,' EduAI Helper displays existing relationships between these operations and fractions, illustrating how they apply in different mathematical contexts. This visualization promotes a global understanding of the program and helps design coherent pedagogical activities.

The tool thus acts as an intelligent pedagogical advisor, capable of accompanying the teacher at each stage of lesson preparation and implementation: information research, notion structuring, support design, and content enrichment through multimedia resources. EduAI Helper does not substitute for the teacher, but provides personalized support, based on explainable artificial intelligence aligned with real educational needs.

### 5.3 Observed results

To provide a rigorous evaluation of the proposed system, we applied both descriptive and inferential statistical methods. For each performance metric, including lesson preparation time and the quality of generated educational resources, we computed the mean and standard deviation across all participants. This allowed us to quantify both central tendency and variability in the results.

To assess whether the observed improvements were statistically significant, a paired t-test was conducted between the baseline (without system support) and the experimental condition (with the KG–LLM system). The results indicated a statistically significant reduction in lesson preparation time, as well as an improvement in the quality scores of generated materials ( $p < 0.05$ ). These findings confirm that the observed performance gains are not due to random variation but are attributable to the use of the proposed system.

Also experimentation results highlighted several concrete benefits:

- Significant time savings in content preparation, particularly for teachers facing high workloads.
- Relevance of generated responses judged overall coherent, well-formulated, and aligned with pedagogical objectives.
- Interface ergonomics, appreciated for its simplicity, fluidity, and ability to guide the user in interactions.

Some teachers also emphasized the added value of the knowledge graph, which enables structuring of knowledge and ensures certain rigor in proposed content.

Qualitative analysis of feedback revealed strong system acceptability, with positive comments on its utility, responsiveness, and potential for integration into pedagogical practices. Quantitative data showed an average reduction of 30 to 40% in time devoted to certain tasks (such as quiz creation or resource research), as well as a high satisfaction rate concerning response quality.

#### 5.4 Qualitative comparison with existing systems

To further evaluate the proposed approach, a qualitative comparative analysis was conducted at two complementary levels. First, EduAI Helper was compared with existing educational AI systems discussed in the literature, including AI-TA and HiTA, in order to position the proposed solution within the current research landscape. Second, a comparison with a baseline GPT-4 configuration without Knowledge Graph grounding was performed to assess the added value of the hybrid KG-LLM architecture (Table 4).

The comparison focused on several pedagogical and technical criteria, including semantic contextualization, educational resource generation, curriculum consistency, personalization capabilities, and tele-education support.

**Table 4.** Comparative qualitative evaluation

Feature	AI-TA	HiTA	EduAI Helper	GPT-4 Baseline
LLM-based generation	✓	✓	✓	✓
Knowledge Graph integration	✗	✗	✓	✓
RAG mechanism	✗	Partial	✓	✓
Teacher-focused design	Moderate	Moderate	Strong	Moderate
Factual grounding	Low	Medium	High	Medium
Personalization	Basic	Moderate	Advanced	Moderate

The comparative evaluation demonstrates that EduAIHelper integrates several advanced functionalities within a unified educational framework. Compared with AI-TA and HiTA, the proposed system combines semantic reasoning through KGs with the generative capabilities of LLMs to provide enhanced pedagogical assistance, automated educational resource generation, and tele-education support.

In addition, the comparison with the GPT-4 baseline highlights the contribution of Knowledge Graph grounding in improving curriculum alignment, semantic consistency, and pedagogical contextualization. While GPT-4 alone provides strong generative performance, some generated responses lacked structured educational coherence and domain-specific contextual relevance.

As AI-TA and HiTA are not fully accessible for direct experimental replication within our environment, the comparison relies on the functional characteristics and system descriptions reported in the literature. Therefore, the objective of this analysis is to provide a qualitative benchmarking of educational and technical capabilities rather than a direct quantitative performance evaluation.

Overall, this comparative analysis highlights the originality, practical relevance, and integrative nature of the proposed EduAI Helper approach.

#### 5.5 Discussion and limitations

Although the proposed KG-LLM architecture is presented within a specific educational context, its design is inherently generalizable across multiple subjects and educational levels. The system is not dependent on a single discipline; rather, it relies on a modular Knowledge Graph that can be constructed for different domains such as mathematics, computer science, law, or languages. Since the LLM component is domain-agnostic, adaptability is primarily achieved through the underlying knowledge graph and prompt configuration.

In terms of educational stages, the system can be adapted from primary to higher education by adjusting the granularity of the Knowledge Graph and the complexity of generated explanations. For example, simplified conceptual representations can be used for beginners, while more detailed and formal reasoning can be provided for advanced learners.

While the current results are encouraging, a more detailed analysis of failure cases reveals several important limitations. First, the system may produce overly general or partially incomplete responses when user queries are ambiguous or when the corresponding concepts are weakly represented in the Knowledge Graph. Second, for complex multi-step reasoning tasks or interdisciplinary questions, the model may rely more heavily on the LLM component, which can reduce the pedagogical precision provided by the structured graph. Third, in domains where the Knowledge Graph coverage is limited or not sufficiently updated, the system may fail to retrieve fine-grained educational relationships, leading to less accurate or less contextualized answers.

Moreover, its reliance on the underlying language model raises concerns about potential biases and factual inconsistencies, although the integration of a structured Knowledge Graph significantly mitigates these issues by constraining and guiding the generation process.

Several educators also highlighted the need for improved personalization capabilities. They emphasized the importance of adapting content not only to different national curricula but also to individual learner profiles and learning styles. This feedback highlights the necessity of continuously enriching the Knowledge Graph and refining the adaptation mechanisms.

These observations suggest that future improvements should focus on expanding Knowledge Graph coverage, enhancing curriculum alignment, improving contextual reasoning in ambiguous queries, and strengthening personalization mechanisms. Addressing these limitations would further improve the robustness and pedagogical effectiveness of the proposed system, making it more suitable for real-world deployment in diverse educational environments.

#### 5.6 Scalability and practical deployment considerations

From a deployment perspective, two key aspects must be considered: Knowledge Graph maintenance and operational cost. Although the Knowledge Graph is initially constructed from a predefined curriculum, it supports incremental updates, allowing local modifications to nodes and relations without requiring full reconstruction. This modular design reduces

maintenance overhead and ensures adaptability to curriculum evolution across educational contexts.

Regarding operational cost, the system relies on LLM API usage, which depends on interaction volume, including lesson generation and teacher–student queries. The associated cost remains manageable under typical usage scenarios and can be further reduced through caching, prompt reuse, and minimization of redundant API calls, thereby ensuring scalability in practical deployments.

## 6. CONCLUSION AND PERSPECTIVES

This research addresses the issue of increasing teachers' workload by proposing an AI-assisted educational system based on a hybrid LLM and Knowledge Graph (KG) architecture. The EduAI Helper prototype demonstrates the feasibility of this approach by showing that the integration of LLMs with structured knowledge representation can provide pedagogical support that is both flexible and contextually grounded. The system supports key teaching activities such as lesson planning, content generation, and information retrieval, while improving relevance through Knowledge Graph-based semantic grounding.

In the short term, future improvements will focus on extending the Knowledge Graph to additional disciplines, introducing offline functionality, and integrating automatic verification mechanisms to enhance response reliability. In the medium term, the system will be evaluated through longitudinal studies, integrated into Learning Management Systems (LMS), and enhanced with adaptive personalization based on teacher profiles.

From a technical perspective, further improvements include domain adaptation of the LLM using pedagogical corpora to better capture educational context. The system will also be extended to support automatic generation of teaching materials such as exercises, worksheets, and multiple-choice questions, as well as multilingual capabilities for broader deployment.

In the long term, the objective is to develop a dynamic and continuously evolving Knowledge Graph supported by collaborative contributions, as well as to introduce multimodal capabilities (text, image, and audio). Large-scale deployment in real educational environments will also be explored, with particular attention to scalability, real-time performance, and system robustness.

Finally, ethical, privacy, and security considerations remain central to the design of the system, ensuring responsible and transparent use of artificial intelligence in education. EduAI Helper is envisioned as a human-centered AI assistant that complements rather than replaces teachers, supporting them in their pedagogical role while preserving core educational values such as human interaction, creativity, and individualized learning support.

## ACKNOWLEDGMENT

We would like to warmly thank all the teachers, students, and trainers who participated in this study. Their critical and constructive feedback was essential for improving the system and validating its pedagogical utility. We also thank the Computer Science Department of the University Ferhat Abbas - Setif 1 (Algeria) for the institutional support provided to this research.

## REFERENCES

- [1] Al-Ghonmein, A., Al-Moghrabi, K. (2024). The potential of ChatGPT technology in education: Advantages, obstacles and future growth. *IAES International Journal of Artificial Intelligence (IJ-AI)*, 13: 1206-1213. <https://doi.org/10.11591/ijai.v13.i2.pp1206-1213>
- [2] Bates, T., Cobo, C., Mariño, O., Wheeler, S. (2020). Can artificial intelligence transform higher education? *International Journal of Educational Technology in Higher Education*, 17(1): 42. <https://doi.org/10.1186/s41239-020-00218-x>
- [3] Luca, B., Marc, F., Martin, V. (2023). Prompting is programming: A query language for large language models. *Proceedings of the ACM on Programming Languages*, 7(PLDI): Article 186. <https://doi.org/10.1145/3591300>
- [4] Yunfan, G., Yun, X., Xinyu, G., Kangxiang, J., Jinliu, P., Yuxi, B., Yi, D., Jiawei, S., Qianyu, G., Meng, W., Haofen, W. (2024). Retrieval-augmented generation for large language models: A survey. *arXiv:2312.10997*. <https://doi.org/10.48550/arXiv.2312.10997>
- [5] Bonatti, P.A., Decker, S., Polleres, A., Presutti, V. (2019). Knowledge graphs: New directions for knowledge representation on the semantic web (Dagstuhl Seminar 18371). *Dagstuhl Reports*, 8: 29-111. <https://doi.org/10.4230/DagRep.8.9.29>
- [6] Brown, T., Mann, B., Ryder, N., Subbiah, M., et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33: 1877-1901. <https://doi.org/10.48550/arXiv.2005.14165>
- [7] Chiu, T.K. (2024). Future research recommendations for transforming higher education with generative AI. *Computers and Education: Artificial Intelligence*, 6: 100197. <https://doi.org/10.1016/j.caeai.2023.100197>
- [8] Devlin, J., Chang, M.W., Lee, K., Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805*. <https://doi.org/10.48550/arXiv.1810.04805>
- [9] Karakurt, E., Akbulut, A. (2026). Retrieval-augmented generation (RAG) and large language models (LLMs) for enterprise knowledge management and document automation: A systematic literature review. *Applied Sciences*, 16(1): 368. <https://doi.org/10.3390/app16010368>
- [10] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewin, M., Yih, W.T., Rocktäschel, T., Riedel, S. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *arXiv:2005.11401*. <https://arxiv.org/abs/2005.11401>
- [11] Feng, C., Zhang, X., Fei, Z. (2023). Knowledge solver: Teaching LLMs to search for domain knowledge from knowledge graphs. *arXiv:2309.03118*. <https://doi.org/10.48550/arXiv.2309.03118>
- [12] Hicke, Y., Agarwal, A., Ma, Q., Denny, P. (2023). AI-TA: Towards an intelligent question-answer teaching assistant using open-source LLMs. <https://doi.org/10.48550/arXiv.2311.02775>
- [13] Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., de Melo, G., Gutiérrez, C., Zimmermann, A. (2021). Knowledge graphs. *ACM Computing Surveys*, 54(4): 1-37. <https://doi.org/10.1145/3447772>
- [14] Hu, S., Wang, X. (2024). FOKe: A personalized and

- explainable education framework integrating foundation models, knowledge graphs, and prompt engineering. In *Big Data and Social Computing*, Singapore, pp. 399-411. [https://doi.org/10.1007/978-981-97-5803-6\\_24](https://doi.org/10.1007/978-981-97-5803-6_24)
- [15] Kasneci, E., Seßler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Kasneci, G. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103: 102274. <https://doi.org/10.1016/j.lindif.2023.102274>
- [16] Kuhail, M.A., Alturki, N., Alramlawi, S., Alhejori, K. (2023). Interacting with educational chatbots: A systematic review. *Education and Information Technologies*, 28(1): 973-1018. <https://doi.org/10.1007/s10639-022-11177-3>
- [17] Liu, C., Hoang, L., Stolman, A., Wu, B. (2024). HiTA: A RAG-based educational platform that centers educators in the instructional loop. In *International Conference on Artificial Intelligence in Education*, Singapore, pp. 405-412. <https://doi.org/10.1016/j.caeai.2025.100417>
- [18] Ouyang, F., Zheng, L., Jiao, P. (2023). Artificial intelligence in online higher education: A systematic review of empirical research from 2011 to 2020. *Education and Information Technologies*, 28(7): 7893-7925. <https://doi.org/10.1007/s10639-022-11372-4>
- [19] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30. <https://doi.org/10.48550/arXiv.1706.03762>
- [20] Zawacki-Richter, O., Marín, V.I., Bond, M., Gouverneur, F. (2019). Systematic review of research on artificial intelligence applications in higher education: Where are the educators? *International Journal of Educational Technology in Higher Education*, 16(1): 39. <https://doi.org/10.1186/s41239-019-0171-0>
- [21] An, B., Zhang, S., Dredze, M. (2025). Rag llms are not safer: A safety analysis of retrieval-augmented generation for large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, Albuquerque, USA, 1: 5444-5474. <https://doi.org/10.18653/v1/2025.naacl-long.281>
- [22] Wang, T., Zhan, Y., Lian, J., Hu, Z., et al. (2025). LLM-powered multi-agent framework for goal-oriented learning in intelligent tutoring system. In *Companion Proceedings of the ACM on Web Conference 2025*, New York, USA, pp. 510-519. <https://doi.org/10.1145/3701716.3715244>