Building a Buzzer-Quiz Answering System

Naoya Sugiura **Kosuke Yamada** **Ryohei Sasano**
Koichi Takeda **Katsuhiko Toyama**
Graduate School of Informatics, Nagoya University, Japan
{sugiura.naoya.e7,yamada.kosuke.v1}@s.mail.nagoya-u.ac.jp
{sasano,takedasu,toyama}@i.nagoya-u.ac.jp

Abstract

A buzzer quiz is a genre of quiz in which multiple players simultaneously listen to a quiz being read aloud and respond it by buzzing in as soon as they can predict the answer. Because incorrect answers often result in penalties, a buzzer-quiz answering system must not only predict the answer from only part of a question but also estimate the predicted answer's accuracy. In this paper, we introduce two types of buzzer-quiz answering systems: (1) a system that directly generates an answer from part of a question by using an autoregressive language model; and (2) a system that first reconstructs the entire question by using an autoregressive language model and then determines the answer according to the reconstructed question. We then propose a method to estimate the accuracy of the answers for each system by using the internal scores of each model.

1 Introduction

We use the term "buzzer quiz" to refer to a genre of quiz in which questioner reads quiz questions aloud and players answer by buzzing in as soon as they can predict the answer. A well-known example of a similar format to what we call a buzzer quiz here is the U.S. TV program *Jeopady!*, in which contestants must buzz in with a lock-out device before trying to answer a question. However, in *Jeopady!*, answers are only allowed after all the questions have been read aloud, whereas we assume a format in which answers are allowed while the questions are being read out. Because of the importance of buzzing in quickly, players normally answer incomplete questions in buzzer quiz.

Quizzes have been studied as open-domain question answering (QA) tasks because they do not limit the scope of knowledge. However, the major datasets for open-domain QA tasks, like Natural Questions (Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017) cointain complete questions. Consequently, systems built using those datasets (Karpukhin et al., 2020; Yamada et al., 2021; Izacard and Grave, 2021) are not designed to answer incomplete questions. Furthermore, it is certainly crucial in buzzer quizzes to give correct answers, but it is also essential to consider the plausibility of a predicted answer based on the given question at that moment and to decide whether to actually respond. For example, consider the question listed in Table 1 if it has not been read past the phrase "200-hit." At that point, because other baseball players also hold records comparable to that of Pete Rose, it is difficult to narrow the answer down to a single candidate. This makes the predicted answer at that moment more likely to be incorrect, so it would be better not to answer at that point. On the other hand, once the question has been read further, the predicted answer converges to the correct answer, "Ichiro Suzuki." Hence, to construct a more effective buzzer-quiz answering system, we need an indicator of a predicted answer's likelihood of being correct, which call a "confidence score."

We believe that the capability to respond to buzzer quizzes by answering incomplete questions could help replicate the human capacity to smoothly generate responses in a conversation by

Q (75% completeness): Pete Rose and this player are tied with ten 200-hit seasons each. This Japanese outfielder played most of his career with the Mariners, and currently plays for the Marlins.
Confidence score: 0.991 **A:** Ichiro Suzuki *correct*
Q (25% completeness): Pete Rose and this player are tied with ten 200-hit seasons each. This Japanese outfielder played most of his career with the Mariners, and currently plays for the Marlins.
Confidence score: 0.125 **A:** Ty Cobb *incorrect*

Table 1: Examples of quiz question text and output of answering system. Gray texts indicate the unread portions of the question text. "Completeness" denotes the percentage of the question text that has been read, and the "confidence score" refers to a value indicating the likelihood of the predicted answer being correct.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics - Student Research Workshop, pages 194–199

sequentially predicting the content of the dialogue. In this study, we first constructed a buzzer-quiz answering system that produces appropriate answers for incomplete questions, and we propose the methods for calculating the confidence scores for two different models. Specifically, we constructed two systems: the **GPT-only** system, which directly generates answers in response to a question by using GPT (Radford et al., 2018); and the **GPT+DPR** system, which generates answers through a retriever-reader approach using Dense Passage Retrieval (DPR) (Karpukhin et al., 2020), after completing the question via GPT. For the former system, we calculate a confidence score by using token output probabilities during answer generation, while for the latter system, we use scores that are used in the output of the model.

2 Proposed Method

We propose two types of buzzer-quiz answering systems based on open-domain QA systems. We also propose methods to estimate the accuracy of the answers in each system by using the internal scores in each model.

2.1 Open-Domain QA System

In open-domain QA, there are two mainstream approaches. The first is a generation-based approach that generates answers directly in response to input questions. A representative model is GPT (Radford et al., 2018), which is a pre-trained language model that is based on the Transformer decoder (Vaswani et al., 2017) and is trained to predict word sequences from a context by using a large text corpus. Because of this property, GPT can be used in language generation tasks that involve generating text in response to input text. In the case of QA, GPT can generate answers by formatting the input in such a way as to infer only the answer to a question. Furthermore, because GPT often achieves higher performance through fine-tuning with datasets from downstream tasks, such fine-tuning can be applied to build QA models.

The second major open-domain QA approach is a retriever-reader approach that searches for documents related to a question and extracts the answer from the documents. A representative model is the retriever-reader model, which uses DPR as the retriever. DPR uses a dual encoder network with different BERT models (Devlin et al., 2019) for questions and documents. When sentences are input to BERT, a special token `[CLS]` is inserted at the beginning of a document, and the embedding representations for the question text and each document are obtained. Then, documents are selected according to the semantic similarity calculated as the inner product of the obtained representations (Karpukhin et al., 2020). In the reader, BERT predicts the relevant documents containing the correct answer and extracts the answer portion within a document. Specifically, it predicts the document that is most likely to contain the answer at the position of the token `[CLS]`. Then, it performs the answer-portion extraction from the predicted document and determines the start and end points of the token sequence that forms the answer.

2.2 Buzzer-Quiz Answering Systems

The effectiveness of the open-domain QA systems that answer complete questions has been confirmed, but their effectiveness for a buzzer-quiz answering system remains unclear because such a system requires to answer incomplete questions. Generally, when only part of a question is given, the nature of the problem differs significantly from the case of a complete question, because there may be multiple possible answers, or the necessary information to determine the answer might not be available yet.

In this study, we constructed two buzzer-quiz answering systems: one that relies solely on inference via GPT, called the GPT-only system, and another that uses GPT for question completion and applies the retriever-reader approach with DPR, called the GPT+DPR system. For the GPT-only system, the designed input format is "[question text] + '/the answer is'," which prompts the model to generate the answer within the single quotation marks, which is then used as the predicted answer. The purpose of inserting a slash '/' between the question text and "the answer is" is to make the model recognize the boundary of the question text, which prevents the completion of incomplete questions. For the GPT+DPR system, an incomplete question is input to the GPT to complete the question text, and the resulting complete question is then used as input for the DPR-based retriever-reader model to generate the answer.

2.3 Confidence Scores

Next, we propose to calculate the confidence scores for predicted answers by using the internal scores that each model uses when it generates the outputs for the buzzer-quiz answering system. Here, the

confidence score means an indicator for judging whether a predicted answer is correct. For higher values of our proposed confidence scores, we expect a higher percentage of correct answers.

For the GPT-only model, we use the generation probability of the first token in the predicted answer (referred to as the **generation score**) as the confidence score. When given a sentence's first n tokens during sentence completion, GPT outputs the $(n+1)$-th token from the vocabulary with the highest generation score. The first token largely determines the direction of the answer in the buzzer quiz, because the answer often comprises a small number of tokens. Hence, we adopt only the first token's generation score as the confidence score.

As for the GPT+DPR model, three internal scores can be used as confidence scores: the **document score** and the **extraction score** calculated by the reader, as well as their arithmetic mean, the **average score**. In the reader, each `[CLS]` token in a document is scored through a learned linear layer, and the document with the highest score is selected; this is the document score. Then, the model extracts the span containing the answer from the selected document by calculating a span score, which comprises a start score and an end score. The extraction score is the sum of these start and end scores.

3 Experiments

We conducted two experiments: an evaluation of the proposed buzzer-quiz answering system's accuracy, and an investigation of the effectiveness of the confidence scores for each model. We define question completeness as $x\%$ when a question is truncated after the first $x\%$ of the text in terms of the character count. For the accuracy verification, we applied the GPT-only and the GPT+DPR models to questions with completeness levels of 25%, 50%, 75%, and 100%. For investigation of the confidence scores' effectiveness, we evaluated the confidence scores for each model by examining the relationship between the confidence scores and the accuracy at each level of question completeness.

3.1 Settings

Datasets We used the 2nd AIO Official Dataset (AIO),[1] which contains past questions from Japanese quiz competitions. The AIO dataset is officially divided into a training set, a development set, and a test set. In addition, we collected past questions from the Japanese quiz application "Minna de Hayaoshi Quiz" (Minhaya)[2] as additional training data. Table 2 shows the number of quiz-answer pairs and the average number of characters in the questions for the datasets. Note that the training of DPR required positive and negative documents in addition to quiz-answer pairs. Accordingly, DPR was trained using only the AIO dataset, whereas the Minhaya dataset was used only for training GPT.

Subset	Source	Size	Length
Train	AIO	17,735	48.2
	Minhaya	35,149	64.8
Dev	AIO	1,000	46.9
Test	AIO	2,000	51.6

Table 2: Overview of the datasets. "Length" means the average number of characters for the questions.

Comparison Models We compared both models, GPT-only and GPT+DPR, in the accuracy verification. In the investigation of confidence score effectiveness, for GPT-only, we used the generation score; in contrast, for GPT+DPR, we used all three scores, i.e., the document score, extraction score, and average score.

We used the Japanese GPT model[3] on Hugging Face Hub (Wolf et al., 2020) and a DPR model[4] based on Japanese BERT-large,[5] which is pre-trained the Japanese Wikipedia corpus. For GPT-only, we fine-tuned the model on the training set with the input format "[question text] + '/ the answer is' [answer]." For GPT+DPR, GPT was fine-tuned using only the questions from the training set. In both cases, the training was conducted for 5 epochs. DPR was based on Japanese BERT-large for both the retriever and reader components. The retriever was trained for 5 epochs with a batch size of 128 and a learning rate of 1e-5, and the reader was trained for 3 epochs with a batch size of 8 and a learning rate of 2e-5.

Metrics In the accuracy verification, the correctness of the predicted answer was assessed in terms

[1] https://sites.google.com/view/project-aio/dataset

[2] https://livequiz.work/minhaya1/

[3] https://huggingface.co/rinna/japanese-gpt-1b

[4] https://github.com/cl-tohoku/AIO2_DPR_baseline

[5] https://huggingface.co/cl-tohoku/bert-large-japanese

Model	25%	50%	75%	100%
GPT-only	11.9	27.9	45.6	56.2
GPT+DPR	11.9	28.8	45.9	62.0

Table 3: Results of accuracy verification. The x% represents the question completeness.

of exact matching. In the investigation of confidence score effectiveness, we created curves of the correct answer rate with respect to the answer generation rate, and we evaluated the effectiveness in terms of the area under the curve (AUC). Here, the answer generation rate was the proportion of times that the system actually provided an answer. If the models only answered questions for which the confidence score exceeded a threshold α, we can control the answer rate by changing α. On the other hand, the correct answer rate was the proportion of correct answers among the answers output by the models. If α is set to a value below 0, the answer rate will coincide with the overall correct answer rate of the system. As α increases, only questions with high confidence scores will be answered, so the correct answer rate will be expected to increase.

3.2 Accuracy Verification

Table 3 lists the accuracies for the GPT-only and GPT+DPR models for each level of question completeness. As the question completeness decreased, the correct answer rate also decreased, but the rate of decrease was not proportional. From 100% to 75%, the decline was relatively gentle. This was likely because many important words that determine the answer appear in the first half of a question, whereas cases with information-rich words appearing in the latter half of a question are relatively rare. Comparing the scores of the two models, we see that GPT+DPR performed better when the question completeness was 100%. When the questions were incomplete, however, there was no significant difference in performance between the two models was observed.

3.3 Confidence Score Effectiveness

Table 4 lists the AUC values for each level of question completeness. Among the three confidence scores for GPT+DPR, using the document score yielded the highest AUC. Furthermore, among all the results, the generation score for GPT-only achieved the highest AUC.

Next, because the document score had the highest AUC for GPT+DPR, we used it to compare

Model	Score	25%	50%	75%	100%
GPT-only	generation score	**41.4**	**63.6**	**81.2**	**85.9**
GPT+DPR	document score	31.8	58.1	77.3	84.8
	extraction score	25.0	51.3	70.0	84.1
	average score	29.1	56.1	75.8	84.0

Table 4: AUC values for each level of question completeness. "Score" means the internal scores we used.

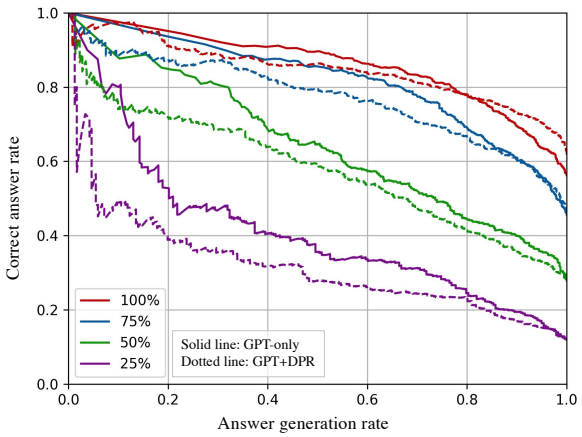

Figure 1: Curves of the correct answer rate vs. the answer generation rate. The x% represents the question completeness.

the correct answer rate vs. answer generation rate curves of the GPT-only model and the GPT+DPR models. Figure 1 shows the results. For all settings, we can observe that the accuracy was increased by limiting the questions to be answered to only those with high confidence scores, thus confirming the effectiveness of the confidence scores. Comparing GPT-only and GPT+DPR, as listed in Table 3, the accuracy at an answer rate of 1.0 was higher for GPT+DPR when the question completeness was 100%, and equivalent in for less-complete questions. When the answer rate was less than 0.8, however, GPT-only had higher accuracy in all cases. This difference was more obvious when both the question completeness and the answer rate were low. For example, in the case of 25% question completeness and an answer rate of 0.1, the accuracy of GPT+DPR is around 0.5, whereas that of GPT-only was around 0.8, thus showing a significant difference. Accordingly, we can conclude that the GPT-only model is more suitable for buzzer quizzes.

Table 5 shows examples of quiz question text and output from the GPT-only system. Examples (a) and (b) are cases with 25% question completeness, while Examples (c) and (d) are cases with 75% question completeness. In Examples (a) and

	Examples
(a)	**Q (25% completeness):** ごはんの上にハンバーグと目玉焼きを乗せ、グレービーソースをかけたハワイの名物料理は何でしょう? (This is a rice dish topped with a hamburger steak and a fried egg, which is covered with gravy sauce and originated in Hawaii. What is this?) **Confidence score:** 0.996 **A:** ロコモコ (loco moco) *correct*
(b)	**Q (25% completeness):** オーストリアの首都はウィーンですが、オーストラリアの首都はどこでしょう? (The capital of Austria is Vienna, but what is the capital of Australia?) **Confidence score:** 0.982 **A:** キャンベラ (Canberra) *incorrect*
(c)	**Q (75% completeness):** 約5年の歳月をかけてシスティーナ礼拝堂の祭壇に描かれた、ミケランジェロの代表作である絵画は何でしょう? (This painting was created over the span of about five years in the Sistine Chapel. Now, this is known as one of Michelangelo's masterpieces. What is this?) **Confidence score:** 0.991 **A:** 最後の審判 (The Last Judgment) *correct*
(d)	**Q (75% completeness):** 1985年に発売され、全世界で 4000万本以上を売り上げたという任天堂ファミリーコンピュータのゲームで、「スーマリ」などと略されるものは何? (This game was launched for the Nintendo Family Computer in 1985 and has sold 40 million copies, which is often referred to by the abbreviation "Su-Mari." What is this?) **Confidence score:** 0.955 **A:** ドンキーコング (Donkey Kong) *incorrect*

Table 5: Examples of quiz question text and output from the GPT-only system. Since the actual data are in Japanese, English translations are given in parentheses.

(c), the system predicted correct answers with high confidence scores because sufficient information was provided to narrow down the answer. In contrast, in Examples (b) and (d), the system predicts the answers with high confidence scores, but the answers are incorrect. Example (b) is a question text with contrasting first and second halves, which would be difficult to answer in a situation where only the first half of the question is given. Example (d) is incorrect because the question text is mostly clear, but does not contain the key information that determines one answer.

4 Conclusion

In this study, we constructed two models for answering buzzer quiz questions, which have not been considered in previous research: GPT-only and GPT+DPR. Then, we evaluated the accuracy for various levels of question completeness. Furthermore, we investigated the relationship between the model's internal scores, which were treated as confidence scores, and the accuracy; as a result, the validity of using the internal scores of the models as confidence scores was confirmed.

In the future, we consider the use of more powerful models like FiD (Izacard and Grave, 2021) or GPT-4 (OpenAI, 2023) to improve the correct answer rate for quizzes. We also would like to validate the differences in performance between our systems and humans.

Limitations

We built buzzer quiz answering systems. However, they do not take into account the time required to respond, and these systems do not have the ability to generate real-time responses, which is essential in actual buzzer quizzes. Additionally, the experiments in this study were conducted only in Japanese, and it remains unclear whether similar results would be obtained in other languages. Particularly, English has a significantly different sentence structure compared to Japanese, hence further investigation is necessary to confirm whether appropriate results can be achieved.

Acknowledgements

This work was partly supported by JSPS KAKENHI Grant Number 21H04901.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2019)*, pages 4171–4186.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th*

Conference of the European Chapter of the Association for Computational Linguistics: Main Volume (EACL 2021), pages 874–880.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020)*, pages 6769–6781.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics (TACL)*, 7.

OpenAI. 2023. Gpt-4 technical report.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. Open AI Technical Report.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pages 5998–6008.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (EMNLP 2020)*, pages 38–45.

Ikuya Yamada, Akari Asai, and Hannaneh Hajishirzi. 2021. Efficient passage retrieval with hashing for open-domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers) (ACL-IJCNLP 2021)*, pages 979–986.