

Hobert, Sebastian

## Chat with your Lecture Recording : Creating easy-to-use Chatbots for Learning Videos

In:

Schmid, Ute; Leidner, Jochen L.; Kohlhase, Michael; Wolter, Diedrich (Eds.), Proceedings of the Second Workshop on Artificial Intelligence for Artificial Intelligence Education (AI4AI Learning 2024), Bamberg: University of Bamberg Press, p. 2-22. 2025. DOI: 10.20378/irb-107661

### Bookpart - Published Version

DOI of the Article: 10.20378/irb-108884

Date of Publication: 07.07.2025

### Legal Notice:

This work is protected by copyright and/or the indication of a licence. You are free to use this work in any way permitted by the copyright and/or the licence that applies to your usage. For other uses, you must obtain permission from the rights-holder(s).

This document is made available under the **Creative Commons License CC BY**.



This licence information is available online:  
<https://creativecommons.org/licenses/by/4.0/>

# Chat with your Lecture Recording – Creating easy-to-use Chatbots for Learning Videos

Sebastian Hobert 

TH Lübeck

sebastian.hobert@th-luebeck.de

## Abstract

Educational chatbots represent a promising technology for improving learning. Prior research studies revealed the benefits of using chatbots in educational settings, e.g., to enable students to engage with learning content. However, the technology is not widely used in teaching practice. In response to this gap between scientific findings and usage of the technology in practice, we aim to enable lecturers to create their own customized chatbots based on educational videos by developing a chatbot creation tool that can create tailored chatbots (semi-)automatically. Following a design science research approach, we iteratively designed a software prototype that can convert lecture recordings or similar learning videos into chatbots. The specialty of the prototype is that it is designed to make the chatbot technology accessible to lecturers by reducing technical barriers. With the help of the prototype, it is possible to adapt the chatbot to the lecturer's teaching content without the need for a time-consuming, manual creation of a knowledge base. Our evaluation with experienced lecturers, instructional designers, and developers of educational technology indicate positive results. In this paper, we present the results of our development and evaluation process.

**Keywords** Chatbot, Technology-Enhanced-Learning, Video-based Learning, Large Language Model, Education

## Introduction

Chatbots have been used and researched in educational settings for years (see, e.g., Hobert & Meyer von Wolff, 2019; Winkler & Söllner, 2018; Wollny et al., 2021). Many different use cases have been analyzed in the past. Use cases for chatbots in educational settings are, for instance, supporting students in language learning (Huang et al., 2022), supporting students in learning programming (Hobert, 2023), or improving video-based teaching by integrating chat interactions (Winkler et al., 2020).

When integrating chatbots into online learning settings, the engagement of users with the learning content typically changes. Typical passive interactions with learning content like reading textual learning modules or watching videos can be extended by conversations with a chatbot. This has been found to be supporting for the learning process, e.g., for engagement, learning, and self-efficacy (Chang et al., 2022).

Although positive effects of well-designed chatbots on the learning process are to be expected, and many use cases have already been researched, no widespread use of chatbots to support learning can be identified in teaching practice. Grounded in our experience of integrating educational chatbots into multiple field settings in the past (e.g., Hobert, 2023; Hobert & Berens, 2024), we consider various reasons to be particularly important: (1) Teachers need easy-to-use tools to create their own chatbots. (2) Creating sufficient materials for training a chatbot manually is time-exhaustive. (3) Available chatbot tools are often aimed at technology-affine users.

The emergence of large language models (LLMs) seems to be a catalyst for educational chatbots. Available tools make it easy even for less technology-affine users to create customized chatbots for various settings. However, a prerequisite for this is the availability of textual content (such as slides, books, or scripts) specifically aligned to the course. In teaching practice in formal learning scenarios, however, it can be experienced that although the available learning materials (often slides) provide a basic building block, much additional knowledge is conveyed orally in face-to-face lectures, video conferences, or on the spoken track of recorded

learning videos. If teachers are required to reproduce all of the knowledge conveyed orally in full in writing, this, in turn, involves a great deal of additional effort.

Based on this problem and with the aim of easily integrating chatbots into our own AI courses, we aim to make chatbots easy to create and easy to customize based on lecture recordings. To this end, we follow a design science research approach (Hevner, 2007; Hevner et al., 2004) to design and evaluate an educational technology tool that allows lecturers to create customized chatbots based on learning videos. The underlying idea is that lecturers do not have to spend any additional time creating written teaching materials but can simply use a recording of their lecture as a basis. Thus, we aim to answer the following research question: How to enable lecturers to create their own customized chatbots based on educational videos? To answer this research question, we build on our experiences from various previous chatbot studies. In contrast to these prior studies (e.g., Hobert, 2019, 2023; Hobert et al., 2023; Hobert & Berens, 2024; Winkler et al., 2020), in which we spent a lot of time and effort to manually create knowledge bases, the further development in this project is to automate the creation of a knowledge base using the technology of LLMs. In doing so, we want to take a small but relevant step towards the further dissemination of chatbots in educational settings.

## **Theoretical Background**

Educational chatbots can be defined as software programs that interact with learners using natural language (Følstad & Brandtzaeg, 2020; Hobert & Meyer von Wolff, 2019) in educational settings (like lectures or online courses). By providing a chat-based user interface, educational chatbots aim to mimic human-like interactions (Følstad & Brandtzaeg, 2020; Weizenbaum, 1966), e.g., about learning content or organizational aspects of learning.

In recent years, educational chatbots received substantial research interest, which is reflected in the emergence of several literature reviews (e.g., Hobert & Meyer von Wolff, 2019; Winkler & Söllner, 2018; Wollny et al.,

2021). In prior research studies, many purposes and use cases have been researched (like language learning (Huang et al., 2022), programming (Hobert, 2023), enhancing video-based learning (Winkler et al., 2020), or corporate training (Hobert et al., 2023)). Summarizing the results of prior research shortly, it can be concluded that educational chatbots seem to be able to foster students' learning in many different use cases.

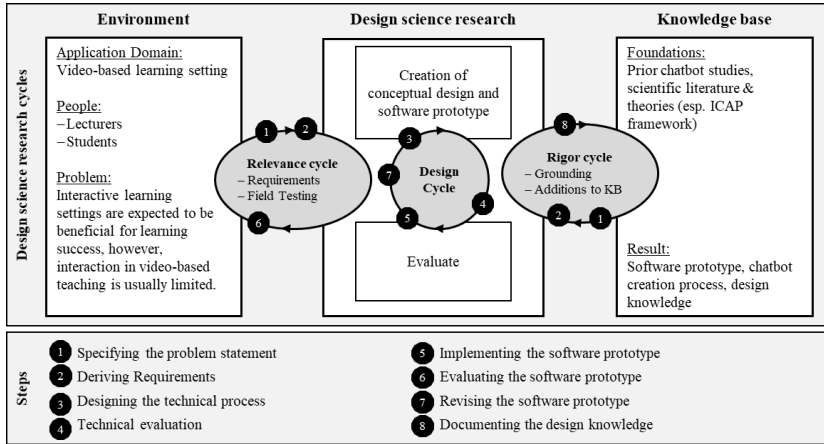
From a theoretical point of view, the benefit of including chatbots in educational settings for supporting learning processes can, for instance, be explained with the ICAP framework (Chi & Wylie, 2014). According to the ICAP framework, it is to be expected that learning success can be fostered by increasing learning engagement (Chi & Wylie, 2014). Purely text-based or video-based online learning formats can often be regarded as rather passive. This can also apply to (large) lecture formats with limited interaction. According to the framework, however, interactive formats, such as discussions, are considered to be beneficial for successful learning processes. This is where educational chatbots can support the learning process, as they can discuss and interact with users (even in purely online formats) using natural language. Therefore, it is to be expected that more interactive learning settings can be created with the help of educational chatbots.

Building on the promising prior research results and the theoretical grounding in the ICAP framework (Chi & Wylie, 2014), this research project aims to support lecturers in integrating educational chatbots in their educational settings to improve students' learning.

## Research Design

In this project, we follow a design science research-oriented research design. We specifically base our methodology on the three-cycle design science research framework (Hevner, 2007; Hevner et al., 2004). To develop our software prototype and the underlying technical process to create chatbots (semi-)automatically from learning videos, we first ground our design process on our own prior research studies on chatbots in the educational sector (e.g., Hobert, 2019, 2023; Hobert et al., 2023; Hobert &

Berens, 2024; Winkler et al., 2020) and related learning theories (i.e., the ICAP framework (Chi & Wylie, 2014)). Additionally, we ground our research results on the application context (i.e., the environment) and the expertise of experienced educators.



**Fig. 1.** Design science research process (adapted from (Hevner, 2007; Hevner et al., 2004)).

As displayed in **Fig. 1**, the research process consists of eight consecutive steps. First, we derive the underlying research problem from the educational practice (relevance cycle) as well as from prior research studies and learning theory (rigor cycle). Based on this, we derive software requirements and design the technical process to (semi-)automatically create a chatbot from an educational video. In the fourth step, we technically evaluate the capabilities of the technical process before we implement a web-based software prototype in the fifth step. In an evaluation study with experienced lecturers, instructional designers, and developers of educational technology, we evaluate the software prototype to identify suggestions for improvements. Finally, we revise the prototype and document the design knowledge in this paper.

## Design and Evaluation

### Specifying the Problem Statement

As motivated in the introduction section, we aim to improve the interaction of students with learning content by enabling lecturers to create educational chatbots easily. In typical video-based learning settings, students watch learning videos (e.g., lecture recordings, micro learning videos, or recordings of hands-on tutorials) in a rather passive way. Interacting with learning materials in a passive way is, however, expected to be less successful in terms of fostering learning success compared to active, constructive, or even interactive learning settings, according to the ICAP framework (Chi & Wylie, 2014).

Fostering the interactivity in asynchronous video-based learning is, however, not easy as students typically watch videos not in a formal learning environment (e.g., at home for the exam preparation). One possibility to increase the interactivity is the integration of chatbots (i.e., pedagogical conversational agents). For instance, in a prior research study it could be shown that learning success in a video-based learning setting can be fostered when specifically designed chatbots are integrated to interact with students about the video's learning content (Winkler et al., 2020).

Integrating chatbots in a similar way as it was done in Winkler et al., (2020) is, however, challenging for lecturers as it requires a lot of additional (manual) work. This can particularly be explained by the required design process of a tailored chatbot. So far, as in Winkler et al., (2020), the learning materials must be specifically prepared to be accessible by a chatbot. In a learning video and specifically in a lecture recording, the learning contents are typically not fully available in written form and can, thus, not easily be integrated into common chatbot platforms. To address this problem, we aim to extract not only the learning content from lecture slides but also the lecturer's voice-based instructions from learning videos. Using this extracted information, we aim to automatically create a knowledge base that can be used by an LLM-based chatbot to answer students' questions. In doing so, we are focusing on a content level in which a chatbot

is provided that can answer content-related questions about the given learning video (e.g. “What is the definition of term XY?”, “Please explain term XY again in different words”, “Please summarize the most important aspects of the video.”).

## **Deriving Requirements**

As outlined in the introduction section and based on the problem statement, the software to be designed should be able to create chatbots in an easy-to-use way even by less technology-affine lecturers based on educational videos. To this end, we derived requirements as the basis for the subsequent implementation process (see **Table 1**). First, the software to be designed needs suited functionalities to select and upload a learning video (R1). Ideally, the lecturer should not need to care about specific video codecs or formats as this would hinder less technology-affine lecturers. After uploading the video, a (semi-)automated process should start to create the chatbot’s knowledge base and initiate the chatbot. To this end, the learning content of the uploaded video needs to be extracted (R2). This should include the learning content from slides as well as the lecturer’s voice-based instructions. Based on the extracted learning content, a suitable knowledge base should be created automatically (R3) that can be used by state-of-the-art chatbot technology (i.e., LLMs) as an input to provide chatting functionalities using an easy-to-use user interface for students (R4). The whole creation process should be provided in an easy-to-use user interface guiding lecturers step-by-step through the creation process without needing specific technical knowledge (R5).

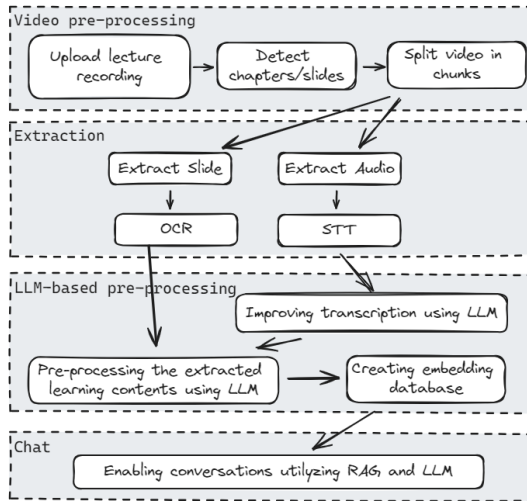
**Table 1.** Overview of the core requirements for the software prototype.

#	Description
R1	Provide functionalities to select and upload arbitrary learning videos
R2	Provide functionalities to automatically extract the learning content from a learning video (including the content from slides and the lecturer’s voice)
R3	Provision of functionalities for the automatic creation of a suitable knowledge base based on the automatically extracted learning content
R4	Provide chatting functionalities using an easy-to-use user interface for students
R5	Provide an easy-to-use user interface guiding lecturers through the creation process without needing specific technical knowledge

## Designing the Technical Process

Based on the derived problem statement and the corresponding requirements, we designed the technical process visualized in **Fig. 2**. We divided the chatbot creation process into four subprocesses: (1) video pre-processing, (2) extraction, (3) LLM-based pre-processing, and (4) chat.

At the beginning of the video pre-processing, the video must first be uploaded. After completing the upload, the video can be analyzed to detect the different slides. To do this, the video stream is analyzed and it is detected when the visual content changes substantially. This is interpreted as a slide transition and is used for splitting the video into chunks. After splitting the video into smaller chunks (aligned with the slides), the first subprocess terminates. In the second subprocess, the learning content is extracted from the video chunks. For each video chunk, the textual content of the slide is extracted using optical character recognition (OCR). Additionally, the audio of the video chunk is extracted and transcribed using speech-to-text (STT).



**Fig. 2.** Derived process to automatically create a chatbot from educational videos

Using the extracted textual information from OCR and STT as the input, the LLM-based pre-processing starts. First, using an LLM the STT-based textual extraction of the speaker’s voice is improved. To this end, the LLM is prompted to “correct only spelling mistakes and grammatical errors”. This step seems particularly helpful as automatic transcriptions using STT are – according to our experience – often prone to grammar and spelling issues. The reason for this is that lecture recordings are often recorded spontaneously in the lecture hall without following a detailed manuscript. The spoken language, therefore, often contains mistakes that do not typically occur in written learning materials. Using an LLM for improving automatic transcriptions works well – at least for our tests during this research project. The improved transcriptions, as well as the extracted texts from the slides (OCR), are then further processed. Creating summaries of all slides and the entire lecture video and extracting key points seem to be particularly helpful. To this end, we prompt the LLM to create such summaries for the extracted content. For instance, we prompt the LLM to “summarize the text in one sentence” or to “list the most

important key points”. These summaries and key points are then used to create an embedding database together with the extracted and improved content of the OCR and STT steps.

Finally, a chatbot can be provided using retrieval augmented generation (see, e.g., [aws.amazon.com](https://aws.amazon.com), 2024). With this method, the chatbot is then able to respond to the user’s questions. To this end, the user first need to ask the chatbot a question. This question is then passed to the embeddings database to conduct a similarity search. The most relevant results from the similarity search are then passed to the LLM, which is promoted to “answer the question briefly in two sentences” and to “use the following information from [the similarity search]” as a basis. This has the effect that the chatbot’s response depends to a large extent on the results of the similarity search and the answer generation by the LLM. Since the processed content of the learning video is stored in the embedding database (see previous steps), the generation of the response is aligned to the content of the learning video.

Based on the technical design of the process, we implemented it as a collection of multiple small Python and bash scripts using common open-source tools and libraries (like pytesseract (Github, 2024c) for OCR, whisper (Github, 2024a) for STT, chroma ([trychroma.com](https://trychroma.com), 2024) for the embedding database, and ffmpeg ([ffmpeg.org](https://ffmpeg.org), 2024) for analyzing and splitting the video). This collection of scripts was used as the input for the technical evaluation and allowed the designed technical process to be fully tested.

## Technical Evaluation

Before implementing a full user interface to execute the designed technical process, we aimed to test the technical process manually in a first technical evaluation. To this end, we selected ten different learning videos from our own lectures and ran the full process manually. Afterward, we manually tested the quality of chat conversations.

To generate exemplary chat messages, we use exemplary user messages retrieved from prior chatbot studies that we conducted in previous years (e.g., (Hobert, 2023; Hobert & Berens, 2024)). For instance, we tested the following messages: *Please define the term \$topic. What is meant by \$topic? What is the video about? On which slide can I find more information about \$topic?*

Whereas questions about getting a summary or about defining specific terms mentioned in the videos resulted in good-quality answers, getting information about where to find specific information (e.g., on which slide) was less successful. The reason for this was that we did not provide the slide number as an information to the LLM. After adding this information to the database, the answer quality increased, and the LLM was capable of responding to these types of questions as well. After getting a sufficiently high answer quality with our ten exemplary learning videos, we decided to continue with the implementation of a fully functional software prototype.

## Implementing the Software Prototype

To enable lecturers – even without specific technical knowledge – to create chatbots for their lecture recordings easily, we implemented a web-based software prototype using Python’s Flask framework (Github, 2024b). To design an easy-to-use user interface, we rely on the MIT-licensed Tabler UI kit (The Tabler Authors, 2024).

To implement the technical process, we refined our script collection implemented during the design of the technical process and integrated it into our web-based user interface.

As visualized in **Fig. 3**, the resulting user interface guides the lecturers step-by-step through the creation process. The procedure consists of one introductory welcome screen (step #1), six intermediate steps (#2 – #7) for video pre-processing, extraction, and LLM-based pre-processing, and finally, the user may directly interact with the resulting chatbot while being able to watch the video at the same time (#8).

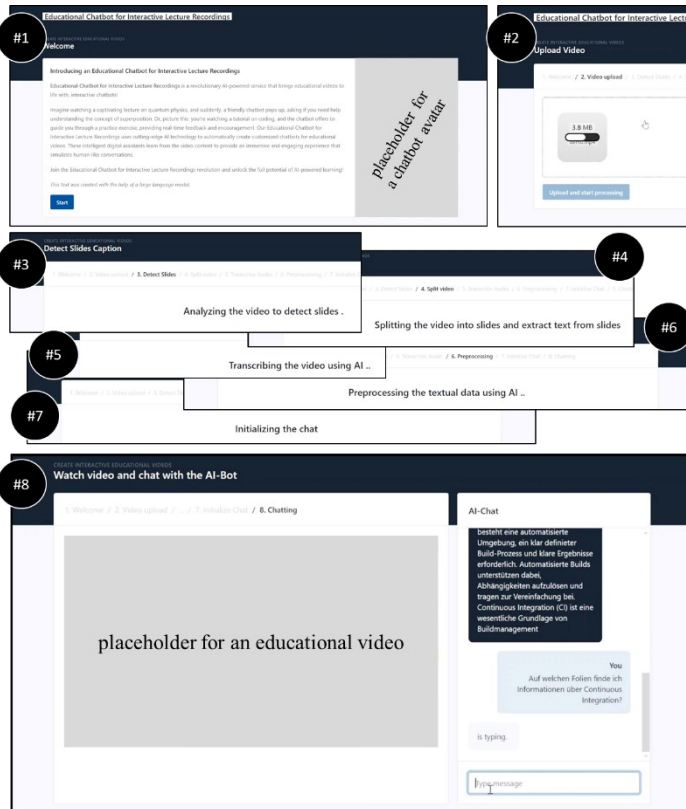


Fig. 3. Overview of the implemented software prototype

## Evaluating the Software Prototype

To evaluate the software prototype, we created an online survey and distributed it among experienced lecturers, instructional designers, and developers of educational technology. Hereby, we chose a mixed-method evaluation, as similar approaches have been shown to be suited for evaluating and developing chatbot-based system in prior studies (e.g., Hobert, 2019; Hobert & Berens, 2024).

Within the online evaluation, the participants first received a written introduction to the overall context of the research project, including further

information on the scientific background. Afterward, we asked the participants to self-evaluate their AI literacy using the Meta AI Literacy Scale (Carolus et al., 2023). We included the measurement of AI literacy to ensure that the participants had sufficient prior knowledge of AI. In the next step, the usage scenario was introduced in an executive summary and the software prototype with an exemplary learning video was demonstrated in a video screencast. Demonstrating the prototype in a screencast ensured that all participants had the chance to get insights into all steps of the prototype. After this demonstration step, the participants were asked to evaluate the software prototype using the user experience questionnaire (UEQ) scale (Laugwitz et al., 2008). Finally, we asked the participants for qualitative feedback to give us additional insights on how to improve the prototype. Hereby, we were particularly interested in aspects of the prototype that were considered to be useful and valuable, as well as possible suggestions for improvement.

### **Quantitative Feedback on the User Experience**

The participants evaluated the user experience of the developed software prototype positively in all six sub-scales of the UEQ (Laugwitz et al., 2008). According to the benchmark included in the UEQ Data Analysis Tools (The UEQ Team, 2024), the software prototype was rated as excellent in terms of attractiveness, perspicuity, efficiency, simulation, and novelty. Only in the scale dependability, it receives lower but still positive ratings. This can maybe be explained by the nature of the creation process. During the semi-automated process, most steps are automatically executed by the system, and the results are dependent on the results of the large language model. This can maybe result in a feeling that the next steps are less predictable compared to traditional, not AI-supported, systems. Overall, the results of the user experience evaluation do not indicate any major problems.

## Qualitative Feedback

To analyze the qualitative feedback collected as part of the evaluation procedure, we aggregated and categorized the mentioned aspects. We translated selected quotes into English to include them in this paper.

### 1. *Positive Aspects*

First, we analyzed the aspects of the prototype that were particularly named as useful and valuable. Related to this question, the participants named 33 aspects, which we aggregated. Many aspects were mentioned by several participants, so a total of four major aspects emerged after we finished our qualitative analysis.

The simplicity of the usage of the creation process of the chatbot was named by many of the participants. For instance, the participants highlighted the simplicity of the “automatic creation of the chatbot” and the “simple usage and process”.

Closely related to simplicity is the intuitive user interface. Hereby, the participants stated that “the individual steps are displayed clearly”, and “the look and feel correspond to the familiar interface of chats”. The qualitative statements related to the user interface are in line with the quantitative feedback on the user experience.

The possibility to interact with the chatbot about the specific learning content of the uploaded video was named by multiple participants as a major benefit. For instance, one participant highlighted “the ability to interact with the chatbot, e.g., to give instructions and ask questions”. The natural language conversation with the chatbot (“interaction using free text in the user’s natural language”) was also highlighted as a benefit.

Finally, the (semi-)automatic processing of the video and its content was evaluated as particularly valuable. The participants mentioned this aspect the most. For instance, participants highlighted that “the slides are automatically recognized from the video and [that] the text is automatically

extracted”, “the chatbot can specify the slides on which the answer is based on when asked”, “the ability to extract and analyze both written and spoken language”. These quotes highlight once again the usefulness of the implemented process to (semi-)automatically generate the chatbot based on the video.

## 2. *Suggestions for improvement*

The participants made 39 suggestions for improvement. Many of them were named multiple times. After aggregating them, we resulted in four aspects related to the creation process of the chatbot and two aspects related to the learners’ interaction with the chatbot.

First, multiple participants requested the possibility to enable uploads for further documents in addition to the learning video. For instance, “additional textual lecture materials or literature sources” could be uploaded. One participant also suggested including the software artifact directly into a learning management system and enabling the import of the content of the entire course into the video.

In addition, multiple participants asked whether it would be possible to provide a multilingual chatbot that is capable to converse, for instance, in either German or English. This form of internationalization would enable non-native speakers to interact more easily with the learning content.

To make the creation process of the chatbot more transparent, it was suggested to provide “more detailed explanations in the intermediate steps” while the chatbot is created. This suggestion is somehow contradicting to the suggestion to “automate the next buttons for the intermediate steps”, which was stated by another participant.

One participant requested to offer better possibilities for quality assurance. For instance, she or he suggested offering the possibility to read the automatic transcription from the video. This seems to be an interesting suggestion at first glance, but it is – according to our opinion –

questionable whether this really would improve the quality of the output as the transcription are only one building block for the generation of the chatbot's answer using the LLM.

In addition to these suggestions for improvement targeting the creation process, two further aspects were suggested related to the learners' interaction with the chatbot.

First, it was requested by several participants to include the possibility to jump directly to the slide when the chatbot mentioned a specific slide: "It would be good if I [...] can jump directly to the point in the video where the information is mentioned". This would enable learners to watch the specific section of the video once again.

Finally, some participants asked to proactively offer learners the possibility to get a summary or the key message of the video. This could for instance be included in the starting point of the conversation.

According to our assessment, the suggestions for improvement seem to be helpful and might contribute to further improve our developed software artifact.

Nevertheless, we also want to disclose one mostly critical aspect mentioned by one participant who responded that "I can also just take the slides and search for terms using the search function. That would be more efficient on the one hand and would also show me 1:1 what is relevant without additional information." This statement seems controversial but misses – according to our estimation of the benefits of chatbots in educational settings – that chatbots allow learners a different way of interacting with the learning content using natural language conversations.

## **Revising the Software Prototype**

Based on the suggestions for improvements revealed in the qualitative feedback, we revised our software implementation. In particular, we revised the prototype in four areas. (1) We enabled the possibility to upload not only one video during the creation process but users are now allowed

to also upload further text documents at the same time. These text documents are processed similarly to the video without executing the transcription and OCR steps. This enables lecturers to integrate additional materials like handouts or speaker notes. (2) We added more explanatory information on the processing steps and also automated the next buttons at the same time (by moving to the next step after the processing steps has finished either when the user clicks the next button or after a timer of 10 seconds expired). (3) For learners, we revised the initial welcome message of the chatbot. We integrated a button to switch to English in the conversation. Additionally, the chatbot proactively offers the possibility to show a short summary of the learning video. (4) We added links to the chatbot's answer every time a specific slide number is referenced. By clicking on the slide number, the video jumps directly to the corresponding slide, and the user may replay the video again at this point.

Overall, we believe that the implemented suggestions for improvement make sense and that the revision improved the prototype. We believe that the implemented improvements do not require another evaluation with lecturers at this time. Instead, we suggest that the next step would be to test automatically generated chatbots in real learning settings with learners. This is subject to future research. At this point in time, we are ending this development process as part of our research project at this point.

## **Discussion and Conclusion**

In this design science research project, we aimed to enable lecturers to create their own customized chatbots based on educational videos. To achieve this, we derived requirements, designed the underlying technical process, and developed a software prototype in multiple design cycles. The software prototype was evaluated by experienced lecturers, instructional designers, and developers of educational technology. Both the quantitative and qualitative evaluation results were encouraging and predominantly positive, so we assume that we have developed a helpful first prototype. Nevertheless, this paper presents only an intermediate step and educational chatbots should continue to be the subject of future research.

## **Implications for Spreading Educational Chatbots**

Although research has targeted educational chatbots for many years and shows their benefits, we observe that they are still used rarely. With this research and development project, we aim to address this issue by enabling lecturers to create chatbots for their teaching easily. We see particularly two main use cases: (1) Lecturers in in-class teaching settings can record their lectures and use those lecture recordings to create tailored chatbots for their students. Students can then use the provided chatbot for their follow-up work and for the preparation for the final exam. By giving them an additional interactive way to engage with the learning content, to ask content-related questions, and to discuss relevant aspects of the lecture, a positive impact on the learning process is to be expected according to the ICAP framework (Chi & Wylie, 2014). (2) The developed software prototype can be used to improve online-based courses (like MOOCs or online study programs). These courses are, in many cases, pre-dominated by a substantial part of passive learning when watching videos. When integrating the ability to discourse with a chatbot about the learning content, it is to be expected that this could have a positive impact on the students' learning.

We assume that both use cases can be implemented with our developed software prototype by lecturers - even without specific technical expertise. We hope - also on the basis of the positive feedback from the evaluation - that we can contribute to the spread of the chatbot technology in the educational sector.

## **Implications for our AI Education**

In addition to the implications for spreading the educational chatbot technology, we plan to integrate the developed software prototype as a show case of an AI use case in two AI courses. To this end, chatbots for supporting in-class teaching in both courses should be provided to the students. This enables them to interact with a tailored LLM-based chatbot and gives them the opportunity to interact with an LLM to analyze its

possibilities and limitations. Afterward, the underlying architectures (e.g., LLM, RAG, STT) should be discussed.

## Limitations and Future Research

The software developed as part of this project is currently a software prototype. Bugs known to us have been fixed as part of the iterative development process, but the prototype should be tested in more detail before going into production. For instance, it would be interesting to test in more detail how accurate the individual process steps (e.g., STT, TTS, or LLM-based improvements of the transcriptions) work. In addition to that, an evaluation with students is still pending. Here it would be particularly interesting to analyze how accurate the chatbot's answers are. As part of this step, it could also be further investigated how the system should deal with incorrect answers. For example, a retro perspective quality control, or a way for students to report answers that are then reviewed by a lecturer, could be feasible here. As a follow up study, we are interested in testing the chatbot in a field setting. We expect that this will provide us with further insights on how to improve the student-centered chat interface. By conducting a subsequent discourse analysis, we might be able to get additional insights into the students' interaction with the chatbot and its answer quality. Overall, we also hope to gain further insights into whether the automatic creation of the knowledge base using our derived technical process can lead to a change in the discourse quality. As a comparison, we could rely on insights that we gained from several prior chatbot studies (e.g., Hobert, 2019, 2023; Hobert et al., 2023; Hobert & Berens, 2024; Winkler et al., 2020) in which knowledge bases were created manually.

## References

- aws.amazon.com. (2024). *What is RAG? - Retrieval-Augmented Generation AI Explained* - AWS. Amazon Web Services, Inc. <https://aws.amazon.com/what-is/retrieval-augmented-generation/>
- Carolus, A., Koch, M. J., Straka, S., Latoschik, M. E., & Wienrich, C. (2023). *MAILS - Meta AI literacy scale: Development and testing of an AI literacy questionnaire based on well-founded competency models and psychological change- and meta-competencies*. *Computers in Human Behavior: Artificial Humans*, 1(2), 100014. <https://doi.org/10.1016/j.chbah.2023.100014>

- Chang, C., Hwang, G., & Gau, M. (2022). Promoting students' learning achievement and self-efficacy: A mobile chatbot approach for nursing training. *British Journal of Educational Technology*, 53(1), 171–188. <https://doi.org/10.1111/bjet.13158>
- Chi, M. T. H., & Wylie, R. (2014). The ICAP Framework: Linking Cognitive Engagement to Active Learning Outcomes. *Educational Psychologist*, 49(4), 219–243. <https://doi.org/10.1080/00461520.2014.965823>
- ffmpeg.org. (2024). *FFmpeg*. <https://www.ffmpeg.org/>
- Følstad, A., & Brandtzaeg, P. B. (2020). Users' experiences with chatbots: Findings from a questionnaire study. *Quality and User Experience*, 5(1). <https://doi.org/10.1007/s41233-020-00033-2>
- Github. (2024a). *GitHub—Openai/whisper: Robust Speech Recognition via Large-Scale Weak Supervision*. <https://github.com/openai/whisper>
- Github. (2024b). *GitHub—Pallets/flask: The Python micro framework for building web applications*. <https://github.com/pallets/flask/>
- Github. (2024c). *H/pytesseract*. <https://github.com/h/pytesseract>
- Hevner, A. R. (2007). A Three Cycle View of Design Science Research. *Scandinavian Journal of Information Systems*, 19(2), 87–92.
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28(1), 75–105. <https://doi.org/10.2307/25148625>
- Hobert, S. (2019). Say hello to 'coding tutor'! Design and evaluation of a chatbot-based learning system supporting students to learn to program. *ICIS 2019 Proceedings*.
- Hobert, S. (2023). Fostering skills with chatbot-based digital tutors – training programming skills in a field study. *i-com*, 22(2), 143–159. <https://doi.org/10.1515/icom-2022-0044>
- Hobert, S., & Berens, F. (2024). Developing a digital tutor as an intermediary between students, teaching assistants, and lecturers. *Educational Technology Research and Development*, 72(2), 797–818. <https://doi.org/10.1007/s11423-023-10293-2>
- Hobert, S., Følstad, A., & Law, E. L.-C. (2023). Chatbots for active learning: A case of phishing email identification. *International Journal of Human-Computer Studies*, 179, 103108. <https://doi.org/10.1016/j.ijhcs.2023.103108>
- Hobert, S., & Meyer von Wolff, R. (2019). Say Hello to Your New Automated Tutor – A Structured Literature Review on Pedagogical Conversational Agents. *Wirtschaftsinformatik 2019 Proceedings*, 301–314.
- Huang, W., Hew, K. F., & Fryer, L. K. (2022). Chatbots for language learning—Are they really useful? A systematic review of chatbot-supported language learning. *Journal of Computer Assisted Learning*, 38(1), 237–257. <https://doi.org/10.1111/jcal.12610>
- Laugwitz, B., Held, T., & Schrepp, M. (2008). Construction and Evaluation of a User Experience Questionnaire. In A. Holzinger (Ed.), *HCI and Usability for Education and Work* (Vol. 5298, pp. 63–76). Springer Berlin Heidelberg. [https://doi.org/10.1007/978-3-540-89350-9\\_6](https://doi.org/10.1007/978-3-540-89350-9_6)
- The Tabler Authors. (2024). *Tabler/tabler*. <https://github.com/tabler/tabler>
- The UEQ Team. (2024). *UEQ Data Analysis Tools*. [https://www.ueq-online.org/Material/Data\\_Analysis\\_Tools.zip](https://www.ueq-online.org/Material/Data_Analysis_Tools.zip)
- trychroma.com. (2024). *chroma—The AI-native open-source embedding database*. <https://www.trychroma.com>

Weizenbaum, J. (1966). ELIZA—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1), 36–45. <https://doi.org/10.1145/365153.365168>

Winkler, R., Hobert, S., Salovaara, A., Söllner, M., & Leimeister, J. M. (2020). Sara, the Lecturer: Improving Learning in Online Education with a Scaffolding-Based Conversational Agent. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–14. <https://doi.org/10.1145/3313831.3376781>

Winkler, R., & Söllner, M. (2018). Unleashing the Potential of Chatbots in Education: A State-Of-The-Art Analysis. *Academy of Management Annual Meeting (AOM)*.

Wollny, S., Schneider, J., Di Mitri, D., Weidlich, J., Rittberger, M., & Drachsler, H. (2021). Are We There Yet? - A Systematic Literature Review on Chatbots in Education. *Frontiers in Artificial Intelligence*, 4, 654924. <https://doi.org/10.3389/frai.2021.654924>