Table 2: 1 example showing how the pretrained ICAE ($k = 128$) restores the original context.

Origin Context	Restoration
Large pretrained language models have shown surprising In-Context Learning (ICL) ability. With a few demonstration input-label pairs, they can predict the label for an unseen input without additional parameter updates. Despite the great success in performance, the working mechanism of ICL still remains an open problem. In order to better understand how ICL works, this paper explains language models as meta-optimizers and understands ICL as a kind of implicit finetuning. Theoretically, we figure out that the Transformer attention has a dual form of gradient descent based optimization. On top of it, we understand ICL as follows: GPT first produces metagradients according to the demonstration examples, and then these meta-gradients are applied to the original GPT to build an ICL model. Experimentally, we comprehensively compare the behavior of ICL and explicit finetuning based on real tasks to provide empirical evidence that supports our understanding. The results prove that ICL behaves similarly to explicit finetuning at the prediction level, the representation level, and the attention behavior level. Further, inspired by our understanding of meta-optimization, we design a momentum-based attention by analogy with the momentum-based gradient descent algorithm. Its consistently better performance over vanilla attention supports our understanding again from another aspect, and more importantly, it shows the potential to utilize our understanding for future model designing.	Large pretrained models have shown surprising In-Context Learning (ICL) ability. With a few demonstration input-label pairs, they can predict the label for an unseen input without additional parameter updates. Despite the great success in performance, the working mechanism of ICL still remains an open problem. In order to better understand how ICL works, this paper explains how language models as meta-optimizers and understands ICL as a kind of implicit finetuning. Theoretically, we figure out that the Transformer attention has a dual form of gradient descent based on optimization. On top of it, we understand ICL as follows: GPT first produces metagradients according to the demonstration examples, and then these meta-gradients are applied to the original GPT to build an ICL model. Experimentally, we comprehensively compare the behavior of ICL and explicit finetuning based on real tasks to provide empirical evidence that supports our findings. The experimental evidence proves that ICL behaves like us to the same extent. Prediction at the explicit finetuning level, the representation level, and the attention behavior level. Further, inspired by our understanding of meta-optimization, we design a momentum-based attention by analogy with the gradient descent-based momentum gradient algorithm. Its consistently better performance against vanilla attention supports us again from another aspect, and more importantly, it shows the potential to use our understanding for future modeling tasks.

Table 3: Restoration performance for different types of 512-token content with 128 memory slots. Patterned random text is obtained by adding 1 to each token_id in a normal text.

Content type	Loss	BLEU
Normal text	0.01	99.3
Patterned random text	1.63	3.5
Completely random text	4.55	0.2

Based on this intuition, it is very likely that a more powerful LLM may support a higher compression ratio without significant forgetting. We will discuss it in Section 3.3.1.

3.2.2 Fine-tuned ICAE

In order to evaluate the fine-tuned ICAE's performance, we evaluate on the PwC test set. We use the GPT-4 to compare the outputs of the two systems to determine which one performs better or if they are on par with each other, following Mu et al. (2023). Table 4 shows the comparison of results of the LLMs conditioned on memory slots and original contexts. For Llama-7b (fine-tuned ICAE), we compare with Alpaca and StableLM-tuned-alpha-7b since there is no official instruction-tuned Llama-1 model. The Llama-7b (ICAE) conditioned on 128 memory slots largely outperforms both Alpaca and StableLM which can access original contexts (∼512 tokens), with a win rate of 56.7% and 74.1% respectively and a win+tie rate of 73%∼81%. However, when compared to the GPT-4 (we regard it as the gold standard), there is still a significant gap, with around 70% of the cases underperforming the GPT-4's results, and a win+tie ratio of about only 30%.

When we switch the base model to Llama-2-chat, we observe ICAE's performance becomes much better than its counterpart based on Llama-1: when $k = 128$, its win+tie rate can reach around 75% againt the GPT-4 although it still lags behind its counterpart conditioning on the original context as the compression is lossy. As k increases, the win+tie rate further improves while the compression rate decreases. We perform the same comparative studies on Llama-2-13b-chat and observe better results of ICAE, supporting our assumption in Section 3.2.1 that the ICAE can benefit more on larger LLMs.

We investigate the impact of memory length on results. Table 5 shows pairwise comparisons between ICAE models with varying memory slot lengths. A higher compression ratio makes it harder to ensure response quality, but a larger ratio doesn't always lead to worse performance. Table 5 highlights that a pretrained ICAE with 8× compression (k=64) can match a non-pretrained ICAE with 4× compression (k=128). Under the same ratio, the pretrained ICAE performs much better than its