No Need To Talk 🤐: Asynchronous Mixture of Language Models

Anastasiia Filippova, Angelos Katharopoulos, David Grangier, Ronan Collobert
ICLR 2025 · Apple

Motivation

Training & inference of LLMs requires high-bandwidth 💸 & low-latency interconnect 💰

But what if we don't have access to state-of-the-art interconnect hardware?

Prior works to reduce communication cost:

- require some form of synchronization [1, 2]
- relay on full corpus clustering [3]
- often under-perform or does not scale [4, 5]

How can we achieve efficient training and inference **without**

- relying on fast interconnect
- compromising model performance?

Background

Asynchronous training & sparse inference with Mixture of Experts [6]

$$\mathscr{L}(\mathbf{x};\theta) = -\sum_{s=1}^{S-1}\log\sum_{e=1}^{E} p(x_{s+1} \mid \mathbf{x}_{1:s}, e\,;\,\theta^e)\,p(e \mid \mathbf{x}_{1:s}\,;\,\theta^r),$$

Can be sparse by using hard assignments:

$$\mathscr{L}(\mathbf{x};\theta) \leq -\sum_{s=1}^{S-1}\log p(x_{s+1} \mid \mathbf{x}_{1:s}, e^\star\,;\,\theta^{e^\star})\,p(e^\star \mid \mathbf{x}_{1:s}\,;\,\theta^r),$$

Compared to Switch Transformer MoE [7]

Routing is done per token ⇒

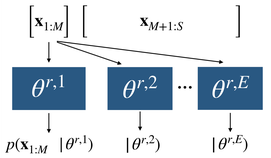
- all experts **must be retained in RAM** during both training and inference
- **high communication overhead** is required to synchronise gradients

CITATIONSSS

SmallTalk LM

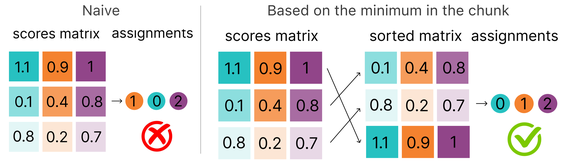
Routing With Independent Language Models

Router — independent small LLM per expert



$$\begin{aligned} e^\star &= \arg max_e p(e \mid \mathbf{x}_{1:s}\,;\,\theta^r) \\ &= \arg max_e \frac{p(\mathbf{x}_{1:s} \mid e, \theta^r)\,p(e)}{\sum_{i=1}^{E} p(\mathbf{x}_{1:s} \mid i\,;\,\theta^r)\,p(i)} \\ &= \arg max_e p(\mathbf{x}_{1:s} \mid \theta^{r,e}) \\ &\approx \arg max_e p(\mathbf{x}_{1:M} \mid \theta^{r,e}) \quad (1) \end{aligned}$$

Assignments balancing



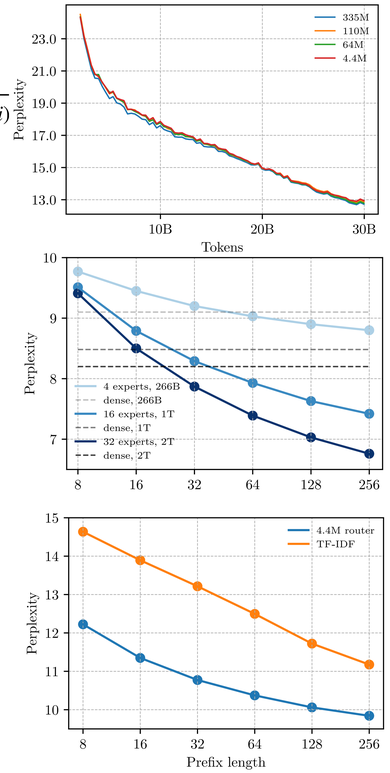
Training Procedure

1) Train the routers with EM **without using the experts at all,** alternating between log-likelihood optimization & assignments
2) Train the experts as E **independent** lms:

Train the routers
$\mathbf{X} \leftarrow N$ new sequences from the dataset
$\mathbf{X}_{1:E} = \text{random\_assignments}(\mathbf{X})$
for $i = 1 \ldots T$ **do**
  for $e = 1 \ldots E$ **do**
    $\theta^{r,e} \approx \arg\min_{\theta^{r,e}} \mathcal{L}(\mathbf{X}_e; \theta^{r,e})$
  end for
  $\mathbf{X} \leftarrow N$ new sequences from the dataset
  $\mathbf{X}_{1:E} = \text{balanced\_assignments}(\mathbf{X}, \theta^r)$
end for

Train the experts
$\mathbf{X} \leftarrow M$ new sequences from the dataset
$\mathbf{X}_{1:E} = \text{balanced\_assignments}(\mathbf{X}, \theta^r)$
for $e = 1 \ldots E$ **do**
  $\theta^e \approx \arg\min_{\theta^e} \mathcal{L}(\mathbf{X}_e; \theta^e)$
end for

Computational cost & Routing Analysis

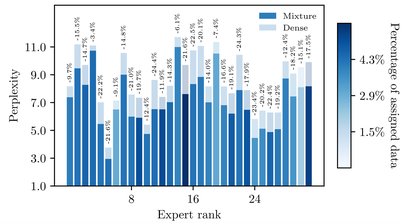


(1) The size of the router **does not impact** the performance ⇒ reduce computational cost of routing with only **4.4M** params routers!

(2) With 32 tokens prefix length **SmallTalk LM** still outperforms dense baseline!

(3) Simpler methods such as TF-IDF encoding + K-Means under performs compared to routing with 4.4M language model.

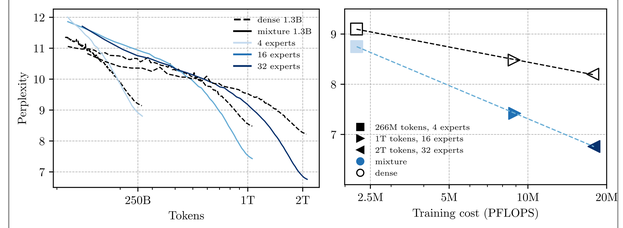
Results



Experts do specialize and all contribute to the overall improvement instead of only a few dominant experts.

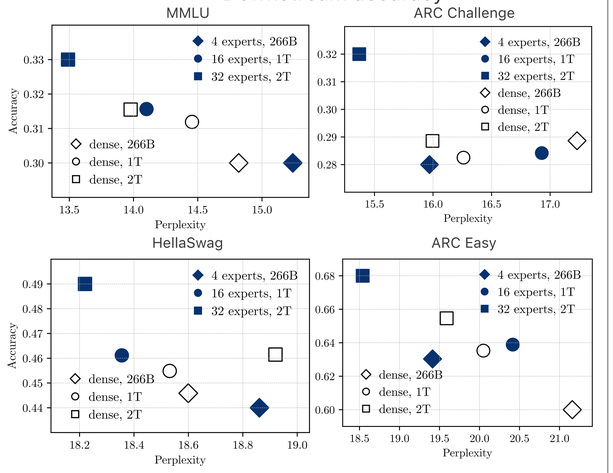
Results

Training FLOPS / tokens vs Perplexity



Significant performance boost with near-identical FLOPS, identical data volume and **no reliance on fast interconnect** 🎉!

Downstream accuracy



Conclusion: No Need To Talk 🤐

Compared to dense baseline **SmallTalk LM**:

- **reduces bandwidth** requirement to zero
- archives significantly **lower perplexity** with near-identical training and inference FLOPs and identical training data volume
- achieves **better accuracy** on downstream tasks