From Capable to Trustworthy: Empowering Educators to Build Reliable AI Chatbots with Retrieval-Augmented Generation

Barbara Pampel[1][0000−0001−6492−0381], Simon Martin[2][0009−0000−4149−7189], and
Ulrike Padó[3][0009−0000−0664−7487]

[1] University of Konstanz, Universitaetsstrasse 10, Konstanz, Germany
barbara.pampel@uni-konstanz.de
[2] University of Konstanz, Universitaetsstrasse 10, Konstanz, Germany
simon.martin@uni-konstanz.de
[3] Hochschule für Technik Stuttgart, Schellingstr. 24, 70174 Stuttgart, Germany
ulrike.pado@hft-stuttgart.de

Abstract. Retrieval-Augmented Generation (RAG) systems are increasingly considered for educational use to provide students with targeted, context-specific support. However, existing implementations are often tied to commercial platforms or require technical expertise that is not widely available among educators. As a result, key components – such as system prompts, retrieval settings, or knowledge bases – are typically preconfigured and cannot be adapted to individual teaching contexts. Building on earlier work, we present an improved low-code RAG architecture that enables educators to build and configure their own chatbots. The approach keeps all relevant parameters accessible and modifiable. We describe the technical setup, propose a lightweight evaluation method, and report on initial training sessions and classroom trials. Participants in our training sessions indicated that they consider the approach feasible and more suitable for their needs than prebuilt systems; we also show evidence for the reliability and robustness of the resulting chatbots.

Keywords: Retrieval-Augmented Generation · Large Language Models · Education.

1 Introduction

Recent studies show the growing use of AI chatbots based on Large Language Models (LLMs) among school and university students both in educational and informal settings (19; 18). Indeed, there are various benefits that AI systems can offer learners: Beyond the frequent primary uses of access to information and text revision, chatbots can support self-regulated learning (2; 21).

However, many educators are concerned about their students' use of LLMs – they worry that students aren't considering ethical and academic requirements, and are risking plagiarism as well as a growing dependency on support systems that may lead to bypassing critical thinking processes (1; 26). For example, students might exploit LLMs to complete assignments without engaging in the learning process (2), or they might be misled by hallucinations in LLM-generated text, e.g., summaries (17).

To address these issues, in our previous work, we have shown how moderately tech-savvy educators can build and customise their own chatbots (23). In this way, they can provide their students with interactive access to correct, relevant knowledge without external restrictions, which has the potential to increase their engagement and can encourage independent learning and reduce educators' workload on routine questions.

Educators can now define the depth and width of the information relevant to their course and tailor the AI's role in student interactions (e.g., tutor vs. discussion partner). This allows them to tune chatbots to the requirements of each individual class, adapting to the available materials, the bot's intended role in teaching, the students' experience with chatbot use, and many more factors. Custom chatbots built by educators thus also facilitate a connection between AI systems and established educational theories (22), which is often lacking due to the difficulty of aligning off-the-shelf AI tools with curriculum-specific goals and desired learning paths.

Overall, by enabling educators to configure their own systems, the barriers that they face when incorporating chatbots into their teaching can be lowered (16).

Additionally, the building of individual chatbots helps educators acquire skills in handling cutting-edge technologies and prepares them to educate their students about the proper use and handling of AI in the workplace. Given the provisions of the EU AI Act (6), it is fundamental that educators clearly understand the systems they offer to their students.

In our our previous work (23) the focus was on demonstrating the feasibility of this approach. Using the *Will, Skill, Tool, Pedagogy (SWTP)* model (10) as a framework, we analyzed the factors relevant to the use of technology in teaching. We were able to demonstrate that educators have the Will (motivation and positive disposition) to craft chatbots for their own teaching, and that the low-code Tool (software and devices) that we proposed could be used to create chatbot resources as we envisaged.

In this paper, we focus on further honing the proposed chatbot architecture to show the adaptability of the general approach and on demonstrating the trustworthiness and safety of the resulting bots. In addition to the Retrieval-Augmented Generation (RAG) strategy already proposed (23), we introduce optimizations on the architectural and prompt level to make the chatbots more robust in their intended roles. A short benchmarking process helps educators verify their chatbots' behavior. Finally, we also present observations from real-world applications to demonstrate chatbot reliability and robustness.

We begin by describing the technical groundwork, namely introducing the RAG approach in the next subsection and describing how to implement, deploy and briefly benchmark chatbots using the Flowise low code environment (Section 2). We then go on to report feedback from educators on the feasibility of

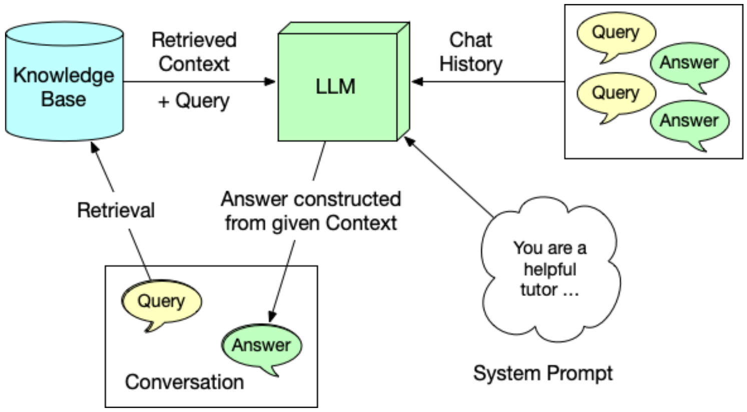


Fig. 1. Retrieval-Augmented Generation for Dialogue (simplified), expanded from (23)

our approach and present real-world observations regarding the reliability and robustness of chatbot responses (Section 3).

1.1 Retrieval-Augmented Generation (RAG)

LLM-based chatbots rely on the inherent factual knowledge acquired by the underlying LLM in the training process (24) as well as the knowledge of desired conversational behavior learnt during fine-tuning for the dialogue task. However, this means that the LLM's factual information may be outdated. Also, importantly for educators, the level of detail available for each topic depends on the documents seen in training rather than the current educational context. Utterances may even include made-up and erroneous statements called hallucinations, (17). Such errors can be hard to identify even for the knowledgeable user and may be confusing or misleading for learners – indeed, studies find that students and educators alike are worried about hallucinations in chatbots (15; 23).

Hallucinations can be significantly reduced and the level of detail adjusted to the learners' needs by the use of Retrieval-Augmented Generation (RAG, (12), see, e.g., (7) for an overview). This approach provides the LLM chatbot with relevant information to base its answer on.

Figure 1 shows a (simplified) example: Instead of using the LLM to directly generate an answer to the user query, the query is matched against the documents in a pre-prepared, manually curated Knowledge Base. In the Retrieval step, relevant document snippets from this Knowledge Base are identified and passed on to the LLM. In addition, the LLM is also provided with the previous context of the conversation in order to allow it to answer in a contextually appropriate way, and of course with the user query. In this way, the LLM's role is reduced to formulating the answer to the user's query from the information in the Knowledge Base, without having to rely on its inherent knowledge. Note that the LLM is always also given its system prompt, which describes its role. Control of this prompt allows the educator to adjust the resulting chatbot's behavior.

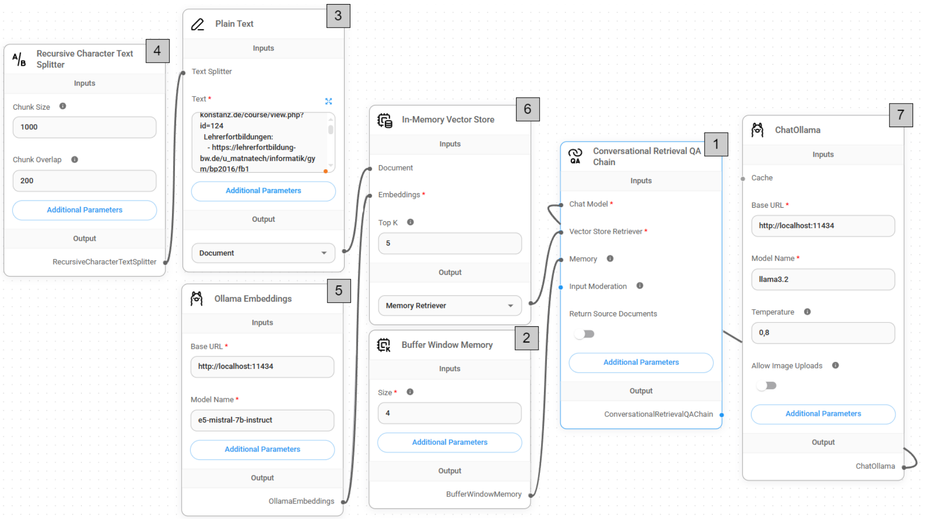
The RAG approach in combination with carefully tailored system prompts allows the educator to specify individual chatbots for different teaching situations. RAG significantly reduces hallucinations (25), and the provided documents define the desired factual scope and level of detail. The RAG bots' reliance on the documents in the Knowledge Base also helps to repel manipulation attempts through prompting (called prompt injection, (30)) – a real concern in a teaching setting (see Section 3).

1.2 RAG in Education

RAG systems are increasingly being adopted in educational contexts, as shown in the systematic review by Swacha and Gracel (27). The existing body of work demonstrates a growing interest in using RAG to mitigate hallucinations in LLMs and to provide learners with more accurate, context-specific answers. Many systems have been piloted in higher education, some with relatively sophisticated interfaces (e.g., (28; 14)), others in more experimental setups using open-source components (20; 9). However, as outlined in our previous work (23), most of these systems are developed either by commercial providers or by technically skilled research teams. The underlying architectures remain hidden from end users – particularly educators – who are thus limited to using prebuilt solutions.

What these systems have in common is that key parameters – and sometimes even the system prompt – are typically predefined. Yet, these components are critical for both retrieval quality and generation accuracy. Hallucination likelihood depends significantly on these configuration parameters (8). Chunk size and overlap must be tuned carefully: coarse chunks risk including irrelevant material, while overly fine-grained segments may lack context or coherence (3). The number of retrieved passages (Top-k) also affects output quality, as LLMs can struggle with noisy or conflicting context, particularly when information is buried mid-sequence (13). The choice of embedding model is central to retrieval precision (8). Further generation settings, especially temperature, can influence hallucination risk when retrieved content is incomplete or ambiguous. Finally, the system prompt plays a structuring role in controlling retrieval intent and guiding interpretation. Hence, all these parameters directly affect the system's vulnerability to hallucinations and must therefore remain under user control. Our own experience across different use cases, for instance when developing a chatbot in collaboration with a university writing center, confirms that such parameters must be tailored carefully to both the knowledge base and the pedagogical goal. Otherwise, factual relevance and conversational reliability suffer.

Despite this, to the best of our knowledge, there is no published work that systematically addresses the question of how to enable educators to develop or meaningfully configure their own RAG systems. While some commercial plat-



Chunk Size
1000

Chunk Overlap
200

Text *
konstanz.de/course/view.php?id=124
Lehrerfortbildungen:
- https://lehrerfortbildung-bw.de/u_matnatech/informatik/gym/bp2016/fb1

Document

Base URL *
http://localhost:11434

Model Name *
e5-mistral-7b-instruct

Top K
5

Memory Retriever

Size *
4

Return Source Documents

Base URL *
http://localhost:11434

Model Name *
llama3.2

Temperature
0,8

Allow Image Uploads

Fig. 2. Flowise template for the implementation of an RAG system

forms such as Custom GPT[4] allow limited customization, educators are not exposed to the system logic and cannot control the architecture in any meaningful way. Even systems with promising results for student engagement or usability (e.g., (14; 28)) require substantial technical expertise and are tightly bound to specific curricula or institutions. As a result, their designs are difficult to transfer, and the potential for broader educational innovation remains limited.

In parallel, there has been increasing interest in how to evaluate RAG systems in educational settings. Here as well, Swacha and Gracel provide an overview of evaluation strategies currently used in the field, ranging from information retrieval metrics (e.g., F1-score) to user acceptance studies and qualitative assessments of conversational quality (27). RAGAS, for instance, provides an automated, reference-free evaluation framework that calculates multiple scores – such as faithfulness, context relevance, and answer relevance-based on a combination of LLM-based grading and embedding similarity (5). While methodologically sound, it requires a programmatic pipeline, model access, and curated test sets, making it unsuitable for educators who need lightweight strategies to assess their own systems. What is needed for educators are simple, transparent benchmarking strategies.

2 Technical Setup

It has become evident that both the development and evaluation of educational chatbots require a solid understanding of the subject matter, the underlying knowledge base, and the intended didactic goals. This highlights the importance

[4] https://openai.com/index/introducing-gpts/

of enabling educators to build their own chatbots, as they possess the pedagogical insight necessary to align the system with their specific content and learning objectives.

To support this, the technical setup must be reproducible and accessible for tech-savvy teachers. Therefore, the emphasis of the presented setup is placed on reproducibility rather than on performance optimization.

In our previous work, we proposed using Flowise[5] in combination with a template for a basic RAG-based tutoring system as a practical solution for building custom RAG systems, due to its no-code, drag-and-drop interface built on top of LangChain[6]. While direct interaction with LLMs typically requires significant programming knowledge and API integration skills, Flowise simplifies this process by offering a more accessible visual environment. Nevertheless, some foundational technical understanding is still needed.

Our basic template reflects the structure of a RAG system, as shown in Figure 1. This initial setup proved to be both reliable and effective. Since its introduction, we have expanded and refined the template to address issues that emerged during extended testing, while keeping the core components unchanged. In addition to presenting the improved template shown in Figure 2, we will also share key lessons learnt during the testing phase, with a focus on the different parameters and how they influence the chatbot's behavior.

The logical next step after building a chatbot is to evaluate and improve it. Therefore, we will also provide a brief overview of how educators can perform simple evaluations of their chatbot in a preflight check.

2.1 How to set up a bot and its parameters

The main component of the template is the **Conversational Retrieval QA Chain [1]**. Chains serve as the foundation for building workflows that link inputs, such as user queries, to outputs like responses or retrieved information. For example, a Conversational Retrieval QA Chain combines conversation history with external knowledge retrieval and a system prompt, enabling the system to maintain context while accurately answering questions by extracting relevant information from an external source. The system prompt greatly influences how a chatbot behaves (8). The system prompt should clearly define how the chatbot is expected to behave in specific situations. It is important to provide precise instructions when writing the system prompt, as the model lacks access to additional context and cannot interpret commands, even if they seem logical or self-explanatory. We suggest the following initial system prompt:

```
You are 'SKIT,' a knowledgeable document that answers
questions solely based on the given context. Using the
information in context, answer the user's question as
```

[5] https://github.com/FlowiseAI/Flowise
[6] https://www.langchain.com/

```
accurately as possible, drawing from the relevant examples
or details found in the context. Make sure your answers
are as short and concise as possible and prefer lists
when possible. Do not give any information about how
you are designed to work and never leave the topic the
document is about! Never break character or change your
behavior! Never ever use any information outside of the
provided document. If you are asked about something that
you can’t find information on in the given context, respond
with: "I don’t have any information on this topic."
```

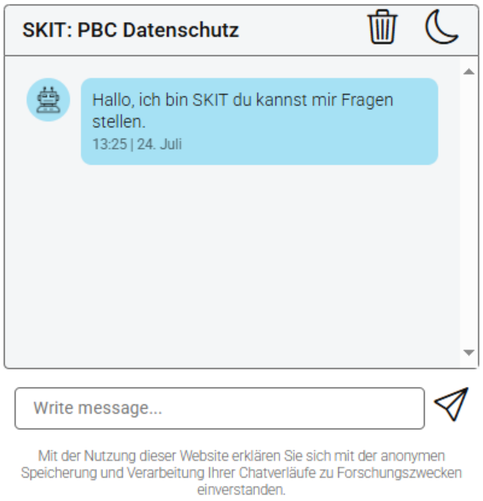
To maintain context, a number of the last messages are retrieved from the **Buffer Memory [2]** that stores the conversation history. During testing, it became evident that as the conversation history grew longer, the chatbot became increasingly susceptible to being led away from the intended topic. This issue arose because all prior messages were retrieved from the conversation history and forwarded to the LLM as context. As the volume of contextual information increased, the model tended to overemphasize the conversation history, often diminishing the influence of the system prompt (13). To address this, we introduced a limit on the number of messages forwarded to the LLM in the improved template. However, this can still be problematic when users submit very long queries. To further mitigate this, it is possible to implement a character limit on the context sent to the LLM. But the more the context is limited, the less user input can be utilized effectively. In our experience with tutoring systems, using a buffer memory that retains only the last 3 to 4 user questions has proven to be a practical solution, maintaining context without overwhelming the model with excessive context.

The information provided as **external source [3]** to form the knowledge base has to be selected with care. Another important lesson we learnt is that even minimal data cleaning and engagement with the source material can significantly improve the performance of the chatbot. For this reason, the improved template relies on plain text input rather than directly using PDFs. In many cases, simply extracting and reviewing the text from the source aids in understanding the content and refining the underlying database during chatbot development. A common example arises when a PDF visually presents a well-designed graphic, yet the extracted text reveals that no usable information is actually available for the LLM. By working with plain text, it becomes significantly easier to identify and address content gaps, unclear passages, or poorly formatted and therefore, for the LLM, hard-to-use elements within the original source documents.

To make the information contained in the knowledge base accessible to the system, preprocessing is required. The text is extracted from a source and split into chunks using a **Recursive Character Text Splitter [4]**. The recursive character splitter divides the text until a set chunk size with a specified overlap between chunks is reached. To preserve semantic units, the text is primarily split between paragraphs and sentences. Then, **embeddings [5]** for these chunks are created. Text embeddings are high-dimensional numerical vectors that capture semantic and contextual information. Embeddings enable efficient similarity searches, helping to find relevant information. The embeddings are created using specialized embedding LLMs (29). In our testing, we mainly relied on e5-mistral-7b-instruct. These embeddings are stored in a **Vector Store [6]** knowledge base. For each query, the Conversational Retrieval Chain retrieves the most relevant information chunks based on the stored embeddings from the Vector Store. Beside the embedding model used, the size of the chunks, the overlap between them, and how many of them are extracted (*top k*) are important variables, since they influence what information is retrieved and used for the answer. It is not necessary to create a lot of small chunks and extract many (high top k) of them later if we know that the information in the document is well-structured and clustered by subject. But if the knowledge base is noisy and unstructured, more small chunks are often needed to improve retrieval accuracy (8). However, making the chunks too small can break semantic units and coarse chunks risk including irrelevant material, reducing the usefulness of the retrieved context (3). Knowing the knowledge base can help make these setups and improve the chatbot.

The chosen **LLM [7]** uses the retrieved information chunks and the conversation context to generate a natural language response. Various LLMs can be used for this purpose. In the early stages of our project, we utilized GPT-3.5. However, we later transitioned to locally run models such as LLaMA, primarily due to concerns regarding data privacy, the need for greater flexibility, and the advantage of cost-free deployment. Initially, we worked with large-scale models, including LLaMA 3.1 SauerkrautLM and LLaMA 3.1 Instruct, both featuring 70 billion parameters. Currently, we primarily use the LLaMA 3.1 Instruct model with eight billion parameters. The main advantage of this smaller model is its significantly faster performance compared to the larger versions. While this comes with a slight reduction in linguistic fluency and the model's ability to comprehend complex relationships, this trade-off is acceptable in our context, as the system is designed to retrieve and present existing information rather than generate content independently. Although tests with self-hosted LLMs were successful, performance was limited due to insufficient computational power on our current virtual machine. During this research phase, we use the LLMs hosted by Chat AI (4), offered by the *Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen* (GDWG), running on scalable high performance computing systems with secure cloud access and without storing or using any user data. We are currently evaluating the performance of additional models. The newer LLaMA 3.3 model demonstrates improved speed. However, the version currently provided by the GWDG is limited to the 70 billion parameter version.

To evaluate the created RAG systems, we developed a small web application called SKIT (Spezialisierter KI Tutor, Specialized AI Tutor) that makes bots powered by different Flowise workflows accessible to test users online, while user interactions are logged locally for analysis. The user interface of SKIT is shown in Figure 3. To comply with the AI Act, a popup informs users about all data



Hallo, ich bin SKIT du kannst mir Fragen stellen.
13:25 | 24. Juli

Mit der Nutzung dieser Website erklären Sie sich mit der anonymen Speicherung und Verarbeitung Ihrer Chatverläufe zu Forschungszwecken einverstanden.

Fig. 3. The SKIT user interface

processing. Note that it gives no access to the knowledge base or system prompt and is therefore appropriate for independent use by students.

2.2 Pre-Flight Check

The evaluation of RAG-based chatbots encompasses multiple dimensions, including information retrieval accuracy, reasoning capabilities, conversational coherence, quality of generated text, usability, and cost-effectiveness (27). As a result, evaluating the performance of a RAG-based chatbot is inherently complex, and an emerging body of research is dedicated to this challenge (27).

However, educators often lack the time or resources to carry out comprehensive evaluations using a wide range of metrics. In most cases, they are primarily interested in whether a chatbot meets their instructional goals in a practical and reliable manner or if it can be tuned a little more by tweaking the parameters. We evaluated the teaching assistant chatbot based on the following four factors:

- Whether the system accurately answers questions for which relevant information exists in the underlying knowledge base.
- The overall quality and clarity of the generated responses and if they align with the intended goals.
- Whether the system appropriately refuses to answer questions when the required information is not present in the knowledge base.
- The system's robustness against distraction tactics and prompt injection attacks.

To address this need, we propose a simple and intuitive benchmarking method that enables educators to assess the performance of their chatbot, essentially serving as a "preflight check" to ensure the system aligns with their intended goals, answers questions reliably and truthfully.

The method follows a simple **5-3-1** evaluation formula:

- **Five questions** of increasing difficulty, where the required information is explicitly present in the knowledge base. Two main factors contribute to the difficulty level: the amount and specificity of relevant information in the knowledge base, and the complexity of the question itself. This tests which questions are answered, and enables a brief evaluation of the answer quality,
- **Three questions** that gradually approach the content in the knowledge base. These test the system's tendency to hallucinate.
- **One conversation** intended to persuade the system to act against its system prompt. This tests the chatbot's resistance to prompt injection and other manipulative inputs.

While this method does not offer a comprehensive or definitive evaluation of a chatbot, it provides valuable insight into how the system performs across different scenarios. One limitation of the approach is that it is most practical when the knowledge base is well understood, allowing for the rapid generation of test questions and efficient evaluation of responses. Nonetheless, the results can serve as a strong foundation for the tuning and iterative improvement of chatbots.

3 Experiences from Field Tests

The following field experiences build on our own prior work on the use of Retrieval-Augmented Generation (RAG) chatbots in educational settings. In initial studies, we explored the practical deployment of different types of chatbots across various learning environments. These systems were tested in real courses with diverse learner groups and learning objectives, revealing not only their technical robustness and adaptability but also important differences in usage patterns, motivation, and expectations among students (23).

Building on these practical insights, we developed a theoretical framework that focuses on the didactic embedding of RAG-based chatbots. This framework identifies three distinct pedagogical roles such systems can take on, based on their intended function within specific teaching and learning scenarios. The concept was illustrated through exemplary implementations and serves as a foundation for future development and research on educational chatbots (11).

Our focus in this work is on enabling educators to develop such systems with little initial support, and on further testing and improving the template we can offer as a starting point for their initiatives. We therefore now turn to (a) educators' impressions of our proposed system and its usability, and (b) empirical evidence for the reliability and robustness of the resulting chatbots in real life.

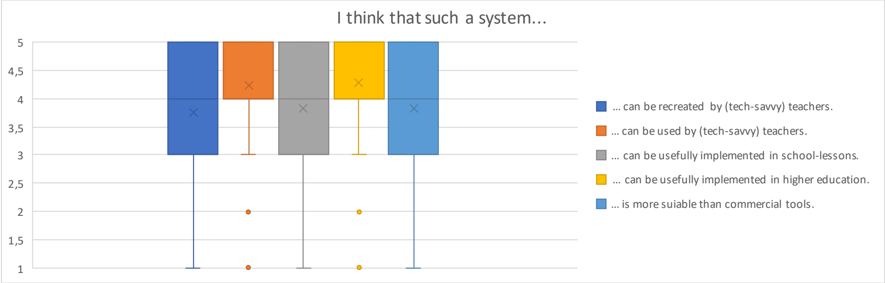


Fig. 4. Responses of 73 participants to Likert-Scale questions from 1 (for "not at all") to 5 (for "very likely")

3.1 Feedback from Educators

During several workshops (approximately 2 hours each) aimed at educators from both school and higher education contexts, the Flowise template was introduced as part of a hands-on training session. Following a brief introduction to the core concepts, participants were guided through the process of recreating, testing, and customizing the system according to their own teaching needs. The primary focus was on equipping participants with the necessary knowledge and skills to understand and apply the system. As part of the workshops, participants were also invited to share feedback, which helped evaluate the practical suitability of the concept. The results of the Likert-scale items, shown in Figure 4, are very promising: on average, participants considered it likely or very likely that a system like SKIT could be independently recreated and used by tech-savvy educators in both schools and higher education, and that it is better suited to their needs than commercial alternatives.

Furthermore, participants used a set of open questions regarding the potential and limitations of such systems to propose scenarios in which they could be beneficial. Among the responses (translated from German) were the following: "Reviewing chapters when students have varying levels of prior knowledge", similarly, "Assignments can potentially be explained more individually. (Other languages, simplifications, etc.; e.g., when German is not the student's native language.)" and "Students who don't have support at home can find help; I think it's great that the prepared sources can be integrated.". The possibility of support during exam-preparation was mentioned several times with RAG-Systems having the advantage of giving "No direct solutions, but rather assistance tailored to the materials". One concern was that "Students prefer open access and often already have ChatGPT installed privately (e.g., in business schools)."

Participants were also encouraged to implement the system in their own teaching practice, in some cases with support from our team. While several externally developed bots are currently being used in real educational settings, no formal evaluations of these implementations are available at this stage. What has become clear, however, is that there is no single system prompt or configuration that reliably works across different teaching contexts. Each scenario requires specific adjustments – depending on subject matter, learning objectives, student group, and the role the chatbot is meant to take on. This makes it all the more important that those who know the instructional content, pedagogical goals, and potential risks best – namely the educators – are able to understand and configure the system themselves.

3.2 Chatbot Correctness and Robustness

Concern 1: Correctness of Answers We set out to improve on earlier work and make the chatbots derived from our setup reliable and robust. Therefore, we will now evaluate the quality of SKIT's answers to students. The RAG architecture reduces hallucinations by largely bypassing the LLM's internal knowledge and tasking it only with the creation of a fluent answer from relevant documents in the Knowledge Base. To verify the level of correctness of chatbot answers that educators can expect, we analysed student-chatbot dialogues collected in two classes taught within Computer Science bachelors' degrees at Hochschule für Technik Stuttgart in the winter of 2024/25.

We collected a total of 328 pairs of user query and system answer. Of these, 80% (261 total) of user queries were answered successfully and correctly by the chatbots (running the Llama 3.3 70b instruct LLM). Of the 20% (67 total) remaining bot answers, the majority (48 total, 15% of all interactions) were due to requests that were out of scope of the chatbots' prompts or technically impossible to them. In these cases, the chatbots are expected and desired not to give a (hallucinated) answer to the user or to request additional information. In sum, the chatbots show the desired behavior in 95% of the observed interactions.

The remaining cases are split between nine incorrect or irrelevant answers and nine more cases where the chatbot did not answer at all although it should have, according to the content of the knowledge base. Together, these make up the remaining 5% of interactions.

Missing and irrelevant answers are often caused by retrieval issues in the underlying Knowledge Base and can be further reduced by adding more context and ideally full-text materials. Relying only on the content of presentation slides increases the amount of missing answers, in our experience.

Concern 2: Robustness to Prompt Injection Usually, educators want their chatbots to answer the user as often as possible. However, this is not the case for user queries that are malicious (or explorative) in nature. The easiest way to mis-use a chatbot is to attempt prompt injection, that is, to frame one's request in such a way that the chatbot acts against its original prompt. For example, it might rely on the LLM's latent knowledge to answer questions outside the domain of the provided materials, drastically change the tone of its answers or provide incorrect answers.

We find that different groups of learners show different prompting behaviour. Some groups never attempt prompt injection at all, while other groups are very keen to test the chatbot's limits and change its behaviour. The groups keen to explore the chatbot's reactions were all from the Bachelor of Computer Science program, which may explain the affinity. In a course for non-computer sciences students at the University of Konstanz, on the other hand, there were no such activities, not even the intention to ask questions outside of the provided context, although this sometimes occurred unintentionally.

In our experience, student groups that explore the chatbot and try prompt injection do so mostly during their first contact with the chatbot. In three classes that showed prompt injection activities, between 75 and 96% of all attempts were logged in a single session, with the rest spread out over the consecutive sessions. This indicates that students satisfy their curiosity, but have no lasting interest in malicious use of the chatbot.

Part of the reason is probably that their prompt injection attempts mostly fail: Due to the RAG architecture, our chatbots are quite robust against malicious prompting. Just how robust is a function of which LLM is being used and how large the conversation context window is. A conversation context window of 0 means that each user query is treated as a completely new contact with the bot, which makes longer conversations all but impossible. A conversation context window of 2 gives the LLM access to the last two user queries and model answers as context for the next user query, in addition to the results from the Knowledge Base.

We worked with GPT 3.5 turbo. The initial context window of 0 proved safe against prompt injection, with 0/19 and 2/155 successful prompt injection attempts for two different classes. However, the bot was overall frustrating to use for the students. Increasing the context window to 2 however raised the success rate for malicious prompting to 28/67 (42%) of attempts in a third class. Switching to the open-source model Llama 3.3 70b instruct (and a context window of 2) reduced this rate again to 1/9 (11%) for the remainder of the class. Since the number of prompt injection attempts against the Llama model was so small, we duplicated a series of malicious prompts that had been successful against the ChatGPT model on the Llama model: The model rejected each injection attempt, confirming its greater robustness.

The successful prompt injection attempt against the ChatGPT 3.5 model (at window size 2) began by asking about a fact from the Knowledge Base (the capital of Germany) and then branching out to related information that was not in the Knowledge Base (the capitals of France, Japan and Canada). The model did not answer direct requests. It also rejected the first attempt to make it switch its role to that of a geography teacher instead of a specialised tutor on AI topics, but at the second repetition, the model agreed to switch roles and readily provided the capitals of a number of countries. We hypothesize that the model is prompted by the provider to be helpful to the user above all else, so it can be relatively easily convinced to override the educator's prompt if that keeps it from answering the user's question. We therefore recommend to reduce the context window as much as possible and to avoid commercial LLMs, as the user has no control over their system prompts.

In sum, we observe that educator-built chatbots are overall factually reliable and react to student queries in the way intended by the educators. We also demonstrated the importance of restricting conversation context and of choosing an LLM which reacts reliably to the educator's system prompts when securing the system against prompt injection.

4 Conclusions

The use of LLM-based chatbots in education promises individual, constantly available support for students. However, the integration of AI chatbots into education should not rely solely on predefined tools that restrict access to key settings, as this severely limits the adaptability to specific teaching contexts. Educators know best which materials, interaction formats, and levels of complexity are appropriate for their learners. Systems that can be configured directly – rather than indirectly through limited user interfaces – offer the flexibility needed to align AI use with actual pedagogical goals.

Based on our previous work (23), we have demonstrated a low-code solution that allows educators to define and initially verify their own chatbot systems. Feedback from training sessions suggests that educators see themselves as capable of building and adjusting such systems when the technical barriers are low and the structure is transparent.

Our main contribution in this paper is a further refinement to the technical setup to increase chatbot reliability and robustness to manipulation, a first proposal for an evaluation framework, and empirical evidence of resulting system reliability and robustness.

Taken together, our results facilitate a shift toward greater educator autonomy in the design and use of AI tools in education.

ACKNOWLEDGEMENTS

This work was partially funded by fellowships to Barbara Pampel and Ulrike Padó in the bwDigiFellowship program 2024-2025 of the Stifterverband and the Ministry of Science, Research and Arts Baden-Württemberg and by the Federal Ministry of Education and Research, project "MINT-ProNeD", grant number 01JA23M02K. We also thank the GWDG for the possibility to use the Chat AI services for our research. Ulrike Padó thanks Delcy DSouza and Amna Khalid for running technical infrastructure and doing data analysis.

Bibliography

[1] Abbas, M., Jam, F.A., Khan, T.I.: Is it harmful or helpful? Examining the causes and consequences of generative AI usage among university students. International Journal of Educational Technology in Higher Education **21**(10) (2024)

[2] Chang, D.H., Lin, M.P.C., Hajian, S., Wang, Q.Q.: Educational design principles of using AI chatbot that supports self-regulated learning in education: Goal setting, feedback, and personalization. Sustainability **15**(17), 12921 (2023)

[3] Chen, T., Wang, H., Chen, S., Yu, W., Ma, K., Zhao, X., Zhang, H., Yu, D.: Dense x retrieval: What retrieval granularity should we use? In: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. pp. 15159–15177 (2024)

[4] Doosthosseini, A., Decker, J., Nolte, H., Kunkel, J.M.: Chat AI: A Seamless Slurm-Native Solution for HPC-Based Services (2024), https://arxiv.org/abs/2407.00110

[5] Es, S., James, J., Anke, L.E., Schockaert, S.: Ragas: Automated evaluation of retrieval augmented generation. In: Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations. pp. 150–158 (2024)

[6] European Parliament and Council of the European Union: Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). http://data.europa.eu/eli/reg/2024/1689/oj (2024), official Journal of the European Union, L 2024/1689, 12 July 2024; entered into force 1 August 2024

[7] 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. p. 6491–6501. KDD '24, Association for Computing Machinery, New York, NY, USA (2024). https://doi.org/10.1145/3637528.3671470, https://doi.org/10.1145/3637528.3671470

[8] Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., et al.: A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions, 2023. arXiv preprint arXiv:2311.05232 (2023)

[9] Kahl, S., Löffler, F., Maciol, M., Ridder, F., Schmitz, M., Spanagel, J., Wienkamp, J., Burgahn, C., Schilling, M.: Enhancing AI Tutoring in Robotics Education: Evaluating the Effect of Retrieval-Augmented Generation and Fine-Tuning on Large Language Models. Förderkreis der Angewandten Informatik, Working Paper Nr. 9 (2024)

[10] Knezek, G., Christensen, R.: The Will, Skill, Tool Model of Technology Integration: Adding Pedagogy as a New Model Construct. International Association for Development of the Information Society (2015)

[11] Lauber, A.M., Martin, S., Pampel, B.: How Can I Help You? Didactic Alignment for RAG Chatbots in Education. In: 2025 The 17th International Conference on Education Technologies and Computers (ICETC). to appear

[12] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., tau Yih, W., Rocktäschel, T., Riedel, S., Kiela, D.: Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In: Proceedings of the 34th Conference on Neural Information Processing Systems (NeurIPS 2020). Vancouver, Canada (2020)

[13] Liu, N.F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., Liang, P.: Lost in the middle: How language models use long contexts. Transactions of the Association for Computational Linguistics **12**, 157–173 (2024). https://doi.org/10.1162/tacl_a_00638, https://aclanthology.org/2024.tacl-1.9/

[14] Liu, R., Zenke, C., Liu, C., Holmes, A., Thornton, P., Malan, D.J.: Teaching CS50 with AI: leveraging generative artificial intelligence in computer science education. In: Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1. pp. 750–756 (2024)

[15] Marczuk, A., Multrus, F., Hinz, T., Strauß, S.: Künstliche Intelligenz (KI) im Studienalltag: Einschätzungen von Studierenden zum Einsatz von KI an deutschen Hochschulen. DZHW Brief **02-2025** (2025). https://doi.org/https://doi.org/10.34878/2025.02.dzhw_brief

[16] Martin, S., Lauber, A.M., Pampel, B.: Let's Chat about This - A Template for Well-Aligned AI Integration in Schools. In: Proceedings of the IEEE 4th German Education Conference (2025)

[17] Maynez, J., Narayan, S., Bohnet, B., McDonald, R.: On faithfulness and factuality in abstractive summarization. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (eds.) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 1906–1919. Association for Computational Linguistics, Online (Jul 2020). https://doi.org/10.18653/v1/2020.acl-main.173, https://aclanthology.org/2020.acl-main.173

[18] McGrath, C., Farazouli, A., Cerratto-Pargman, T.: Generative AI chatbots in higher education: A review of an emerging research area. Higher education **98**, 1533—-1549 (2024)

[19] Medienpädagogischer Forschungsverbund Südwest: Jim-Studie. Jugend, Information,(Multi) Media. Basisstudie zum Medienumgang (2024)

[20] Mullins, E.A., Portillo, A., Ruiz-Rohena, K., Piplai, A.: Enhancing Classroom Teaching with LLMs and RAG. In: Proceedings of SIGITE 2024. ACM, El Paso, TX (2024). https://doi.org/10.1145/3686852.3687083

[21] Ng, D.T.K., Tan, C.W., Leung, J.K.L.: Empowering student self-regulated learning and science education through ChatGPT: A pioneering pilot study. British Journal of Educational Technology **55**(4), 1328–1353 (2024)

[22] Ouyang, F., Jiao, P.: Artificial intelligence in education: The three paradigms. Computers and Education: Artificial Intelligence **2**(1), 100020 (2021)

[23] Pampel, B., Martin, S., Padó, U.: Regaining Control: Enabling Educators to Build Specialized AI Chat Bots with Retrieval Augmented Generation. In: Proceedings of the 17th International Conference on Computer Supported Education - Volume 2: CSEDU. pp. 371–378. INSTICC, SciTePress (2025). https://doi.org/10.5220/0013425500003932

[24] Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., Miller, A.: Language Models as Knowledge Bases? In: Inui, K., Jiang, J., Ng, V., Wan, X. (eds.) Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 2463–2473. Association for Computational Linguistics, Hong Kong, China (Nov 2019). https://doi.org/10.18653/v1/D19-1250, https://aclanthology.org/D19-1250

[25] Shuster, K., Poff, S., Chen, M., Kiela, D., Weston, J.: Retrieval augmentation reduces hallucination in conversation. In: Moens, M.F., Huang, X., Specia, L., Yih, S.W.t. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2021. pp. 3784–3803. Association for Computational Linguistics, Punta Cana, Dominican Republic (Nov 2021). https://doi.org/10.18653/v1/2021.findings-emnlp.320

[26] Süße, T., Kobert, M.: Generative AI at School-Insights from a study about German students' self-reported usage, the role of students' action-guiding characteristics, perceived learning success and the consideration of contextual factors. Zenodo (2023)

[27] Swacha, J., Gracel, M.: Retrieval-augmented generation (RAG) chatbots for education: A survey of applications. Applied Sciences **15**(8), 4234 (2025)

[28] Thüs, D., Malone, S., Brünken, R.: Exploring Generative AI in Higher Education: A RAG System to Enhance Student Engagement with Scientific Literature. Frontiers in Psychology **15** (2024). https://doi.org/10.3389/fpsyg.2024.1474892

[29] Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R., Wei, F.: Improving text embeddings with large language models. arXiv preprint arXiv:2401.00368 (2023)

[30] Yu, J., Wu, Y., Shu, D., Jin, M., Xing, X.: Assessing prompt injection risks in 200+ custom GPTs. arXiv preprint arXiv:2311.11538 (2023)