where $\boldsymbol{o} = (w_{L+1}, \ldots, w_{L+N})$ denotes the continuation of context $\boldsymbol{c}$. This objective helps improve generalization and circumvent excessive reliance on, and overfitting to, the autoencoding task.

2.3 Instruction Fine-tuning

After pretraining, the memory slots produced by the pretrained ICAE are expected to represent the original context. However, for LLMs, the purpose of providing a context extends beyond rote memorization or continuation; instead, the more common use scenario is using the provided context as a basis for accurately and appropriately responding to various prompts, ultimately accomplishing the tasks we want it to perform (Wei et al., 2021; Ouyang et al., 2022).

To enhance the interaction of memory slots produced by the ICAE with diverse prompts, we further fine-tune the ICAE with the PwC dataset (**P**rompt-with-**C**ontext), a dataset[1] introduced in this paper consisting of thousands of (context, prompt, response) samples (as shown in Figure 1).

Formally, the ICAE is fine-tuned for learning to encode the context into the memory slots based on which the decoder (i.e., the target LLM) can produce a desirable response $r_1 \ldots r_n$ according to a given prompt $p_1 \ldots p_m$, as shown in Figure 8 in Appendix A:

$$\begin{aligned}\mathcal{L}_{\text{FT}} &= \max_{\widetilde{m_1} \ldots \widetilde{m_k}} P(r_1 \ldots r_n | \widetilde{m_1} \ldots \widetilde{m_k}, p_1 \ldots p_m; \Theta_{LLM}) \\ &= \max_{\Theta_{LoRA}, e_m} P(r_1 \ldots r_n | m_1 \ldots m_k, p_1 \ldots p_m; \Theta_{LLM}, \Theta_{LoRA}, e_m)\end{aligned}$$

3 Experiments

3.1 Experimental Setting

Data We pretrain the ICAE with the Pile (Gao et al., 2020). For instruction fine-tuning, we use the PwC dataset, as introduced in Section 2.3, which contains 240k (context, prompt, response) samples for training and 18k samples for testing. The context length distribution of test samples is shown in Figure 10. By default, the maximal token length (excluding memory slots) we set during training is 512 in both the ICAE's encoder and decoder in our experiments.

Model Configuration We use the LlaMa (Touvron et al., 2023a;b) as the target LLM to test the ICAE's performance in context compression. For the encoder of the ICAE, LoRA is applied to the query and value projections of the LLM's multi-head attention. In our default setting, the memory slot length k is set to 128, and the LoRA rank r is set to 128 unless otherwise specified. The resulting ICAE only adds about 1% learnable parameters on top of the target LLM.

3.2 Results

3.2.1 Pretrained ICAE

We first evaluate the autoencoding performance of the pretrained ICAE (without instruction fine-tuning) using the following three metrics to understand how well it restores the original context from its produced memory slots: BLEU (Papineni et al., 2002), Exact-Match (EM)[2] and cross entropy loss.

Figure 4 presents the autoencoding results of the ICAE based on the Llama-7b. The ICAE demonstrates a very low overall loss, below 0.05, indicating that the produced memory slots retain almost all the information of the original context. When the context length is within 300, the ICAE can almost perfectly reconstruct the original context, achieving nearly 100% BLEU and EM scores. As the context length increases beyond 400, both BLEU and EM scores start to decline, indicating insufficient capacity of the 128-length memory slots. However, even at a context length of 500, the median BLEU remains over 0.98, and the median EM approaches 0.6 (e.g., perfectly reconstructing about the first 300 words of a 512-token context), showing remarkable performance of ICAE.

We then analyze the effect of the memory size k on the result. According to Figure 5, as the memory slot length k decreases, the ICAE's ability to memorize longer samples significantly deteriorates.

[1] Despite some (prompt, response) datasets such as Self-Instruct (Wang et al., 2022), most of their samples either have no context or very short contexts, which are not suitable for evaluation in our setting. Therefore, we establish the PwC dataset with the help of the GPT-4 (OpenAI, 2023). We include the details in Appendix C.

[2] EM denotes the proportion of the exact matching prefix length to the total length. For a context of 512 tokens, if its first 256 tokens are perfectly restored but its 257th token is not, the EM score is $256/512 = 0.5$.