Analyzing and Optimizing LayerSkip Models

AILA Project Proposal

Motivation

What is LayerSkip?

LayerSkip is an approach that finetunes existing large language models (LLMs) with a specific recipe that consists of early exit loss and layer dropout. These models are more robust to prediction at earlier layers and to skipping intermediate layers.

For inference, the LayerSkip paper proposed a self-speculative solution: generate tokens sequentially using a subset of layers, and verify tokens in parallel using remaining layers. However, we believe that with the LayerSkip checkpoints, novel inference solutions could be explored. Hence, this AILA proposal encourages students and researchers to use those checkpoints to analyze or develop different or novel inference approaches.

Llama checkpoints finetuned with this approach should be open sourced soon at the links below:

- Paper
- Model Checkpoints
- Inference Code

Who cares? If you are successful, what difference will it make?

What sets these LayerSkip checkpoints apart is their robustness to exiting at earlier layers or to skipping intermediate layers and the uniformity of activations across layers. These unique features pave the way for innovative research in optimization and interpretability. Hence, a lot of research ideas could be applied on those models to either make them more efficient, or to perform interpretability analysis to understand the nature of transformers and LLMs.

Ideas

This proposal does not present a single idea but rather presents a list of possible ideas that can utilize the LayerSkip Llama checkpoints.

For each idea below, we provided papers that implemented similar ideas to either non-Llama models, or to base Llama models that haven't been finetuned with the LayerSkip recipe. Applying those ideas to the LayerSkip checkpoints could lead to higher speedups, or could pave the way to developing novel ideas.

- **Efficiency Ideas**
  - **Dynamic Early Exit:** when decoding each sample or each token, determine which layer to exit at.
    - *Papers to Read*:
      - Depth-Adaptive Transformer, ICLR 2020
      - Confident Adaptive Language Modeling, NeurIPS 2022
  - **Dynamic Layer Skipping:** when decoding each sample or each token, skip layers randomly, or determine which layers to skip.
    - *Papers to Read:*
      - SkipNet, ECCV 2018
      - Draft & Verify, ACL 2024
      - LayerDrop, ICLR 2020
  - **Layer Pruning:** remove layers and perform some finetuning to get more accurate, smaller models.
    - *Papers to Read*:
      - LayerDrop, ICLR 2020
      - The Unreasonable Ineffectiveness of the Deeper Layers, 2024
      - ShortGPT, 2024
      - LLM-Pruner, NeurIPS 2023
  - **Novel Speculative Decoding:** speculative decoding (a.k.a assisted generation) is a set of approaches that speed up LLM inference by using a quick but inaccurate draft stage to generates tokens sequentially, and a more accurate but slower verification/correction stage that verifies the tokens in parallel. In the LayerSkip paper, the draft stage was to exit the original model at a specific early layer. You may consider different ideas to improve open that:
    - **Early Exit Schedule:** each decoding step exits at a different layer (similar to HuggingFace's schedule for number of draft tokens)
    - **Skip Intermediate Layers for Draft Stage:** this idea was already explored for Llama base models in Draft & Verify. Since the LayerSkip checkpoints were finetuned with layer dropout, perhaps applying the Draft & Verify approach on top of the LayerSkip checkpoints will lead to higher speedup. You may also consider skipping layers randomly during the draft stage.
    - *Papers to Read*:
      - Introduction to Speculative Decoding, 2024
      - HuggingFace Tutorial, 2023
      - Draft & Verify, ACL 2024
      - LayerSkip, ACL 2024
      - SmartSpec, 2024

- **Interpretability Research Ideas**
    - **Compare Per-Layer Analysis on LayerSkip and Base Model:** you can use LLM Transparency Tool or TransformerLens to compare the evolution of predictions across layers between a Llama model and its LayerSkip counterpart. Understanding the evolution of predictions, the norm of embeddings, and the behavior of attention heads across different models and models trained with and without early exit loss and dropout could provide some interesting insights. You can choose any of the papers below, and compare their insights with the insights you obtain when analyzing the LayerSkip checkpoints:
        - *Papers to Read*:
            - LLM Transparency Tool, ACL 2024
            - Information Flow Routes: Automatically Interpreting Language Models at Scale, 2024
            - Neurons in Large Language Models: Dead, N-gram, Positional, 2023

Helpful Codebases

- For evaluating accuracy: LLM Evaluation Harness for natural language tasks and BigCode Evaluation Harness for coding tasks
- For finetuning: TRL or TorchTune

Compute Requirements

- For efficient inference approaches like **Dynamic Early Exit**, **Dynamic Layer Skipping**, **Novel Speculative Decoding**, 1 V100 GPU model may be sufficient. Faster A100 GPU, or the availability of more GPUs will enable running multiple inference and evaluation experiments in parallel, and hence speed up iteration on different ideas.
- For **Layer Pruning**, if you want to finetune the model you will probably need more GPUs. Finetuning approaches like LoRA or QLoRA may enable you to finetune with 1 GPU. But more GPUs will make finetuning faster.
    - You may consider choosing a special domain dataset, (e.g., one of the datasets in Table 1 of this paper), finetune on its training set and evaluate on its validation set. This way, the pruned finetuned model may not perform well on arbitrary prompts, but could enable you to obtain decent accuracies on the validation set with low a training budget.
- For **Interpretability Research Ideas**, one GPU may be enough.

Contact

You can email the first author of LayerSkip: Mostafa Elhoushi, melhoushi@meta.com