CS 224V Project: Building a CS106B Virtual Lecture Assistant

Naama Bejerano & Liv Sun | Fall 2024

Key Question

How do we create a conversational agent to enhance student engagement when watching asynchronous CS lectures?

Problem

Many Stanford courses, such as CS106B, provide lectures asynchronously, making them more accessible to students. However, students lose the ability to actively engage with the content and get answers to their questions on-demand, as they would in a live setting which negatively impacts their learning.

Background

- As pedagogy transitions towards virtual, a dialog-based system that leverages natural language has been shown to most effectively support a lecture to improve learning outcomes (Wölfel, 2021).
- LLM-based chatbots may address lower teacher-student ratios, as they offer a immediate, customized help without placing an extra burden on teachers (Chen et al., 2023).
- Prior research has demonstrated that students perceive pedagogical chatbots in a positive way, for 3 main reasons: 24/7 responsiveness, interactivity, and confidentiality (Chen et al., 2023).

User Study Findings

- 100% of users indicated they would use this chatbot in practice and it would help their learning.
  - User feedback was overwhelmingly positive for correctness, intelligence, coherence, and conversationality. Users expressed a preference for more concise answers.
- Most commonly cited use cases were:
  - summarizing lectures when reviewing for exams.
  - verifying their understanding of concepts during lecture.
  - requesting additional examples of concepts
- A commonly asked prompt was "*Summarize minutes x to y in lecture z*". This question surpasses the capabilities of a basic LLM, underscoring the value of this lecture tool.

Evaluation Results

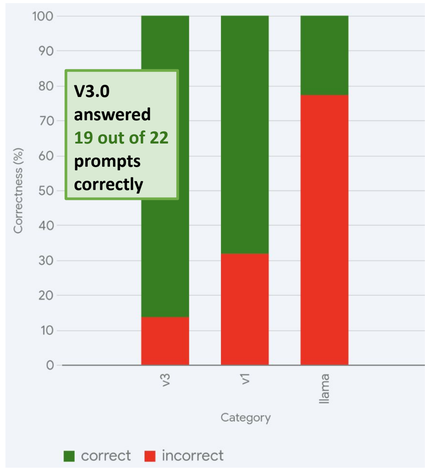
Our v3.0 chatbot achieves 55% better performance than a baseline model using Genie retrieval.

Our evaluation framework was based on two cumulative criteria:

Criteria #1: Correctness

- Evaluated correctness of v3.0 against 22 prompts. Compared percentage against v1.0 (baseline with Genie retrieval and Llama integration) and Llama (without retrieval).
- Prompts covered a diverse array of lecture topics, retrieval capabilities, and edge cases, including "Summarize…", "What is an example of …", an "what is X in Professor's own words…"

Correctness Results



v3.0	**19**/22 correct
v1.0 (baseline model)	**7**/22 correct
Llama	**5**/22 correct

Criteria #2: Style

- Conducted user studies with 12 current & former CS106B students.
- Evaluated the style of responses considering 1) coherence, 2) conversationality and 3) intelligence, and 4) length.

Style Results

- On a scale of 1 (poor) to 5 (excellent):
  - **Coherence** received an overall average score of **4.25**/5.
  - **Conversationality** received an average score of **4.38**/5.
  - **Intelligence** of the chatbot overall received a **4.42**/5.
  - **Length** received an overall average score of **3.917**/5.
- Overall, users were most impressed with the style of **summarized information.**

Tech Stack & Implementation

Tech Stack:

- Streamlit: chatbot web interface.
- Genie: LLM-based search endpoint for lecture transcripts.
- Together API: process queries.
- Meta Llama 3.1 8b (LLM): generate responses.

Implementation:

1. Process the user query & screen for homework-related questions.
2. If query includes timestamp information, extract and complete manual retrieval of relevant snippets from the transcript data.
3. Otherwise, use Genie to retrieve relevant snippets of transcript data.
4. Pass the following information into LLama with an engineered prompt:
   a. Retrieved data
   b. User query
   c. Past conversation history
5. Store and display the response to the user.

Learnings & Future Directions

Our chatbot demonstrates substantial improvements over baseline models, excelling particularly in intelligence and summary-based tasks.

However, challenges remain. A primary concern is mitigating the propagation of false or misleading information, which could impede student learning and pose significant ethical risks, like in exams. Addressing these issues is essential to preserving the integrity of Stanford's Computer Science Department and fostering trust in AI-driven educational tools.

Moreover, we must consider broader implications. If these capabilities are expanded, how might they impact the balance between online and in-person learning? **Does this technology genuinely enhance education, or could it inadvertently erode the value of face-to-face instruction?**

Future Directions:

- User Testing with a Panopto-integrated UI: Integrate v3.0 chatbot into the Panopto asynchronous lecture interface. Conduct rigorous user testing to observe the use of this tool in context.
- Faculty Engagement: Collaborate with CS professors to understand their perspectives on the tool's integration into coursework. This dialogue will help identify essential guardrails for ethical and responsible use.
- Expanded Dataset: Enhance the chatbot's capabilities by integrating additional content from the course website (e.g. logistics) to improve scope, relevance, and accuracy in responses.

Wölfel, M. (2021). Towards the Automatic Generation of Pedagogical Conversational Agents from Lecture Slides. In: Fu, W., Xu, Y., Wang, SH., Zhang, Y. (eds) Multimedia Technology and Enhanced Learning. ICMTEL 2021. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 388. Springer, Cham. https://doi.org/10.1007/978-3-030-82565-2_18

Chen, Y., Jensen, S., Albert, L.J. et al. Artificial Intelligence (AI) Student Assistants in the Classroom: Designing Chatbots to Support Student Success. Inf Syst Front 25, 161–182 (2023). https://doi.org/10.1007/s10796-022-10291-4

suno@stanford.edu, n20bej@stanford.edu