E-QGen: Educational Lecture Abstract-based Question Generation System

Mao-Siang Chen , An-Zi Yen
Department of Computer Science, National Yang Ming Chiao Tung University, Taiwan
siang.cs09@nycu.edu.tw, azyen@nycu.edu.tw

Abstract

To optimize the preparation process for educators in academic lectures and associated question-and-answer sessions, this paper presents E-QGen, a lecture abstract-based question generation system. Given a lecture abstract, E-QGen generates potential student inquiries. The questions suggested by our system are expected to not only facilitate teachers in preparing answers in advance but also enable them to supply additional resources when necessary.

1 Introduction

Teachers, when preparing their lecture content, are tasked with anticipating students' comprehension and any potential queries. Accordingly, to ensure the effectiveness of their instruction, they may adapt the course design and content to address these queries, for instance by preparing additional examples or allocating more time for explanation. In this case it would be helpful to have the use of a pedagogical support system capable of generating questions students might ask regarding lecture content.

Several studies generate questions based on specific text passages. Wikipedia is widely used as a resource based on which to generate questions [Liu *et al.*, 2020]. Moreover, several studies [Zhou *et al.*, 2018; Yuan *et al.*, 2017; Zhao *et al.*, 2018] utilize the SQuAD dataset [Rajpurkar *et al.*, 2016], which consists of questions from Wikipedia paragraphs written by crowd workers. As SQuAD was specifically created for machine reading comprehension, many of the answers to the questions are found within the paragraphs. As such, question generators trained on such a dataset may produce overly uniform questions, often involving inquiries about the properties or descriptions of specific terms mentioned in the given article and predominantly feature questions starting with "what". The nature of such questions whose answers are readily found in the paragraphs may not accurately reflect the variety or depth of questions students tend to ask [Ko *et al.*, 2020].

Recently, large language models (LLMs) have demonstrated remarkable capabilities in natural language comprehension and generation, and have become foundational for natural language processing tasks across a wide range of domains [Bommasani *et al.*, 2021]. Through prompt engineering, we instruct LLMs to generate any content that meets certain criteria or needs. Nonetheless, LLMs employed to directly generate questions based on the provided lecture content may not fully meet the nuanced requirements of educational application scenarios, because student inquiries often concern intricate details, or seek clarification of complex concepts, whereas LLM-generated questions typically focus more on general concepts of terminology in the lecture content. This divergence may stem from the models' lack of context-specific understanding or training data that aligns with genuine student concerns.

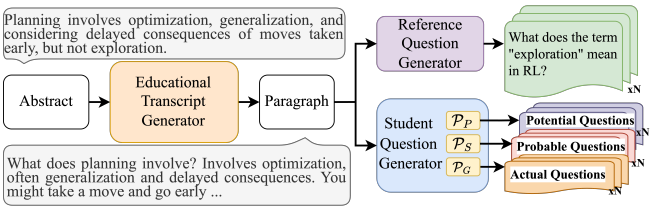

Figure 1: E-QGen system overview

In this work, we construct E-QGen, a pilot system that produces questions that closely resemble those a student might ask, accurately reflecting their genuine learning needs and areas of confusion. To address the challenges associated with generating such questions, we propose a student question generator that uses multitask learning and LoRA [Hu *et al.*, 2021] fine-tuning. Figure 1 shows an overview of E-QGen. E-QGen offers a service that allows teachers to input an abstract of the lecture content. Based on this abstract, the educational transcript generator automatically generates a complete lecture script, and the reference and student question generators subsequently produce suggested questions related to the generated script, aiding in comprehensive lesson planning. The reference question generator and student question generator are responsible for producing general questions and extended or in-depth questions, respectively. Our student question generator will produce three types of questions: actual student questions, probable student questions, and potential student questions. Actual student questions closely align with what students typically ask; probable student questions reflect

topics students may care about; and potential student questions estimate the inquiries students might have about course concepts, offering teachers high-quality suggestions without the need for costly APIs.

We construct a dataset by collecting real inquiries posed by students regarding the content of individual class sessions to fine-tune the model. However, acquiring real-world student questions is challenging, primarily due to the inherent difficulty in obtaining a comprehensive collection of educational materials and their corresponding student-posed questions. We collect publicly available lecture videos and transcripts uploaded by universities and research institutions to YouTube, along with user comments, after which we extract the comments that contain inquiries specifically regarding the education transcripts. Our contributions are threefold:

- We present a pioneer work on a pedagogically supported LLM that assists teachers in course preparation.
- We construct a novel system to generate educational transcripts and associated student inquiries based on lecture abstracts.[1][2]
- In addition to the demonstration system, we also construct a dataset comprising publicly available course transcripts and questions posed by students.[3]

2 Dataset Construction

Educational Transcript and Comment Collection. Inspired by work [Wang *et al.*, 2023] that crawls user comments concerning educational transcripts, we collect both subtitles and comments from the MIT OpenCourseWare[4] and Stanford Online[5] YouTube channels. These comments contain student thoughts, opinions, and inquiries. The videos are sourced from 35 distinct playlists (courses), encompassing a total of 19,013 user comments from 963 videos. Note that the majority of comments express gratitude, which is consistent with statistics from SIGHT [Wang *et al.*, 2023]. Hence, we extract questions related to course content from the comments, and align those questions to the corresponding sections of the educational transcripts.

Question Extraction. Given the large amount of comments, manual identification of questions would be labor-intensive and time-consuming. Therefore, we leverage the language understanding capabilities of LLMs as annotators to assist in question extraction by instructing them to label whether a comment is a question related to the course content. We employ three LLMs—`PaLM 2` [Anil *et al.*, 2023], `GPT-3.5` [Schulman *et al.*, 2022], and `GPT-4` [Achiam *et al.*, 2023]—for a labeling process that regards unanimously approved comments as questions. Afterwards, we further manually verify that these comments are indeed questions about the course. Comments that do not receive unanimous support are removed. This results in a total of 1,685 questions extracted from 19,013 comments.

Paragraph Segmentation. Since the majority of videos exceed 40 minutes in duration, the subtitles are lengthy. To generate questions more precisely for individual concepts throughout the course content, we segment the transcripts into multiple paragraphs using the text tiling toolkit.[6] Important hyperparameters include w and k, which represent the pseudosentence size and the number of sentences used in the block comparison method, respectively. A smaller value of w leads to finer segmentation. For k, 10 is suggested as a proper value. To ensure each paragraph conveys a complete and specific concept, w and k are set to 30 and 10, respectively. We segment a total of 14,422 paragraphs.

Question Alignment. As YouTube enables users to input timestamps in their comments to indicate the specific video segment their comment refers to, 419 questions can be mapped to a paragraph based on the timestamps. For the 1,266 questions lacking timestamps, we propose question alignment using classification and an assessment of semantic relatedness. For the classification method, we employ `PaLM 2` to determine whether a question is relevant to a specific paragraph. For the semantic relatedness assessment, we measure the cosine similarity between each paragraph and question. The paragraph and question representations are obtained through the `all_minilm_l6_v2` [Reimers and Gurevych, 2019] and `PaLM 2` embedding models. For each question, we calculate the cosine similarity with each paragraph using two embeddings. Given these two sets of scores, we rank them separately, and take the top 10 matches from each ranking as potential question-paragraph pairs suggested by their respective embedding models. These suggestions, combined with the classification results, are then submitted to a majority vote. The question-paragraph pair that receives a majority of votes is used to identify the correct alignment. In this way, we can align questions with multiple paragraphs. The question-paragraph pairs identified based on user-typed timestamps and our alignment strategy are referred to as "golden pairs" and "silver pairs", respectively. Golden pairs consist of actual questions accurately matched to specific paragraphs, and silver pairs, though derived from student questions, represent probable questions with less certainty regarding their specific paragraph alignment. This results in 356 golden pairs and 4,434 silver pairs, with an average paragraph aligned with 1.2 and 2.3 questions, respectively.

Data Augmentation. Due to limited question-paragraph pairs, we employ an LLM to augment our dataset and then use these to fine-tune our question generation model. Specifically, we instruct `GPT-4` to generate 20 potential student questions related to the course content based on the given transcript paragraphs. We collect 4,829 question-paragraph pairs, which are referred to as "platinum pairs".

3 E-QGen Implementation

This section details the implementation of E-QGen, consisting of an educational transcript generator, followed by a student question generator and a reference question generator.

Educational Transcript Generator. To develop the educational transcript generator, we leverage the natural language

[1] E-QGen system: https://nlplab.cs.nycu.edu.tw/demo
[2] https://youtu.be/SuiroLobtEU
[3] https://github.com/NYCU-NLP-Lab/E-QGen
[4] https://www.youtube.com/@mitocw
[5] https://www.youtube.com/@stanfordonline
[6] https://www.nltk.org/_modules/nltk/tokenize/texttiling.html

comprehension and generation capabilities of LLMs, along with their extensive knowledge.

Simply by entering an abstract of the lecture content, our system automatically generates a paragraph that encapsulates the key aspects of that abstract.

Student Question Generator. We adopt a multitask learning framework in which LoRA is applied to fine-tune the LLM. As mentioned in Section 2, since the number of golden pairs is limited, we postulate that fine-tuning with auxiliary tasks enhances the model's ability to generate high-quality questions that resemble those of students. Specifically, we complete three distinct subtasks with specific prompts: (1) $\mathcal{P}_G$ for generating actual student questions based on golden pairs G, (2) $\mathcal{P}_S$ for formulating probable student questions from silver pairs S, and (3) $\mathcal{P}_P$ for suggesting potential student questions associated with platinum pairs P.[7]

Formally, given an LLM parameterized by Φ and the task-specific parameter increment $\Delta\Phi = \Delta\Phi(\Theta)$ encoded by a much smaller set of parameters Θ, the fine-tuning process boils down to optimizing Θ:

$$\max_{\Theta} \sum_{(x_i^k, y_i^k) \in \mathcal{D}^k} \sum_{t=1}^{|y|} (\log p_{\Phi_0 + \Delta\Phi(\Theta)}(y_{i,t}^k | x^k, y_{i,<t}^k)), \quad (1)$$

where $\mathcal{D}^k = \{(x_i^k, y_i^k)\}_{i=1,\ldots,N}$ is the training data of subtask k, x_i^k is the i-th input generated by the task-specific prompt, and y_i^k denotes the corresponding questions. For instance, $x_i^G = \mathcal{P}_G(h_i)$ is the input of the first subtask to generate actual student questions y_i^G for the i-th paragraph h_i.

Reference Question Generator. Questions generated by the student question generator tend to delve deeper into specific concepts. To assist teachers in preparing a more comprehensive set of course questions, a reference question generator is constructed to produce general conceptual questions.

4 Experiment and Discussion

4.1 Experimental Setup

We focus on evaluating the results of actual student question generation. To assess whether the generated questions align most closely with those that students would pose, we ensure that questions from the same video are not included in both the training and test sets: this prevents the model from being exposed to similar questions during the training phase. The resulting question-paragraph pairs in the golden pairs are divided into training, validation, and test sets, with counts of 300, 10, and 46, respectively. All silver and platinum pairs are used as training data. `Vicuna-7b-v1.5` [Zheng *et al.*, 2023] and `GPT-3.5` are used as the base models for the student question generator and reference question generator, respectively. In our experiments, we set each generator in E-QGen to produce 20 questions for a given paragraph.

4.2 Experimental Results

Table 1 shows the results of the models that generate actual student questions for the given paragraph. ROUGE-1, ROUGE-2, ROUGE-L [Lin, 2004], and BERTScore [Zhang *et al.*, 2019] are adopted as the evaluation metrics. Given that a paragraph may be aligned with multiple questions and the generators produce 20 questions for each paragraph, to evaluate the results of multiple candidates against multiple references, we compute the scores for all combinations of references and candidates across all metrics for each instance. For each reference, we select the candidate that achieves the highest average score in all metrics to calculate the overall score and standard deviation for that metric. The scores before and after $\pm$ denote the overall score and the standard deviation. Higher overall scores indicate better-quality questions generated by the model, whereas higher standard deviations suggest greater diversity in the questions produced.

Models	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore
GPT-3.5	0.2328 $\pm$ 0.0565	0.0589 $\pm$ 0.0202	0.1865 $\pm$ 0.0438	0.8619 $\pm$ 0.0074
GPT-4	0.2505 $\pm$ 0.0656	0.0658 $\pm$0.0210	0.1967 $\pm$ 0.0507	0.8615 $\pm$ 0.0080
SQ Only	0.2418 $\pm$ **0.0660**	0.0645 $\pm$ 0.0210	0.2019 $\pm$ **0.0528**	0.8625 $\pm$ **0.1191**
E-QGen	**0.2667** $\pm$ 0.0652	**0.0866** $\pm$ **0.0238**	**0.2160** $\pm$ 0.0503	**0.8642** $\pm$ 0.0857

Table 1: Experimental results

Models	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore
SQ only	**0.2418**	**0.0645**	**0.2019**	**0.8625**
w/o fine-tuning on $\mathcal{D}^S$	0.2245	0.0488	0.1905	0.8570
w/o fine-tuning on $\mathcal{D}^P$	0.2197	0.0455	0.1779	0.8578

Table 2: Ablation results for student question generator

"SQ Only" denotes results generated using only the student question generator. Experimental results show that "SQ Only" outperforms `GPT-3.5`. In addition, E-QGen outperforms other LLMs, especially `GPT-4`. This suggests that our multitask learning approach achieves better performance than models with more parameters. Incorporating the reference question generator, the question generated by E-QGen not only aligns with relevant student inquiries but also provides a diverse range of questions for reference.

We also conducted an ablation study for the student question generator to investigate the impact when introducing different pairs of data. In Table 2, the performance degrades most when excluding $\mathcal{D}^P$ in the training set. This shows that pseudo-training data produced by the powerful LLM is effective for fine-tuning relatively small LLMs.

5 Conclusion and Future Work

This work demonstrates E-QGen, a pilot system that generates educational transcripts and questions that students are likely to ask, assisting teachers in proactively preparing course content. To account for the limited number of question-paragraph pairs, we construct a dataset and implement a multitask learning framework to fine-tune the model. Experimental results indicate that E-QGen achieves promising performance. Moreover, the questions generated by our system surpass `GPT-4` both in similarity to student-posed questions and diversity. At the current stage, our primary focus is on courses related to computer science. In the future, we will broaden the applications of our method by extending it to cover courses across various fields.

[7] All prompts are shown at https://github.com/NYCU-NLP-Lab/E-QGen.

Acknowledgements

We thank the reviewers for their insightful comments. This work was partially supported by the National Science and Technology Council, Taiwan, under grant NSTC 111-2222-E-A49-010-MY2, and by the Higher Education Sprout Project of the National Yang Ming Chiao Tung University and the Ministry of Education (MOE), Taiwan.

References

[Achiam *et al.*, 2023] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[Anil *et al.*, 2023] Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. PaLM 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

[Bommasani *et al.*, 2021] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[Hu *et al.*, 2021] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models, 2021.

[Ko *et al.*, 2020] Wei-Jen Ko, Te-yuan Chen, Yiyan Huang, Greg Durrett, and Junyi Jessy Li. Inquisitive question generation for high level text comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6544–6555, 2020.

[Lin, 2004] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[Liu *et al.*, 2020] Bang Liu, Haojie Wei, Di Niu, Haolan Chen, and Yancheng He. Asking questions the human way: Scalable question-answer generation from text corpus. *Proceedings of The Web Conference 2020*, 2020.

[Rajpurkar *et al.*, 2016] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, 2016.

[Reimers and Gurevych, 2019] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[Schulman *et al.*, 2022] John Schulman, Barret Zoph, Christina Kim, Jacob Hilton, Jacob Menick, Jiayi Weng, Juan Felipe Ceron Uribe, Liam Fedus, Luke Metz, Michael Pokorny, et al. ChatGPT: Optimizing language models for dialogue. *OpenAI blog*, 2022.

[Wang *et al.*, 2023] Rose E Wang, Pawan Wirawarn, Noah Goodman, and Dorottya Demszky. SIGHT: A large annotated dataset on student insights gathered from higher education transcripts. In *18th Workshop on Innovative Use of NLP for Building Educational Applications*, June 2023.

[Yuan *et al.*, 2017] Xingdi Yuan, Tong Wang, Caglar Gulcehre, Alessandro Sordoni, Philip Bachman, Saizheng Zhang, Sandeep Subramanian, and Adam Trischler. Machine comprehension by text-to-text neural question generation. In Phil Blunsom, Antoine Bordes, Kyunghyun Cho, Shay Cohen, Chris Dyer, Edward Grefenstette, Karl Moritz Hermann, Laura Rimell, Jason Weston, and Scott Yih, editors, *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 15–25, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[Zhang *et al.*, 2019] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*, 2019.

[Zhao *et al.*, 2018] Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. Paragraph-level neural question generation with maxout pointer and gated self-attention networks. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3901–3910, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[Zheng *et al.*, 2023] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and Chatbot Arena, 2023.

[Zhou *et al.*, 2018] Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. Neural question generation from text: A preliminary study. In *Natural Language Processing and Chinese Computing: 6th CCF International Conference, NLPCC 2017, Dalian, China, November 8–12, 2017, Proceedings 6*, pages 662–671. Springer, 2018.