

Personalized LLM Response Generation with Parameterized User Memory Injection

Anonymous ACL submission

Abstract

Large Language Models (LLMs) have exhibited remarkable proficiency in comprehending and generating natural language. On the other hand, personalized LLM response generation holds the potential to offer substantial benefits for individuals. However, existing work struggles with efficiently incorporating user information for LLM personalization. In this study, we draw inspirations from real-world bionic memory mechanism to propose a novel parameterized **Memory-injected** approach using parameter-efficient fine-tuning (PEFT), combined with a Bayesian Optimization searching strategy to achieve **LLM Personalization (MiLP)**. Our MiLP takes advantage from the alignment between real-world memory mechanism and the LLM’s architecture. Extensive experiments have shown the superiority and effectiveness of MiLP. To encourage further research into this area, we are releasing our implementations¹.

1 Introduction

The undeniable capability of large language models in comprehending and producing natural language has been underscored by various studies (Brown et al., 2020; Chowdhery et al., 2022; Touvron et al., 2023). Simultaneously, there exists untapped potential to customize these models for delivering personalized responses to users, enabling them to receive tailored and fitting replies according to their individual requirements (Bender and Koller, 2020). For instance, in an LLM-based medical dialogue scenario, an assistant capable of recognizing the patient’s medical history can generate more tailored responses, rather than offering generic and potentially inappropriate suggestions. Individuals in regions with limited access to the medical resources can benefit significantly from such applications, highlighting the imperative needs for personalized LLM response generation (Chen et al., 2023).

¹A copy is attached with this submission

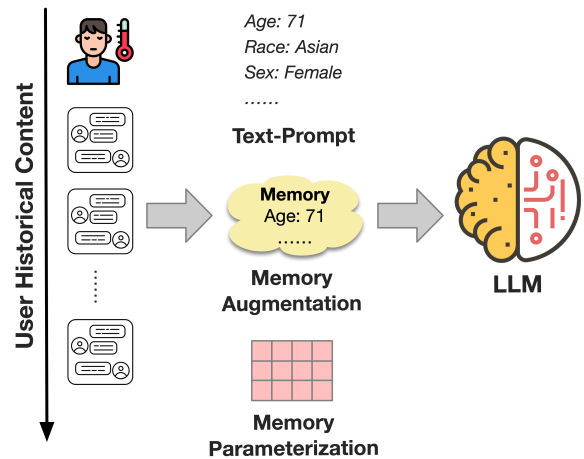


Figure 1: Three types of methods for incorporating user historical content to achieve personalized LLM.

Incorporating user historical information properly to LLM can be a key towards LLM personalization. Existing works can be concluded into three lines as illustrated in Fig. 1. Text-Prompt based methods draw inspiration from in-context learning to organize the user historical content as prompts, providing them to LLM so that personal information can be considered (Petrov and Macdonald, 2023; Kang et al., 2023; Liu et al., 2023). However, this paradigm suffers from the long-context window limitation of LLM (Liu et al., 2024). Memory-based approaches provide a solution by leveraging a memory to store user historical content. When a new user query comes, a retriever will first retrieve relevant user information from the memory to prompt LLM produce personalized responses (Dalvi et al., 2022; Madaan et al., 2022; Lewis et al., 2020; Zhang et al., 2023). Unfortunately, they are limited in capturing fine-grained information due to the nature of similarity comparison retrieval process (Zhang et al., 2023). Additionally, user historical content can be complex and noisy, posing difficulties for LLMs to focus on the most relevant information without a proper learnable process. To address this, recent studies

066 have proposed parameterizing and projecting user
067 historical content into a learnable representation
068 space (Ning et al., 2024; Deng et al., 2022; Zhong
069 et al., 2022). Instead of using text to prompt LLMs,
070 the learned user representations can be neglected
071 in the LLM’s decoding process via cross-attention
072 to enable personalized response generation. In this
073 study, we take a further step by investigating a
074 memorization process that mimics real-world mem-
075 ory mechanisms to incorporate user information,
076 aiming to achieve personalization while mitigating
077 associated challenges.

078 Previous studies in neuroscience have indicated
079 that memory is stored in different parts of the brain
080 and is activated accordingly when needed (Leven-
081 son and Sweatt, 2005; Nadel et al., 2012). Concur-
082 rently, efforts have revealed that the Feed Forward
083 Layers (FFL) of Transformer architecture serve as
084 a memory bank, storing both shallow patterns (e.g.,
085 sentences ending with a certain word) and seman-
086 tic patterns (e.g., sentences about a certain topic)
087 (Tay et al., 2022; Geva et al., 2021; Chen et al.,
088 2024). Subsequent attempts have been made to in-
089 ject external knowledge into LLMs via parameter-
090 efficient fine-tuning (PEFT) (Houlsby et al., 2019;
091 Pfeiffer et al., 2020b; Li and Liang, 2021; Hu et al.,
092 2021), maintaining a modular and adaptable struc-
093 ture without compromising the LLM’s original ca-
094 pabilities compared to fully fine-tuning (Ye et al.,
095 2023; Wang et al., 2020; Diao et al., 2023; Yao
096 et al., 2022; Wang et al., 2020). Drawing valuable
097 inspiration from the alignment between real-world
098 bionic memory mechanisms and LLM’s memory
099 mechanisms, we propose to first parameterize user
100 historical content and store it as memory in the
101 LLM via adapters, followed by fine-tuning the
102 LLM for personalized response generation.

103 However, different memories, with different
104 characteristics highlighting the distinct sensitiv-
105 ity to the allocated parameter budget and the lo-
106 cation of the injected layers(He et al., 2021; Zhang
107 et al., 2023). Unfortunately, most PEFT applica-
108 tions are limited to a single PEFT architecture with
109 fixed decisions on its components (e.g. hidden
110 size, insertion layers) which can not store and ac-
111 tivate different memories for personalization. To
112 address this, we propose to leverage multiple PEFT
113 modules (e.g., LoRAs) for different memory stor-
114 age and utilize a high-dimensional multi-objective
115 Bayesian optimization (BO) approach to determine
116 the optimal configurations for memory storage. In

117 tandem, we draw inspirations from the alignment
118 between real-world bionic memory mechanism
119 and the LLM’s memory mechanism to propose
120 a novel parameterized **Memory-injected** method
121 that capitalizes on PEFT, complemented by a novel
122 Bayesian Optimization-based searching strategy
123 to handle multi-PEFT settings for achieving **LLM**
124 **Personalization (MiLP)**. Our contributions can be
125 outlined as follows:

- In contrast to previous studies, we leverage
126 the alignment between real-world memory mecha-
127 nisms and the LLM’s architecture to inject parame-
128 terized user memory directly into the LLM which
129 offers a fresh perspective for the community on the
130 topic of LLM personalization.
131
- To achieve parameterized memory injection,
132 we propose the MiLP framework, which integrates
133 a comprehensive searching space and a Bayesian
134 optimization-based approach to handle multi-PEFT
135 settings for LLM personalization.
136
- Our empirical results demonstrate significant
137 improvements over three datasets compared to
138 three baselines which validates the effectiveness
139 and superiority of our MiLP approach.
140

141 2 Methodology

142 **Overview** Our proposed MiLP takes user’s content
143 including user profile, historical content (e.g., di-
144 alogues, posts) $\mathcal{U} = \{c_0, \dots, c_n\}$ and a query x as
145 input and the goal is to inject and search for proper
146 memory to output personalized response y . The
147 parameterized user memory injection is achieved
148 by applying multiple Low-Rank Adaption (LoRA)
149 modules into the FFL of the base LLM Φ under an
150 optimal configuration and a modified Bayesian Op-
151 timisation approach is utilized to handle the multi-
152 LoRA setting as illustrated in Fig 2. The LLM’s
153 performance p (e.g., loss, metrics) will be targeted
154 by the optimal search and this process will iterate
155 until it converges. Finally, an instruction-tuning
156 will be performed for aligning the generated re-
157 sponse with human intents.

158 2.1 LoRA Module

159 Previous work have provided insights in the success
160 of injecting knowledge into the LLM via PEFT tun-
161 ing Yao et al. (2022); Wang et al. (2020). Inspired
162 by the function analysis of feed-forward layers in
163 Transformer(Geva et al., 2021), our MiLP modified
164 the usage of Low-Rank Adaption (LoRA) Hu et al.
165 (2021) to the feed-forward layers of the base LLM.

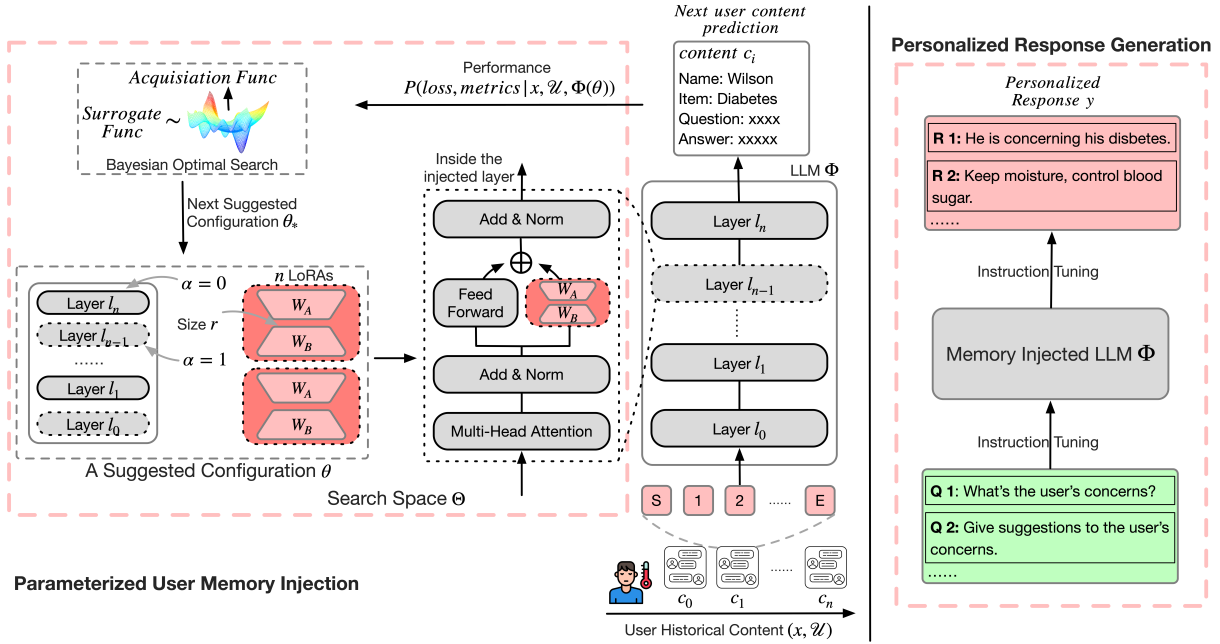


Figure 2: Illustration of the proposed MiLP: The search space encompasses *the number of LoRAs, inserted layers and the size of injected LoRA*. Given a suggested configuration (upper right) (e.g., two LoRAs with fixed size r are injected into the 0-th and $(n-1)$ -th layer, respectively), the base LLM trains on this configuration and take the performance as target. Then the BO search will make a new suggestion and iterate the process until it converges.

For a feed-forward layer $h = W_l x$, the forward process is modified to be:

$$h = W_l x + B A x \quad (1)$$

where $W_l \in \mathbb{R}^{d \times k}$ denotes the weights of the l -th feed-forward layer, $B \in \mathbb{R}^{d \times r}$, $A \in \mathbb{R}^{r \times k}$ are the low-rank decomposed matrices and the rank $r \ll \min(d, k)$.

2.2 Parameterized Memory Injection

Determining how to properly store and activate pertinent information presents a challenge. Inspired by the success of neural architecture search, we start by defining a search space. Subsequently, we employ a Bayesian optimization (BO) approach to identify the optimal configuration for generating personalized responses. In the following sections, we provide a detailed explanation of our search space design, along with the rationale behind it, and describe the process of conducting BO.

2.2.1 Search Space

Inserted Layer Prior work has shown that different feed-forward layers in the LLM store distinct information (Geva et al., 2021) where the shallow layers store shallow patterns (e.g., sentences end with a certain word), deep layers store semantic patterns (e.g., sentences about a

certain topic). Consequently, applying LoRA to all layers can lead to suboptimal results. Thus, we introduce a binary parameter α at each layer l_i that controls whether the layer is active (i.e., to be inserted) or inactive.

Number of LoRAs It is intuitive that the volume of distinct user content can vary, resulting in a range of learnable spaces for injecting such content (Wang et al., 2020). Utilizing a fixed number of LoRAs may therefore result in suboptimal learning spaces for accommodating different user content. To address this, we incorporate the number of LoRAs, denoted as n , into our search space.

Low-Rank Size Prior studies have demonstrated that the performance of LoRA is greatly influenced by the number of adjustable parameters (Chen et al., 2022). Therefore, it is crucial to dynamically adjust its capacity to align with the demands of the specific task to achieve optimal performance. To address this, we include the rank r as a parameter in our search space, which signifies LoRA's capability to store user-specific content in memory.

2.2.2 Bayesian Optimal Search

While much existing work concentrates on identifying a single PEFT module with the best performance, real-world applications often involve opti-

mizing multiple PEFT models, a scenario that has been rarely explored (Zhou et al., 2023). To address this gap, we opt to employ a modified Bayesian Optimization (BO) approach to determine how different parts of injected memory should be utilized in response to a user’s query.

Bayesian Optimization leverages two key components: **1)** A probabilistic *surrogate model* to approximate the objective function using previous observations. **2)** An *acquisition function* that suggests which point in the search space should be evaluated next. The fundamental principle of Bayesian Optimization (BO) is to iteratively select points for evaluation, striking a balance between exploration (searching different areas) and exploitation (focusing on areas likely to yield the best results). The surrogate model estimates the objective function and its uncertainty, while the acquisition function identifies the most promising points to evaluate. By continuously updating the surrogate model and selecting points expected to improve the objective, BO efficiently explores the space for the optimal solution while minimizing the number of evaluations of the costly objective function.

Surrogate Function Applying BO to our defined search space is non-trivial. Thus, we opt for the usage of Sparse AxisAligned Subspace (SAAS-GP) (Eriksson and Jankowiak, 2021) to serve as the surrogate function. SAAS-GP employs robust, sparsity-inducing priors to address the challenge of modeling high-dimensional data. It assumes that despite the nominal high dimensionality, the effective dimensionality is significantly lower, thereby simplifying the modeling process. Given the user’s content \mathcal{U} , a query x and the base LLM Φ with a suggested configuration θ , the performance $p(l, rl|x, \mathcal{U}, \Phi(\theta))$ can be represented by the CrossEntropyLoss $l = -\frac{1}{N} \sum_{i=1}^N \log P(y_i|y_{<i}, \mathcal{U}, x)$, where N is the length of targeted length, and ROUGE-L score rl between generated \hat{y} and targeted response y . Thus, we can give the surrogate function in our settings:

$$p(\theta) \sim \mathcal{N}(\mu(\theta), \sigma^2(\theta)) \quad (2)$$

where $\theta \in \Theta$ is a suggested configuration from our defined search space Θ as elaborated in Section 2.2.1, $\mu(\theta)$ is the mean and $\sigma^2(\theta)$ is the variance. For the kernel function, we use log-Normal distribution as the kernel. Then given a new configuration θ_* , the posterior distribution of $p(\theta_*)$ can be

updated as follows:

$$p(\theta_*)|\{\theta_i, f(\theta_i)\}_{i=1}^n \sim \mathcal{N}(\mu_*, \sigma_*^2) \quad (3)$$

where n is the number of observed points. The mean and variance of the posterior distribution are computed using the Gaussian process regression.

Acquisition Function For acquisition function, we use the Negative Expected Hypervolume Improvement (NEHVI) (Daulton et al., 2021) since it quantifies the negative expected improvement in hypervolume when including a new point in the solution set which in nature is suitable for handling multi-objective optimization setting. The function in our setting can be described as:

$$NEHVI(\theta) = -\mathbb{E}[H(p(\theta^+) \cup p(\theta)) - H(p(\theta^+))] \quad (4)$$

where $H(\cdot)$ is the hypervolume function, $p(\theta^+)$ is a reference point representing the best-known objective values achieved so far and $p(\theta)$ is the predicted function value at θ calculated by the surrogate function.

2.3 Personalized Response Generation

Upon on the learned user representation from historical content, the LLM can be fine-tuned to generate personalized response. We resort to the usage of instruction tuning which has shown great ability for leading LLM to generate desired response in just a few samples (Stiennon et al., 2020; Min et al., 2021; Ouyang et al., 2022). In this work, we fine-tune the memory injected model on instruction-following examples in a supervised manner to aligned the LLM’s response with human intents with respect to the user historical content.

In contrast to prior approaches, we emulate the real-world bionic memory mechanism using multiple LoRA modules and employ a novel BO optimization strategy to manage the multi-LoRA setting. This allows us to achieve effective parameterized memory injection. Coupled with instruction fine-tuning, this approach ultimately leads to LLM personalization using user historical content, as depicted in Figure 2.

3 Experimental Settings

MiLP is tailored to fine-tune the base LLM to generate personalized responses. To evaluate its effectiveness, we compare our method across three public datasets that contain user historical content.

	AmazonQA	Reddit	MedDia
# User	46,923	46,818	60
# Samples	51,936	95,881	10,920
# Len(History)	30.7	72.4	182
Avg. Len(Content)	23.6	22.8	27.8
Avg. Len(Response)	50.2	9.1	23.7

Table 1: Statistics comparison of the datasets

For this evaluation, we utilize four different base LLMs of varying scales.

3.1 Datasets

AmazonQA/Products(Deng et al., 2022) is a public E-commerce dataset of which each data sample contains user’s historical posted content, including questions, answers and reviews as well as the corresponding product’s description².

Reddit(Zhong et al., 2022) is a public dataset collect from social media platforms where a user can post question and respond to other users. Each data sample contains a query, a response and a sequence of this user’s dialogue history³.

MedicalDialogue(Zhang et al., 2023) is a medical dialogue dataset derived from open-source medical corpus of which each data sample contains a patient’s profile, preference and the historical dialogues between the patient and the doctor⁴.

The detailed comparisons can be seen in Table 1. For our experiments, we split the dataset in a user-oriented manner and format each user’s historical content into a fixed text phrase which allows us to perform next user content prediction task to learn the user’s preference.

3.2 Baselines

We opt to compare our MiLP with three different configurations for LLM Personalization: Text-prompt (TpLP), Memory-Augmented(Zhang et al., 2023) (MaLP), User-embedding (UeLP)(Ning et al., 2024) in terms of four LLMs as the base models⁵: DialoGPT(Zhang et al., 2020), RoBERTa(Liu et al., 2019), LLaMA2-7B and LLaMA2-13B(Touvron et al., 2023). For a fair comparison, we use the configuration with the best performance as reported in their paper.

²<https://cseweb.ucsd.edu/~jmcauley/datasets.html>

³<https://github.com/bangbangbang12315/MSP/tree/release/data>

⁴<https://github.com/MatthewKKai/MaLP/tree/main/data>

⁵Due to the resources limitation, we are unable to test larger scale LLMs.

3.3 Evaluation Metrics

Automatic Evaluation We resort to the usage of **ROUGE-1** and **ROUGE-L** to measure the word overlaps between the generated response and the ground truth. Further, since the goal is to generate personalized response, the **Persona F1 (P-F1)**(Ma et al., 2021) is also used to measure the unigram F1 between the generated response and the user’s content (e.g., historical dialogues, profile etc.).

Human Evaluation Automatic evaluation can assure the quality of the generated response with respect to the ground-truth, however, we recognize that human evaluation is needed. Thus, we follow the scoring method of Wang et al. (2023) and calculate the **Win Rate**, scoring the generated response and compare the scores between different settings and the standard generation of the Text-prompt based method.

3.4 Implementation Details

For implementation details, we leverage the Transformers (Wolf et al., 2020) and Adapters(Pfeiffer et al., 2020a) as the base code and conduct extensive experiments with the DialoGPT, RoBERTa, LLaMA2-7B and LLaMA2-13B. We use the AdamW optimizer(Loshchilov and Hutter, 2018; Paszke et al., 2017) with a learning rate of 5e-4 and also a linear warm-up scheduler initialized with 10% of the total training steps as warm-up steps and a weight decay of 1e-4 to avoid over-fitting for all the experiments. The batch size per device is set to 8. Further, for all the LLMs, we follow their default settings from the Transformers (Wolf et al., 2020) and add search space factors in the their configurations. We modified the forward logic of injected layers by combing the outputs from both the base model layer and the injected lora module. For BO algorithm implementation, we resorted to the usage of BoTorch (Balandat et al., 2020) and follow the suggested settings from Zhou et al. (2023) for both surrogate function and acquisition function. For prior distributions, we randomly sample 100 initialisation points for all the experiments. For all datasets used, we split 70% of them as the training set, 10% of them as the validation set and 20% of them as the testing set. For search space, each factor is an integer from different ranges. The details can be seen in Table 2. All the experiments are conducted on a computation node configured with four 80G Tesla A100 GPUs.

Factor	Range
α	[0, 1]
n	[0 ~ 32]
r	[8, 16, 32, 64, 96]

Table 2: Search range for each factor from the space.

4 Experimental Results

4.1 Comparative Study

Table 3 presents the automatic evaluation comparative results between baselines and our proposed MiLP on three datasets. It is evident that the inclusion of memory improves performance across all baseline models, highlighting the ability of provided personal information to enhance personalized response generation. MaLP, which incorporates long- and short-term memory, outperforms text-prompt based methods, indicating the effectiveness of differentiating stored information. However, our proposed MiLP exhibits superior performance compared to them. MiLP achieves average relative improvements of 4.38%, 5.05% and 2.09% in ROUGE-L scores over all base LLMs against the best baselines on three datasets, respectively. One interesting thing we found is that as the base LLMs goes deeper (e.g., the number of hidden layers is larger), the relative improvements will also increase. For example, LLaMA2-13B equipped with MiLP achieves a relative improvement of 0.44% in ROUGE-L score while DialogGPT equipped with MiLP only achieves a relative improvement of 1.19% on MedicalDialogue dataset against the best baseline. We attribute this to the deeper layers learning more semantic features (Geva et al., 2021).

Moreover, the improvements in persona-F1 score confirm that incorporating a memory mechanism allows for the integration of more user-specific information into the generated response, thereby enhancing personalization. However, we observed that MiLP demonstrates better coverage of personalized information compared to the best baselines, achieving average increases of 0.090, 0.088, and 0.117 in persona-F1 score across the three datasets, respectively. This can be attributed to the fact that the approach of retrieving pre-stored memory to augment LLM personalization relies on the quality of retrieval and the LLM’s understanding of the retrieved prompts, which may lead to sub-optimal results. In contrast, user-embedding-based method anticipated in the LLM’s decoding

process leading to a better performance. Additionally, our MiLP injects memory directly into the intricate LLM and achieves a better understanding of the injected information through our proposed BO approach, thereby producing more relevant user-specific information when generating personalized responses. The comprehensive results validate the effectiveness and superiority of our proposed MiLP.

4.2 Quality Study⁶

We further conduct quality study to examine the quality of generated responses as illustrated in Table 4. We observed that leveraging a memory achieves above 50% win rate for all base LLMs over three datasets. We attribute this to the nature that introducing user-specific information as prompts can enhance LLM response generation in terms of personalization. However, relying solely on memory can lead to misunderstandings by the LLM when generating responses, resulting in sub-optimal outcomes. Our MiLP not only utilizes user-specific information from the user’s historical content but also leverages the natural language understanding and inference abilities of the LLM itself through our proposed BO method. This enables the LLM to comprehend which information should be considered when generating a response, leading to optimal performance compared to other baselines in most scenarios. However, we also notice that as the base LLM becomes more complicated, its greater natural language understanding and inferring ability are not always accompanied with better performance. For example, the results of LLaMA2-13B are incomparable with LLaMA2-7B on MedicalDialogue dataset. We attribute this disparity to the greater sparsity of user-specific information in the historical user content of the MedicalDialogue dataset compared to the other two datasets. Consequently, the configuration space for LLaMA2-13B is relatively sparser than that of other models, leading to suboptimal performance. The base LLM size selection with respect to the scale of input information (e.g., user numbers, the memory size etc.) is worthy to be explored in the future. Despite these challenges, the increase in win rate confirms the effectiveness of our proposed MiLP.

Human validation To validate the alignment of our automatic scoring schema with human judgments, we adopted the methodology of Wang et al.

⁶We further provide a Case Study which can be seen in Appendix A.

Model	Type	AmazonQA			Reddit			MedDia		
		ROUGE-1	ROUGE-L	P-F1	ROUGE-1	ROUGE-L	P-F1	ROUGE-1	ROUGE-L	P-F1
DialoGPT	TpLP	16.44	14.63	0.741	14.57	13.89	0.337	15.47	14.31	0.890
	MaLP	17.02	16.31	0.843	16.12	13.40	0.399	17.15	15.87	0.929
	UeLP	18.02	17.74	0.901	15.95	13.71	0.389	16.92	15.04	0.916
	MiLP	18.61	17.83	0.925	16.38	14.51	0.409	17.67	15.94	1.072
RoBERTa	TpLP	17.35	15.41	0.704	13.91	12.81	0.391	14.81	13.99	0.947
	MaLP	18.50	15.76	0.828	14.17	13.96	0.462	17.79	16.80	1.141
	UeLP	18.97	16.19	0.899	15.96	14.86	0.491	16.21	14.33	0.971
	MiLP	19.73	17.59	0.974	16.83	15.09	0.531	18.96	17.18	1.187
LlaMA2-7B	TpLP	19.61	17.71	1.817	14.37	13.70	0.533	17.19	16.77	1.818
	MaLP	19.80	17.06	1.834	13.91	13.09	0.533	19.98	18.89	1.917
	UeLP	20.91	18.79	2.083	16.61	14.74	0.613	18.27	16.73	2.081
	MiLP	21.69	19.96	2.176	18.63	16.81	0.756	20.98	19.73	2.274
LlaMA2-13B	TpLP	24.91	23.36	2.107	20.87	20.19	0.678	22.77	21.32	2.009
	MaLP	22.61	21.29	2.061	21.18	20.78	0.671	23.77	22.69	2.250
	UeLP	25.02	23.74	2.089	22.03	21.80	0.704	22.18	20.88	2.131
	MiLP	25.51	24.25	2.283	22.28	21.83	0.864	24.13	22.96	2.337

Table 3: Comparative results on different datasets using automatic metrics.

Model	Type	AmazonQA	Reddit	MedDia	Space	AmazonQA	Reddit	MedDia
DialoGPT	TpLP	-	-	-	Num	2.011	0.604	2.027
	MaLP	57.37	51.95	69.33	Size	2.017	0.601	2.034
	UeLP	63.20	60.17	75.02	Layer	1.921	0.597	2.001
	MiLP	63.97	60.76	75.78	Num+Size	2.016	0.604	2.073
RoBERTa	TpLP	-	-	-	Num+Layer	2.130	0.731	2.196
	MaLP	57.91	56.39	63.83	Size+Layer	2.195	0.767	2.197
	UeLP	59.99	60.11	66.75	Num+Size+layer	2.283	0.864	2.337
	MiLP	61.97	60.19	67.63				
LlaMA2-7B	TpLP	-	-	-				
	MaLP	64.74	59.67	88.93				
	UeLP	65.91	61.870	89.43				
	MiLP	66.17	59.81	91.83				
LlaMA2-13B	TpLP	-	-	-				
	MaLP	71.82	72.96	87.89				
	UeLP	74.37	75.13	89.18				
	MiLP	75.48	76.61	90.67				

Table 4: Quality study results on different datasets using the Win Rate metric.

(2023) for point-wise evaluation. We hired two master’s students to evaluate 50 response pairs, consisting of responses generated by standard settings and MiLP using LLaMA2-13B, along with the corresponding user content for each pair. The students were asked to indicate which response they deemed better by selecting 1 (win), 0 (tie), or -1 (lose) for each pair. Subsequently, we calculated the Pearson Correlation Coefficient (P.C) and the accuracy between human selections and automatic selections. The high P.C of 0.86 and an accuracy of 91% collectively indicate the feasibility and high confidence of our evaluation method.

4.3 Ablation Study on Search Space

To explore the effectiveness of each factor within the search space, an ablation study is conducted. As can be observed in Table 5, search number n or size

Table 5: Ablation study of using LlaMA2-13B as the base on different search space using Persona-F1 score as the metric.

r only achieves similar personal information coverage which we attribute to these two factors are more related to the scale of stored memory. In contrast, the choice of which layer to inject influences how the LLM understands the injected memory. During our experiments, when only searching which layers to inject, as the number of LoRAs and LoRA size become larger, the overall performance will be better. However, performance eventually dropped once n and r reached a threshold. This observation aligns with our understanding that there should be a balance between the size of input information and trainable parameters. Meanwhile, without being aware of what memory should be used when generating responses will lead to a sub-optimal result. These findings verify the necessity and effectiveness of the comprehensive search space.

4.4 Ablation Study on MiLP Components

To determine whether the effectiveness of MiLP is due to instruction-tuning or the parameterized memory injection component, we conducted an ablation study. The results are presented in Table 6. As shown, neither instruction-tuning alone

Components	ROUGE-1	ROUGE-L	P-F1
Instruction Tuning	23.67	22.07	2.107
Memory Injection	22.81	19.90	2.331
Full MiLP	24.13	22.96	2.337

Table 6: Ablation study of using LLaMA2-13B as the base on different components on MedDia dataset.

nor parameterized memory injection alone can outperform the fully configured MiLP. Additionally, we observed that ROUGE scores are higher with instruction-tuning, while the Persona-F1 score is higher with parameterized memory injection. We attribute this to instruction-tuning aligning the generated response with human intents, while parameterized memory provides more personal information. With all components incorporated, our MiLP demonstrates the highest effectiveness.

5 Conclusion & Future Work

In a nutshell, we propose a novel frame called MiLP to achieve Memory-injected LLM personalization. MiLP uses LoRA as the base PEFT module and leverages a Bayesian Optimisation based approach to iterative inject and search user historical information towards personalized response generation from our defined search space. Additionally, we conduct extensive experiments to compare our method with three baselines on three datasets and the results verify the effectiveness and superiority of our MiLP. Further, an ablation study is conducted for validating the the necessity of each factor within the defined search space.

In the future, exploring scalability with a larger number of users and larger LLMs is essential. Additionally, enhancing the inference ability to better understand user-specific needs is crucial. This includes integrating shared information and user graphs into the LLM to improve personalized response generation.

6 Related Work

Memory-Augmented LLM refers to apply a memory that stores extra information for enhancing LLM’s output (Ouyang et al., 2022). Various efforts have been made to utilize memory in this context. Tandon et al. (2021) proposed leveraging a corrector that can rectify the model’s output based on similar mistakes stored in memory. However, this method focuses on repairing wrong outputs. In contrast, Madaan et al. (2022) argued that stored experiences can be used to prevent incorrect out-

puts by incorporating feedback into the new query. Another usage of memory is to include the memory into a learning frame such as self-learning or teacher-student paradigm so that the LLM can learn by iterative refinement (Madaan et al., 2023; Dalvi et al., 2022). In tandem, the key for better usage of memory is to equip powerful retrievers (Guu et al., 2020; Lewis et al., 2020; Yuan et al., 2022) and improve the effectiveness of storing memory (Zhang et al., 2023). Unlike previous studies, our MiLP framework parameterizes and injects memory directly into the LLM through PEFT modules while accounting for memory budgets.

Personalized LLM has garnered increasing attention for its ability to provide tailored experiences that align with user expectations and needs (Salemi et al., 2023). Previous works focused on identifying user preferences using Ceteris Paribus (CP)-nets (Asher et al., 2010) and modeling user historical content into language models (Zhong et al., 2022; Deng et al., 2022). However, these methods suffer from limited natural language understanding ability of language models. With the emergence of LLMs, prompt-based methods have been developed to design detailed prompts that guide LLMs in producing desired outputs while being aware of user status and contextual content (Wang et al., 2023; Wu et al., 2023; Aher et al., 2023). Another line of approach attempts to leverages memory to store user relevant information. When a new user query arises, a retriever will retrieve relevant user information from the memory to prompt the LLM to produce personalized responses (Dalvi et al., 2022; Madaan et al., 2022; Lewis et al., 2020; Zhang et al., 2023). Moreover, recent studies have explored projecting user information into embeddings, allowing parameterized user data to participate in the decoding process of the LLM to generate personalized responses (Korbak et al., 2023; Salemi et al., 2023; Xu et al., 2023; Ning et al., 2024). In contrast to previous works, we build on the alignment between real-world bionic memory mechanisms and LLM memory mechanisms. We leverage a novel Bayesian Optimization strategy to inject parameterized user memory into the LLM, enabling it to produce personalized responses.

In tandem, our work stands out from previous research as we pioneer a parameterized memory injection method. By leveraging this novel method, user information can be stored and activated effectively to produce personalized responses.

621 Limitations

622 Despite its empirical success, our approach has two
623 notable limitations that warrant attention. Firstly,
624 it has high experimental requirements. Our search
625 space demands significant computational resources;
626 for instance, a minimum of 4 A100 GPUs is needed
627 to run our method. Additionally, our approach re-
628 lies on the user’s historical content, and the sparsity
629 of user-specific information within this content can
630 influence the quality of the generated response. In
631 the near future, recent efficient search strategies on
632 LoRAs, such as LoNAS (Munoz et al., 2024) and
633 Shears (Muñoz et al., 2024), should be considered
634 for more efficient memory injection.

635 Secondly, our method is designed for a single
636 user. Therefore, it would be valuable to explore
637 how the number of users and the scale of the LLM
638 can impact the generated response. For example,
639 when dealing with a larger group of users, it would
640 be important to consider how to assign Personal-
641 ized Fine-Tuning (PEFT) modules (e.g., type, num-
642 ber) for each user and how to select the base LLM
643 (e.g., one single small LLM for one user or one sin-
644 gle layer within a large LLM for one user). How-
645 ever, due to computational and data resource lim-
646 itations, we are unable to explore this at present.
647 We hope to address this in future work and see
648 increasing attention given to this aspect.

649 Ethics Statement

650 After carefully reviewing the ACL Ethics Policy,
651 we are committed to show our respect and obey to
652 consent all.

653 Acknowledgements

654 References

655 Gati V Aher, Rosa I Arriaga, and Adam Tauman Kalai.
656 2023. Using large language models to simulate mul-
657 tiple humans and replicate human subject studies.
658 In *International Conference on Machine Learning*,
659 pages 337–371. PMLR.

660 Nicholas Asher, Elise Bonzon, and Alex Lascarides.
661 2010. Extracting and modelling preferences from
662 dialogue. In *International Conference on Informa-
663 tion Processing and Management of Uncertainty in
664 Knowledge-Based Systems*, pages 542–553. Springer.

665 Maximilian Balandat, Brian Karrer, Daniel R. Jiang,
666 Samuel Daulton, Benjamin Letham, Andrew Gordon
667 Wilson, and Eytan Bakshy. 2020. *BoTorch: A Frame-
668 work for Efficient Monte-Carlo Bayesian Optimiza-
669 tion*. In *Advances in Neural Information Processing
670 Systems 33*.

Emily M. Bender and Alexander Koller. 2020. *Climbing
towards NLU: On meaning, form, and understanding
in the age of data*. In *Proceedings of the 58th Annual
Meeting of the Association for Computational Lin-
guistics*, pages 5185–5198, Online. Association for
Computational Linguistics.

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

Bowen Chen, Namgi Han, and Yusuke Miyao. 2024. A
multi-perspective analysis of memorization in large
language models. *arXiv preprint arXiv:2405.11577*.

Guangzheng Chen, Fangyu Liu, Zaiqiao Meng, and
Shangsong Liang. 2022. Revisiting parameter-
efficient tuning: Are we really there yet? *arXiv
preprint arXiv:2202.07962*.

Lingwei Chen, Ting Chen, Tianjiao Lan, Chu Chen,
and Jay Pan. 2023. The contributions of population
distribution, healthcare resourcing, and transportation
infrastructure to spatial accessibility of health care.
*INQUIRY: The Journal of Health Care Organization,
Provision, and Financing*, 60:00469580221146041.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin,
Maarten Bosma, Gaurav Mishra, Adam Roberts,
Paul Barham, Hyung Won Chung, Charles Sutton,
Sebastian Gehrmann, et al. 2022. Palm: Scaling
language modeling with pathways. *arXiv preprint
arXiv:2204.02311*.

Bhavana Dalvi, Oyvind Tafjord, and Peter Clark. 2022.
Towards teachable reasoning systems: Using a dy-
namic memory of user feedback for continual system
improvement. In *Proceedings of the 2022 Confer-
ence on Empirical Methods in Natural Language
Processing*, pages 9465–9480.

Samuel Daulton, Maximilian Balandat, and Eytan Bak-
shy. 2021. Parallel bayesian optimization of multiple
noisy objectives with expected hypervolume improve-
ment. *Advances in Neural Information Processing
Systems*, 34:2187–2200.

Yang Deng, Yaliang Li, Wenxuan Zhang, Bolin Ding,
and Wai Lam. 2022. Toward personalized answer
generation in e-commerce via multi-perspective pref-
erence modeling. *ACM Transactions on Information
Systems (TOIS)*, 40(4):1–28.

Shizhe Diao, Tianyang Xu, Ruijia Xu, Jiawei Wang,
and Tong Zhang. 2023. Mixture-of-domain-adapters:
Decoupling and injecting domain knowledge to pre-
trained language models memories. *arXiv preprint
arXiv:2306.05406*.

David Eriksson and Martin Jankowiak. 2021. High-
dimensional bayesian optimization with sparse axis-
aligned subspaces. In *Uncertainty in Artificial Intel-
ligence*, pages 493–503. PMLR.

727	Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories . In <i>Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing</i> , pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.	784
728		785
729		786
730		787
731		788
732		
733		
734	Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In <i>International conference on machine learning</i> , pages 3929–3938. PMLR.	789
735		790
736		791
737		792
738		793
739	Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. 2021. Towards a unified view of parameter-efficient transfer learning. <i>arXiv preprint arXiv:2110.04366</i> .	794
740		795
741		796
742	Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In <i>International Conference on Machine Learning</i> , pages 2790–2799. PMLR.	797
743		798
744		799
745		800
746		801
747		802
748	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 <i>International Conference on Learning Representations</i> .	803
749		804
750		805
751		806
752		807
753	Wang-Cheng Kang, Jianmo Ni, Nikhil Mehta, Maheswaran Sathiamoorthy, Lichan Hong, Ed Chi, and Derek Zhiyuan Cheng. 2023. Do llms understand user preferences? evaluating llms on user rating prediction. <i>arXiv preprint arXiv:2305.06474</i> .	808
754		809
755		810
756		811
757		812
758	Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Vinayak Bhalerao, Christopher Buckley, Jason Phang, Samuel R Bowman, and Ethan Perez. 2023. Pretraining language models with human preferences. In <i>International Conference on Machine Learning</i> , pages 17506–17533. PMLR.	813
759		814
760		815
761		
762		
763		
764	Jonathan M Levenson and J David Sweatt. 2005. Epigenetic mechanisms in memory formation. <i>Nature Reviews Neuroscience</i> , 6(2):108–118.	816
765		817
766		818
767	Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. <i>Advances in Neural Information Processing Systems</i> , 33:9459–9474.	819
768		820
769		821
770		822
771		823
772		824
773	Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In <i>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)</i> , pages 4582–4597.	825
774		826
775		827
776		828
777		829
778		830
779		831
780	Junling Liu, Chao Liu, Peilin Zhou, Renjie Lv, Kang Zhou, and Yan Zhang. 2023. Is chatgpt a good recommender? a preliminary study. <i>arXiv preprint arXiv:2304.10149</i> .	832
781		833
782		834
783		835
	Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. <i>Transactions of the Association for Computational Linguistics</i> , 12:157–173.	836
		837
		838
		839
	Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. <i>arXiv preprint arXiv:1907.11692</i> .	
	Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In <i>International Conference on Learning Representations</i> .	
	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 <i>Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval</i> , pages 555–564.	
	Aman Madaan, Niket Tandon, Peter Clark, and Yiming Yang. 2022. Memory-assisted prompt editing to improve gpt-3 after deployment. In <i>Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing</i> , pages 2833–2861.	
	Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2023. Self-refine: Iterative refinement with self-feedback. <i>arXiv preprint arXiv:2303.17651</i> .	
	Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2021. Metaicl: Learning to learn in context. <i>arXiv preprint arXiv:2110.15943</i> .	
	J Pablo Muñoz, Jinjie Yuan, and Nilesh Jain. 2024. Shears: Unstructured sparsity with neural low-rank adapter search. <i>arXiv preprint arXiv:2404.10934</i> .	
	Juan Pablo Munoz, Jinjie Yuan, Yi Zheng, and Nilesh Jain. 2024. Lonas: Elastic low-rank adapters for efficient large language models. In <i>Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)</i> , pages 10760–10776.	
	Lynn Nadel, A Hupbach, R Gomez, and K Newman-Smith. 2012. Memory formation, consolidation and transformation. <i>Neuroscience & Biobehavioral Reviews</i> , 36(7):1640–1645.	
	Lin Ning, Luyang Liu, Jiaying Wu, Neo Wu, Devora Berlowitz, Sushant Prakash, Bradley Green, Shawn O’Banion, and Jun Xie. 2024. User-llm: Efficient llm contextualization with user embeddings. <i>arXiv preprint arXiv:2402.13598</i> .	
	Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. <i>Advances in Neural Information Processing Systems</i> , 35:27730–27744.	

840	Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.	896
841		897
842		898
843		899
844	Aleksandr V Petrov and Craig Macdonald. 2023. Generative sequential recommendation with gptrec. <i>arXiv preprint arXiv:2306.11114</i> .	900
845		901
846		902
847	Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020a. Adapterhub: A framework for adapting transformers. In <i>Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations</i> , pages 38–45, Online. Association for Computational Linguistics.	903
848		904
849		905
850		906
851		907
852		908
853		909
854	Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020b. Mad-x: An adapter-based framework for multi-task cross-lingual transfer. In <i>Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)</i> , pages 7654–7673.	910
855		911
856		912
857		913
858		914
859		915
860	Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. 2023. Lamp: When large language models meet personalization. <i>arXiv preprint arXiv:2304.11406</i> .	916
861		917
862		918
863		919
864	Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. <i>Advances in Neural Information Processing Systems</i> , 33:3008–3021.	920
865		921
866		922
867		923
868		924
869		925
870	Niket Tandon, Aman Madaan, Peter Clark, and Yiming Yang. 2021. Learning to repair: Repairing model output errors after deployment using a dynamic memory of feedback. <i>arXiv preprint arXiv:2112.09737</i> .	926
871		927
872		928
873		929
874	Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022. Transformer memory as a differentiable search index. <i>Advances in Neural Information Processing Systems</i> , 35:21831–21843.	930
875		931
876		932
877		933
878		934
879	Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. <i>arXiv preprint arXiv:2302.13971</i> .	935
880		936
881		937
882		938
883		939
884		940
885		941
886	Hongru Wang, Rui Wang, Fei Mi, Zezhong Wang, Ruifeng Xu, and Kam-Fai Wong. 2023. Chain-of-thought prompting for responding to in-depth dialogue questions with llm. <i>arXiv preprint arXiv:2305.11792</i> .	942
887		943
888		944
889		945
890		946
891	Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Guihong Cao, Daxin Jiang, Ming Zhou, et al. 2020. K-adapter: Infusing knowledge into pre-trained models with adapters. <i>arXiv preprint arXiv:2002.01808</i> .	947
892		948
893		949
894		950
895		950
	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. <i>Transformers: State-of-the-art natural language processing</i> . In <i>Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations</i> , pages 38–45, Online. Association for Computational Linguistics.	
	Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas Funkhouser. 2023. Tidybot: Personalized robot assistance with large language models. <i>arXiv preprint arXiv:2305.05658</i> .	
	Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. <i>arXiv preprint arXiv:2304.01196</i> .	
	Yunzhi Yao, Shaohan Huang, Li Dong, Furu Wei, Huajun Chen, and Ningyu Zhang. 2022. Kformer: Knowledge injection in transformer feed-forward layers. In <i>CCF International Conference on Natural Language Processing and Chinese Computing</i> , pages 131–143. Springer.	
	Qichen Ye, Junling Liu, Dading Chong, Peilin Zhou, Yining Hua, and Andrew Liu. 2023. Qilinmed: Multi-stage knowledge injection advanced medical large language model. <i>arXiv preprint arXiv:2310.09089</i> .	
	Xingdi Yuan, Tong Wang, Yen-Hsiang Wang, Emery Fine, Rania Abdelghani, Pauline Lucas, Hélène Sauzéon, and Pierre-Yves Oudeyer. 2022. Selecting better samples from pre-trained llms: A case study on question generation. <i>arXiv preprint arXiv:2209.11000</i> .	
	Kai Zhang, Fubang Zhao, Yangyang Kang, and Xiaozhong Liu. 2023. Memory-augmented llm personalization with short-and long-term memory coordination. <i>arXiv preprint arXiv:2309.11696</i> .	
	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 <i>ACL, system demonstration</i> .	
	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. <i>arXiv preprint arXiv:2204.08128</i> .	
	Han Zhou, Xingchen Wan, Ivan Vulić, and Anna Korhonen. 2023. Autopeft: Automatic configuration search for parameter-efficient fine-tuning. <i>arXiv preprint arXiv:2301.12132</i> .	

Background: A patient with *diabetes*, who has previously experienced *demopathy*, is consulting a doctor about recent symptoms of *red spots* appearing on his skin. He also has a history of *skin disease* due to *respiratory tract infection*. The LLM with different settings will respond to the following query.
New Query: I got *red spots on my skin* recently, can you help me with this?

TpLP

Response:

As a doctor, I would recommend scheduling an appointment for a thorough examination. *Red spots on the skin* can be caused by a variety of factors, including *infections, allergic reactions,* or other skin conditions. It's important to determine the underlying cause in order to provide the appropriate treatment. In the meantime, avoid scratching or irritating the affected area, and keep the skin clean and moisturized. If you experience any other symptoms or if the red spots worsen, seek medical attention promptly.

MaLP

Response:

The sudden appearance of red spots on your skin could probobaly due to the recurrence of *respiratory infections. Red spots on the skin* can be caused by a variety of factors, including *infections, allergic reactions,* or other underlying health issues. It's important to have these spots examined by a healthcare professional to determine the exact cause and appropriate treatment. I recommend scheduling an appointment with a dermatologist as soon as possible. In the meantime, avoid scratching or irritating the affected area, keep your skin clean and moisturized.

MiLP

Response:

Given your history of *diabetes* and previous experience with *demopathy*, the appearance of *red spots on your skin* raises additional concerns. In individuals with *diabetes*, skin conditions can sometimes be more complex and may indicate underlying issues with circulation, immune function, or nerve damage. *Red spots on your skin* could signify a range of possibilities, including *skin infections, allergic reactions, or even a complication related to your diabetes.* Please monitor the spots closely and avoid any further irritation. Ensure that your *blood sugar* levels are well-controlled.

Figure 3: Case study of generated responses by LLaMA2-13B under three different settings based on provided background information.

A Case Study

As can be seen in Figure 3, Tex-prompt-based memory only perceives that this patient has *skin disease* and give general reasons as well suggestions. MaLP, on the other hand, captures the *respiratory infection* information due to its long- and short-term memory coordination and gives a more targeted analysis. However, the suggestions are still in general. Our MiLP, with the ability of understanding and inferring user-specific information, produces more detailed responses such as "blood sugar control", "circulation" etc. It analyzes the potential causes and gives suggestions from both diabetes and skin infection aspects. The response of MiLP covers the most personal information against other two settings which show the high quality of generated responses in terms of penalization and further confirm the power of our proposed method.