OTCE: Hybrid SSM and Attention with Cross Domain Mixture of Experts to construct Observer-Thinker-Conceiver-Expresser

Jingze Shi [*1] **Bingheng Wu** [*1 2] **Ting Xie** [2] **Chunjun Zheng** [3]

Abstract

We have found that combining Mamba with Transformer architecture outperforms using Mamba or Transformer architecture alone in language modeling tasks. We propose a position information injection method that connects the selective state space model with the quadratic attention, and integrates these two architectures with hybrid experts with cross-sharing domains, so that we can enjoy the advantages of both. We design a new architecture with a more biomimetic idea: Observer-Thinker-Conceiver-Expresser (OTCE), which can compete with well-known medium-scale open-source language models on a small scale in language modeling tasks.

1. Introduction

The Transformers (Attention is All You Need (**?**)) architecture is popular in modern deep learning language modeling, which can directly capture the relationship between any two elements in a sequence, effectively handle long-distance dependencies, however, the architecture has two main drawbacks. First, when processing long sequences, its self-attention mechanism's quadratic complexity and cache size limit the ability to handle long contexts. Second, Transformer lacks a single summary state, which means that each generated token must compute over the entire context, which exacerbates the model's computational burden. Meanwhile, the Selective State Model (Mamba (**?**)) has emerged. Mamba achieves linear scaling of sequence length during training and maintains a constant state size during generation through its selective state update mechanism. Moreover, due to its linear recursive state update mechanism, Mamba has a single summary state. However, Mamba also has a major drawback, that is, its positional information depends on the implicit local positional information provided by the causal convolution, while long-distance dependencies depend on the matrix D that skips the connection between input and output. This makes Mamba perform poorly in capturing long-distance dependencies, such as correctly capturing input-output formats in context learning (ICL). An efficient model must have a small state, and an effective model must have a state that contains all the necessary information from the context. To build a model that is both efficient and effective, the key is to design a state that is both compact and comprehensive in capturing the necessary context information. Our main goal is to combine self-attention and the Selective State Model to overcome their respective limitations, further combine them with a mixed expert with extensive general and cross-domain knowledge to build a better basic model architecture than Transformers or Mamba. The model has the ability to learn long context dependencies, aggregate states, and efficient reasoning. This paper proposes a bionic perspective, aiming to explore new model architectures by cleverly combining the Selective State Model with self-attention mechanism. This approach can fully utilize the advantages of the two mechanisms and promote language modeling in a more efficient and effective direction.

1.1. Positional Information

We first identified a key challenge in combining the Selective State Model with self-attention: the effective integration of positional information. In Mamba, positional information is provided by the implicit local positional information from the causal convolution, while self-attention itself cannot provide positional information, it relies on positional encoding to provide global positional context. To address this issue, we designed a relative positional information injection method that connects the inner product state of the Selective State Space with the inner product state of self-attention, allowing the input gate-state-output gate of the Selective State Space to make filtered relevant information have discrete relative positional information, and in self-attention, the discrete relative positional information is re-continuous to build long-term dependency relationships of relevant information. This method not only enables our model to have the ability to selectively process input sequences but also effectively capture long-distance dependency relationships. In complex multi-query associative recall tasks, our model trained on the same dataset outperforms larger-scale Mamba, Transformer models, and models that mix Mamba and Transformer without our proposed relative positional information injection method.

1.2. Cross-Domain Mixture of Experts

In human society, knowledge is widely distributed across different domains, and these domains are interconnected through common foundational knowledge and cross-domain connections. To simulate this phenomenon in the model, we designed two types of cross-domain mixed experts: Cohesive Cross-Domain Expert and Expansive Cross-Domain Expert. These experts (multi-layer perceptrons) store and transfer common foundational knowledge and cross-domain knowledge between different domains by sharing parameters. The Cohesive Cross-Domain Expert achieves close integration between domains by sharing linear layer parameters within all experts, which is more suitable for small-scale models with fewer experts because of its faster computation speed. The Expansive Cross-Domain Expert shares a complete MLP parameter, adding a common domain knowledge gate in each expert to control the flow of common domain knowledge into private MLP parameters, which is more suitable for large-scale models with more experts because it allows for more flexible adjustment and utilization of common knowledge. Experimental results show that the performance of these two cross-domain mixed experts on the same dataset is better than shared expert isolated mixed experts, confirming the effectiveness of our design in promoting cross-domain knowledge transfer and improving model generalization.

1.3. Architecture Design

From a biological perspective, the relationship between input and output can be described by observing, thinking, conceiving, and expressing four stages. Inspired by this idea, we designed a new architecture with a bionic perspective: Observer-Thinker-Conceiver-Expresser (**OTCE**). The OTCE architecture mimics the natural process of information processing in biology, aiming to optimize information processing and transmission in a modular way. In the Observer module, we use the selective state space's selection ability to filter out irrelevant information in the sequence to retain relevant information. In the Thinker module, we use the self-attention's ability to capture dependencies between any two elements in a sequence of any length, regardless of their position in the sequence, to build long-term dependency relationships. In the Conceiver module, we use the linear recursive state update mechanism of the state space to build a single summary state information. In the Expresser module, we combine the context-aware state information produced by self-attention considering all elements with the summary state information to build a context-aware summary state. We also studied the combination of the Selective State Space and self-attention with ordinary multi-layer perceptrons and cross-domain mixed experts, and finally determined a model combination with the lowest perplexity at the same parameter scale.

We empirically validated OTCE on multiple tasks, including semantic similarity evaluation, long-short text classification, natural language inference, keyword recognition, different domain selection tasks, context learning, and multi-query associative recall tasks. These experiments demonstrate the effectiveness of the OTCE architecture in handling complex language tasks.

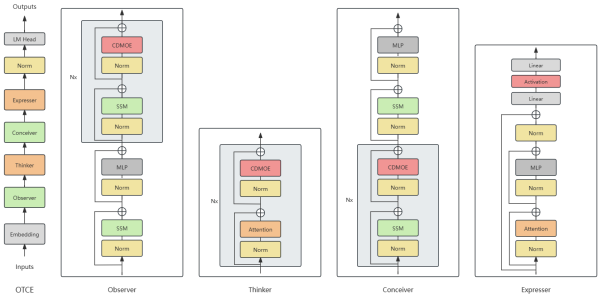

Figure 1. (**OTCE Architecture**.) OTCE demonstrates the overall combined architecture and process of using Observer, Thinker, Conceiver, and Expresser modules in language modeling tasks. Observer, Thinker, Conceiver, and Expresser show their internal combination of selective state space, self-attention, multi-layer perceptron, and cross-domain mixed experts.

2. Background

2.1. Selective State Space Models

Selective state space models (**?**) consider that a fundamental problem in sequence modeling is to compress the context into a smaller state. From this perspective, the attention mechanism of Transformers explicitly stores the entire context information, as if reviewing all previous inputs and generated tokens before writing each token. In contrast, RNNs only refer to a fixed number of previous tokens each time, which allows for faster writing but may forget key tokens.

In the precursor state space models of selective state space models, there are 4 parameters$(\Delta t, A, B, C)$, and they do not change with the input. These parameters control the following two stages:

$$h'(t) = Ah(t) + Bx(t) \quad \text{(1a)}$$
$$y(t) = Ch(t) \quad \text{(1b)}$$
$$h_t = \overline{A}h_{t-1} + \overline{B}x_t \quad \text{(2a)}$$
$$y_t = Ch_t \quad \text{(2b)}$$
$$\overline{K} = (C\overline{B}, C\overline{AB}, \ldots, C\overline{A}^{k-1}\overline{B}, \ldots) \quad \text{(3a)}$$
$$y = x * \overline{K} \quad \text{(3b)}$$

- In the first stage (1a & 1b), the continuous param-

eters $h'(t)$ are transformed into discrete parameters h_t through a fixed formula $\overline{A} = f_A(\Delta t, A)$ and $\overline{B} = f_B(\Delta t, A, B)$, where (f_A, f_B) is called the discretization rule. The most common is the zero-order hold (ZOH) rule, defined as $\overline{A} = \exp(\Delta tA)$ and $\overline{B} = (\Delta tA)^{-1}(\exp(\Delta tA) - I) \cdot \Delta tB$.

- In the second stage (2a & 2b and 3a & 3b), after the parameters are transformed from $(\Delta t, A, B, C)$ to $(\overline{A}, \overline{B}, C)$, linear recursion or global convolution can be used for computation.

State space models cannot update states selectively based on different input information.

The solution of selective state space models (Mamba (**?**)) is that, compared to state space models that compress all historical information, they design a simple selection mechanism that parameterizes the input of the state space model,making $(\Delta t, B, C)$ a function of the input.

2.2. Theorems and such

The preferred way is to number definitions, propositions, lemmas, etc. consecutively, within sections, as shown below.

Definition 2.1. A function $f : X \to Y$ is injective if for any $x, y \in X$ different, $f(x) \neq f(y)$.

Using **??** we immediate get the following result:

Proposition 2.2. *If f is injective mapping a set X to another set Y, the cardinality of Y is at least as large as that of X*

Proof. Left as an exercise to the reader. □

?? stated next will prove to be useful.

Lemma 2.3. *For any $f : X \to Y$ and $g : Y \to Z$ injective functions, $f \circ g$ is injective.*

Theorem 2.4. *If $f : X \to Y$ is bijective, the cardinality of X and Y are the same.*

An easy corollary of **??** is the following:

Corollary 2.5. *If $f : X \to Y$ is bijective, the cardinality of X is at least as large as that of Y.*

Assumption 2.6. The set X is finite.

Remark 2.7. According to some, it is only the finite case (cf. **??**) that is interesting.

2.3. Citations and References

Please use APA reference format regardless of your formatter or word processor. If you rely on the LaTeX bibliographic facility, use `natbib.sty` and `icml2025.bst` included in the style-file package to obtain this format.

Citations within the text should include the authors' last names and year. If the authors' names are included in the sentence, place only the year in parentheses, for example when referencing Arthur Samuel's pioneering work (**?**). Otherwise place the entire reference in parentheses with the authors and year separated by a comma (**?**). List multiple references separated by semicolons (**???**). Use the 'et al.' construct only for citations with three or more authors or after listing all authors to a publication in an earlier reference (**?**).

Authors should cite their own work in the third person in the initial version of their paper submitted for blind review. Please refer to **??** for detailed instructions on how to cite your own papers.

Use an unnumbered first-level section heading for the references, and use a hanging indent style, with the first line of the reference flush against the left margin and subsequent lines indented by 10 points. The references at the end of this document give examples for journal articles (**?**), conference publications (**?**), book chapters (**?**), books (**?**), edited volumes (**?**), technical reports (**?**), and dissertations (**?**).

Alphabetize references by the surnames of the first authors, with single author entries preceding multiple author entries. Order references for the same authors by year of publication, with the earliest first. Make sure that each reference includes all relevant information (e.g., page numbers).

Please put some effort into making references complete, presentable, and consistent, e.g. use the actual current name of authors. If using bibtex, please protect capital letters of names and abbreviations in titles, for example, use {B}ayesian or {L}ipschitz in your .bib file.

Accessibility

Authors are kindly asked to make their submissions as accessible as possible for everyone including people with disabilities and sensory or neurological differences. Tips of how to achieve this and what to pay attention to will be provided on the conference website `http://icml.cc/`.

Software and Data

If a paper is accepted, we strongly encourage the publication of software and data with the camera-ready version of the paper whenever appropriate. This can be done by including a URL in the camera-ready copy. However, **do not** include URLs that reveal your institution or identity in your submission for review. Instead, provide an anonymous URL or upload the material as "Supplementary Material" into the

OpenReview reviewing system. Note that reviewers are not required to look at this material when writing their review.

Acknowledgements

Do not include acknowledgements in the initial version of the paper submitted for blind review.

If a paper is accepted, the final camera-ready version can (and usually should) include acknowledgements. Such acknowledgements should be placed at the end of the section, in an unnumbered section that does not count towards the paper page limit. Typically, this will include thanks to reviewers who gave useful comments, to colleagues who contributed to the ideas, and to funding agencies and corporate sponsors that provided financial support.

Impact Statement

Authors are **required** to include a statement of the potential broader impact of their work, including its ethical aspects and future societal consequences. This statement should be in an unnumbered section at the end of the paper (co-located with Acknowledgements – the two may appear in either order, but both must be before References), and does not count toward the paper page limit. In many cases, where the ethical impacts and expected societal implications are those that are well established when advancing the field of Machine Learning, substantial discussion is not required, and a simple statement such as the following will suffice:

"This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here."

The above statement can be used verbatim in such cases, but we encourage authors to think about whether there is content which does warrant further discussion, as this statement will be apparent if the paper is later flagged for ethics review.

References

Duda, R. O., Hart, P. E., and Stork, D. G. *Pattern Classification*. John Wiley and Sons, 2nd edition, 2000.

Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Michalski, R. S., Carbonell, J. G., and Mitchell, T. M. (eds.). *Machine Learning: An Artificial Intelligence Approach, Vol. I.* Tioga, Palo Alto, CA, 1983.

Newell, A. and Rosenbloom, P. S. Mechanisms of skill acquisition and the law of practice. In Anderson, J. R. (ed.), *Cognitive Skills and Their Acquisition*, chapter 1, pp. 1–51. Lawrence Erlbaum Associates, Inc., Hillsdale, NJ, 1981.

Samuel, A. L. Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 3(3):211–229, 1959.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

A. You *can* have an appendix here.

You can have as much text here as you want. The main body must be at most 8 pages long. For the final version, one more page can be added. If you want, you can use an appendix like this one.

The `\onecolumn` command above can be kept in place if you prefer a one-column appendix, or can be removed if you prefer a two-column appendix. Apart from this possible change, the style (font size, spacing, margins, page numbering, etc.) should be kept the same as the main body.