Retrieval-Augmented Language Model for Long-Term Conversation via Weakly-Supervised Learning from Perplexity Improvements

Kosuke Nishida, Naoki Yoshinaga, and Masashi Toyoda

Abstract Consistency of the response between the past dialogue is important for the dialogue system so that the system becomes the daily conversation partner. We tackle the long-term open-domain conversation task by focusing on retrieval-augmented language models that retrieve a characteristic utterance from their dialogue history and generate a response by referring to the retrieved utterance. We propose a weakly-supervised training algorithm of the retrieval model for the long-term conversation. Here, to identify the useful past utterance for the user-specific response, we use the improvement of the perplexity when an utterance is fed to the response model as the pseudo labels for the retrieval model training. Experimental results showed that our model generates more consistent responses than baseline models.

1 Introduction

Dialogue systems, such as ChatGPT [22], Apple Siri, and Amazon Echo, attract much attention as the daily conversation partners. Consistency in dialogue systems is important to provide engaging conversations for use cases where users are talking to a dialogue system for a long time [19, 25, 13]. To ensure consistency, previous studies incorporated speaker information in training a dialogue system; examples of such speaker information include ID [19], attributes [25], profile text [43], and role-play-based question-answering [6].

Kosuke Nishida
The University of Tokyo. He also works for NTT Human Informatics Laboratories.
e-mail: nishidak@tkl.iis.u-tokyo.ac.jp

Naoki Yoshinaga and Masashi Toyoda
Institute of Industrial Science, The University of Tokyo.
e-mail: ynaga@iis.u-tokyo.ac.jp, toyoda@tkl.iis.u-tokyo.ac.jp

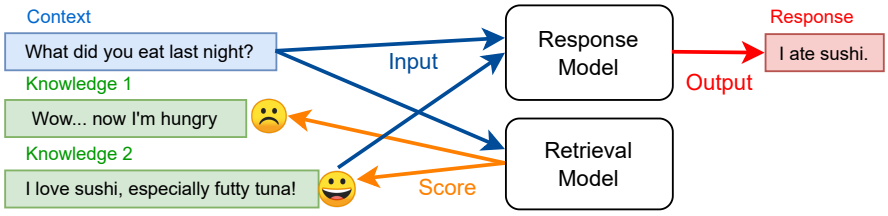

Fig. 1: Overview of our training algorithm for a retrieval-augmented response generation model; The retrieval model extracts knowledge by scoring it and the response model generates a response on the basis of the original context and the retrieved knowledge. Our goal is to train response and retrieval models that refer to and extract knowledge to ensure dialogue consistency. First, the response model is trained to generate the response. Then, we compute the contribution of each knowledge to the response generation. Then, the contribution scores are used for the weakly-supervised learning of the retrieval model so that the retrieval model can predict the contribution scores.

In this study, we focus on the consistency of the response with the past dialogue in the long-term conversation because the speaker's information appears in the dialogue. In this task, it is challenging to extract useful information for the response because there are many utterances in the past dialogue. To solve this problem, previous work [32] performed retrieval-augmented generation that retrieves past useful dialogues from their dialogue history and trains the retriever by assuming utterances in the same session are more useful, which are not always true.

Therefore, we follow the retrieval-augmentation studies and propose the training algorithm tailored for the long-term conversation task, as shown in Fig. 1. Our key idea is to quantify the contribution of knowledge to the response generation by calculating the improvement of the perplexity when knowledge is fed. First, we run n-gram-based knowledge retrieval to train the response model that refers to knowledge. Second, we compute the perplexity improvement for each knowledge with the trained response model. Then, we train the retrieval model in the weak-supervision setting by using the perplexity improvement as the pseudo-label.

To evaluate our method, we focus on conversation logs on microblogs because of their two advantages: (i) they cover broad topics and thus they are useful for the development of open-domain dialogue systems [27, 35, 1], and (ii) they include long-term conversation which is suitable to train and evaluate the personal consistency of the response. Experimental results on an existing large-scale Multi-session Twitter Dialogue Dataset [32] showed that the proposed model generates responses that are consistent with the retrieved knowledge. Also, we observed that our retrieval model extracts knowledge with a higher perplexity improvement score.

Our contributions are summarized as follows:

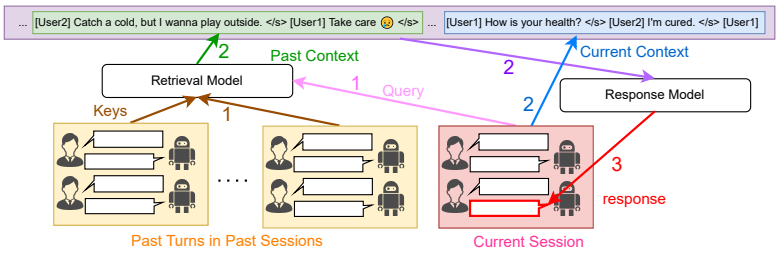

Fig. 2: Notations and the model overview. (1) The retrieval model scores past turns $\{t\}$ in past sessions by using the current session x as a query. (2) The concatenation of the current session x and the retrieved past context t is fed to the response model. (3) The response model generates a response y.

- We propose the training algorithm of retrieval-augmented language models for long-term conversation by focusing on the perplexity improvement caused by the knowledge.
- We confirmed that responses of the proposed model are more similar to the reference response and more consistent with retrieved knowledge than a baseline.
- Qualitative analysis showed that the proposed model extracts the speaker's information and generates a response that reflects the information.

2 Preliminaries

Here, we explain the task definition, the notations, and the model overview in Fig. 2.

2.1 Task Definition and Notations

The long-term conversation task is a task to generate a response for a given past dialogue between the user and the system, which finishes with a user's utterance and involves multiple *sessions* over a long period of time. A session is a sequence of the utterances in the past dialogue and is distinguished into a *current session* and *past sessions*. A *turn* is a pair of a user's utterance and its next system response.

2.2 Long-Term Conversation as Retrieval-Augmented Generation

We cast the above task as a retrieval-augment generation task [40, 32]. We adopt a RALM consisting of two components, a retrieval model and a response model (Figure 2), and the input of them is the current session x. A *retrieval model* uses x_{ret} that is a part of x as a query. The retrieval model has a key-value database in which turn t is converted to a key-value pair. The retrieval model extracts turns as text on the basis of the similarity between the query and the keys in the database. A *response model* generates a response y on the basis of x and the extracted turns as a past context c.

For each response, the retrieval model finds turns (*i.e.,* pairs of user's and system's utterances) related to x_{ret} from the past sessions. The retrieval model is a pre-trained encoder such as BERT [4]. The encoder maps the current dialogue session to $h_q \in \mathbb{R}^d$ and the past turns to $\{h_{k,i}|h_{k,i} \in \mathbb{R}^d, i = 1, \cdots, N_k\}$, where d is the hidden size of the language model and N_k is the number of keys.

The response model outputs the system response. The response model is a pre-trained decoder. The input of the response model, the current context x and the past context c, is converted to a prompt with a pre-defined format. In this paper, we set the number of extracted turns to one for simplification.

3 Related Work

Here, we review retrieval models used for the dialogue system. Then, we discuss previous approaches for the long-term open-domain conversation task.

3.1 Retrieval-guided Response Generation

Traditional dialogue systems have introduced a retrieval-based model to extract a response candidate from dialogue corpora. Previous work [11, 7] returned the retrieved response as is. Since their responses are limited to those in the corpora, the generative models are used for the dialogue systems [29, 34]. However, the generative models suffer from the safe response problem: the models generate a generic and dull response (*e.g.,* "I don't know.") [18]. To solve the problem, retrieval models are used to guide generation [31, 37, 42]. Although those studies used general dialogue corpora and textual knowledge bases as the retrieval pool, we retrieve the user-specific knowledge from past sessions to personalize responses.

3.2 Long-term Open-domain Conversation

In the long-term open-domain conversation task, we must process large amounts of past utterances and extract useful information from them. Previous studies can be classified into two types: summarization-based models and retrieval-based models.

Among summarization-based models, some work [41, 2] provides the persona information for the model and edits them as the dialogue progresses. Other work [40] generates the dialogue summaries of past sessions and feeds them to the model. However, those studies require an additional annotation to the dialogue corpus.

Retrieval models are based on retrieval-augmented language models [5, 17, 30, 3, 10]. Previous work [32] proposed the retrieval model designed for the task, which is trained with the triplet loss by distinguishing the utterances in the same session and another session. Other work [16] used a large language model (LLM) as is with various techniques such as chain-of-thought [36]. It used a pre-trained dense passage retriever [12] as the retrieval model. We revise the training algorithm of the retrieval model by utilizing the difference of the perplexity of each utterance with and without past utterance as the supervision of the usefulness of the past utterance.

4 Baseline: KeyNext Retriever

Here, we describe the baseline model and its training method [32]. Then, we provide our analysis of the baseline model and explain the motivation of our algorithm.

4.1 Model

Previous work [32] proposed the KeyNext retriever which encodes a user's utterance in a turn t to a key state $h_k \in \mathbb{R}^d$. Also, the query x_{ret} is defined as the current user's utterance and encoded to h_q. In the inference phase, the model retrieves the previous turn that minimizes the distance between the key and query states.

4.2 Training

First, the retrieval model is trained with a triplet loss [26],

$$L_{\text{triplet,L2}} = \max(\|h_q - h_p\|_2 - \|h_q - h_n\|_2 + \varepsilon, 0), \tag{1}$$

where $h_p, h_n \in \mathbb{R}^d$ are the key embeddings of a positive utterance and negative utterance, respectively. $\varepsilon = 1$ is the minimum margin required.

The KeyNext retriever is trained by using the topic consistency The KeyNext retriever is trained to distinguish whether utterances belong to the same session or not. The query utterance x_{ret} is randomly selected from the dataset. The positive utterance is the user's randomly selected utterance in the query's session. The negative utterance is randomly selected from the user's utterances in the query's past sessions.

Then, the response model is trained with a cross-entropy loss in the teacher-forcing fashion:

$$L_{\mathrm{CE}}(x,y,c) = -\frac{1}{|y|}\sum_{t=1,\cdots,|y|}\log f(y_t|x,y_{1:t-1},c). \tag{2}$$

4.3 Analysis

In our pilot experiments, we clarified that the past turns retrieved by KeyNext do not always assist the response generation.

Here, we introduce a new metric, the perplexity improvement (PI) score. The PI score is defined as

$$\mathrm{PI}(t|x,y) = L_{\mathrm{CE}}(x,y,c=\phi) - L_{\mathrm{CE}}(x,y,c=t). \tag{3}$$

We note that $L_{\mathrm{CE}}(x,y,c=\phi)$ is the cross-entropy loss of the response model without the past context. This model is only fed x in both the training and inference phases.

We calculate the PI score of an instance (x,y,t) in the evaluation data to estimate the importance of a turn t to generate the reference response y from the input x. A larger PI score indicates that t reduced the likelihood of generating y more. Also, a negative PI score indicates that t obstructed the generation. Because the perplexity, one of the major metrics in the dialogue system, is the exponential of L_{CE}, the PI score measures the difference between the logarithm of perplexity. This metric is inspired by Toolformer [28], which is a pre-trained language model (PLM) using external text-based APIs. The authors created its pre-training corpus by inserting the special tokens indicating each API call into the original text. Whether or not to insert special tokens was determined by the difference in the perplexity score before and after the insertion. Although we used perplexity improvement, our method and analysis can be extended to other metrics. For example, an external natural language inference model can be used to evaluate the consistency between the response and the persona [44].

Here, we summarize our experiments, and the details are given in §6. We extract a turn t_{KN} for (x,y) with the KeyNext retriever in the development set. The histogram of the PI scores is shown in Figure 3.

We found that even among the turns with the highest KeyNext scores, 36.7% of them obstructed the generation of the reference response y. We consider that this phenomenon is due to the diversity of the conversation and not to the KeyNext

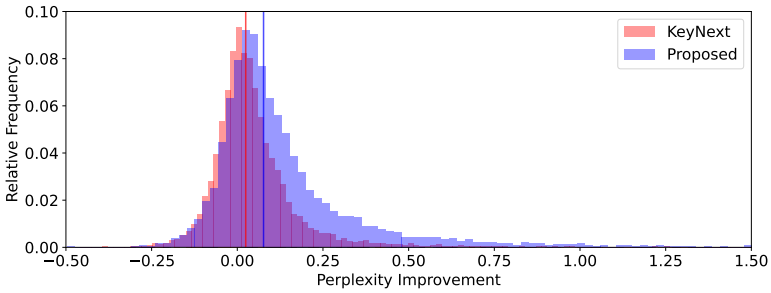

Fig. 3: Histogram of the perplexity improvement. The vertical lines indicate the median. The ratio of positive instances is 63.3% in the KeyNext baseline and 78.4% in the proposed model.

retriever; That is, if t_{KN} encourages one of interesting responses, the reference response y is often irrelevant to t_{KN}. For instance, a user, who said that his favorite food is sushi in the past sessions, may report that his dinner is a hamburger today in the reference response.

The above observation raised our hypothesis: The dataset consisting of (x, y, t_{KN}) could not train the response model that leverages the past context sufficiently because 36.7% of the dataset leads to ignoring the past context to generate the response.

5 Proposed Training Algorithm

In this section, we describe the proposed model. The proposed model consists of two components. First, we propose a method to automatically create the dataset that trains a response model referring to a past context. This method is motivated by our hypothesis discussed in §4.3. Second, we train a retriever using the PI score as a supervision that teaches the model what turn is useful for generation by definition.

5.1 Response Model Training

Our motivation in this subsection is to create dataset $\{(x, y, t)\}_{(x,y)\in\mathscr{D}_{\text{Train}}}$ with which we can train the response model that refers to a past context to generate the response, where $\mathscr{D}_{\text{Train}}$ is the training data. Thus, we used an oracle retriever to extract the most similar system utterance u_t in turn t to the reference response y, because the information of the reference response is available to create the training dataset. As the metric of the similarity, we use the coverage of the bag-of-words $\text{BoW}(\cdot)$:

Algorithm 1 Proposed Training Algorithm

1: Train a retrieval model like KeyNext on half of the training data
2: Create training data for the response model with the BoW coverage and the KeyNext retriever
3: Predict responses for the rest of the training data
4: Calculate the PI scores for the generated response
5: Train the retrieval model on the PI ranking task
6: Generate responses with the trained response and retrieval models

$$t_{\mathrm{BoW}} = \operatorname{argmax}\left(\mathrm{BoW}(u_t) \cap \mathrm{BoW}(y)\right). \tag{4}$$

This formulation is designed so that the model learns to copy words from u_t, which may include the system's character information and character-specific expression. If tied utterances exist, we use the turn with a higher KeyNext score.

5.2 Retrieval Model Training

Our motivation in this subsection is to create dataset $\{(x,y,t)\}_{(x,y)\in\mathscr{D}_{\mathrm{Eval}}}$ with which the response model can generate a consistent and interesting response, where $\mathscr{D}_{\mathrm{Eval}}$ is the evaluation data.

We train the retrieval model with the perplexity improvement ranking task. We define the positive and negative utterance by their PI scores $\mathrm{PI}(t|x,y)$, and thus the model can learn fine-grained signal that measures the extent to which t affects the generation. The loss function is a triplet loss with dot product similarity:

$$L_{\mathrm{triplet,DP}} = \max(h_q^\top h_n - h_q^\top h_p + \varepsilon, 0). \tag{5}$$

In the implementation of this task, we found that the model cannot learn this task when we use the reference response for y. This is because the reference responses in the dataset often require no past context due to the diversity of the conversation, which is a similar phenomenon discussed in §4.3. To solve this problem, we use the predicted response $\hat{y}$ in the PI score by feeding (x,t) to the trained response model. We expect that the predicted response has a relation to t because the response model is trained to copy u_t through our training algorithm.

In addition, we split the training data D_{Train} into two splits to avoid leakage through the response model training. We use one split for training the response model. Then, we generate $\hat{y}$ from (x,t) in the other split with the response model. We calculate the PI score of $(x,\hat{y},t)$ and use the resulting PI score to define the ranking task.

Table 1: Data statistics.

	Train	Dev.	Test
# Episodes	60,000	1778	2682
Periods	2011 – 2017	2018	2019
Average of # sessions in a episode	15.92	15.41	15.49
# Current utterances	150,747	4666	7113
Average of # turns in a session	6.65	6.89	6.89
Average of # tokens in a session	146.37	151.51	146.66

6 Experiments

6.1 Dataset

We used the Multi-session Twitter dialogue dataset (MSTD) [32], which is built from X (formerly, Twitter). Table 1 shows the statistics of the dataset. They defined a reply tree as a dialogue session and all sessions between two specific users as an *episode*. The training, development, and test data were split on the basis of the collection period. The speakers in each split did not overlap with the other splits. The dataset includes episodes only with 11-25 sessions, each of which consists of 5-30 turns. Following [32], we used the last session as the current session and the rest sessions as the past sessions.

6.2 Metrics

To evaluate the whole model, we measured the similarity between the predicted response and the reference response with ROUGE-1/2/L [21][1] and BLEU-2/3 [23].[2] We also evaluated the perplexity of the reference response generation. Moreover, we measured the similarity between the predicted response and the retrieved response in each model with the same metrics. These metrics represent the extent to which the response model refers to the past context for the generation. In the above metrics, we used the MeCab tokenizer with UniDic 2.1.2 [15] as the previous work did [32].

To evaluate the retrieval model, we used recall@1 as the metric.

[1] `https://github.com/pltrdy/rouge`

[2] `https://www.nltk.org/_modules/nltk/translate/bleu_score.html`

Table 2: Hyperparameters.

	Retrieval Model	Response Model
Batch size	512	64
# Epochs	3	1
Max length	128	512
Learning rate	5e-5	1e-4

6.3 Compared Models

To evaluate the whole model, we used two baseline models for comparison. The **No Past** model does not use any information of past sessions. The **KeyNext** model [32] is described in §4.1. We note that the original paper reported that the KeyNext model outperformed a naive baseline that retrieves the most recent session as the past context. For a fair comparison, to train the response model in the No Past and KeyNext models, we did not split the training dataset and used all of it for training.

Implementations. The backbone PLM of the retrieval model was the pre-trained Japanese BERT-base-uncased model [4],[3] and that of the response model was the pre-trained Japanese Transformer decoder [33][4] with 3.6B parameters. We fine-tuned the full parameters of the retrieval model. For the response model, we added and fine-tuned LoRA parameters [8] with fixing all parameters of the response model. For the LoRA modules, we set the target modules to the key, query, and value of linear layers in the self-attention, lora r to 8, lora α to 16, and lora dropout ratio to 0.05.

The training of the retrieval model and the response model took 10 minutes and 1 hour on eight NVIDIA A6000 (48GB) GPUs, respectively. For computational efficiency, we used 100,000 (x, y) instances and top-3 past turns per instance. Thus, we obtain three $\hat{y}$ and $\mathrm{PI}(x, \hat{y}, t)$ scores for each x. That is, the size of the retrieval pool for each x in the PI ranking task is three. The total time consumption to obtain them was 30 hours. The hyperparameter settings are listed in Table 2. We used the Adam optimizer [14], PyTorch (ver. 1.13.1) [24],[5] and transformers (ver. 4.33.2) [38].[6]

The input format is '*The following conversation is between A and B on Twitter:* {Past Context} *In reference to this conversation, write a tweet following the next conversation:* {Current Context}' for the proposed and KeyNext model.[7] For the NoPast model, we used '*The following conversation is between A and B on Twitter. Write a tweet following the next conversation:* {Current Context}.' The format of the context repeats '*A:* {Utterance} *B:* {Utterance} ...'

[3] https://huggingface.co/cl-tohoku/bert-base-japanese-whole-word-masking

[4] https://huggingface.co/line-corporation/japanese-large-lm-3.6b

[5] https://pytorch.org/

[6] https://github.com/huggingface/transformers

[7] The original prompts are written in Japanese, and we translated them to English.

Table 3: Main results and ablation studies. PPL indicates perplexity, R-1/2/L does ROUGE-1/2/L, and B-2/3 does BLEU-2/3, respectively.

		Response Similarity					Knowledge Consistency				
	PPL	R-1	R-2	R-L	B-2	B-3	R-1	R-2	R-L	B-2	B-3
NoPast	31.05	13.55	4.30	12.01	4.83	3.40	—	—	—	—	—
KeyNext	**30.16**	14.20	4.84	12.59	5.35	3.89	13.03	4.57	11.66	5.40	4.05
Proposed	31.23	14.09	5.02	12.51	5.44	4.00	**19.49**	**8.23**	**17.38**	**9.23**	**6.74**
– Response Model	30.18	**14.24**	4.98	12.59	5.43	3.95	13.69	4.95	12.18	5.82	4.42
– Retrieval Model	30.30	14.21	**5.10**	**12.65**	**5.51**	**4.05**	18.82	7.76	16.90	8.52	6.08

6.4 Main Results

Table 3 shows the main results.

Does the retrieval augmentation improve the response similarity? The proposed model and KeyNext model outperformed the NoPast model on the response similarity metrics. We consider that the past context effectively guided the generation.

Does the proposed model improve knowledge consistency? The proposed model outperformed the KeyNext model on the knowledge consistency metrics. The proposed model refers more to the past context than the KeyNext model, which resulted in responses that are more consistent with the past sessions. In other words, we can control the response generated by the proposed model by providing the past context.

Does the proposed model improve the response similarity? The response similarity of the proposed model is comparable with that of the KeyNext model. This result is not surprising because the past context t with a higher PI score of $(x, \hat{y}, t)$ does not always promote the reference response y by definition.

6.5 Ablation Studies

Table 3 also shows the results of the ablation studies. In each ablation, we replaced the corresponding model in the proposed model with that in the KeyNext baseline. All models achieved comparable performance on the response similarity metrics.

Does our response model improve the knowledge consistency? Our response model contributed to the knowledge consistency. Our response model was trained with the past turn highly overlapped with the reference response. Thus, our response model learned to refer to the past context.

Does our retrieval model training improve the knowledge consistency? The proposed model outperformed the model without our retrieval model on all metrics of knowledge consistency. Our retrieval model was trained to retrieve turns with high

Table 4: Results on the perplexity improvement ranking task. The size of the retrieval pool is three.

	Recall@1
Chance Rate	33.33
Proposed	56.25
$-\hat{y}$ generation	32.94

PI scores. We consider that such turns were easy for the model to use as guidance to generate responses.

6.6 Evaluation as the Retrieval Model

Here, we evaluate our retrieval model on knowledge retrieval tasks.

Does our retrieval model retrieve past turns with high PI scores? Fig. 3 shows the histograms of the PI scores. The proposed and KeyNext models retrieve a turn t from the past sessions by using x as a key and then compute the PI score to the reference y. We found that the proposed model retrieved turns with higher PI scores. Although both retrievers were agnostic of the reference response, the turns performed as guidance to generate the reference response. We consider that the proposed retrieval model learned to retrieve turns that include user-specific characteristics (*e.g.,* personal information and a favorite phrase). Qualitative analysis in §6.8 shows an example where the user's name was retrieved and generated.

Can we define the perplexity improvement ranking task for the reference response? The proposed retrieval model training consists of three steps: the generation of $\hat{y}$ by the response model, the calculation of the $PI(t|x,\hat{y})$, and the ranking task itself. Therefore, our question is whether we can remove the first step and directly rank the $PI(t|x,y)$. Table 4 shows the results. We found that the performance of the model was comparable to the chance rate on the perplexity improvement ranking task for the reference response. We consider that this is because of the diversity of the dialogue. That is, the model, the inputs of which are only x and t, cannot distinguish a turn t that leads to the unknown reference response y because a turn t that is irrelevant to y can lead to other possible responses. Meanwhile, on our ranking task, the model outperformed the chance rate because the model is required to predict the contribution of a turn t to the generation of $\hat{y}$ that is actually generated by the response model from t.

Table 5: Human evaluation results. All results were not significant ($p > 0.1$) by the Wilcoxon signed rank test.

	Consistency	Coherence	Engagingness
Baseline	4.73	4.10	3.71
Proposed	4.71	4.01	3.71

6.7 Human Evaluation

For human evaluation, we asked three annotators to evaluate system responses on a scale from one to five (poor, marginally poor, fair, marginally good, good). The annotators are graduate students and men in their 20s and 30s. We sampled 100 responses in the last turn of the last session from development episodes. We evaluated the KeyNext baseline and the proposed model. In each example, we provided the current session and the system response to annotators. The metrics were three-fold:

Consistency Whether the response does not include a grammatical error or contradiction inside the response [20].

Coherence Whether the response is relevant to and continuous with the context [39].

Engagingness Whether the response includes interesting content [43].

We also asked whether annotators rated a response with confidence. We finally obtained 92 responses after filtering unconfident responses.

Table 5 lists the results. All results were not significant. Fleiss's κ were 0.72, 0.42, and 0.34 on the consistency, coherence, and engagingness, respectively.

In the human evaluation, we could not evaluate the coherence between the past sessions and the response because a response has 14.49 past sessions on average and it is difficult and burdensome to read all of them and evaluate the response on the basis of them. Thus, the proposed model did not outperform the KeyNext baseline on all metrics. We also found that the consistency and the coherence were higher than four (marginally good) on average. The dialogue system has achieved a certain level due to the advent of LLMs [9], and the development of the evaluation methods is an important direction for future work.

6.8 Qualitative Analysis

Here, we describe the typical results of the proposed model. Table 6 shows an example of the generation. In the proposed model, the retrieval model extracted the turns containing speaker-specific features, such as names, emoticons, and habits of saying, while the response model copied them. Thus, the proposed model generated responses that included more information. Because the proposed model generated informative responses, the generation occasionally produced a hallucination, which would cause low consistency and coherence in the human evaluation.

Table 6: Qualitative analysis. U and S indicate the user and the system, respectively. The original data is in Japanese, and we translate it to English. Emoticons and emojis are omitted.

Context	S: I want to work hard on the new book in February and make it more pages long, but... do you think I'll have enough time? U: How many pages? S: Thirty-eight to forty pages... No, that's a lot for me...! U: It's easy for you, Riu-chan!
Reference	S: What!? I can't do it!
Baseline	
Retrieved turn	U: If it's all right with Riu-chan, we'll do it later! S: Wow! You too! Let's do it next time!
Generation	S: Thanks! I'll do my best!
Proposed	
Retrieved turn	U: Riu-chan, good morning! S: Uru-chan, good morning! Are you feeling okay? Don't take it too hard.
Generation	S: Oh, oh... I'll do my best... Take care of yourself too, Uru-chan.

7 Conclusion

We studied the consistency of the response with the past dialogue sessions in the long-term conversation. We proposed the training algorithm of the retrieval-augmented language models by defining the perplexity improvement as the usefulness of past utterances for response generation. Experimental results showed that the proposed model retrieved a past turn that improved the perplexity and generated a response that is consistent with the retrieved knowledge. However, it did not increase the human evaluation score because the consistency with the whole past session is difficult to evaluate. The development of evaluation methods in terms of advanced dialogue skills is an important direction for future work.

Acknowledgements This work was partially supported by the special fund of Institute of Industrial Science, The University of Tokyo, by JSPS KAKENHI Grant Number JP21H03494, JP21H03445, and by JST, CREST Grant Number JPMJCR19A, Japan.

References

1. Al-Rfou, R., Pickett, M., Snaider, J., Sung, Y.H., Strope, B., Kurzweil, R.: Conversational contextual cues: The case of personalization and history for response ranking. arXiv preprint arXiv:1606.00372 (2016)
2. Bae, S., Kwak, D., Kang, S., Lee, M.Y., Kim, S., Jeong, Y., Kim, H., Lee, S.W., Park, W., Sung, N.: Keep me updated! memory management in long-term conversations. In: Findings of the Association for Computational Linguistics: EMNLP 2022, pp. 3769–3787 (2022). URL https://aclanthology.org/2022.findings-emnlp.276

3. Borgeaud, S., Mensch, A., Hoffmann, J., Cai, T., Rutherford, E., Millican, K., Driessche, G.v.d., Lespiau, J.B., Damoc, B., Clark, A., et al.: Improving language models by retrieving from trillions of tokens. arXiv preprint arXiv:2112.04426 (2021). DOI https://doi.org/10.48550/arXiv.2112.04426
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: NAACL, pp. 4171–4186 (2019). DOI 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423
5. Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M.: Retrieval augmented language model pre-training. In: ICML, pp. 3929–3938 (2020). URL https://proceedings.mlr.press/v119/guu20a.html
6. Higashinaka, R., Mizukami, M., Kawabata, H., Yamaguchi, E., Adachi, N., Tomita, J.: Role play-based question-answering by real users for building chatbots with consistent personalities. In: Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue, pp. 264–272 (2018). DOI 10.18653/v1/W18-5031. URL https://aclanthology.org/W18-5031
7. Hu, B., Lu, Z., Li, H., Chen, Q.: Convolutional neural network architectures for matching natural language sentences. In: Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, NIPS'14, p. 2042–2050 (2014)
8. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: LoRA: Low-rank adaptation of large language models. In: International Conference on Learning Representations (2022). URL https://openreview.net/forum?id=nZeVKeeFYf9
9. Hudeček, V., Dusek, O.: Are large language models all you need for task-oriented dialogue? In: Proceedings of the 24th Meeting of the Special Interest Group on Discourse and Dialogue, pp. 216–228 (2023). URL https://aclanthology.org/2023.sigdial-1.21
10. Izacard, G., Lewis, P., Lomeli, M., Hosseini, L., Petroni, F., Schick, T., Dwivedi-Yu, J., Joulin, A., Riedel, S., Grave, E.: Atlas: Few-shot learning with retrieval augmented language models. arXiv preprint arXiv **2208** (2022)
11. Ji, Z., Lu, Z., Li, H.: An information retrieval approach to short text conversation. arXiv preprint arXiv:1408.6988 (2014)
12. Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W.t.: Dense passage retrieval for open-domain question answering. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 6769–6781 (2020). DOI 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550
13. Kim, D., Ahn, Y., Kim, W., Lee, C., Lee, K., Lee, K.H., Kim, J., Shin, D., Lee, Y.: Persona expansion with commonsense knowledge for diverse and consistent response generation. In: Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pp. 1139–1149 (2023). URL https://aclanthology.org/2023.eacl-main.81
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (Poster) (2015). URL http://arxiv.org/abs/1412.6980
15. Kudo, T., Yamamoto, K., Matsumoto, Y.: Applying conditional random fields to Japanese morphological analysis. In: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, pp. 230–237 (2004). URL https://aclanthology.org/W04-3230
16. Lee, G., Hartmann, V., Park, J., Papailiopoulos, D., Lee, K.: Prompted LLMs as chatbot modules for long open-domain conversation. In: Findings of the Association for Computational Linguistics: ACL 2023, pp. 4536–4554 (2023). DOI 10.18653/v1/2023.findings-acl.277. URL https://aclanthology.org/2023.findings-acl.277
17. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., Riedel, S., Kiela, D.: Retrieval-augmented generation for knowledge-intensive NLP tasks. In: NeurIPS, pp. 9459–9474 (2020). URL https://proceedings.neurips.cc/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf

18. Li, J., Galley, M., Brockett, C., Gao, J., Dolan, B.: A diversity-promoting objective function for neural conversation models. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 110–119 (2016). DOI 10.18653/v1/N16-1014. URL https://aclanthology.org/N16-1014
19. Li, J., Galley, M., Brockett, C., Spithourakis, G., Gao, J., Dolan, B.: A persona-based neural conversation model. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 994–1003 (2016). DOI 10.18653/v1/P16-1094. URL https://aclanthology.org/P16-1094
20. Li, J., Sun, X.: A syntactically constrained bidirectional-asynchronous approach for emotional conversation generation. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 678–683 (2018). DOI 10.18653/v1/D18-1071. URL https://aclanthology.org/D18-1071
21. Lin, C.Y.: ROUGE: A package for automatic evaluation of summaries. In: Text Summarization Branches Out@ACL, pp. 74–81 (2004). URL https://aclanthology.org/W04-1013
22. OpenAI: Introducing chatgpt. OpenAI blog (2022)
23. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: ACL, pp. 311–318 (2002)
24. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in PyTorch. In: Autodiff@NIPS (2017). URL https://openreview.net/forum?id=BJJsrmfCZ
25. Qian, Q., Huang, M., Zhao, H., Xu, J., Zhu, X.: Assigning personality/profile to a chatting machine for coherent conversation generation. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, pp. 4279–4285 (2018). DOI 10.24963/ijcai.2018/595. URL https://doi.org/10.24963/ijcai.2018/595
26. Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 3982–3992 (2019). DOI 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410
27. Ritter, A., Cherry, C., Dolan, W.B.: Data-driven response generation in social media. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pp. 583–593 (2011). URL https://aclanthology.org/D11-1054
28. Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Zettlemoyer, L., Cancedda, N., Scialom, T.: Toolformer: Language models can teach themselves to use tools. arXiv preprint arXiv:2302.04761 (2023)
29. Shang, L., Lu, Z., Li, H.: Neural responding machine for short-text conversation. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 1577–1586 (2015). DOI 10.3115/v1/P15-1152. URL https://aclanthology.org/P15-1152
30. Singh, D., Reddy, S., Hamilton, W., Dyer, C., Yogatama, D.: End-to-end training of multi-document reader and retriever for open-domain question answering. In: NeurIPS, pp. 25,968–25,981 (2021). URL https://proceedings.neurips.cc/paper/2021/file/da3fde159d754a2555eaa198d2d105b2-Paper.pdf
31. Song, Y., Yan, R., Li, X., Zhao, D., Zhang, M.: Two are better than one: An ensemble of retrieval-and generation-based dialog systems. arXiv preprint arXiv:1610.07149 (2016)
32. Takasaki, M., Yoshinaga, N., Toyoda, M.: Effective dialogue-context retriever for long-term open-domain conversation. In: The 13th International Workshop on Spoken Dialogue Systems Technology (2023). URL https://www.tkl.iis.u-tokyo.ac.jp/~takasa-m/iwsds2023.html
33. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NIPS, pp. 5998–6008 (2017)

34. Vinyals, O., Le, Q.: A neural conversational model. arXiv preprint arXiv:1506.05869 (2015)
35. Wang, H., Lu, Z., Li, H., Chen, E.: A dataset for research on short-text conversations. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 935–945 (2013). URL https://aclanthology.org/D13-1096
36. Wei, J., Wang, X., Schuurmans, D., Bosma, M., ichter, b., Xia, F., Chi, E., Le, Q.V., Zhou, D.: Chain-of-thought prompting elicits reasoning in large language models. In: Advances in Neural Information Processing Systems, vol. 35, pp. 24,824–24,837 (2022). URL https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf
37. Weston, J., Dinan, E., Miller, A.: Retrieve and refine: Improved sequence generation models for dialogue. In: Proceedings of the 2018 EMNLP Workshop SCAI: The 2nd International Workshop on Search-Oriented Conversational AI, pp. 87–92 (2018). DOI 10.18653/v1/W18-5713. URL https://aclanthology.org/W18-5713
38. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M.: Transformers: State-of-the-art natural language processing. In: ACL: System Demonstrations, pp. 38–45 (2020). URL https://www.aclweb.org/anthology/2020.emnlp-demos.6
39. Wu, W., Guo, Z., Zhou, X., Wu, H., Zhang, X., Lian, R., Wang, H.: Proactive human-machine conversation with explicit conversation goal. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 3794–3804 (2019). DOI 10.18653/v1/P19-1369. URL https://aclanthology.org/P19-1369
40. Xu, J., Szlam, A., Weston, J.: Beyond goldfish memory: Long-term open-domain conversation. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 5180–5197 (2022). DOI 10.18653/v1/2022.acl-long.356. URL https://aclanthology.org/2022.acl-long.356
41. Xu, X., Gou, Z., Wu, W., Niu, Z.Y., Wu, H., Wang, H., Wang, S.: Long time no see! open-domain conversation with long-term persona memory. In: Findings of the Association for Computational Linguistics: ACL 2022, pp. 2639–2650 (2022). DOI 10.18653/v1/2022.findings-acl.207. URL https://aclanthology.org/2022.findings-acl.207
42. Yang, L., Hu, J., Qiu, M., Qu, C., Gao, J., Croft, W.B., Liu, X., Shen, Y., Liu, J.: A hybrid retrieval-generation neural conversation model. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM '19, p. 1341–1350 (2019). DOI 10.1145/3357384.3357881. URL https://doi.org/10.1145/3357384.3357881
43. Zhang, S., Dinan, E., Urbanek, J., Szlam, A., Kiela, D., Weston, J.: Personalizing dialogue agents: I have a dog, do you have pets too? In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 2204–2213 (2018). DOI 10.18653/v1/P18-1205. URL https://aclanthology.org/P18-1205
44. Zhou, J., Pang, L., Shen, H., Cheng, X.: SimOAP: Improve coherence and consistency in persona-based dialogue generation via over-sampling and post-evaluation. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 9945–9959. Association for Computational Linguistics, Toronto, Canada (2023). DOI 10.18653/v1/2023.acl-long.553. URL https://aclanthology.org/2023.acl-long.553