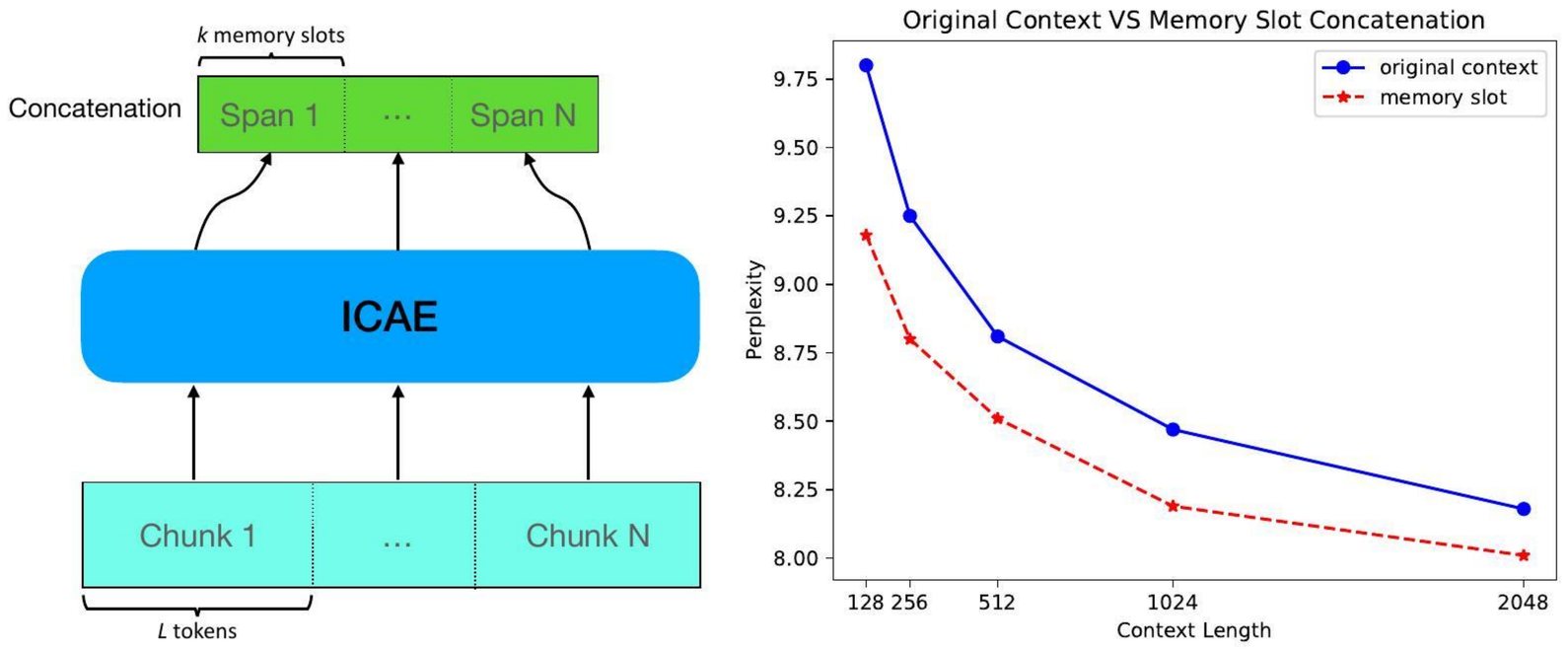

Figure 6: **Left:** Individually compress then concatenate multiple spans of memory slots; **Right:** Perplexity comparison with original contexts and $4\times$ compressed memory slots – for example, 1024-length memory slots are obtained by compressing the original context with a length of 4096 tokens.

thus does not address the real issue of long contexts. Also, this method requires fine-tuning the LLM, and the obtained gist tokens also need to be used within the specially tuned LLM (for gist tokens) and seem not compatible with the untouched LLM. AutoCompressors for recursively compressing long text into summary vectors. Like Mu et al. (2023), the LLM must be tuned to work with generated summary vectors and its training is sophisticated as it involves recursive compression. In contrast, we propose a very simple, straightforward and scalable approach to generating memory slots that can be used in the target LLM with different prompts for various purposes. Moreover, our approach is much more parameter-efficient (i.e., LoRA) for tuning on top of the existing LLM. Additionally, some recent work studies how to compress prompts into more concise natural language (Jiang et al., 2023a), and approaches the context limit with divide-and-conquer methodology (Bertsch et al., 2023; Chen et al., 2023; Song et al., 2024).

Also, there is related work studying compressing indescribable concepts into (vector) tokens for later use in other contexts. Representative work includes Gal et al. (2022) which compresses a vision object into a token and Ge et al. (2023) which compresses a text style into a token.

Considering related work from a boarder perspective of compression, Jiang et al. (2023b) examines kNN-based prediction using general-purpose compressors, such as gzip. Delétang et al. (2023) extensively investigates the compression abilities of LLMs, uncovering their potential as versatile predictors, which also provides insights into recent developments in scaling laws and tokenization.

5 Conclusion and Future Work

We propose the In-context Autoencoder (ICAE) to leverage the power of an LLM to highly compress contexts. By generating compact and informative memory slots to represent the original context, the ICAE enables an LLM to acquire more information with the same context length or represent the same content with a shorter context, thereby enhancing the model's capability to handle long contexts as well as reducing computation and memory overheads for inference in many practical scenarios like Retrieval Augmented Generation (Lewis et al., 2020) and advanced prompting methods (Wei et al., 2022; Wang et al., 2023; Zhang et al., 2024). Moreover, ICAE provides insight into how an LLM performs memorization, offering a novel perspective on the connection between the memory of LLMs and humans, and suggesting future research in LLM context management.

Due to computational limitations, our experiments were conducted on Llama models up to 13 billion parameters. As discussed in the paper, ICAE is expected to benefit even more from more powerful LLMs, where it should be able to achieve more significant compression ratios. In the future, we hope to have sufficient computational resources to validate the effectiveness of ICAE on larger and stronger LLMs. In addition, we plan to explore the application of ICAE in multimodal LLMs (as the context length for images, videos, and audio is often much longer and has greater compression potential) with discrete memory slots (which can be either continuous or discrete) for helping unify compact representation across modalities in the era of LLM/AGI.