

Keep Me Updated! Memory Management in Long-term Conversations

Sanghwan Bae^{1,2} Donghyun Kwak^{1,2} Soyoung Kang^{1,2} Min Young Lee^{1,2}
Sungdong Kim^{2,3} Yuin Jeong¹ Hyeri Kim¹ Sang-Woo Lee^{1,2,3}
Woomyoung Park^{1,2} Nako Sung¹

NAVER CLOVA¹ NAVER AI Lab² KAIST AI³

Abstract

Remembering important information from the past and continuing to talk about it in the present are crucial in long-term conversations. However, previous literature does not deal with cases where the memorized information is outdated, which may cause confusion in later conversations. To address this issue, we present a novel task and a corresponding dataset of memory management in long-term conversations, in which bots keep track of and bring up the latest information about users while conversing through multiple sessions. In order to support more precise and interpretable memory, we represent memory as unstructured text descriptions of key information and propose a new mechanism of memory management that selectively eliminates invalidated or redundant information. Experimental results show that our approach outperforms the baselines that leave the stored memory unchanged in terms of engagingness and humanness, with larger performance gap especially in the later sessions.

1 Introduction

In human interactions, memory is an important mechanism that helps us hold conversations, develop rapport, and maintain long-term relationships (Alea and Bluck, 2003; Nelson, 2003; Brewer et al., 2017). To this end, recent studies (Wu et al., 2020; Xu et al., 2022a,b) on open-domain dialogues have proposed methods to remember and utilize persona information (Zhang et al., 2018) of the interlocutors obtained from previous conversations. Specifically, they summarize the persona information in an extractive or abstractive way and give it as a condition for generating responses in subsequent conversations. They show that this feature leads to better consistency and engagingness of the chatbot systems.

Despite such progress, an aspect overlooked by previous studies is that memorized information can be invalidated by newly gathered information. They

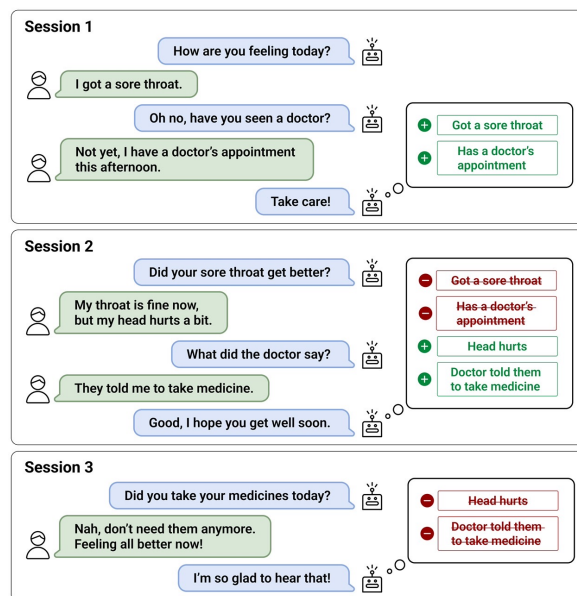


Figure 1: An example of a long-term dialogue. There is information obtained from an early session that is no longer true in a later session, e.g. “Got a sore throat”. This information should be removed from the memory of later sessions in order to correctly follow up with the interlocuter.

simply accumulate and maintain the stored information in memory; once stored, such information has no possibility of getting updated in the future. Memory in real-life conversations, however, can change over time, either in a short period of time (e.g. health status, plans for the weekend, or recently watched movie) or in relatively longer period of time (e.g. age, job, or hobby). Such memory needs to be kept track by asking its status again in subsequent conversations, as exemplified in Figure 1. Therefore, updating previous memory with new relevant information and maintaining it up-to-date are important features of human-like long-term conversations.

In this work, we study the methods of memorizing and updating dynamic information and utilizing them in successive dialogues. We formulate a new

task of memory management in long-term conversations and construct its corresponding dataset¹, by extending an existing Korean open-domain dialogue dataset (Bae et al., 2022) to multiple sessions with changing user information. In each session of our dataset, while the user and the bot have a conversation, information about the user is identified from the dialogue. Then, in successive sessions, the bot keeps in memory only the information valid at that point and utilizes the resulting memory in dialogue.

In addition, we propose a long-term dialogue system including a novel memory management mechanism. In this system, information about the interlocutors revealed in the previous conversation is abtractively summarized and stored in memory. Specifically, the memory management mechanism decides which information to keep in memory. For this purpose, we define four pairwise operations (PASS, REPLACE, APPEND, and DELETE) to find and eliminate the information that can cause confusion or redundancy in later conversations. For example, if the previous memory sentence is “Haven’t got COVID tested yet” and the new incoming summary is “Just got positive results from COVID test”, the two sentences are contradictory, in which the former needs to be replaced in memory by the latter. Through this process, only valid information remains in new memory. Then, in subsequent sessions, a relevant information from this memory is retrieved and given as additional condition for generating chatbot responses.

With extensive experiments and ablations, we show that the proposed memory management mechanism becomes more advantageous in terms of memorability as the sessions proceed, leading to better engagingness and humanness in multi-session dialogues.

Our contributions are as follows:

1. We make a step towards long-term conversations with dynamic memory that must be kept up-to-date.
2. We propose a novel memory management mechanism in the form of unstructured text that achieves better results in automatic and human evaluation over baselines.
3. We release the first Korean long-term dialogue

¹The dataset is available at <https://github.com/naver-ai/carecall-memory>

dataset for further research on memory management in dialogues.

2 Related Work

Personalized Dialogue System Building human-like open-domain chatbots is one of the seminal research topics in the field of natural language processing. Zhang et al. (2020) has provided a strong backbone generator model for dialogue systems, while Adiwardana et al. (2020), Roller et al. (2021) and Thoppilan et al. (2022) have paved the way for the development of more human-like, natural-sounding chatbots. The applications of open-domain chatbots have also widely expanded, including role-specified (Bae et al., 2022) and personalized (Zhang et al., 2018) dialogue systems. In particular, personalized dialogue system has typically been studied either via utilizing predefined, explicitly stated user profile (Zhang et al., 2018), or via directly extracting user profile from dialogue history (Xu et al., 2022a,b). While the latter approach is preferred in recent research works (Zhong et al., 2022), long-term management of the obtained information is yet to be studied.

Long-term Memory in Conversation Because it is inefficient to use the entire dialogue history as long-term memory, techniques for obtaining and managing information from dialogue history have been studied. Representing latent features as neural memory (Weston et al., 2015; Tran et al., 2016; Munkhdalai et al., 2019) used to be a traditional method. Slot-value format in dialogue state tracking (Heck et al., 2020; Hosseini-Asl et al., 2020; Kim et al., 2020), and graph format in Hsiao et al. (2020) have been the two major approaches in handling the memorized information in a structured way. Kim et al. (2020) suggested update operations on fixed-sized slot-value pairs for dialogue states. Wu et al. (2020) extracted user attributes from dialogues in triples. However, such approaches have not been demonstrated in a multi-session setting.

Leveraging the advancement of pre-trained language models (Devlin et al., 2019; Raffel et al., 2020; Brown et al., 2020; Kim et al., 2021), recent studies attempt to use the unstructured form of text as memory, which is expected to be advantageous in terms of generalizability and interpretability. Ma et al. (2021) and Xu et al. (2022b) selectively stored dialogue history with relevant information, while Zhong et al. (2022) employed refiners to extract fine-grained information from

dialogue history. Xu et al. (2022a) summarized the dialogue history to avoid overflow and redundancy. Nevertheless, these works rarely consider that the obtained information may change and become outdated. Specifically, MSC (Xu et al., 2022a) does not reflect the change of information. In other words, information in MSC remains fixed once it is stored. DuLeMon (Xu et al., 2022b) is not formatted in a multi-session manner, making it impossible to track memory changes across multiple sessions.

3 Task and Dataset

This section describes the task of long-term conversations with dynamic memory changes and the process of constructing a new dataset to conduct research on this task.

3.1 Task Definition

An episode consists of multiple consecutive dialogue sessions with a specific user. Dialogue context of the current session is $D_t = \{c_1, u_1, c_2, u_2, \dots, c_t, u_t\}$ at time step t , where c and u represent the chatbot’s and user’s utterance, respectively. Natural language memory sentences $M = \{m_1, m_2, \dots, m_n\}$ contain user information abstracted from the previous sessions of the same episode. Then, given the dialogue context D_t , and memory M , we are interested in predicting the chatbot’s response c_{t+1} . At the end of each session, the entire session D is summarized into several sentences of user information, denoted as $S = \{s_1, s_2, \dots, s_k\}$. Memory sentences M' for the next session are constructed by combining M and S .

3.2 Dataset Construction

To study this task, we build a new dataset based on CareCall dataset² (Bae et al., 2022), which consists of single sessions of open-domain dialogues between bots and users. We choose this dataset because the sessions contain various topics that are likely to change in a short period of time, such as user’s health, sleep, and diet, as well as those in a relatively longer period of time, such as family, pets, and frequently visited places. We extend this single-session dataset to a multi-session setting, which is a similar procedure presented in MSC (Xu et al., 2022a). Our resulting dataset contains more persona updates than other datasets (Xu et al.,

²<https://github.com/naver-ai/carecall-corpus>

Statistics	
Sessions	7,665
Session 1	2,812
Session 2	2,798
Session 3	743
Session 4	674
Session 5	638
Turns	160,191
Avg. turns per session	20.90
Avg. words per turn	4.93
Unique words for all turns	59,434
Distinct-1/2 for all turns	0.0753/0.2891
Avg. memory sentences per session $ M $	3.41
Avg. summary sentences per session $ S $	2.88
Avg. words per summary sentence	4.70
Distinct-1/2 for all summary sentences	0.1425/0.3926

Table 1: Statistics of our CareCall_{mem} dataset. Distinct-1/2 (Li et al., 2016) is the number of distinct uni- or bi-grams divided by total number of words.

2022a,b) (see Section C.1 in Appendix for more details).

3.2.1 Preliminary Step: Dialogue and Summary

To efficiently collect the dataset, we train preliminary models for dialogue summaries and memory grounded dialogues to first automatically generate the dataset, and then a group of annotators revise them. This procedure has shown to be more effective in recent studies (Sun et al., 2021; Bae et al., 2022; Liu et al., 2022; Zheng et al., 2022). In the entire process, we leverage the large-scale language models (LMs) for each step; HyperCLOVA 6.9B as backbone LM.

Dialogue Summary We randomly sample 600 dialogue sessions with more than 15 turns from the CareCall dataset. We ask annotators to summarize each session into several sentences to build S that may be useful to continue the next conversation. Using these summaries, we fine-tune LMs to generate summaries given dialogues $P(S|D)$. The models then generate summaries of unseen dialogues randomly sampled from the CareCall dataset. Finally, annotators edit the generated summaries by filling in missing information or correcting erroneous sentences. Since there is no memory sentence for the first session, i.e. $M = \emptyset$, memory for the second session M' is equal to S .

Memory Grounded Dialogue To build a second session of each episode, annotators write dia-

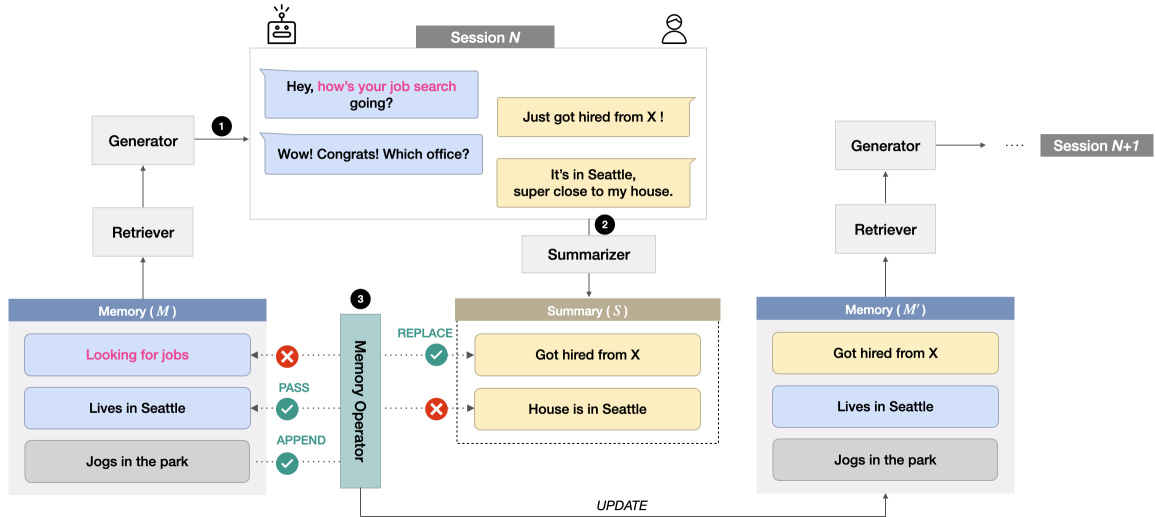


Figure 2: The overview of the proposed system. (1) Memory grounded response generation model (Section 4.1) conditioned on memory sentences M converses with human user. (2) At the end of the session, the dialogue summarizer (Section 4.2) summarizes user information into several sentences S from the session history. (3) Memory operator (Section 4.3) predicts the operations for every (m_i, s_j) pair to select information to leave, which consists the next memory M' .

logue sessions grounded on the 600 human-written summaries from the previous step. Likewise, we fine-tune LMs to generate the entire dialogue sessions given previous memory $P(D|M)$. Then, the fine-tuned models generate memory grounded dialogues from the unseen dialogue summaries in the previous paragraph. Lastly, human annotators revise the generated dialogues, i.e. correcting wrong responses (misuse of memory, not sensible, or out-of-bounds from CareCall’s role described in Bae et al. (2022)).

3.2.2 Interactive Step: Multi-Session Dialogue

From the preliminary step, we obtain the data to build a chatbot that can conduct interactive conversation utilizing the memorized information. To construct a multi-session dialogue system, we train the dialogue summarizer and memory grounded response generator described in Section 4 on previously collected (D, S) pairs with (M, D) pairs respectively.

Then, crowdworkers converse with the resulting system for 5 sessions per episode, starting from the first session. The interval between sessions is assumed to be from 1 to 2 weeks. At the end of each session, the summarizer generates S from the current session. Both generated responses and summaries are edited by annotators to correct errors. Lastly, we ask annotators to select which sentences in M and S should remain in new memory M' for the next session. We provide details of quality

control in Appendix A and an example episode in Figure 4 in Appendix. We name this dataset as CareCall_{mem} and the statistics of the dataset are given in Table 1, which includes all the collected data described in Section 3.2.1-3.2.2.

4 Models

We propose a long-term dialogue system with memory management mechanism. The system consists of three parts: memory grounded response generation, dialogue summarization, and memory update. The overall architecture is shown in Figure 2.

4.1 Memory Grounded Response Generation

Response Generation We consider the response generation model conditioned on memory sentences. Given the memory M and the dialogue history $D_t = \{c_1, u_1, c_2, u_2, \dots, c_t, u_t\}$ at time step t , the conditional probability of the next target response $c_{t+1} = \{w_1, w_2, \dots, w_{|c_{t+1}|}\}$ can be written as the product of a sequence of conditional probabilities:

$$p(c_{t+1}|D_t, M) = \prod_i p_\theta(w_i|D_t, M, w_{<i}), \quad (1)$$

where w_i is i -th token of the sequence and θ is trainable parameters of the model. We use HyperCLOVA 6.9B as the response generation model. The model is fine-tuned using the maximum likeli-

hood estimation (MLE), which minimizes:

$$\mathcal{L}_\theta(c_{t+1}, D_t, M) = - \sum_i \log p_\theta(w_i | D_t, M, w_{<i}). \quad (2)$$

Memory Retrieval In addition, following the previous studies (Xu et al., 2022a,b), we consider that retrieving information relevant to the current dialogue context is effective when dealing with a large collection of sentences in memory. We use an approach almost identical to context persona matching (CPM) method proposed in Xu et al. (2022b), replacing persona sentences to memory sentences in our task (See Appendix B.3 for more details). At the inference time, the retrieved top k sentences constitute M_{ret} , which is the actual input condition of the response generator described in the preceding paragraph.

4.2 Dialogue Summarization

Given the dialogue history of the entire session D , our abstractive summarization model summarizes important user information in the form of several natural language sentences $S = \{s_1, s_2, \dots, s_k\}$. What information to be summarized can be learned based on the human annotation from our newly collected CareCall_{mem} dataset. Here, we train HyperCLOVA 6.9B as the summarizer to generate summary sentences given the dialogue history as an input. Formally, this is done by minimizing loss for each gold summary sentence $s_t = \{w_1, w_2, \dots, w_{|s_t|}\}$:

$$\mathcal{L}_\phi(s_t, D, s_{<t}) = - \sum_i \log p_\phi(w_i | D, s_{<t}, w_{<i}), \quad (3)$$

where ϕ is trainable parameters of the summarizer.

4.3 Memory Update

The memory update process stores the latest user information in memory by combining the old and the new information sentences. At the end of each session, n existing memory sentences $M = \{m_1, m_2, \dots, m_n\}$ and k new summary sentences $S = \{s_1, s_2, \dots, s_k\}$ are given. The memory writer combines them to find $M' = \{m'_1, m'_2, \dots, m'_{|M'|}\}$ that are lossless, consistent, and not redundant in terms of information. Here, we assume that M and S are internally consistent and not redundant.

Our approach finds the sentence set M' by classifying the relationship of the sentence pair (m_i, s_j) , where $m_i \in M$ and $s_j \in S$. We

```

Input  $M = \{m_1, m_2, \dots, m_n\}$ 
Input  $S = \{s_1, s_2, \dots, s_k\}$ 
Input  $O(m, s) \rightarrow \{\text{"P"}, \text{"R"}, \text{"A"}, \text{"D"}\}$ 
Output  $M' = \{m'_1, m'_2, \dots, m'_{|M'|}\}$ 
1:  $M_{del} \leftarrow \emptyset$ 
2:  $S_{del} \leftarrow \emptyset$ 
3: for  $\forall m_i \in M$ :
4:   for  $\forall s_j \in S$ :
5:     if  $O(m_i, s_j) \in \{\text{"R"}, \text{"D"}\}$ :
6:        $M_{del} \leftarrow M_{del} \cup \{m_i\}$ 
7:       if  $O(m_i, s_j) = \text{"D"}$ :
8:          $S_{del} \leftarrow S_{del} \cup \{s_j\}$ 
9:  $M \leftarrow M - M_{del}$ 
10: for  $\forall s_j \in S$ :
11:   for  $\forall m_i \in M$ :
12:     if  $O(m_i, s_j) = \text{"P"}$ :
13:        $S_{del} \leftarrow S_{del} \cup \{s_j\}$ 
14:  $S \leftarrow S - S_{del}$ 
15:  $M' \leftarrow M \cup S$ 
16: return  $M'$ 

```

Algorithm 1: Our memory update algorithm. P, R, A, and D are abbreviations for PASS, REPLACE, APPEND, and DELETE, respectively.

define operations for (m_i, s_j) as $O(m_i, s_j) \rightarrow \{\text{"PASS"}, \text{"REPLACE"}, \text{"APPEND"}, \text{"DELETE"}\}$.

- **PASS** means storing only m_i . It reflects the case in which the information of m_i already contains that of s_j , i.e. $m_i \supseteq s_j$ in terms of information. Only m_i is stored in order to avoid redundancy.
- **REPLACE** means storing only s_j . When s_j and m_i are inconsistent, the more recent information s_j remains. This operation is also useful when s_j has more information than m_i , i.e. $m_i \subset s_j$ in terms of information.
- **APPEND** means storing both m_i and s_j . If m_i and s_j are irrelevant, both are stored in order to avoid loss of information.
- **DELETE** means removing both m_i and s_j . We found that there are cases where this operation is useful. It can reduce the memory confusion by “forgetting” a completed state that no longer needs to be remembered. For example, if $m_i = \text{“having a cold and taking medicine”}$ and $s_j = \text{“cold is all better now”}$, not only m_i should be removed because it is no longer true, but s_j should also be removed because the chatbot doesn’t have to

Statistics	
Training pairs	2,149
Validation pairs	300
Test pairs	300
individual label = gold label (estimated human performance)	87.52%
no gold label	1.96%
PASS	13.3%
REPLACE	37.0%
APPEND	44.6%
DELETE	5.1%

Table 2: Statistics of the collected sentence pairs to train and evaluate memory update methods.

remember the user’s cold anymore. If such information is not forgotten and remained in memory, something like Pink Elephant Paradox (Wegner et al., 1987) can occur, causing hallucination of the dialogue model. Therefore, we decided to delete the information that no longer needs to be remembered.

A formal description of the entire algorithm for memory update is given in Algorithm 1, which use the proposed pairwise operations. We fine-tune a classification model of T5 (Raffel et al., 2020) architecture to perform $O(m, s)$.

5 Experiments

In this section we describe experimental settings and results including the evaluations of the proposed memory update methods of memory management (Section 5.1) and the evaluations of the entire system on multi-session dialogues (Section 5.2).

5.1 Memory Update

In this section, we compare several types of models to perform $O(m_i, s_j)$ described in Section 4.3.

5.1.1 Pairwise Evaluation

To build a dataset to train and evaluate, we ask the annotators to annotate each pair of (m_i, s_j) in CareCall_{mem} dataset into 4 classes {"PASS", "REPLACE", "APPEND", "DELETE"}. If any one of the four labels is chosen by at least two of the three annotators, it is regarded as the gold label. If there is no such consensus, which occur in about 2% of the cases, we discard the example. Table

Model	Pairwise Acc.		Set F1
	Validation	Test	Test
From scratch	84.65 (0.99)	83.65 (2.01)	87.98 (2.11)
NLI zero-shot	72.23 (1.61)	71.50 (1.24)	84.62 (2.20)
NLI transfer (fine-tune)	85.34 (0.81)	84.10 (1.01)	88.69 (1.65)

Table 3: 4-class accuracy for pairwise operations and sentence-level F1 scores for set-level evaluation (standard deviation in brackets).

2 reports some key statistics about the collected dataset. Examples of the annotated pairs are in Table 7 of the Appendix and see Appendix C.2 for discussions of the cases we found that need further research on this direction.

We consider three types of model for this task. A pre-trained T5 architecture is used for all three models (See Appendix B.4 for more details).

- *From scratch*: A model fine-tuned on the collected dataset.
- *NLI zero-shot*: A model fine-tuned on KLUE-NLI (Park et al., 2021), a Korean natural language inference (NLI) dataset. Assuming the old memory sentence and the new memory sentence as a premise and a hypothesis, respectively, we can map the memory update operations as follows. “PASS”: entailment, “REPLACE”: contradiction or reversely entailment, “APPEND”: neutral. “DELETE” is not mapped, so this model has minimum 5.1% of error. The *NLI zero-shot* is used to see if the knowledge from NLI can be transferred to the proposed memory operations.
- *NLI transfer (fine-tune)*: The *NLI zero-shot* model further fine-tuned on the collected dataset.

The results are shown in Table 3. Although *NLI zero-shot* reaches certain level of performance, there is a significant performance drop compared to *From scratch* model (−12.15% on test set). We hypothesize that this is because the memory update requires common sense beyond pure logical reasoning. For example, if $m_i =$ “planning to see a doctor” and $s_j =$ “went to the hospital”, there is no logical inconsistency, but m_i can generally be expected to be replaced by s_j . Eventually, we found that further training the NLI model on the collected dataset is the best (*NLI transfer*).

<i>All turns</i>	Session 2			Session 3			Session 4			Session 5		
	PPL↓	BLEU-1/2↑	F1↑	PPL↓	BLEU-1/2↑	F1↑	PPL↓	BLEU-1/2↑	F1↑	PPL↓	BLEU-1/2↑	F1↑
Without memory	4.023	0.293/0.169	0.334	4.789	0.298/0.174	0.332	4.073	0.289/0.167	0.320	4.221	0.289/0.159	0.334
History accumulate	4.057	0.267/0.161	0.320	4.491	0.263/0.153	0.322	4.652	0.261/0.151	0.321	4.673	0.261/0.154	0.329
Memory accumulate	3.735	0.313/0.189	0.365	3.782	0.314/0.189	0.368	3.875	0.307/0.186	0.358	4.052	0.311/0.193	0.364
Memory update	3.743	0.312/0.187	0.363	3.773	0.316/0.192	0.369	3.794	0.309/0.188	0.360	3.937	0.316/0.198	0.369
Memory gold*	3.680	0.317/0.201	0.375	3.736	0.325/0.206	0.383	3.746	0.318/0.201	0.377	3.878	0.320/0.202	0.375

<i>Memory turns</i>	Session 2			Session 3			Session 4			Session 5		
	PPL↓	BLEU-1/2↑	F1↑	PPL↓	BLEU-1/2↑	F1↑	PPL↓	BLEU-1/2↑	F1↑	PPL↓	BLEU-1/2↑	F1↑
Without memory	6.655	0.285/0.128	0.361	6.779	0.275/0.127	0.358	6.577	0.279/0.120	0.346	7.106	0.269/0.117	0.330
History accumulate	4.246	0.242/0.131	0.339	4.548	0.245/0.135	0.342	5.008	0.228/0.126	0.323	6.617	0.219/0.102	0.275
Memory accumulate	4.439	0.324/0.160	0.381	4.620	0.304/0.136	0.362	5.117	0.293/0.125	0.354	5.725	0.284/0.117	0.333
Memory update	4.487	0.329/0.163	0.382	4.627	0.306/0.145	0.360	4.872	0.297/0.130	0.360	5.308	0.284/0.120	0.339
Memory gold*	4.419	0.352/0.188	0.395	4.421	0.338/0.182	0.393	4.518	0.335/0.178	0.386	4.855	0.322/0.177	0.379

Table 4: Comparison of automatic evaluation metric results among different systems on test set in multi-session dialogues. *Memory turns* refer to the turns annotated to use memory explicitly, which accounts for 22.4% of all turns in the test set.

5.1.2 Set Evaluation

We manually annotate the test set in Table 2 to measure the performance of set level algorithm $f(M, S) \rightarrow M'$ (Alg. 1). The annotation process is the same as the one described in Section 3.2.2, and the compared models are the same as the pairwise evaluation. We measure the sentence-level F1 scores between gold M' and the predicted one. The evaluation results in Table 3 show a trend similar to the pairwise evaluation; *NLI transfer* achieves the best performance.

5.2 Multi-session Dialogues

We evaluate our entire system described in Section 4 in a multi-session dialogue setting. The response generation model (4.1) is trained on both CareCall_{mem} and original CareCall datasets (details are in Appendix D). The memory retrieval (4.1) and dialogue summarization (4.2) models are trained on CareCall_{mem} only. The memory update model (4.3) is *NLI transfer (fine-tune)* from 5.1.

5.2.1 Automatic Evaluation

We randomly sample 60 episodes (300 sessions) from CareCall_{mem} dataset to build a test set. As evaluation metrics, PPL, BLEU-1/2 (Papineni et al., 2002), F1, and Distinct-1/2 (Li et al., 2016) are used. We compare four models in this section.

- *Without memory*: This model consists of only the response generation model in Section 4. The input at inference is current dialogue history D_t only.
- *History accumulate*: This model consists of only a response generation model, with input

of all previous sessions concatenated before D_t for both training and inference.

- *Memory accumulate*: This model consists of all components in Section 4, but it use “APPEND” as the only pairwise operation.
- *Memory update*: This model consists of all components in Section 4. The entire memory update method in Algorithm 1 is used.
- *Memory gold**: This model consists of all components in Section 4, but M for each session is gold memory in the dataset. This serves as the upper limit for memory management mechanism.

Table 4 shows the results. First, *Without memory* shows a relatively high PPL in memory turns of all sessions. *History accumulate* is competitive in all metrics in early sessions, but its performance drops significantly as the session progresses. This is conjectured that there are many distractions in accumulated dialogue history which make it difficult to track changing information. Furthermore, the performance gain of *Memory update* over *Memory accumulate* becomes larger as the sessions progress. This is because *Memory accumulate* has a relatively high possibility of utilizing outdated information, which can be a noise to the model. *Memory update* achieves the best performance in most metrics especially in later sessions (Session 4-5), showing advantage of up-to-date memory.

5.2.2 Human Evaluation

We also perform human evaluation in multi-session dialogues. For a reliable evaluation, we use con-

Model	Coherence	Consistency	Engagingness	Humanness	Memorability
Without memory	-0.0450	0.5907	-0.4625	-0.2445	-1.3057
Memory accumulate	0.1892	0.6301	-0.2871	0.0831	0.0030
Memory update	0.2248	0.6272	-0.1770	0.1917	0.4351

Table 5: Average standardized scores of human evaluation metrics on live conversation for 5 sessions in each episode. Absolute scores are in Table 9 in Appendix.

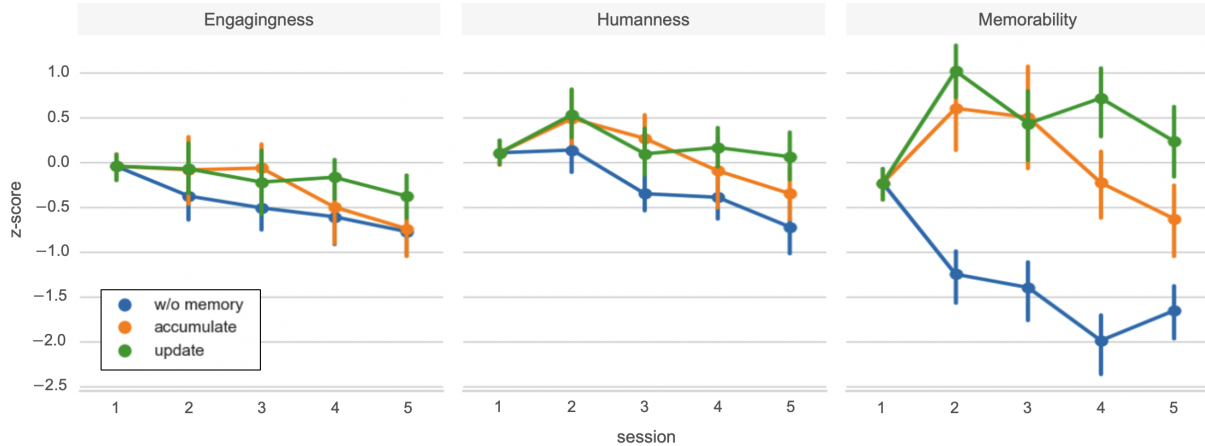


Figure 3: Average standardized scores per session. The dots indicate the average scores and the vertical lines indicate the 95% confidence interval. The averaged scores outside the vertical line of other models mean significant differences at the 95% confidence level. The models in the first session are all equivalent.

tinuous rating for the live conversations proposed in Ji et al. (2022), which is shown to be easily reproducible with high correlation among repeated experiments. We extend this evaluation process to multi-session dialogues. A crowdworker conducts live conversations for five sessions per episode (assuming 1-2 weeks are elapsed between sessions) with a randomly selected model among the compared models. At the end of each session, we ask the crowdworkers to rate the degree to which they agree with the statements on each evaluation metric on a scale of 0-100. After an episode is over, they repeat another episode with another randomly selected model. The score distribution of each crowdworker is standardized, removing the potential bias of each worker. The evaluation metrics are summarized in Appendix E and the interface used for evaluation is shown in Figure 5 in Appendix. There are three models compared in this evaluation: *Without memory*, *Memory accumulate*, and *Memory update* described in Section 5.2.1. A total of 155 episodes and 775 sessions are evaluated through this process.

Overall results are shown in Table 5. *Memory update* shows a clear advantage over *Memory accumulate* in memorability (p-value < 0.05 for pairwise significance test). We also discover that memorabil-

ity positively correlates with engagingness (Pearson correlation 0.68 at p-value < 0.01). Furthermore, there is some positive correlation between memorability and humanness (Pearson correlation 0.47 at p-value < 0.01). Accordingly, *Memory update* shows the highest engagingness and humanness compared to the other two models. Additionally, coherence and consistency have no statistically meaningful difference among the models (p-value > 0.1 for pairwise significance test). This allows us to conclude that *Memory update* has better ability to remember while still preserving general conversational abilities like coherence and consistency.

Figure 3 shows scores of each session for the three metrics (results for all five metrics are in Table 8 in Appendix). In the case of *Without memory*, it is observed that engagingness and humanness consistently drop as the sessions progress. In contrast, *Memory accumulate* and *Memory update* maintain engagingness and humanness to some level in the subsequent sessions. However, in the later sessions (Session 4-5), the difference between *Memory accumulate* and *Memory update* grows with regards to all three metrics. It seems that the crowdworkers feel as if the chatbot doesn't remember well when it brings up an information that has become no longer true in previous sessions, resulting in a

lower engagingness and humanness. This is likely the reason why crowdworkers rate *Memory update* as the most engaging and human-like.

6 Conclusion

We present a novel task of long-term conversation with dynamic memory changes and build the corresponding dataset. We propose a memory management method that performs operations between old and new memory information in the form of unstructured text. Through an extensive series of experiments, we demonstrate the effectiveness of the proposed method in terms of improving memorability of a chatbot system. We also show that keeping memory up-to-date in long-term conversations is important for engaging and human-like dialogues. We release the newly collected dataset, looking forward to further research on this promising direction.

Limitations

For the sake of simplicity and clarity in our current research study, we only considered remembering and updating information of a single interlocutor. However, our future studies should aim to include memorized information from both sides and bringing it up in conversations, just as Xu et al. (2022a,b) did by duplicating the proposed memory management for both sides.

Regarding the generalizability of our results, it should be noted that the experiment was performed on data collected in Korean language. Although we do not use a Korean-language-specific approach in our experimental settings, whether or not our results would extend across different languages is yet to be determined.

Our experiments do not cover extremely long conversations where memory reaches its maximum capacity. In this case, removing the oldest memories (i.e. first in, first out) could be a plausible approach, just as human memory fades over time. Still, the amount of computation can grow large. Since pairwise operation occurs $O(|M||S|)$ times and each operation is predicted by T5 in our experiments (it takes about 80ms on 1 NVIDIA V100 for a single inference), the memory update can be costly when $|M|$ gets large.

Finally, our experiments require large GPU resources, at least 1 NVIDIA A100 or multiple GPUs equivalent to it. The specifications of GPUs used for training the models are provided in Appendix

B.

Ethical Considerations

Our dataset is created by authors, crowdworkers, and large-scale language models. Throughout the interactive data collection process, we instructed crowdworkers to play the role of potential users only, without disclosing any personally identifiable information about workers themselves. Meanwhile, it is known that the generated output from pre-trained language models may contain toxicity (Gehman et al., 2020; Liu et al., 2021; Xu et al., 2021), private information (Carlini et al., 2021), or social biases (Bordia and Bowman, 2019; Shwartz and Choi, 2020; Bender et al., 2021; Garrido-Muñoz et al., 2021). To address these issues, we carefully constructed criteria for harmful texts based on legal and ethical considerations offered by our group’s specialists. We guided all annotators to filter and edit the dataset based on such criteria. In addition, since the users in our dataset might be deemed as a vulnerable social group, our group’s ethical consultation included a review of sensitive subjects and the elimination of sessions involving any mention of such topics. We also had multiple filtering processes by multiple workers for every example to ensure that the final dataset does not contain any potentially malicious or unintended harmful effects.

Furthermore, since the proposed system has the capability of storing the information they learned from the interactions with users, we emphasize that the information in long-term memory remains absolutely private to the individual’s conversation and is not shared with anyone else. Also, if it comes into the wrong hands, chatbots are exposed to the possibility of getting programmed to imitate humans and be used for phishing and fraud. In such conversations, we can expect abusive cases where individuals accidentally disclose important and confidential information. Thus, incorporating the proposed system into real-world applications requires developers to ensure that it is used only in a safe and ethical manner.

Acknowledgements

The authors thank all the members of CLOVA and AI Lab of NAVER for devoted support and discussion. In particular, they would like to thank the members of CLOVA Conversation for their technical support and active discussion. In addition, the

authors thank the members of CLOVA Conversation Planning for guiding and monitoring the data collection process. Finally, they would like to show appreciation for the valuable feedback given by Professor Yejin Choi.

References

- Daniel Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, et al. 2020. Towards a human-like open-domain chatbot. *arXiv preprint arXiv:2001.09977*.
- Nicole Alea and Susan Bluck. 2003. Why are you telling me that? a conceptual model of the social function of autobiographical memory. *Memory*, 11(2):165–178.
- Irwin Altman and Dalmas A Taylor. 1973. *Social penetration: The development of interpersonal relationships*. Holt, Rinehart & Winston.
- Sanghwan Bae, Donghyun Kwak, Sungdong Kim, Donghoon Ham, Soyoung Kang, Sang-Woo Lee, and Woomyoung Park. 2022. **Building a role specified open-domain dialogue system leveraging large-scale language models**. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2128–2150, Seattle, United States. Association for Computational Linguistics.
- Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623.
- Shikha Bordia and Samuel R. Bowman. 2019. **Identifying and reducing gender bias in word-level language models**. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 7–15, Minneapolis, Minnesota. Association for Computational Linguistics.
- Robin N Brewer, Meredith Ringel Morris, and Siân E Lindley. 2017. How to remember what to remember: exploring possibilities for digital reminder systems. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, 1(3):1–20.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Matiusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. **Language models are few-shot learners**. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B Brown, Dawn Song, Úlfar Erlingsson, et al. 2021. Extracting training data from large language models. In *USENIX Security Symposium*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. **BERT: Pre-training of deep bidirectional transformers for language understanding**. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Sarah E. Finch and Jinho D. Choi. 2020. **Towards unified dialogue system evaluation: A comprehensive analysis of current evaluation protocols**. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 236–245, 1st virtual meeting. Association for Computational Linguistics.
- Ismael Garrido-Muñoz, Arturo Montejó-Ráez, Fernando Martínez-Santiago, and L Alfonso Ureña-López. 2021. A survey on bias in deep nlp. *Applied Sciences*, 11(7):3184.
- Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. **RealToxicityPrompts: Evaluating neural toxic degeneration in language models**. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3356–3369, Online. Association for Computational Linguistics.
- Michael Heck, Carel van Niekerk, Nurul Lubis, Christian Geishauser, Hsien-Chin Lin, Marco Moresi, and Milica Gasic. 2020. **TripPy: A triple copy strategy for value independent neural dialog state tracking**. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 35–44, 1st virtual meeting. Association for Computational Linguistics.
- Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. *Advances in Neural Information Processing Systems*, 33:20179–20191.
- Yu-Ting Hsiao, Edwinn Gamborino, and Li-Chen Fu. 2020. A hybrid conversational agent with semantic association of autobiographic memories for the elderly. In *Cross-Cultural Design. Applications in Health, Learning, Communication, and Creativity: 12th International Conference, CCD 2020, Held as*

- Part of the 22nd HCI International Conference, HCI 2020, Copenhagen, Denmark, July 19–24, 2020, Proceedings, Part II*, pages 53–66.
- Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.
- Tianbo Ji, Yvette Graham, Gareth Jones, Chenyang Lyu, and Qun Liu. 2022. [Achieving reliable human assessment of open-domain dialogue systems](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6416–6437, Dublin, Ireland. Association for Computational Linguistics.
- Boseop Kim, HyoungSeok Kim, Sang-Woo Lee, Gichang Lee, Donghyun Kwak, Jeon Dong Hyeon, Sunghyun Park, Sungju Kim, Seonhoon Kim, Dongpil Seo, Heungsub Lee, Minyoung Jeong, Sungjae Lee, Minsub Kim, Suk Hyun Ko, Seokhun Kim, Taeyong Park, Jinuk Kim, Soyoun Kang, Na-Hyeon Ryu, Kang Min Yoo, Minsuk Chang, Soobin Suh, Sookyo In, Jinseong Park, Kyungduk Kim, Hiun Kim, Jisu Jeong, Yong Goo Yeo, Donghoon Ham, Dongju Park, Min Young Lee, Jaewook Kang, Inho Kang, Jung-Woo Ha, Woomyoung Park, and Nako Sung. 2021. [What changes can large-scale language models bring? intensive study on HyperCLOVA: Billions-scale Korean generative pretrained transformers](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3405–3424, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Hanjoo Kim, Minkyu Kim, Dongjoo Seo, Jinwoong Kim, Heungseok Park, Soeun Park, Hyunwoo Jo, KyungHyun Kim, Youngil Yang, Youngkwan Kim, et al. 2018. [Nsml: Meet the mlaas platform with a real-world case study](#). *ArXiv preprint*, abs/1810.09957.
- Sungdong Kim, Sohee Yang, Gyuwan Kim, and Sang-Woo Lee. 2020. [Efficient dialogue state tracking by selectively overwriting memory](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 567–582, Online. Association for Computational Linguistics.
- Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. [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*, pages 110–119, San Diego, California. Association for Computational Linguistics.
- Margaret Li, Jason Weston, and Stephen Roller. 2019. Acute-eval: Improved dialogue evaluation with optimized questions and multi-turn comparisons. *arXiv preprint arXiv:1909.03087*.
- Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021. [DExperts: Decoding-time controlled text generation with experts and anti-experts](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6691–6706, Online. Association for Computational Linguistics.
- Alisa Liu, Swabha Swayamdipta, Noah A Smith, and Yejin Choi. 2022. Wanli: Worker and ai collaboration for natural language inference dataset creation. *arXiv preprint arXiv:2201.05955*.
- Zhengyi Ma, Zhicheng Dou, Yutao Zhu, Hanxun Zhong, and Ji-Rong Wen. 2021. One chatbot per person: Creating personalized chatbots based on implicit user profiles. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 555–564.
- Tsendsuren Munkhdalai, Alessandro Sordani, Tong Wang, and Adam Trischler. 2019. Metalearned neural memory. *Advances in Neural Information Processing Systems*, 32.
- Katherine Nelson. 2003. Self and social functions: Individual autobiographical memory and collective narrative. *Memory*, 11(2):125–136.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [Bleu: a method for automatic evaluation of machine translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Sungjoon Park, Jihyung Moon, Sungdong Kim, Won Ik Cho, Ji Yoon Han, Jangwon Park, Chisung Song, Junseong Kim, Youngsook Song, Taehwan Oh, et al. 2021. Klue: Korean language understanding evaluation. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.
- 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. *Journal of Machine Learning Research*, 21:1–67.
- Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Eric Michael Smith, Y-Lan Boureau, and Jason Weston. 2021. [Recipes for building an open-domain chatbot](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 300–325, Online. Association for Computational Linguistics.

- Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. [Megatron-lm: Training multi-billion parameter language models using model parallelism](#). *ArXiv preprint*, abs/1909.08053.
- Vered Shwartz and Yejin Choi. 2020. [Do neural language models overcome reporting bias?](#) In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6863–6870, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Eric Smith, Orion Hsu, Rebecca Qian, Stephen Roller, Y-Lan Boureau, and Jason Weston. 2022. [Human evaluation of conversations is an open problem: comparing the sensitivity of various methods for evaluating dialogue agents](#). In *Proceedings of the 4th Workshop on NLP for Conversational AI*, pages 77–97, Dublin, Ireland. Association for Computational Linguistics.
- Kai Sun, Seungwhan Moon, Paul Crook, Stephen Roller, Becka Silvert, Bing Liu, Zhiguang Wang, Honglei Liu, Eunjoon Cho, and Claire Cardie. 2021. [Adding chit-chat to enhance task-oriented dialogues](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1570–1583, Online. Association for Computational Linguistics.
- Nako Sung, Minkyu Kim, Hyunwoo Jo, Youngil Yang, Jingwoong Kim, Leonard Lausen, Youngkwan Kim, Gayoung Lee, Donghyun Kwak, Jung-Woo Ha, et al. 2017. [Nsm1: A machine learning platform that enables you to focus on your models](#). *ArXiv preprint*, abs/1712.05902.
- Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. 2022. [Lamda: Language models for dialog applications](#). *arXiv preprint arXiv:2201.08239*.
- Ke Tran, Arianna Bisazza, and Christof Monz. 2016. [Recurrent memory networks for language modeling](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 321–331, San Diego, California. Association for Computational Linguistics.
- Daniel M Wegner, David J Schneider, Samuel R Carter, and Teri L White. 1987. Paradoxical effects of thought suppression. *Journal of personality and social psychology*, 53(1):5.
- Jason Weston, Sumit Chopra, and Antoine Bordes. 2015. [Memory networks](#). In *3rd International Conference on Learning Representations, ICLR 2015*.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Chien-Sheng Wu, Andrea Madotto, Zhaojiang Lin, Peng Xu, and Pascale Fung. 2020. [Getting to know you: User attribute extraction from dialogues](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 581–589, Marseille, France. European Language Resources Association.
- Jing Xu, Da Ju, Margaret Li, Y-Lan Boureau, Jason Weston, and Emily Dinan. 2021. [Bot-adversarial dialogue for safe conversational agents](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2950–2968, Online. Association for Computational Linguistics.
- Jing Xu, Arthur Szlam, and Jason Weston. 2022a. [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)*, pages 5180–5197, Dublin, Ireland. Association for Computational Linguistics.
- Xinchao Xu, Zhibin Gou, Wenquan Wu, Zheng-Yu Niu, Hua Wu, Haifeng Wang, and Shihang Wang. 2022b. [Long time no see! open-domain conversation with long-term persona memory](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2639–2650, Dublin, Ireland. Association for Computational Linguistics.
- Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. [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)*, pages 2204–2213, Melbourne, Australia. Association for Computational Linguistics.
- Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. [DIALOGPT : Large-scale generative pre-training for conversational response generation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, Online. Association for Computational Linguistics.
- Chujie Zheng, Sahand Sabour, Jiaxin Wen, and Minlie Huang. 2022. [Augesc: Large-scale data augmentation for emotional support conversation with pre-trained language models](#). *arXiv preprint arXiv:2202.13047*.
- Hanxun Zhong, Zhicheng Dou, Yutao Zhu, Hongjin Qian, and Ji-Rong Wen. 2022. [Less is more: Learning to refine dialogue history for personalized dialogue generation](#). In *Proceedings of the 2022*

Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 5808–5820, Seattle, United States. Association for Computational Linguistics.

A Data Quality Control

In each annotation process, we provide detailed guidance and training for all annotators in order to optimize our dataset quality. Specifically, we instruct them with a comprehensive manual for each job and clarify all questions in a group chat to eliminate potential misunderstandings. We also give personalized feedback to each annotator every week. We recruited annotators from a freelancing platform and in-house labeling services. The major instructions for the data collection process are summarized as follows.

Dialogue Summary We ask annotators to summarize the core information of the user in a given dialogue session, particularly the information that is worth continuing with the subsequent sessions. The resulting summaries are abstractive summaries, as we instruct them not to copy the utterance itself from the dialogues, but write it into a new abstractive sentence. For the summarized information, we guide the annotators to exclude information about one-off events or overly-detailed information about the user such as “had three eggs for breakfast” or “went to the park at 9:10AM”, given that they are difficult or irrelevant to use in subsequent dialogues. However, if the user’s information has been changed, such information would be included in the summary (e.g. “just got married” or “recovered from flu”) to keep track of. Three different groups of annotators consecutively edited the summaries, only 1.4% of which were edited in the last iteration.

Memory Grounded Dialogue We ask annotators to make sure that the dialogues contain various daily topics and that the information in memory is utilized in a context as naturally as possible, rather than obsessively mentioning memorized information. In addition, for consistency, we ask them to correct cases where the bot generates responses or questions that may contradict information already included in the memory. For example, if the bot already knows that the user is hospitalized for back surgery, the bot wouldn’t ask questions such as “Are there any health issues?”. Also, the bots are not allowed to mention things that are not in memory as if they do remember those things. For other remaining instructions, we refer to the specifications in [Bae et al. \(2022\)](#) for the consistent role of the bot.

Memory Update At the end of each session, annotators are asked to select the sentences in M and S that could be used in the subsequent sessions. In other words, the statements that are no longer valid or redundant are removed, and the statements that do not conflict with others remain. Additionally, every time when m_i and s_j are exactly the same in terms of information, even if it might be okay to leave either of them, we guide to leave m_i for consistency of the dataset.

B Implementation Details

B.1 Pre-trained Language Models

We use three types of Transformer-based pre-trained language models in our experiments. For response generation (Section 4.1) and dialogue summarization (Section 4.2), we use HyperCLOVA ([Kim et al., 2021](#)) with 6.9B parameters. The model specification follows [Kim et al. \(2021\)](#) and the implementation is based on Megatron-LM ([Shoeybi et al., 2019](#)). For the memory update model (Section 4.3), we use a model of T5 ([Raffel et al., 2020](#)) architecture pre-trained on the corpus identical to that of [Kim et al. \(2021\)](#). This model consists of 24 layers, 1024-dimensional embeddings, and 16 attention heads, resulting in total of 822M parameters. Lastly, for retriever (Section 4.1), we pre-train BERT ([Devlin et al., 2019](#)) on a corpus that we collected in-house and a public Korean dialogue corpus³. Our BERT consists of 12 layers, 768-dimensional embeddings, and 12 attention heads, resulting in total of 110M parameters. The models except HyperCLOVA are based on Huggingface Transformers ([Wolf et al., 2020](#)). Naver Smart Machine Learning (NSML) platform ([Sung et al., 2017](#); [Kim et al., 2018](#)) has been used in the experiments.

B.2 Generator

For efficient training, we employ LoRA ([Hu et al., 2021](#)) for fine-tuning of all response generation and dialogue summarization models. We fix adaptor rank to 4 and LoRA α to 32, with learning rate of 5×10^{-4} , weight decay factor of 0.1, and batch size of 8. The maximum training epoch is 3 with early stopping. Training is completed within 10 hours using 1 NVIDIA A100. The maximum sequence length is 2,048 and the inputs that exceed this length are truncated from the front.

³<https://aihub.or.kr/aihub-data/natural-language/about>

B.3 Retriever

Our retriever implementation is similar to context persona matching (CPM) method proposed in Xu et al. (2022b). The current dialogue context D_t is encoded with dialogue encoder $E_D(\cdot)$, and each memory sentence $m_i \in M$ is encoded with memory sentence encoder $E_m(\cdot)$. Here, $E(\cdot)$ refers to each input’s representation, i.e. the encoder’s output on the first input token ([CLS]). The encoders E_D and E_m are initialized with pre-trained BERT (Devlin et al., 2019) architecture. We use triplet loss to fine-tune the encoders as:

$$\max(\text{sim}(D_t, m^+) - \text{sim}(D_t, m^-) + \alpha, 0), \quad (4)$$

where m^+ is a memory sentence matched with D_t in the training dataset, m^- is a memory sentence from other dialogue sessions in the training dataset, and $\alpha = 0.2$ is the margin. The models are trained for 20 epochs with early stopping using a maximum learning rate of 3×10^{-5} and an linear scheduler. This training takes about 3 hours using 1 NVIDIA V100. At inference time, the top k ($k = 5$ in our experiments) memory sentences are retrieved from M using cosine similarity:

$$\text{sim}(D_t, m_i) = \cos(E_D(D_t), E_m(m_i)). \quad (5)$$

B.4 Memory Operator

For memory operation and NLI tasks, we define a unified text-to-text format of input and output to train our T5. The input sequence becomes “sentence 1: [m or premise sentence] sentence 2: [s or hypothesis sentence]” and the target labels are mapped to single tokens corresponding to numeric characters {“0”, “1”, “2”, “3”}; for memory operation, {“0”: PASS, “1”: APPEND, “2”: REPLACE, “3”: DELETE}, and for NLI, {“0”: Entailment, “1”: Neutral, “2”: Contradiction}. This makes it simple to transfer the model trained with NLI to the memory operation task by just replacing inputs and targets.

The models are trained with a batch size of 8. They are trained for 20 epochs with early stopping, a maximum learning rate of 5×10^{-5} , and a linear scheduler. This training takes about 1 hour using 1 NVIDIA A100.

B.5 Hyperparameter Search

For all models, the learning rate was searched in the range of $[3 \times 10^{-5}, 5 \times 10^{-5}, 5 \times 10^{-4}]$ and the batch size in the range of $[8, 16, 32]$. We tried

at least 3 times for each setting to find the best configurations.

C Discussion

C.1 Persona Updates in Our Dataset

Our dataset contains more persona updates, because of 1) relationship setting between interlocutors and 2) duration of episodes.

According to Altman and Taylor (1973), people disclose more private information as their relationship deepens. MSC (Xu et al., 2022a) was collected in a setting of two strangers getting to know each other, where people tend to discuss rather easily shareable information such as their profiles or preferences. Our dataset reflects a more intimate relationship where people disclose more private realms of life, which is more likely to change over a few weeks (e.g. “getting a physiotherapy”, “gained a few pounds”, “arguing with son”) or do not change frequently but will share updates if occurred (e.g. “got laid off”, “got back together with partner”).

Also, the assumed duration of dialogue in our dataset is relatively longer than MSC (Xu et al., 2022a). The average term between sessions is about 2 days in MSC, while 10 days in our dataset. The entire span of each episode is 5 hours to 5 weeks in MSC, while 5 weeks to 10 weeks in our dataset. Thus, more persona information might change.

C.2 Memory Operation

While collecting the dataset in Section 5.1, we found a few cases where the four proposed pairwise operations might not perfectly guarantee that the stored memory is lossless, consistent, and not redundant. For example, when $m_i =$ “Back hurts but hasn’t seen a doctor yet” and $s_j =$ “Receiving physiotherapy at the hospital”, the resulting memory might not be lossless if only one of them is stored, while it might be inconsistent if both are left in memory. In these cases, we ask the annotators to label them as “FUSION”, which might require combining the two sentences into a new sentence like “Receiving physiotherapy at the hospital for back pain”. The cases of gold label = “FUSION” occur in about 1.09% in the collected pairs, so we assume them to be negligible and discard them from our dataset in Table 2, since it would be costly to collect data exclusively for such cases and train a separate generator. It might be helpful to use a generative approach that combines information from the two sentences into a new one $p(m'|m, s)$ selectively for

such cases, though this would require additional consideration about the relationship between the newly generated sentence and other sentences.

Also, dependencies between memory sentences may exist. For example, in the case of $m_i =$ “got gastroenteritis” and $m_j =$ “The doctor banned alcohol, meat, and flour”, if the preceding sentence disappears, the following sentence should also be removed. This case can be resolved simply by clarifying the fact that it is due to enteritis in the second sentence, but there may be some cases in which putting all the dependencies in a sentence is difficult. A graph structure may be an alternative to design dependencies between memories.

D Variants of Response Generation Models

We compare various types of response generation models according to the training dataset and dialogue context type. When using the original CareCall dataset, since this is a single session dataset, the model is always trained to predict c_{t+1} given only D_t , i.e. $p(c_{t+1}|D_t)$. When using the new CareCall_{mem} dataset, the inputs vary depending on the context type.

- *Without memory*: Given only D_t to predict c_{t+1} .
- *Dialogue history grounded*: The dialogue histories from all previous sessions are concatenated before D_t .
- *Memory grounded*: The model described in Section 4.1, in which M is concatenated before D_t .

We evaluate the above models with 300 sessions of the human written dialogues described in Section 3.2.1 as the test set (all D_t ’s are second sessions of each episode).

Table 6 shows the results. The model trained with CareCall dataset has no significant performance improvement in all metrics when the previous session history is given as a context. Since CareCall dataset does not contain utterances that utilize memory, simply giving previous context at inference time does not enable the model converse using memory. On the other hand, the newly collected CareCall_{mem} dialogues are dependent on the previous session. The model trained with such CareCall_{mem} has a significant performance gain when the previous session history is given in the

form of either raw dialogue history or summary. Also, it is better to give a summary than to give a dialogue history, a finding consistent with a previous work (Xu et al., 2022a). Furthermore, using CareCall + CareCall_{mem} shows better or competitive results in all three types of session context. This is probably because the general conversation performance is more advantageous with more data. We also found that the utterances that explicitly mention memorized information are about 24.8% of all utterances in the CareCall_{mem} dataset, which is a reasonable proportion in real-life conversations. This means that the general conversational ability is also important to predict responses in this task. Therefore, we use CareCall + CareCall_{mem} as the training dataset for response generation model on multi-session dialogues (Section 5.2).

E Human Evaluation Metrics

For each session of dialogue between human and a chatbot, we ask the crowdworkers to evaluate the quality of the chatbot by rating the degree to which they agree with the statement on each evaluation metric on a scale of 0-100 (interface is given in Figure 5). The statements for evaluation metrics are as follows.

- *Coherence*: This chatbot understood the context and responded coherently.
- *Consistency*: This chatbot was consistent throughout the conversation.
- *Engagingness*: I would like to chat with this chatbot for a longer time.
- *Humanness*: This chatbot sounded like a human.
- *Memorability*: This chatbot remembered what I said before.

The statements for the first four metrics are referred from previous literature (Li et al., 2019; Finch and Choi, 2020; Ji et al., 2022; Smith et al., 2022) for consistency of evaluation across different works. We additionally defined memorability to evaluate how well the model remembers previous conversations.

Model	PPL↓	BLEU-1/2↑	F1↑	Dist-1/2↑
<i>Without memory</i>				
CareCall	13.214	0.155/0.054	0.214	0.074/0.178
CareCall _{mem}	11.246	0.173/0.067	0.227	0.075/0.178
CareCall + CareCall _{mem}	10.919	0.161/0.061	0.210	0.080/0.185
<i>Dialogue history grounded</i>				
CareCall	17.541	0.156/0.062	0.234	0.122/0.282
CareCall _{mem}	8.380	0.173/0.067	0.232	0.118/0.285
CareCall + CareCall _{mem}	7.966	0.175/ 0.079	0.238	0.107/0.244
<i>Memory grounded</i>				
CareCall _{mem}	7.503	0.179/0.078	0.236	0.118/0.294
CareCall + CareCall _{mem}	7.520	0.186 /0.075	0.239	0.110/0.279

Table 6: Comparison of different context types and training dataset for response generation model.

Memory sentence <i>m</i>	Summary sentence <i>s</i>	label
Lost appetite and doesn't eat much	Lost appetite	PASS
Not sick	Doesn't have any particular health issues	PASS
Goes hiking every weekend	Goes hiking	PASS
Doesn't have any particular health issues	Had back surgery	REPLACE
Couldn't sleep well	Sleeping well after taking sleeping tablets	REPLACE
Living alone	Being with daughter for a while	REPLACE
Has a grandson in elementary school	Grandson enters middle school	REPLACE
Goes to the gym	Body is sore from exercise	APPEND
Eating properly	Receiving physiotherapy	APPEND
Has a dog	The dog likes carrots	APPEND
Has sleeping tablets prescribed	Has a son	APPEND
Gardening as a hobby	Many flowers are growing in the garden	APPEND
Had sore throat	Throat is fully recovered	DELETE
Takes pain relievers for a migraine	Migraine is gone	DELETE

Table 7: Example sentence pairs from the collected dataset in Table 2. All texts are translated into English.

Session	Model	Coherence	Consistency	Engagingness	Humanness	Memorability
Session 1	Without memory	0.1765	0.6926	-0.0440	0.1058	-0.2317
	Memory accumulate	0.1765	0.6926	-0.0440	0.1058	-0.2317
	Memory update	0.1765	0.6926	-0.0440	0.1058	-0.2317
Session 2	Without memory	0.3553	0.6291	-0.3753	0.1402	-0.1257
	Memory accumulate	0.5069	0.6565	-0.1501	0.4418	0.3812
	Memory update	0.6597	0.9033	-0.0724	0.5376	1.0263
Session 3	Without memory	-0.2383	0.4974	-0.5080	-0.3500	-1.4041
	Memory accumulate	0.2036	0.5614	-0.0640	0.2647	0.5017
	Memory update	0.0854	0.5281	-0.2186	0.0946	0.4338
Session 4	Without memory	-0.1135	0.4846	-0.6073	-0.3916	-2.0036
	Memory accumulate	0.2252	0.6787	-0.5143	-0.1339	-0.2023
	Memory update	0.1472	0.6541	-0.1663	0.1660	0.7231
Session 5	Without memory	-0.3995	0.6622	-0.7786	-0.7298	-1.6713
	Memory accumulate	-0.2753	0.4596	-0.7616	-0.3844	-0.6674
	Memory update	0.0598	0.2672	-0.3807	0.0599	0.2391

Table 8: Per session average standardized scores of human evaluation metrics on live conversation. Since the models in the first session are all equivalent, scores are averaged for all models in Session 1.

Session 1

Hello. Is there anything special these days?

No, I'm just lying down.

I see. Any health issues?

I had a really bad stomachache and couldn't move at all.

Oh, no. It must've hurt. Are you okay now?

I stopped eating and it got better. But I feel nauseous even when I drink water.

Oh, did you go see a doctor?

I'm going to not eat today and see how it goes tomorrow. I'll have some porridge tomorrow.

If it doesn't get better tomorrow, go see a doctor.

Okay, I will.

Did you sleep well?

Yes, I did.

Summary 1

Starving because of a stomachache

Sleeping well

Memory 2

Starving because of a stomachache

Sleeping well

Session 2

Hello. This is a care call.

I see.

How's your stomach?

Oh, it's fine.

That's a relief. I hope you stay healthy.

I'm the healthy type. Don't worry.

Good. Did you sleep well?

I did.

Then, how are you planning on spending your day?

I'm planning on going to the lake park right in front.

I see. How's the weather today?

Good.

Okay. Shall we hang up now?

Sure.

Summary 2

Had a stomachache but recovered

Sleeping well

Goes to lake park

Memory 3

Sleeping well

Goes to lake park

Session 3

Hello. How are you feeling today?

Feels good.

That's great. Are you eating properly?

Yes, I cooked noodles today.

That sounds delicious. I'd like to have some, too.

It's spicy and delicious.

I'm glad you enjoyed it. Make sure you eat properly.

Yes, but my back hurts today.

Oh, your back hurts? Did you go see a doctor?

I went to see a doctor and I was told that it's the age. I need to get physiotherapy.

It'll get better if you get physiotherapy regularly.

Okay, I will.

Okay! I'll call you again later. Thank you.

Summary 3

Eating properly

Receiving physiotherapy because of sore back

Memory 4

Sleeping well

Goes to lake park

Eating properly

Receiving physiotherapy because of sore back

Session 4

Hello. Are you continuing with physiotherapy?

Yes. It's working.

That's a relief. For how long do you need to get it?

I don't know. They told me to get it regularly.

Right, you'll get better soon if you get it regularly.

Right.

Oh, do you still go to the lake park?

Yes. It's great because the weather's beautiful.

I'm glad you're enjoying your walks. It'll be nice if you keep going for walks.

Okay. I will.

Figure 4: Example four session conversation from the newly collected CareCall_{mem} dataset. Memory 1 (an empty set) and Summary 4 are omitted. All the texts are translated into English.

Model	Coherence	Consistency	Engagingness	Humanness	Memorability
Without memory	71.79	94.52	54.67	81.37	36.37
Memory accumulate	73.86	96.10	54.98	82.68	50.31
Memory update	75.98	95.90	58.35	84.40	56.59

Table 9: Average absolute scores of human evaluation metrics on live conversation for 5 sessions in each episode.

Dialogue Context

Bot: Hello. This is a care call.
 User: Hello?
 Bot: Hi, do you have a minute?
 User: Sure.
 Bot: You said your back is fine now. How is it these days?
 User: It's great.
 Bot: That's good. Oh, how's your grandchild doing?
 User: She's growing really well.
 Bot: How old is she?

Send Message

She's going into middle school now.

Send Reset

This chatbot understood the context and responded coherently.

 Value: 50

This chatbot was consistent throughout the conversation.

 Value: 50

I would like to chat with this chatbot for a longer time.

 Value: 50

This chatbot sounded like a human.

 Value: 50

This chatbot remembered what I said before.

 Value: 50

Save Session

Reset Episode

Figure 5: Web-based user interface for the human evaluation. All texts are translated into English. For each session, crowdworkers can communicate with the system by sending messages. At the end of the session, they rate the degree to which they agree with the statements on a scale of 0-100 by moving the sliders. They converse with a randomly selected model for 5 sessions, and then reset episode to converse with another model.