

No Need to Talk: Asynchronous Mixture of Language Models

Anastasiia Filippova [†]
EPFL

Angelos Katharopoulos
Apple

David Grangier
Apple

Ronan Collobert
Apple

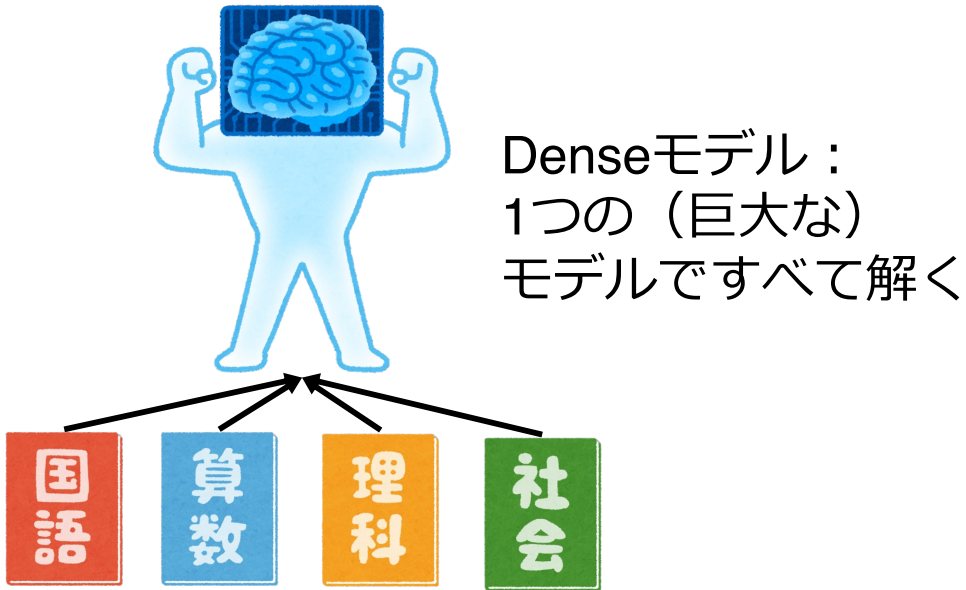
読む人：清野 舜 (SB Intuitions)

Mixture of Experts (MoE) とは何か

何か？

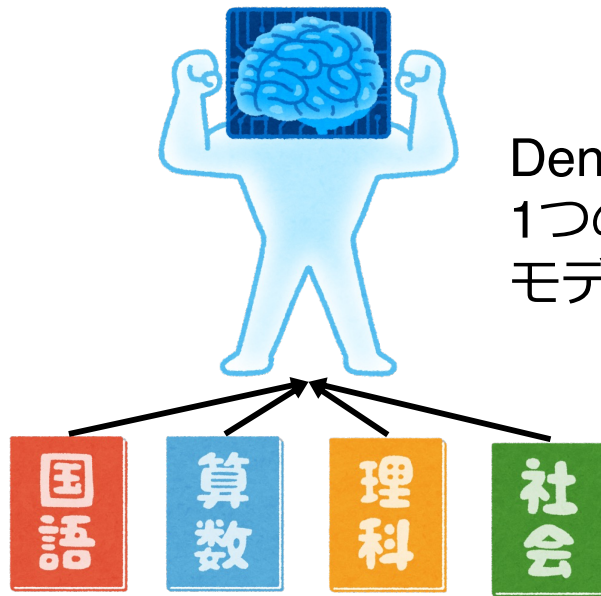
Mixture-of-Experts (MoE) とは (1/2)

- 通常：任意のタスクを1つのモデルで解く
 - MoE の文脈では Dense と呼ばれる

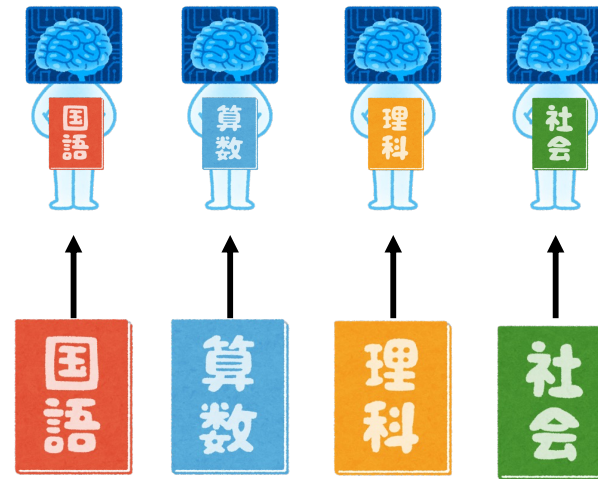


Mixture-of-Experts (MoE) とは (2/2)

- MoE : 各タスクを専属のエキスパートに解かせる
 - 各エキスパートのパラメータ数を減らしても表現力が維持可能
 - 各タスクについての計算量が減らせる



Denseモデル：
1つの（巨大な）
モデルですべて解く

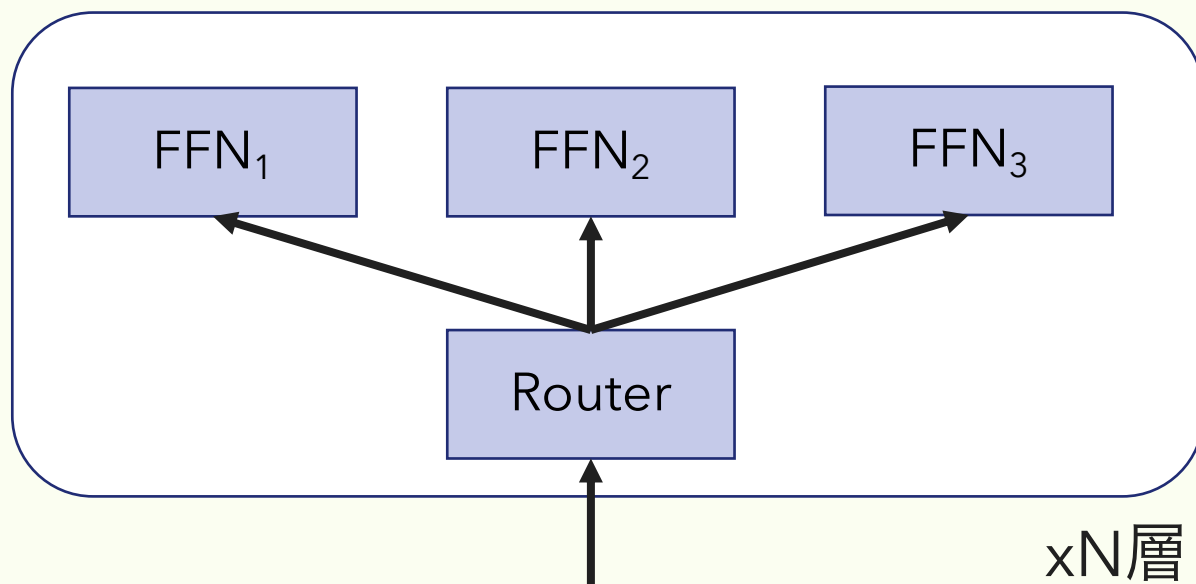


MoE :
各エキスパートに
タスクを割り当て

世は大MoE祭り

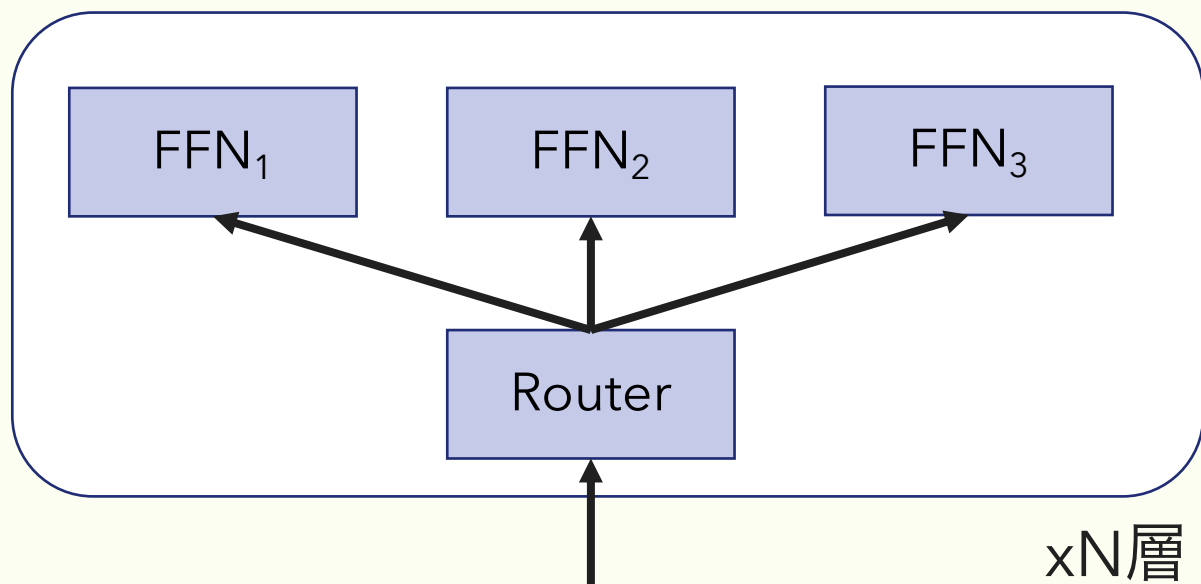
- 最先端のモデルが採用
 - DeepSeek-V3
 - Llama4
 - gpt-oss
- 国内最大規模のモデルも採用
 - Sarashina2-8x70B
- なぜ流行？
 - 同じ推論コストのDenseモデルよりも高い性能だから？
 - 発表者が本当には分かってない...
 - 「性能」のために必要なコストが大きすぎるのでは？という疑問

TransformerにおけるMoE



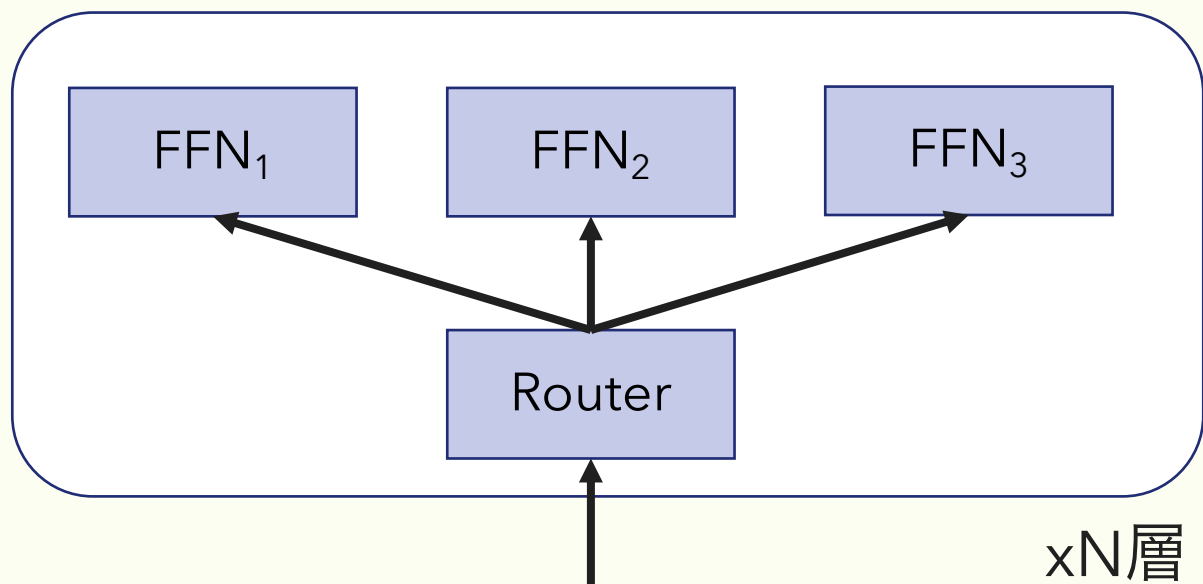
- FFN が Expertに相当
- 各時刻・各層で通信を行う

何が問題か？



- 通信が重い
 - 高速なGPU間通信が必要
 - それは高価
- 大量のメモリを消費する
 - Expertを展開しておく必要
 - 大量のGPUが必要
 - それは高価

何が問題か？

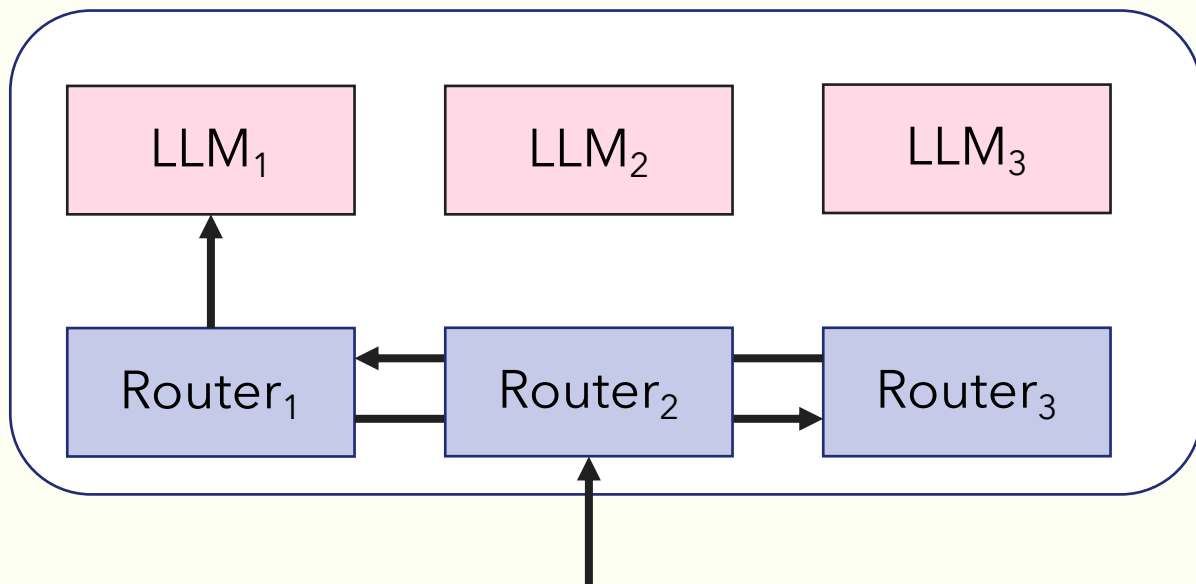


主にこちらを解決

- 通信が重い
 - 高速なGPU間通信が必要
 - それは高価
- 大量のメモリを消費する
 - Expertを展開しておく必要
 - 大量のGPUが必要
 - それは高価

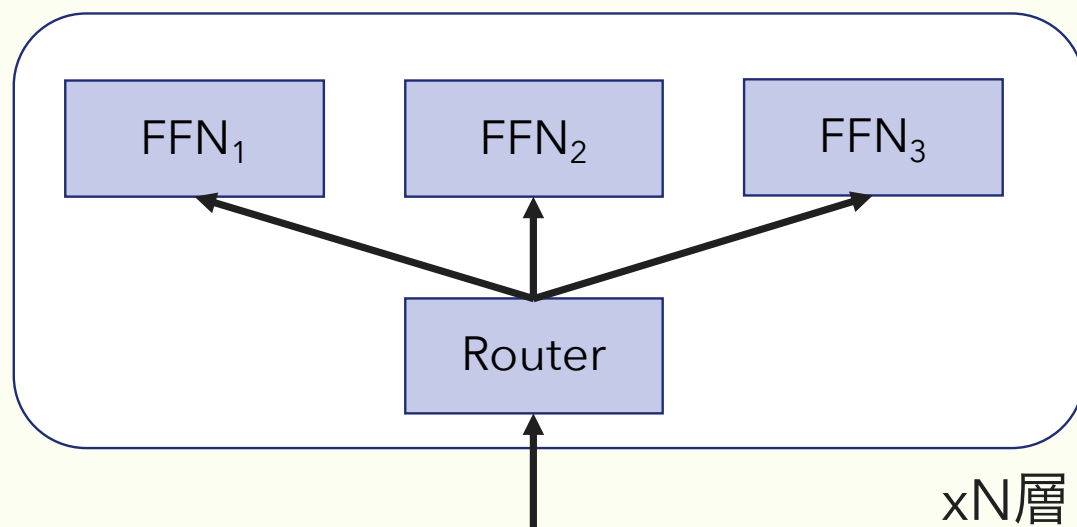
こちらは解決しない？

アイデア：LLMをExpertとみなす

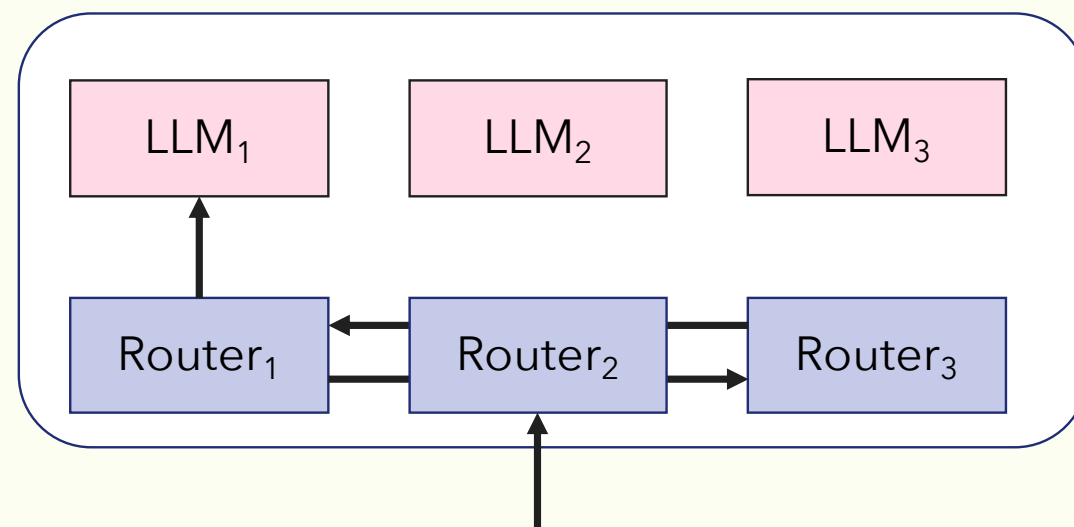


- LLM をExpertとして扱う
- LLM と同じ数のルータを用意する
 - 今回はルータ自体も言語モデル
- 最もスコアの高いルータに対応するLLMを選ぶ
- 選んだLLMで推論
 - Expert 間の通信は必要ない
 - No Need to Talk の伏線回収

既存のMoE vs 提案手法



- FFN が Expert に相当
- 各時刻・各層で通信を行う

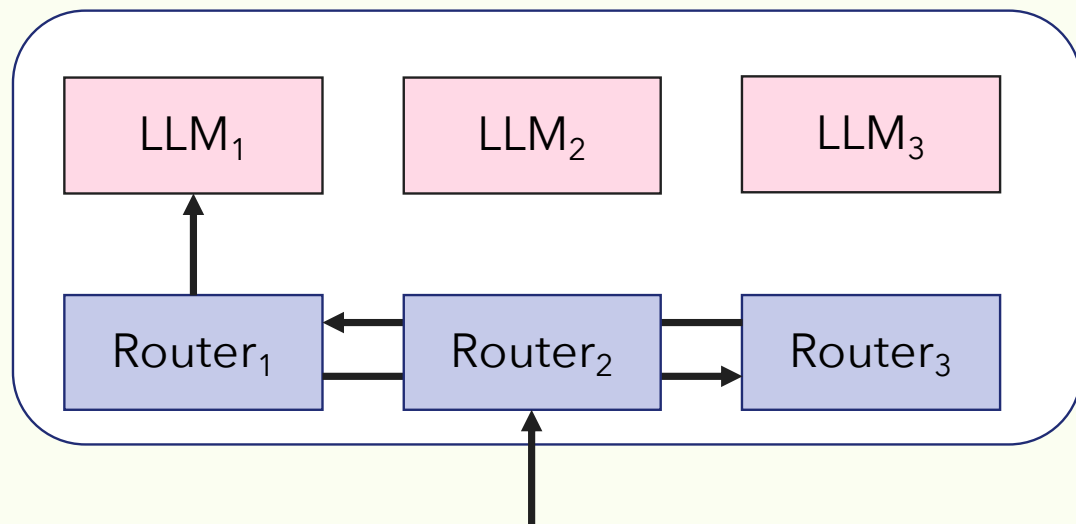


- LLM を Expert として扱う
- 選んだ LLM で一気通貫に推論

高速化のための Router のデザイン

LLM₁

と(ほぼ)同じ推論時間に向けて...



① ルータを軽量にしておく

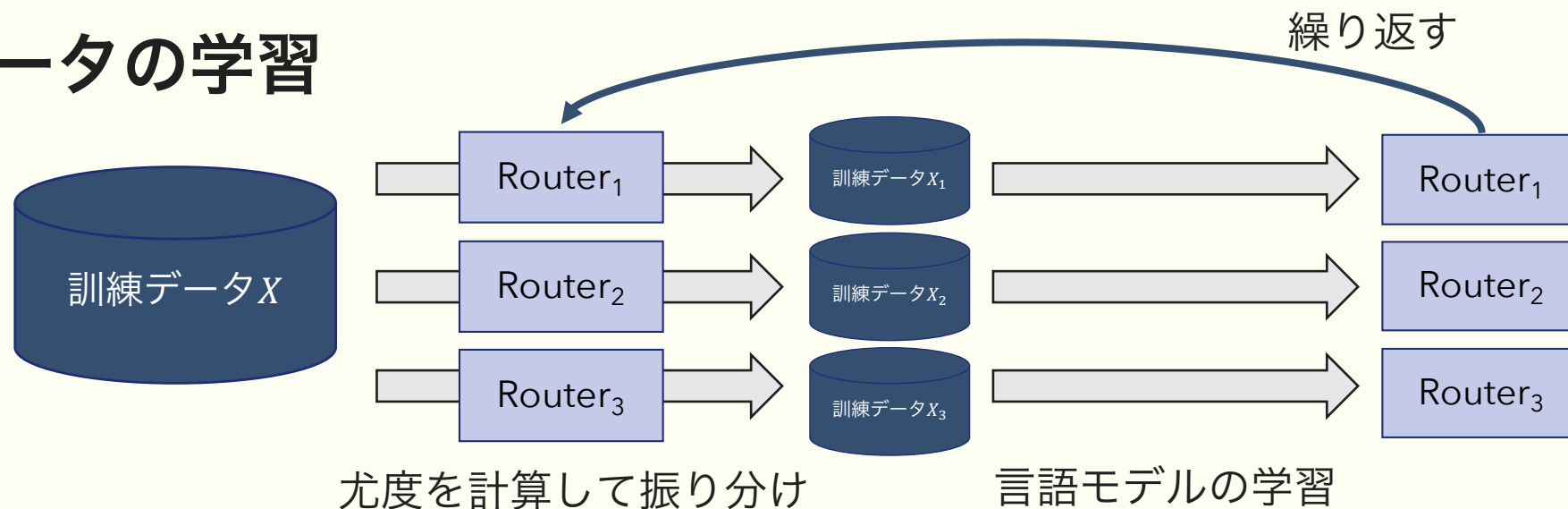
- 計算コストを小さくするため
- 今回は4.4Mパラメタの言語モデル

② 入力系列の接頭辞を使って振り分け

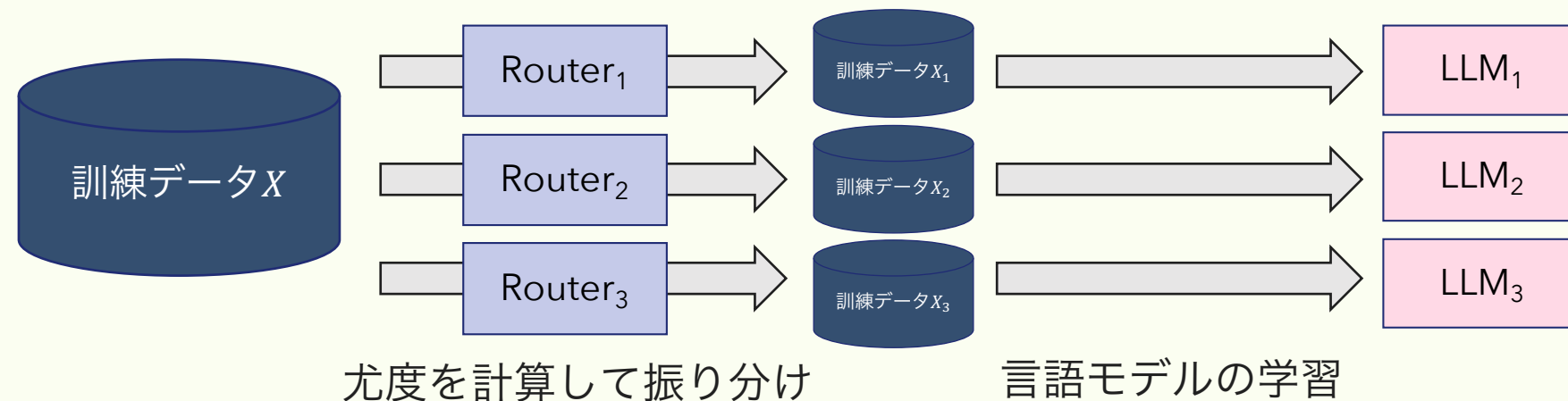
- 入力系列の全てを振り分けに使う必要はない
- 先頭32トークン程度でも十分な性能が出る
- トークン数を増やした実験は後述

ルータ → LLMの順番に学習

① ルータの学習



② Expert (LLM) の学習

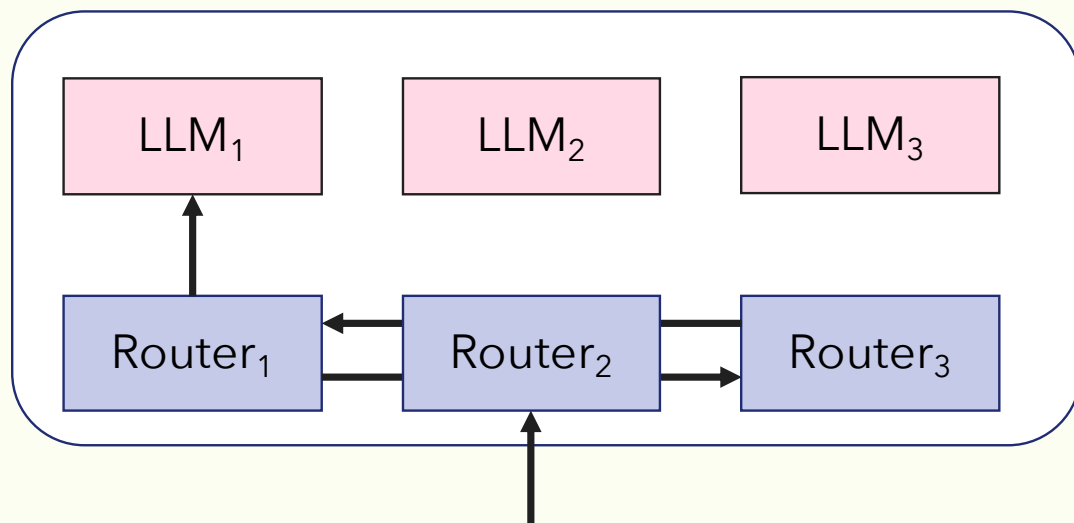


実験：Denseモデルとの比較

ベースライン：Denseモデル

LLM

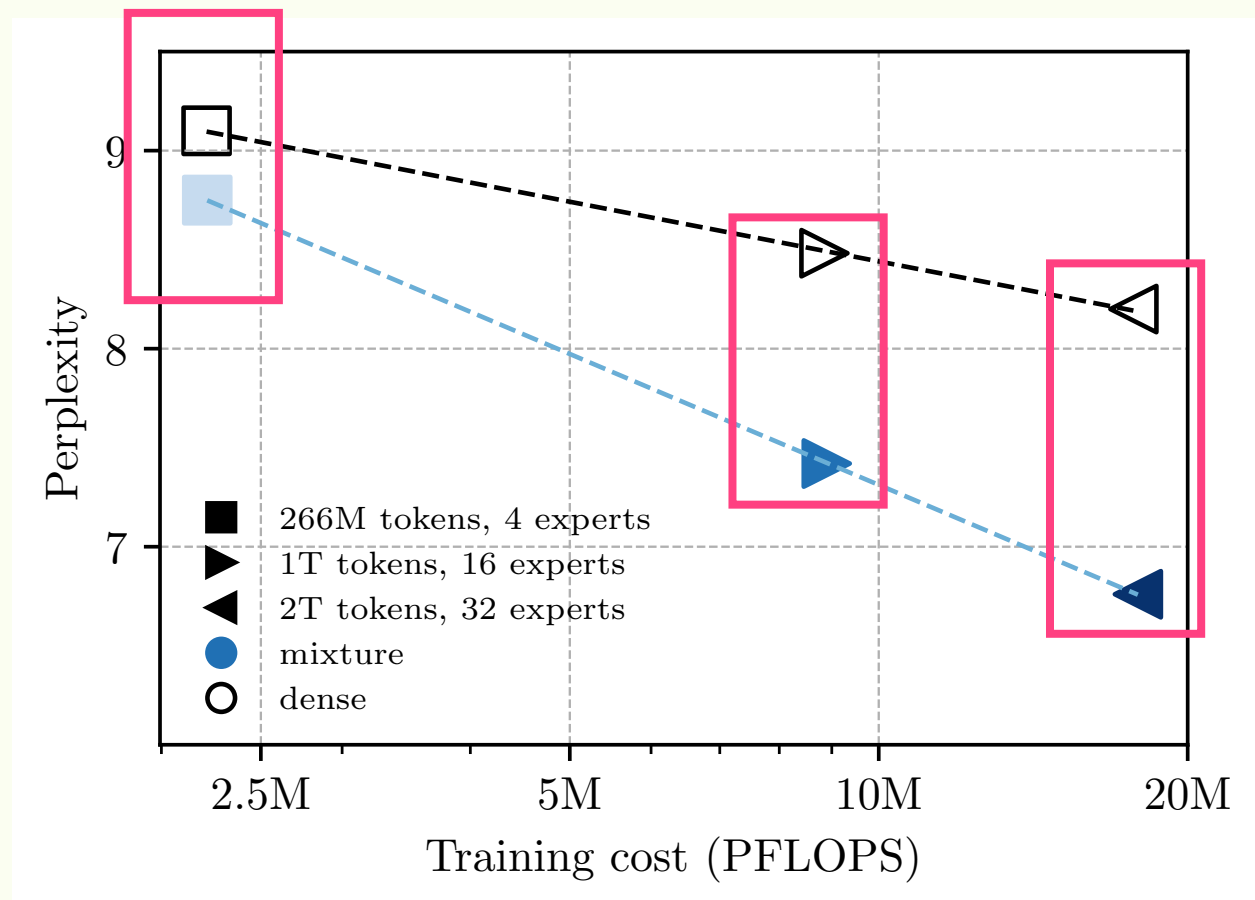
提案手法



LLM のパラメータ数は同じ
→推論にかかる時間を揃えて比較

事前学習で性能が改善した

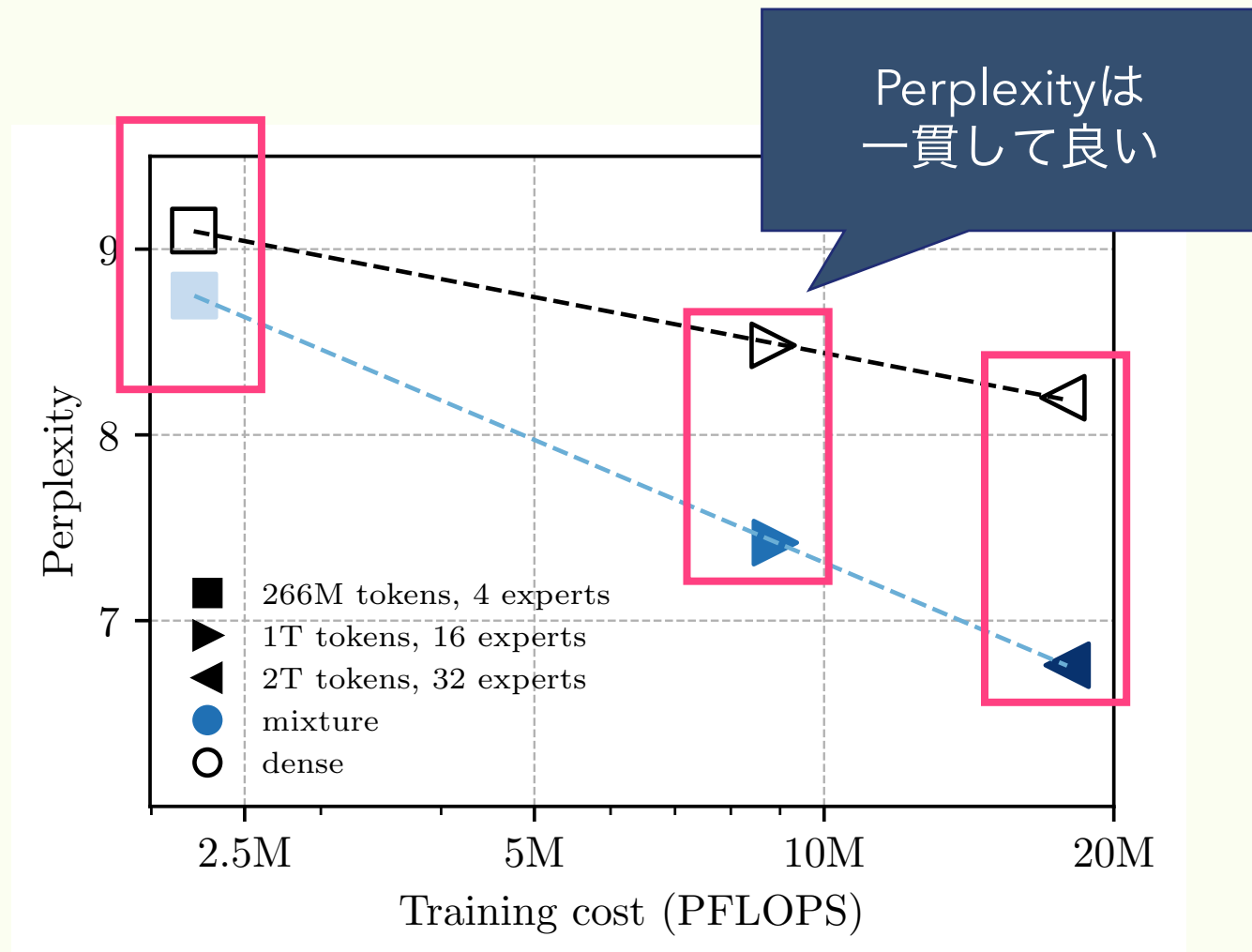
性能



学習コスト

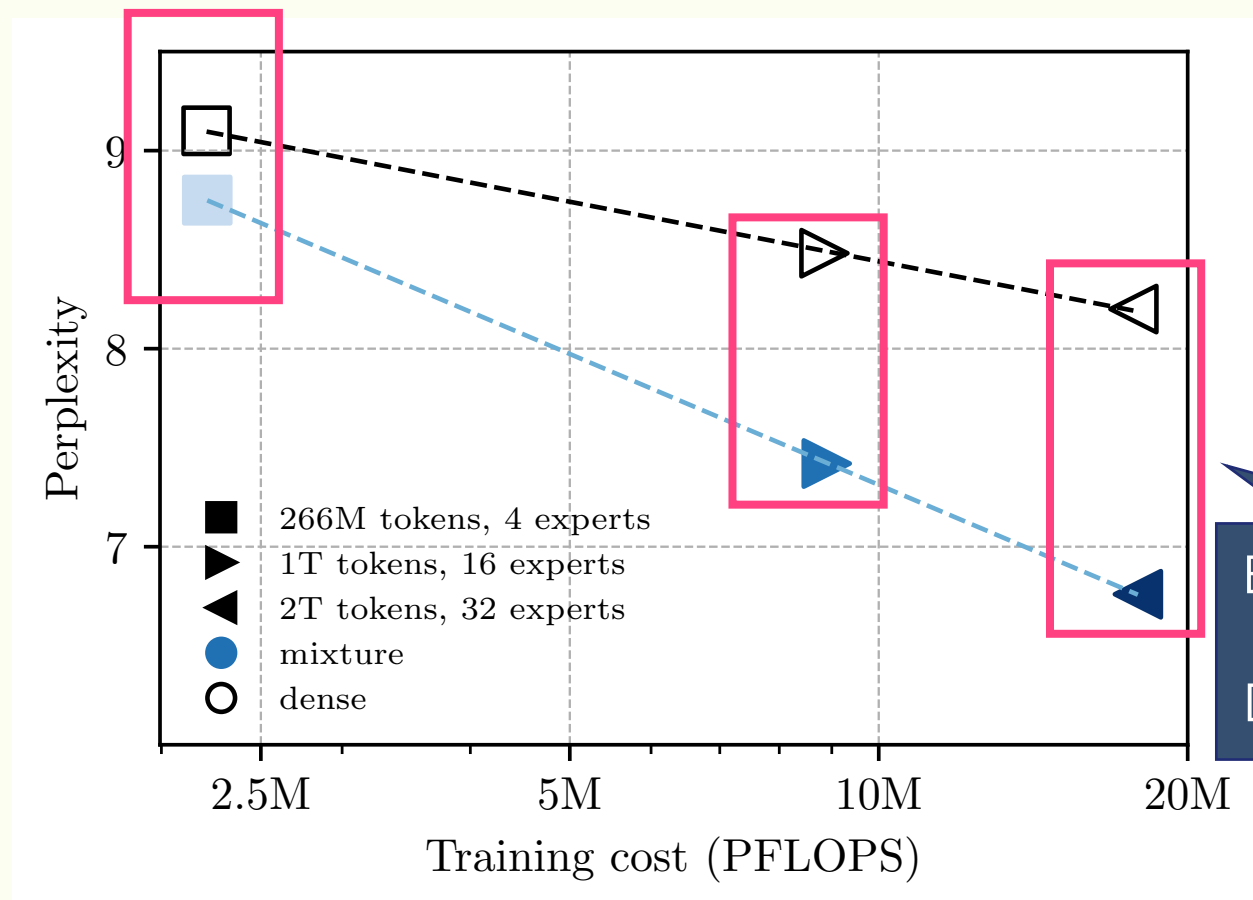
下流タスクでの評価結果は割愛

事前学習で性能が改善した



下流タスクでの評価結果は割愛

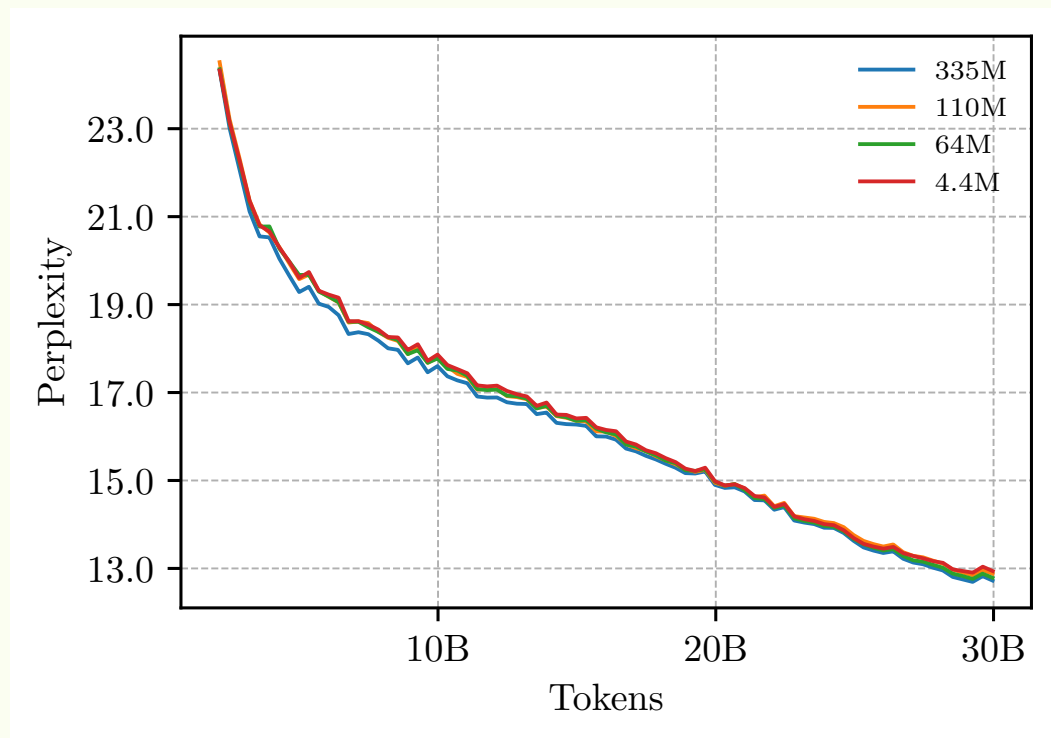
事前学習で性能が改善した



Expertを増やすほど
良さそうだが
Denseに不利な設定

下流タスクでの評価結果は割愛

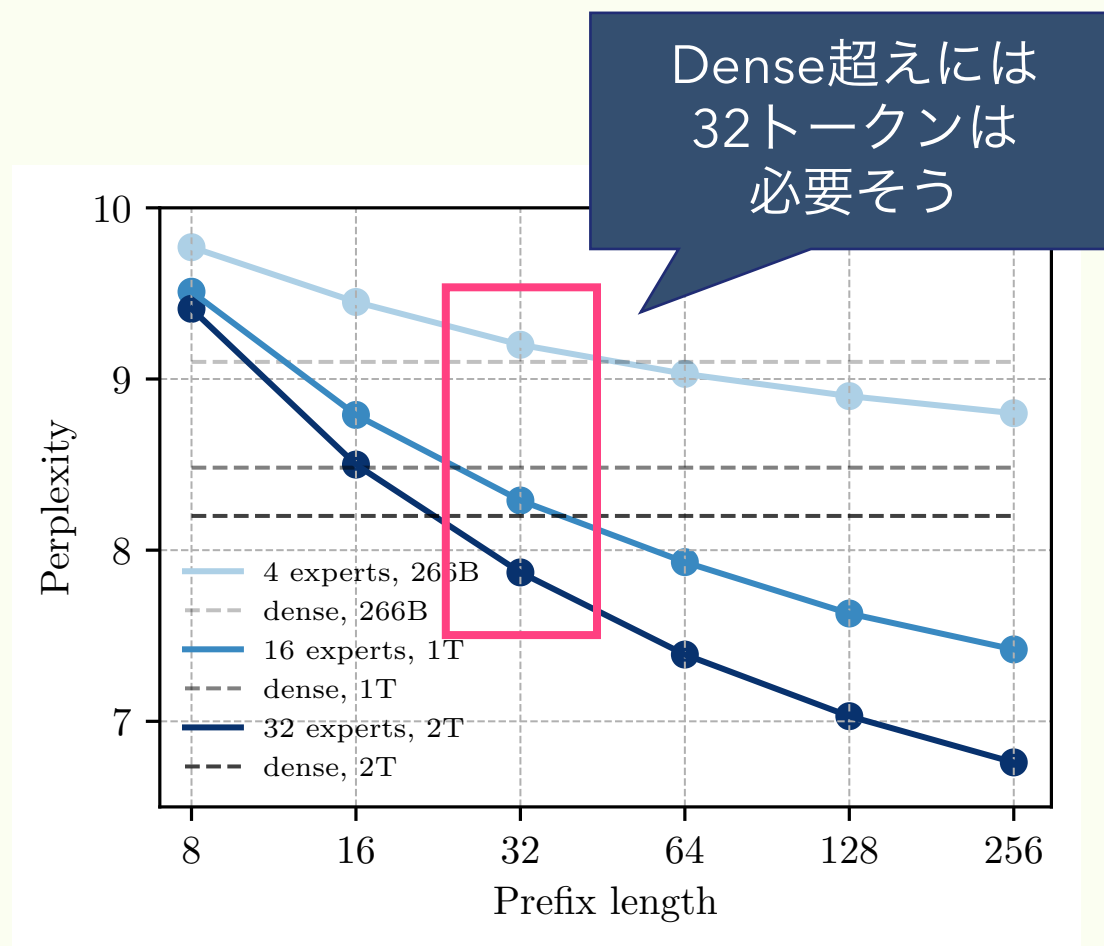
ルータの大きさについて



- ルータ：4.4MパラメタのLM
- 直感：大きなルータにすると、より良い振り分けができそう
- 結果は変わらず
 - 変わらなすぎて実験を疑いたくなる

振り分けに使う接頭辞のトークン数

振り分けに使う接頭辞のトークン数を変えてみた



32トークンってどれくらい？

Tokens	Characters
34	153

You can use the tool below to understand how a piece of text might be tokenized by a language model, and the total count of tokens in that piece of text.

Text

Token IDs

まとめ

- MoEの計算
- LLMをExpertとみなしたMoEを提案
 - 通信コストが削減できる
 - 高価なGPUに頼らないで済む
- ルータには小型LMを使う
 - 小型LMの尤度で入力系列をスコアリングし、最も確率の高いものを選択
- 事前学習タスクでDenseよりも高い性能

感想

- 実験が事前学習に閉じており、事後学習への適用可能性に疑問
 - SFT/DPOで振り分けが破綻しそう
 - ICLRのレビューでも指摘されており、著者も認めている
 - ICLRは気合で押し切った
 - 事前学習と事後学習を一気通貫にやりたい気がする...
 - 本勉強会でそういう論文が紹介される気がしている
- エキスパート（LLM）の数を自由に増減させられると嬉しそう
 - 増やす：ドメイン適用（金融、医療、etc...）
 - 減らす：推論時のコスト削減