Synthesizing Conversations from Unlabeled Documents using Automatic Response Segmentation

Fanyou Wu
Amazon
fanyouwu@amazon.com

Weijie Xu
Amazon
weijiexu@amazon.com

Chandan K. Reddy
Amazon
ckreddy@amazon.com

Srinivasan H. Sengamedu
Amazon
sengamed@amazon.com

Abstract

In this paper, we tackle the challenge of inadequate and costly training data that has hindered the development of conversational question answering (ConvQA) systems. Enterprises have a large corpus of diverse internal documents. Instead of relying on a searching engine, a more compelling approach for people to comprehend these documents is to create a dialogue system. In this paper, we propose a robust dialog synthesising method called SynCARS. We learn the segmentation of data for the dialog task instead of using segmenting at sentence boundaries. The synthetic dataset generated by our proposed method achieves superior quality when compared to WikiDialog, as assessed through machine and human evaluations. By employing our inpainted data for ConvQA retrieval system pre-training, we observed a notable improvement in performance across standard benchmark datasets.[1]

1 Introduction

Conversational Question Answering (ConvQA) is a computational task aimed at modeling the information-seeking processes found in human dialog. The goal of this task is to allow automated systems to understand and respond to questions within a conversational context. Several publicly available datasets, such as QuAC (Choi et al., 2018), CoQA (Reddy et al., 2019), QreCC (Anantha et al., 2021), and OR-QuAC (Qu et al., 2020), have been developed for building ConvQA systems. Despite these resources, the size of the datasets remains relatively limited, posing potential challenges when implementing ConvQA systems in real-world applications.

Meanwhile, a plethora of high-quality documents, including but not limited to sources such as Wikipedia and arXiv, are publicly available. Numerous technological roadmaps have been proposed to leverage the vast wealth of information within these documents to construct a ConvQA system. One approach involves utilizing a Large Language Model (LLM) in conjunction with information retrieval techniques, such as New Bing from Microsoft. Alternatively, a well-trained LLM, such as ChatGPT without a plugin, can be employed independently to achieve similar or even superior results. By utilizing information retrieval tools, LLMs can access up-to-date information, albeit at the expense of increased inference time and latency compared to using LLMs alone.

To effectively employ these documents alongside LLMs for the purpose of constructing a Con-

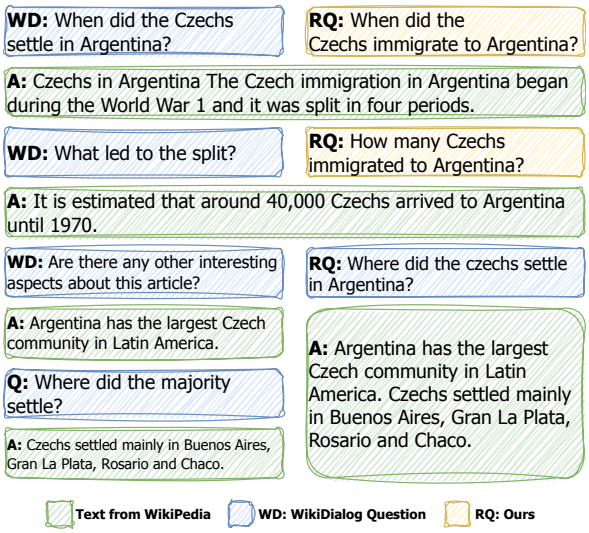

Figure 1: An example dialog from WikiDialog (WD) and ours (RQ). The blue and yellow boxes in the dialog contain the questions generated by our approach, while the green boxes contain the corresponding answers. WD asks a question starting with "Are there any other .." which is not useful to train a question answering system. Besides, some of the answers can be combined such as the last two on the left side. In contrast, our method fixed those problems.

[1]Our model and dataset are publicly available at https://github.com/wufanyou/SynCARS.

Findings of the Association for Computational Linguistics: ACL 2024, pages 8012–8026

vQA system, there are two potential research directions to explore: the decomposition and synthesis of the documents into a dialog format, or the pursuit of improved question embedding representations for the documents. Additionally, conversational question generation (QG) can be utilized in both of these approaches. Dai et al. (2022) pioneered "dialog inpainting," suggesting every sentence in a document can answer a question, leading to the creation of the WikiDialog dataset. In our study, we introduce SynCARS, a novel approach that leverages ConvQA datasets to produce synthetic conversations from unlabeled documents, yielding superior quality compared to WikiDialog. To summarize, our main contributions can be summarized as follows:

- We identified various challenges present within the WikiDialog dataset, a component of FLAN collection (Longpre et al., 2023) used for instruction fine-tuning.
- We designed a new dataset by merging existing datasets and filtered out inadequate data to address issues present in the WikiDialog dataset. Additionally, we have designed a new answer segmentation technique by introducing a special token p_t^m. In comparison to the approach by Dai et al. (2022), we obtained a new and compact model specifically tailored for generating dialogues from documents.
- Our generated dataset exhibits significantly higher answer quality and question specificity, as validated through Human and GPT-4 evaluations, when compared to WikiDialog. Furthermore, the question retrieval system trained on our generated data achieves superior results compared to the system trained on WikiDialog and the standard retrieval-only benchmark method.

2 Related Work

Question generation (QG) is a field that seeks to create natural questions using various types of data sources, including structured knowledge bases (Guo et al., 2018; Chen et al., 2023), text (Rus et al., 2010; Du et al., 2017; Nogueira et al., 2019), images (Li et al., 2018), and tables (Bao et al., 2018). Past research efforts in this area have primarily focussed on producing isolated and disconnected questions from a given passage.

Pan et al. (2019) proposed the Conversational Question Generation (CQG) task as an approach to improve the development of ConvQA systems. This task involves generating the subsequent question by incorporating a passage and a conversation history, thereby requiring a deeper comprehension of the given passage and prior conversation to generate a coherent and relevant question for the next round. Unlike prior QG tasks that only consider the passage, CQG requires an understanding of the previous conversation, making it a more complex task.

Kim et al. (2022) proposed SIMSEEK, a framework that generates ConvQA datasets by modeling the information needs of questioners who may ask incoherent questions due to excessive information. SIMSEEK includes a conversational answer extractor that selects answer candidates from the passage by considering the context of the conversation. However, this method is only suitable for short answers.

In contrast, Dai et al. (2022) introduced dialog inpainting, which assumes that each sentence in a document can be used as an answer to a question. The authors generated a ConvQA dataset called WikiDialog using this approach. This dataset tends to have longer answers as each answer corresponds to a single sentence. This characteristic makes it more suitable for dialog applications. While the proposed method is straightforward and efficient, concerns arise regarding the quality of the WikiDialog dataset. An illustrative example from the WikiDialog dataset is presented in Figure 1. From this example, it is evident that combining certain answers could yield improved responses and questions. Moreover, some questions are overly broad, rendering them less suitable for training a retriever system. In the context of the Open-QA dataset, the "anything else" question serves as a means to transition between topics. However, when examining a brief paragraph, typically containing around six sentences or less, it becomes challenging to discern any significant shifts in the topic.

3 SynCARS

Problem Statement: SynCARS (**Syn**thesizing **C**onversations using **A**utomatic **R**esponse **S**egmentation) aims to generate a high-quality complete dialog from an informative document. It assumes that at most N continuous sentences where $(N > 1)$ from the document can be treated as an answer to an imagery question.

We build our work on top of the dialog inpaint-

ing (Dai et al., 2022), where each sentence is treated as an answer (equivalent to $N = 1$ in our assumption). Our idea stems from the observation that not all sentences are equally informative in WikiDialog, the dataset generated by dialog inpainting. Figure 1 shows some examples that demonstrate the limitations of the WikiDialog dataset.

To synthesize better ConvQA datasets, we implemented a simple sentence segmentation mechanism, along with a few modifications to the dialog inpainting method. In the following section, we will introduce these components in more detail.

3.1 Notations

Formally, a complete dialog $\mathbf{d}$ is a sequence of speaker questions, answers, and optional context, represented by:

$$\mathbf{d} = (\mathbf{c} \oplus \mathbf{q}_1 \oplus \mathbf{a}_1 \oplus \cdots \oplus \mathbf{q}_t \oplus \mathbf{a}_t \oplus \cdots), \quad (1)$$

where $\mathbf{q}_t$ and $\mathbf{a}_t$ are t question and answer in a dialog, respectively. $\mathbf{c}$ is the prefix optional context and $\oplus$ is the sequence joint symbol. We use the same notation for partial dialogs, where unobserved questions are denoted by the $\cdot$ symbol. For example, $(\mathbf{c} \oplus \cdot \oplus \mathbf{a}_1 \oplus \mathbf{q}_2 \oplus \mathbf{a}_2 \oplus \cdot \oplus \mathbf{a}_3)$ is a partial dialog where question $\mathbf{q}_1$ and $\mathbf{q}_3$ are unobserved, and we refer to these as "masked" utterances. Additionally, we use the shorthand $\mathbf{d}_{m(1,3)}$ to denote a dialog $\mathbf{d}$ with masked utterances at positions 1 and 3.

To complete the partial dialog $\mathbf{d}_{m(1,3)}$, we generate predictions for questions 1 and 3, denoted $\hat{q}_1$ and $\hat{q}_3$. The inpainted dialog is then:

$$\text{Inpaint}(\mathbf{d}_{m(1,3)}) = (\mathrm{c} \oplus \hat{\mathbf{q}}_1 \oplus \mathbf{a}_1 \oplus \mathbf{q}_2 \oplus \mathbf{a}_2 \oplus \hat{\mathbf{q}}_3 \oplus \mathbf{a}_3). \quad (2)$$

In this scenario, $\hat{\mathbf{q}}_1$ and $\hat{\mathbf{q}}_3$ are typically questions directed towards the next answer, and our goal is to associate them with all preceding utterances ($\mathbf{q}$ and $\mathbf{a}$).

3.2 Answer Segmentation

Each answer $\mathbf{a}_t$ in Eq. (2) can be further decomposed with sentences, denoted by:

$$\mathbf{a}_t = (\mathbf{s}_t^1 \oplus p_t^1 \oplus \cdots \oplus \mathbf{s}_t^m \oplus p_t^m \cdots), \quad (3)$$

where $\mathbf{s}_t^m$ is the m-th sentence in answer $\mathbf{a}_t$, and p_t^m is its corresponding placeholder. Here we involve placeholder p_t^m to aid in answer segmentation. Specifically, if p_t^m is a special token (e.g., empty string in this paper), then we consider that $\mathbf{s}_t$ and $\mathbf{s}_{t+1}$ should be combined as one answer towards a question $\mathbf{q}_t$. Considering a similar case in Eq. (2), our inpainted dialog with answer segmentation can be written as follows:

$$\text{SegInpaint}(\mathbf{d}_{m(1,3)}) = (\mathbf{c} \oplus \hat{\mathbf{q}}_1 \oplus \mathbf{s}_1^1 \oplus \hat{\mathbf{p}}_1^1 \oplus \mathbf{s}_1^2 \oplus \mathbf{q}_2 \oplus \mathbf{s}_2^1 \oplus \hat{\mathbf{q}}_3 \oplus \mathbf{s}_3^1). \quad (4)$$

Our model is capable of generating questions ($\hat{\mathbf{q}}_1$ and $\hat{\mathbf{q}}_3$) and performing answer segmentation ($\hat{\mathbf{p}}_1^1$) simultaneously. If $\hat{\mathbf{p}}_1^1$ is the special token that we defined, then $\mathbf{q}_1$ is considered as the question to $(\mathbf{s}_1^1 \oplus \mathbf{s}_1^2)$. Otherwise, $(\hat{\mathbf{q}}_1 \oplus \mathbf{s}_1^1)$ and $(\hat{\mathbf{p}}_1^1 \oplus \mathbf{s}_1^2)$ form two question-answer pairs. *By combining some of those sentences, we can generate more comprehensive responses as well as improved questions.*

3.3 Training

To train our model, we utilize a partial dialog and aim to predict two values: $\mathbf{q}_t$ and p_t^i. This task is similar to the masked language modeling used in BERT (Kenton and Toutanova, 2019), where missing tokens in a passage are reconstructed. However, in our case, we aim to reconstruct a missing utterance in a dialog.

Let us assume that the model is a generative model with parameters θ, which specify a probability distribution $P_\theta(\mathbf{q}_t \mid \mathbf{d}_{m(t)})$. Our training objective is to minimize the standard cross-entropy loss:

$$\mathcal{L}(\theta) = -\sum_{\mathbf{d} \in \mathcal{D}} \mathbb{E}_{\mathbf{q}_t \sim \mathbf{d}} \left[\log P_\theta(\mathbf{q}_t \mid \mathbf{d}_{m(t)}) \right] \quad (5)$$

where $\mathcal{D}$ is the set of complete dialogs and $\mathbf{q}_t$ is a randomly sampled question from the dialog $\mathcal{D}$.

Following Dialog Inpainting (Dai et al., 2022), we used T5 (Raffel et al., 2020), a text-to-text encoder-decoder Transformer as our pre-trained model. T5 uses a denoising objective that is slightly different from the original Masked Language Modeling (MLM) used in BERT. We believe that T5's denoising pre-training objective and encoder-decoder architecture are the most suitable for this task. Figure 2 shows the original texts, inputs and targets during our training.

During training, we randomly masked at least one and at most N continuous questions $\mathbf{q}$ within a dialogue or question-answer pairs, as well as all answer segmentation placeholders p. As mentioned earlier in Section 3, our assumption is that N is the maximum number of sentences that can form an answer. To balance the contextual awareness of the model, we decided to randomly add or remove titles of the dialogue or QA during training.

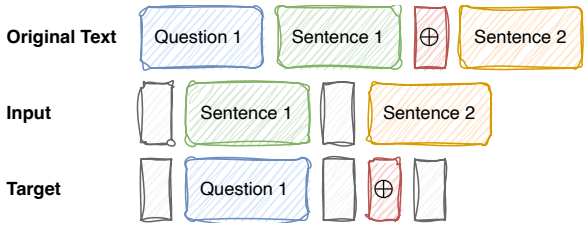

Figure 2: An illustration of preparing the training dataset, considering a training instance with a question and two sentences as answers. Here, grey boxes represent the extra_ids tokens for T5.

3.4 Inference

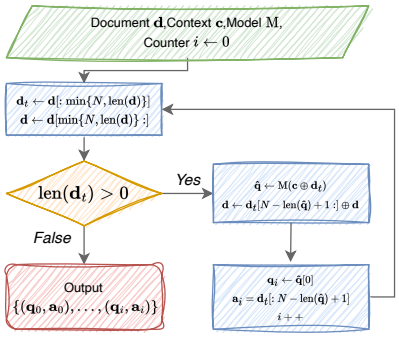

Figure 3: A flowchart illustrating the use of a trained model to convert a document into a dialogue format where, in each iteration, at most N sentences are processed, and only the first generated question is retained.

Figure 3 illustrates the process of using a trained model to convert a document into a dialogue format. The process begins with a document ($\mathbf{d}$) that contains a set of sentences. The trained model takes this document as input, along with a context ($\mathbf{c}$) that includes the document's title. The model then generates a dialogue by predicting questions and their corresponding answers based on the input sentences.

The model processes the document iteratively. In each iteration, it considers a fixed number of sentences (denoted as N in the figure) and generates a question that summarizes these sentences. The generated question and its corresponding answer are then added to the dialogue history. This process continues until all the sentences in the document have been processed.

During the inference process, the model maintains a counter to keep track of the number of sentences processed so far. The output of the model is a dialogue consisting of generated questions and their associated answers, which together provide a comprehensive summary of the input document.

By following this approach, the trained model can effectively convert a document into a more accessible and interactive dialogue format, enabling users to quickly grasp the key points of the document through a series of relevant questions and answers.

4 Experimental Setup

4.1 Datasets

The successful implementation of answer segmentation highly depends on the training dataset. Naturally, if the training dataset only contains short answers or single sentences, the answer segmentation will fail. So we perform some basic data analysis in this section. Table 1 shows the distribution of the number of sentences from the selected training datasets. Dialog Inpainting use OR-QuAC (Qu et al., 2020), and QreCC (Anantha et al., 2021) to conduct semi-supervised training, where more than 90% of the answers consist of only one sentence. In order to improve the answer segmentation capacity, we utilized an additional Dolly dataset (Conover et al., 2023), where answers usually have more than one sentence. Based on the statistics of those datasets, we set $N = 3$ for our experiments reported in this paper.

Dataset	Avg # Sen.	1 Sen.	2 Sen.	$\geq$ 3 Sen.
OR-QuAC	1.08	92.52%	6.99%	0.49%
QreCC	1.10	90.03%	9.29%	0.67%
Dolly	3.44	43.18%	13.83%	42.99%

Table 1: The distribution of the number of sentences from the selected training datasets.

We noticed that the WikiDialog dataset often includes a specific follow-up question – "Are there any other interesting aspects about this article?". Figure 1 also shows this behavior. This follow-up question is a common sentence in the QreCC, and OR-QuAC datasets. For instance, in QreCC, approximately 4% of question-answer pairs and 22.9% of dialogs contain this type of question. The original objective of those "anything else" questions was to indicate shifts in the current topic and to request any new information. However, generating these questions is not ideal because they lack the specificity needed to elicit answers that are meaningfully representative of the content being discussed. Simultaneously, we believe that a short documents should be within a topic. So we decided to cleanup QreCC and OR-QuAC datasets,

using a hand-crafted rule that excludes any question which contain "other interesting". **Thus, our synthetic data is less likely to generate questions regarding the shift of topics.**

Furthermore, each entry in QreCC and OR-QuAC datasets contains two question types, the raw question (RQ) and the rewritten question (WQ). In cases where the raw question includes personal pronouns (such as "she," "they," and "we") and demonstrative pronouns (such as "these," "this," and "that"), a question rewriting model may involve hand-crafted rules to replace those pronouns or rewrite the question entirely. See Qu et al. (2020); Anantha et al. (2021) for more details. In this paper, we chose to use both question types during training as a data augmentation technique. Additionally, to control the desired output question types, we added a prefix to the input as "Type: {*question type*}" to indicate the current question type for the placeholder.

4.2 Models

In this study, we decided to use FLAN-T5-XL (Chung et al., 2022) due to the limitations in our computational resources. Although T5-XXL (Raffel et al., 2020), an 11B parameter model, was used in dialog inpainting (Dai et al., 2022), we opted for a smaller model. In Table A.1 (provided in the Appendix), we show that there is not much difference in performance when applying the proposed method to either FLAN-T5-XL or its counterpart T5-V1_1-XL. Both models are effective and efficient compared to the dialog inpainting approach.

To summarize, we initialized our model with FLAN-T5-XL, which has 3 Billion parameters, and fine-tuned it with 8 V100 16GB GPUs. The training process employed a constant learning rate of 10^{-4}, a dropout rate of 0.1, an equivalent batch size of 32, and ran for 3.5K iterations (equivalent to 4 epochs).

5 Evaluation

Our primary focus in this study centers around conducting a human evaluation to compare the outputs generated by our approach with those produced by WikiDialog (WD) using Dialog inpainting method. To carry out this assessment, we utilize Amazon Mechanical Turk (MTurk) as our platform. Each human annotator is compensated at a rate of 0.036 US dollars per question, and we provide them with identical instructions and sample examples as outlined in Figures 6 to 10 in the work of Dai et al. (2022). A potential issue in Dai et al. (2022) lies in their use of the same raters for evaluations, which could introduce bias due to the evaluators' perception of the subject matter. To mitigate this concern, our evaluation process involves presenting subjective questions to a minimum of three distinct human evaluators for each dialogue turn. We label the answer for each question as the one agreed upon by at least two human annotators. Given that MTurk offers a more diverse pool of human evaluators, we use two-proportion z-test for evaluations.

We report our findings based on dialogs that correspond to a set of 50 selected passages[2]. Recent studies suggest using large language models (LLMs) as reference-free metrics for evaluating natural language generation (NLG) tasks. LLMs have the advantage of being applicable to new tasks that lack human-generated reference texts (Liu et al., 2023). Furthermore, Faysse et al. (2023) demonstrate that LLMs are more aligned with human preferences and exhibit consistent performance across a diverse set of generative tasks. In this study, we utilized the OpenAI chat completion API with the GPT-4 model (Brown et al., 2020) to perform the same evaluation as human annotators. We used the exact same rubric as Dai et al. (2022). The rubrics and prompt templates used for both human and GPT-4 evaluations are provided in Table A.4 in the Appendix. The results of this evaluation are presented in Table 2.

5.1 Quality Comparison

Overall, our approach consistently outperforms WikiDialog (WD) in terms of generating more specific questions and better answers, despite our model's smaller size. To assess the statistical significance of these improvements, we conducted a two-proportion z-test, which is a statistical test used to determine if the proportions of categories in two group variables significantly differ from each other. This means that it is suitable when your variable of interest is categorical and have more than 10 values in each of the populations. If we consider 'Very' as an acceptable answer for the question 'How specific is the question?', then RQ is significantly better than WD with a p-value of 2.5×10^{-2} according to human assessment and

[2]These passages are the first 50 passages in the WD dataset and can be found at https://github.com/google-research/dialog-inpainting

Evaluator	Human			GPT 4		
Answer	**RQ**	**WD**	**WQ**	**RQ**	**WD**	**WQ**
Is the question information seeking?						
Yes (%)	82.6	**87.1**	82.8	**99.3**	99.0	100.0
How relevant is question to the conversation?						
Not at all (%)	3.3	4.1	2.2	4.9	2.4	0.0
Topic only (%)	45.0	45.5	43.0	20.3	26.4	22.0
Follows up (%)	**51.7**	50.4	54.8	**74.8**	71.2	78.0
How specific is the question?						
Not at all (%)	4.5	4.8	5.3	1.0	10.6	0.6
Somewhat (%)	42.4	46.9	47.3	24.4	31.0	16.9
Very (%)	**53.1**	48.3	47.4	**74.6**	58.4	82.5
How well answered is the question?						
Not at all (%)	3.1	2.7	7.7	13.0	11.8	9.0
Incompletely (%)	10.3	13.7	1.9	22.3	38.0	23.7
Sufficiently (%)	40.9	40.4	47.4	21.7	21.6	19.3
Perfectly (%)	**45.7**	43.2	43.0	**43.0**	28.6	48.0

Table 2: Results from a human evaluation of the generated dialog in four variants of our method vs. WikiDialog. In this evaluation, 'RQ' represents the questions generated by our proposed method, 'WQ' indicates rewritten questions, and 'WD' represents questions generated by WikiDialog. Our findings indicate that our proposed method's 'RQ' outperforms WD in 7 out of the 8 cases.

3.0×10^{-4} according to GPT-4. This indicates that our proposed method excels in asking more specific questions compared to WD.

We also conducted a similar test on the criterion 'How well answered is the question?' with 'Perfectly' as an acceptable answer choice. In this evaluation, RQ once again outperforms WD, with a p-value of 4.7×10^{-4} according to human assessment and 2.0×10^{-3} according to GPT-4. This demonstrates that RQ achieves better answer quality than WD. The improved question specificity and answer quality can enhance the utility of the synthesized dataset for downstream tasks, such as information retrieval. While WQ and RQ are both superior to WD in terms of question relevance based on human and GPT-4 evaluations, these differences are not statistically significant. In summary, RQ outperforms WD in 7 out of the 8 cases, demonstrating the superiority of our proposed method compared to the baseline.

5.2 Question Types

Unlike the WikiDialog dataset, we offer rewritten questions (WQ). The expectation is that these rewritten questions are superior based on many proposed criteria, especially since they tend to contain fewer personal and demonstrative pronouns. This assumption is also validated in Table 2. However, for downstream tasks or real-world scenarios where users provide natural inputs, questions are less often in this rewritten style.

5.3 GPT-4 vs Human Evaluation

In general, from our comparison analysis results, GPT-4 evaluation is aligned with our human evaluation in most of the cases, which supports the argument that GPT-4 evaluation is helpful (Liu et al., 2023; Faysse et al., 2023).

We have also observed that GPT-4 become more "binary thinking" than our human evaluators. GPT-4 tends to output the highest ordinal variable while human evaluators seems more conservative. Those evidences could be found from Table 2 that GPT-4 has higher absolute difference between largest ordinal variable and the second largest one compared to the one produced by human.

There is a discrepancy, though no statistical differences, between GPT-4 and humans in determining whether a question is information-seeking or not. The discrepancy can be attributed to the presence of 'anything else' questions in WD. When GPT-4 assesses 'anything else' questions, it cate-

gorizes them as information-seeking with 100%, while other questions have a 98.2% chance of being considered as information-seeking. In contrast, humans treat 79.6% of 'anything else' questions as information-seeking, while 88.0% of the remaining questions are also considered information-seeking. Additionally, the disagreement may arise from the fact that 'anything else' questions often lead to more vague or general answers in the training data for GPT-4 causing it to classify such questions as less information-seeking.

6 Application to Conversational Retrieval

As mentioned earlier in the introduction section, there are few ways to utilize the generated dataset, for e.g., use the dataset to train chatbot directly or transform it as a open-domain conversation retrieval task. In this section, we will focus on open-domain conversational retrieval, as datasets such as OR-QuAC (Qu et al., 2020), Trec-CAsT-19 (Dalton et al., 2020), and Trec-CAsT-20 (Dalton et al., 2021) exist in this domain.

A ConvQA system interacts with a user in a multi-turn dialogue, where the user primarily asks questions and the system responds (with occasional exceptions, such as the system asking for clarification). When it is the system's turn to speak at a given time t, it considers the entire dialogue history, comprising all previous turns, and generates a new utterance as its response. This work focuses on the conversational retriever, showing how to improve it by pre-training on our dataset comparing to WD, leaving improvements to the generator for future work.

6.1 Dual Encoder

In our methodology, we employ a standard dual encoder as described in Ni et al. (2022). The objectives involve optimizing for a combination of factors: maximizing the similarity between a query $\mathbf{q}$ and its corresponding positive passage $\mathbf{p}^*$, while simultaneously minimizing the similarity between query q and all of its negative samples $\mathcal{N}(\mathbf{p})$. This is achieved through the following loss function:

$$l(\boldsymbol{\theta}) = -\log \frac{\exp(s_{\boldsymbol{\theta}}(\mathbf{q},\mathbf{p})/\tau)}{\sum_{\mathbf{p}\in\mathbf{p}^*\cup\mathcal{N}(\mathbf{p})}\exp(s_{\boldsymbol{\theta}}(\mathbf{q},\mathbf{p})/\tau)} \tag{6}$$

Here, $s_{\boldsymbol{\theta}}(\mathbf{q},\mathbf{p})$ represents a standard cosine similarity, defined as:

$$s_{\boldsymbol{\theta}}(\mathbf{q},\mathbf{p}) = \frac{\text{embed}_{\boldsymbol{\theta}}(\mathbf{q})^\intercal \text{embed}_{\boldsymbol{\theta}}(\mathbf{p})}{\|\text{embed}_{\boldsymbol{\theta}}(\mathbf{q})\|\|\text{embed}_{\boldsymbol{\theta}}(\mathbf{p})\|} \tag{7}$$

In this context, $\text{embed}_{\boldsymbol{\theta}}$ refers to an embedding model used to map text into a fixed-dimension embedding vector. We adopt the setup outlined in Dai et al. (2022), utilizing a pre-trained T5-LARGE encoder as the embed model.

6.2 Datasets

The entire WD dataset consists of a total of 11.3 million dialogs. This comprehensive dataset is divided into 100 separate sections, commonly referred to as "dumps." However, due to the sheer size of the WD dataset, utilizing all of it for demonstration purposes can be overwhelming. Therefore, to simplify the demonstration and still provide meaningful insights, we are focusing only on the first five dumps, labeled as #00000 through #00004. These chosen dumps are used to create the RQ and WQ datasets. Notably, each dump from the RQ and WQ datasets represents 1% of the entire WD dataset. For readers interested in a more detailed statistical comparison between our generated datasets (RQ and WQ) and the original WD dataset, please refer to Appendix Table A.2.

6.3 Two-Stage Training Strategy

We employ a two-stage training approach for our dual encoder. Figure 4 provides more details of our training strategy.

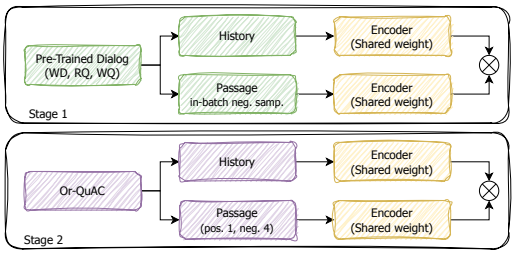

Figure 4: Our dual encoder employs a two-stage training approach. $\otimes$ represents the cosine similarity. Initially, the T5-encoder is pre-trained using the generated dataset (WD, RQ, WQ). Subsequently, it is fine-tuned using the OR-QuAC dataset.

In the first phase, we use our generated dataset, comprising WD, RQ, and WQ, to train embed_{θ}. This is accomplished by implementing in-batch negative sampling, whereby the positive passage for a given instance, represented by i, is treated as a

negative sample for all other instances in the same batch that do not equal to i. This in-batch negative sampling strategy is an efficient, straightforward method to train our sentence embedding model. To summarize our initial training phase, we used the aforementioned datasets with a mini-batch size of 8, a learning rate of 1×10^{-4}, an iteration step of 500 for each 1% subset, and the accumulated gradient batches parameter set to 32 and AdamW optimizer (Loshchilov and Hutter, 2017).

In the second phase, we fine-tune the model using OR-QuAC training set. For simplicity and in the interest of fair comparison, we've chosen not to incorporate the mutli stage hard sampling method outlined in Dai et al. (2022). Instead, we only use the annotated positive sample and negative samples in the dataset. Each model employed here was trained using the following parameters: a mini-batch size of 16, a learning rate of 1×10^{-4}, an iteration step of 250, and the AdamW optimizer for weight adjustments. These parameters were chosen to ensure the robustness of our models.

6.4 Results

System	OR-QuAC MRR@5	CAsT-19 MRR	CAsT-20 MRR
BM25-QR	20.2	58.1	25.0
ANCE-QR	45.7	66.5	37.5
ConvDR	61.6	74.0	50.1
T5-Large DE	57.3	61.1	34.5
+ 1% WD	60.1 (0.53)	62.6 (0.41)	37.2 (0.85)
+ 1% RQ	60.9 (0.71)	63.5 (0.92)	37.4 (0.40)
+ 1% RQ + 1% WQ	62.1 (0.18)	64.3 (0.69)	38.2 (0.34)
+ 5% RQ + 5% WQ	62.8	64.7	38.5

Table 3: Performance analysis of a Retrieval-Only Dual Encoder on the OR-QuAC, Trec-CAsT-19 and Trec-CAsT-20. Key abbreviations include DE for Dual Encoder, WD for WikiDialog Dataset, RQ for generated questions, and WQ for rewritten questions. For subsets labeled with '1%', results from five distinct experiments are averaged and presented using mean (standard deviation). The Retrieval-Only Dual Encoder, fine-tuned on our generated datasets RQ and WQ, demonstrates superior performance compared to the Retrieval-Only Dual Encoder fine-tuned on the original WD dataset and other existing baselines.

During our experimentation, we discovered that using a small portion of the dataset for training can still improve the performance of the dual encoder. This finding offers valuable insights for optimizing computational efficiency without substantially compromising the effectiveness of the model. For example, in our cases, finetuning on 1% of RQ can lead to an average 0.8% performance improvement comparing to using the same WD for OR-QuAC dataset. Upon analysis, our generated datasets (RQ and WQ) consistently exhibited superior performance when compared to the original WD dataset in the context of open-domain conversational retrieval tasks, and this superiority is statistically significant.

In addition, we evaluated our pre-trained dual-encoder retrievers in comparison to three well-known retrieval-only benchmarks: BM25-T5QR by Wu et al. (2022), ANCE-Query Rewriter by Yu et al. (2020) and ConvDR by Yu et al. (2021). Our approach outperforms all of these existing baselines. *This conclusion not only confirms the quality of our data generation methods but also highlights their potential applicability and utility in enhancing the model's ability to address the inherent complexities of such tasks.* Additionally, as we increase the size of the fine-tuning dataset from 1% to 5% percent, the performance improves further.

7 Conclusion

In this paper, we introduce SynCARS, a novel approach that generates high-quality synthetic conversations from unlabeled documents by leveraging ConvQA datasets. Our method outperforms existing benchmarks in terms of the quality of the generated conversations. To generate high-quality synthetic data, we developed an answer segmentation technique that incorporates a special token and curated a new dataset. Despite using a smaller model and lower compute, human evaluations show that our generated dataset surpasses the WikiDialog dataset in terms of answer quality and question specificity. This demonstrates the effectiveness of the proposed approach. Moreover, the question retrieval system trained on our dataset outperforms both the standard retrieval-only benchmark and the same model trained using the WikiDialog dataset. We believe our contributions will facilitate future progress in the development of document-based conversational systems.

Limitations

In our computational setup, we employed machines that were equipped with 8 $\times$ V100 GPUs to train the FLAN-T5-XL model. This training process involved using a mixed dataset comprising three public datasets. To give an estimate of the time required, training the FLAN-T5-XL model typically takes approximately one day. These time estimates

highlight the significant computational resources needed for training both models effectively.

For the retrieval experiments, it is important to note that the performance of the in-batch negative sampling strategy is significantly influenced by the mini-batch size. As the size of the mini-batch increases, model performance typically improves, since each iteration introduces more negative samples. However, due to constraints on the computational resources, we had to keep the mini-batch size relatively small.

In addition, we are unable to replicate the experiment conducted by Dai et al. (2022) and train our model using a full dataset (either 100% WD or 100% RQ and WQ) due to our limited computing resources. It is important to note that this limitation may impact the validation of the model's effectiveness. The absence of evidence in this regard leaves uncertainty about the model's performance in this context.

Ethics Statement

This paper does not present any ethics-related issues. The data and additional resources utilized in this work are open-source and widely used in existing research.

References

Raviteja Anantha, Svitlana Vakulenko, Zhucheng Tu, Shayne Longpre, Stephen Pulman, and Srinivas Chappidi. 2021. Open-domain question answering goes conversational via question rewriting. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Junwei Bao, Duyu Tang, Nan Duan, Zhao Yan, Yuanhua Lv, Ming Zhou, and Tiejun Zhao. 2018. Table-to-text: Describing table region with natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Yu Chen, Lingfei Wu, and Mohammed J Zaki. 2023. Toward subgraph-guided knowledge graph question generation with graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Mike Conover, Matt Hayes, Ankit Mathur, Xiangrui Meng, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free Dolly: Introducing the World's First Truly Open Instruction-Tuned LLM.

Zhuyun Dai, Arun Tejasvi Chaganty, Vincent Y Zhao, Aida Amini, Qazi Mamunur Rashid, Mike Green, and Kelvin Guu. 2022. Dialog inpainting: Turning documents into dialogs. In *International Conference on Machine Learning*, pages 4558–4586. PMLR.

Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. 2020. Trec cast 2019: The conversational assistance track overview. *arXiv preprint arXiv:2003.13624*.

Jeffrey Dalton, Chenyan Xiong, and Jamie Callan. 2021. Cast 2020: The conversational assistance track overview. In *In Proceedings of TREC*.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1342–1352.

Manuel Faysse, Gautier Viaud, Céline Hudelot, and Pierre Colombo. 2023. Revisiting instruction fine-tuned model evaluation to guide industrial applications. *arXiv preprint arXiv:2310.14103*.

Daya Guo, Yibo Sun, Duyu Tang, Nan Duan, Jian Yin, Hong Chi, James Cao, Peng Chen, and Ming Zhou. 2018. Question generation from SQL queries improves neural semantic parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1597–1607, Brussels, Belgium. Association for Computational Linguistics.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Gangwoo Kim, Sungdong Kim, Kang Min Yoo, and Jaewoo Kang. 2022. Generating information-seeking conversations from unlabeled documents. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2362–2378.

Yikang Li, Nan Duan, Bolei Zhou, Xiao Chu, Wanli Ouyang, Xiaogang Wang, and Ming Zhou. 2018. Visual question generation as dual task of visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6116–6124.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. *arXiv preprint arXiv:2301.13688*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375*.

Boyuan Pan, Hao Li, Ziyu Yao, Deng Cai, and Huan Sun. 2019. Reinforced dynamic reasoning for conversational question generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2114–2124, Florence, Italy. Association for Computational Linguistics.

Chen Qu, Liu Yang, Cen Chen, Minghui Qiu, W Bruce Croft, and Mohit Iyyer. 2020. Open-retrieval conversational question answering. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 539–548.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Vasile Rus, Brendan Wyse, Paul Piwek, Mihai Lintean, Svetlana Stoyanchev, and Christian Moldovan. 2010. The first question generation shared task evaluation challenge. In *Proceedings of the 6th International Natural Language Generation Conference*.

Zeqiu Wu, Yi Luan, Hannah Rashkin, David Reitter, Hannaneh Hajishirzi, Mari Ostendorf, and Gaurav Singh Tomar. 2022. Conqrr: Conversational query rewriting for retrieval with reinforcement learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10000–10014.

Shi Yu, Jiahua Liu, Jingqin Yang, Chenyan Xiong, Paul Bennett, Jianfeng Gao, and Zhiyuan Liu. 2020. Few-shot generative conversational query rewriting. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 1933–1936.

Shi Yu, Zhenghao Liu, Chenyan Xiong, Tao Feng, and Zhiyuan Liu. 2021. Few-shot conversational dense retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on research and development in information retrieval*, pages 829–838.

Appendix

A.1 Choice of the base model

We conducted an ablation study to check which base model to use. Table A.1 shows the result for GPT-4 evaluation. Using either T5-XL or FLAN-T5-XL, our method produces successful results in the GPT-4 evaluation. We also added LLAMA-7B based on zero-shot setting as baseline method for reference.

Model	**T5-XXL**	**T5-XL**		**FLAN-T5-XL**		**LLAMA2-7B**
Dataset	**WD**	**RQ**	**WQ**	**WD**	**RQ**	**0-shot**
Is the question information seeking?						
Yes (%)	99.0	100.0	100.0	99.3	100.0	88.8
How relevant is question to the conversation?						
Not at all (%)	2.4	1.5	0.0	4.9	0.0	4.0
Topic only (%)	26.3	15.0	22.1	20.3	22.0	23.7
Follows up (%)	71.2	83.5	77.9	74.8	78.0	71.8
How specific is the question?						
Not at all (%)	10.6	1.1	1.2	1.0	0.6	12.0
Somewhat (%)	31.0	30.1	19.1	24.4	16.9	35.6
Very (%)	58.4	68.9	79.7	74.6	82.5	52.0
How well answered is the question?						
Not at all (%)	11.8	11.5	9.9	13.0	9.0	17.3
Incompletely (%)	38.0	16.9	18.0	22.3	23.7	50.3
Sufficiently (%)	21.6	25.2	22.7	21.7	19.3	20.8
Perfectly (%)	28.6	46.4	49.4	43.0	48.0	11.6

Table A.1: Results from a GPT-4 evaluation of 50 generated dialogs in four variants of our method vs. WikiDialog. Here RQ are generated questions, WQ is rewritten questions and WD is WikiDialog.

A.2 Statistics of the generated dataset

The WikiDialog (WD) dataset consists of a total of 11.3 million dialogues. This dataset is divided into 100 separate partitions for easier handling and processing. We use the first five partitions (#00000 to #00004) to generate new dialogs and label them as RQ and WQ datasets. Appendix Table A.2 shows the statistics of our generated dataset compared to the WD dataset. While there are a few entries for which our model cannot generate outputs in the correct format, the number of dialogs in the RQ and WQ datasets for each partition is slightly fewer than that in the WD dataset. The dialogues in our dataset generally consist of a small number of turns, which aligns with our objective of combining multiple responses to create a single, comprehensive answer.

Stat.	**Part. #**	**WD**	**RQ**	**WQ**
# Dialog	00000	113,678	113,650	113,609
	00001	113,651	113,613	113,574
	00002	113,536	113,498	113,466
	00003	114,286	114,263	114,221
	00004	113,596	113,571	113,542
Avg. Turn	00000	4.93	3.55	3.49
	00001	4.93	3.57	3.50
	00002	4.93	3.56	3.49
	00003	4.93	3.56	3.49
	00004	4.93	3.56	3.50

Table A.2: Statistics of WD, RQ, and WQ datasets. Here, RQ is the dataset with the generated raw questions, WQ is the dataset with rewritten questions and WD corresponds to the WikiDialog dataset.

A.3 Question and Answer Overlapping

Table A.3 shows the overlap between questions and answers using the ROUGE score. We can conclude that our RQ and WQ datasets have higher Rouge scores compared to WD dataset, although the absolute ROUGE scores still indicate a low level of text overlap. This further demonstrates the superior quality of our RQ and WQ datasets.

Dataset	ROUGE-1	ROUGE-2	ROUGE-L
WD	0.111	0.026	0.095
RQ	0.158	0.046	0.130
WQ	0.205	0.080	0.166

Table A.3: ROUGE score for the generated question and the corresponding answers.

A.4 Prompt Template for GPT-4 evaluation

In Table A.4, we show the prompt Template for GPT-4 evaluation. The same rubric is also used for human evaluation.

Question: *Is the question information seeking?*
Prompt Template and Rubric:
Is the QUERY information-seeking based on RUBRIC? Output option only
option:
* Yes
* No
RUBRIC:
* Yes. The user is looking to learn some information from the system. Note: Information-seeking queries don't have to be phrased as questions.
* No. The query is unclear, difficult to understand or not seeking information. Note: Not all questions are information seeking, e.g. questions directed at the system ("how are you", "what do you think") or ones that are nonsensical in the context ("Brian, how is Jill doing?").
CONVERSATION: {conversation}
QUERY: {*query*}
ANSWER: {*answer*}

Question: *How relevant is question to the conversation?*
Prompt Template and Rubric:
How is the QUERY relevant to a CONVERSATION based on RUBRIC? Output option only.
option
* A
* B
* C
RUBRIC:
* A. Follows up on a previous query or response. It is difficult to correctly understand the query without reading the conversation history.
* B. It is difficult to correctly understand the query without reading the conversation history. Only related to the topic of the conversation. The query is topically similar to previous queries or responses, but can be understood without reading them.
* C. Not relevant. The query doesn't appear to be relevant to the topic or a previous query or response. Rule of thumb: if you are surprised by a query, it is probably not relevant.
CONVERSATION: {conversation}
QUERY: {*query*}
ANSWER: {*answer*}

Question: *How specific is the question?*
Prompt Template and Rubric:
How specific is the QUERY based on RUBIC? CONVERSATION is the history context. Only output option text.
option
* Very
* Somewhat
* Not at all
RUBRIC:
* Very. Only a specific answer would satisfy the user. Example: "Why did she make the news in 1999?" likely requires a very specific answer.
* Somewhat. A variety of answers of a specific kind would satisfy the user. Example: While there are many possible answers to "What else does she do?", they are all likely to be a job or activity.
* Not at all. Many topically different answers would satisfy the user. Example: "Tell me something interesting about her." can be answered in many different ways.
CONVERSATION: {conversation}
QUERY: {*query*}
ANSWER: {*answer*}

Question: *How specific is the question?*
Prompt Template and Rubric:
How well does the response ANSWER the QUERY based on RUBRIC? CONVERSATION is history context. Only output option text.
option: * Perfectly
* Sufficiently
* Incompletely
* Not at all
RUBRIC:
* Perfectly. The response completely satisfies the user's information need.
* Sufficiently. The response mostly answers the user's information need, though some additional information could be provided.
* Incompletely. The response provides some information relevant to the user, but doesn't adequately answer the question.
* Not at all. The response does not provide any relevant information for the user's query or is not intelligible.
CONVERSATION: {conversation}
QUERY: {*query*}
ANSWER: {*answer*}

Table A.4: Prompt Template for GPT-4 evaluation.

A.4 Additional Generated Examples

In Figure A.1 and Figure A.2, we show two more examples to compare performance with WD.

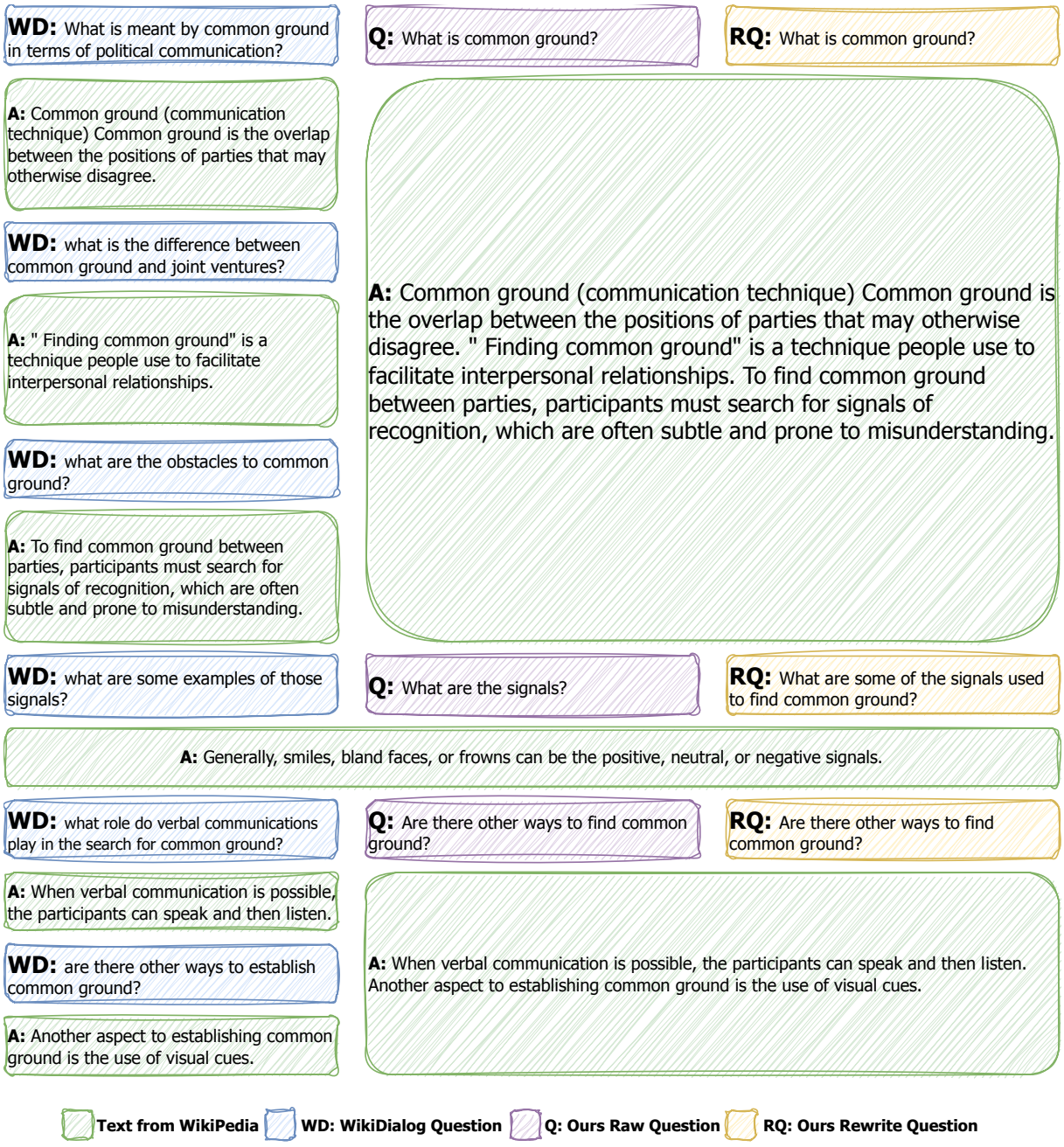

Figure A.1: Comparison examples #1. Here we can find the ability of our proposed method to perform segmentation of the sentences. This link (https://en.wikipedia.org/wiki/Grounding_in_communication) gives the raw Wikipedia web page.

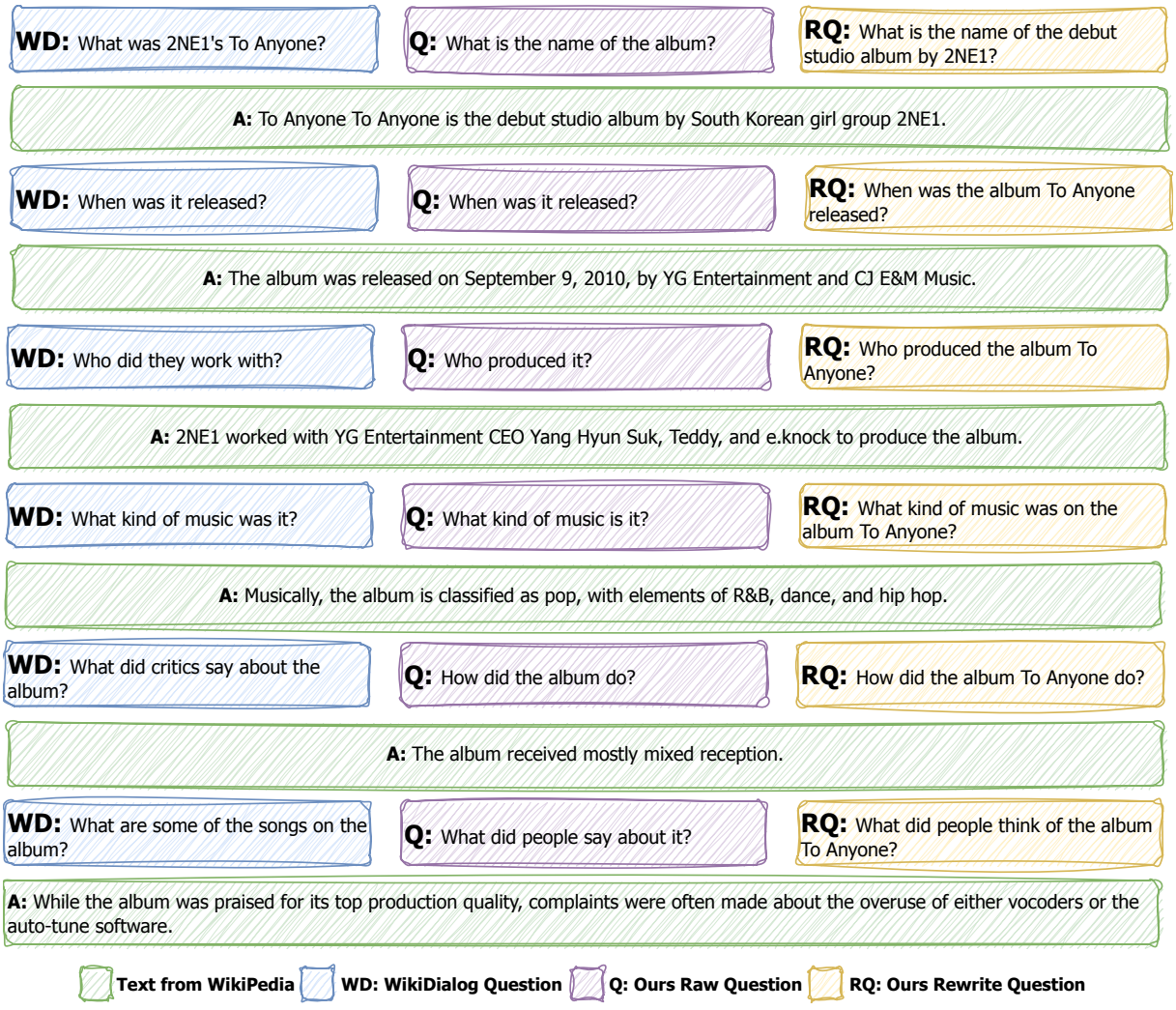

Figure A.2: Comparison example #2. Here we can find a relatively better question generation compared to WikiDialog. This link (https://en.wikipedia.org/wiki/2NE1) gives the raw Wikipedia web page.