Franken-LLAMA: Experimenting with LLama2 surgery to make it more efficient

Project Work in APAI

Presented by Angelo Galavotti

Università di Bologna, December 2024

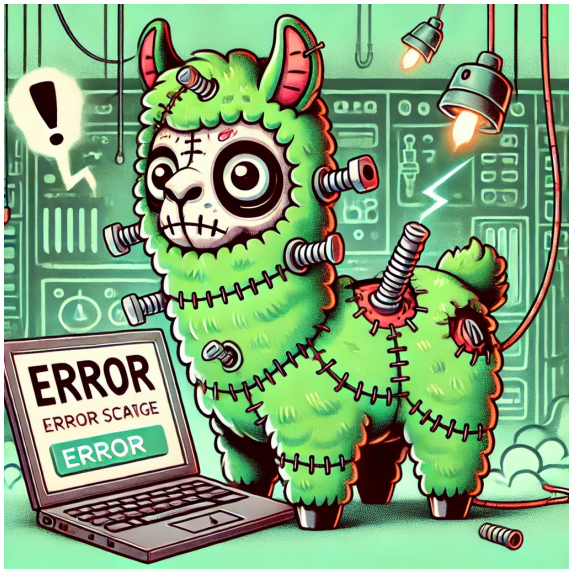
Quick overview on LLaMA 2

LLaMA 2 (Large Language Model Meta AI version 2) is an **open-source** family of foundational language models created by Meta.

- Available in **7B, 13B, and 70B** (billion parameters) versions.
- Performance is comparable to OpenAI's ChatGPT 3.5.
- Comes in a **base version** and **chat version**, fine-tuned with **RLHF** (Reinforcement Learning with Human Feedback) for conversational use.

Franken-LLAMA

Franken-LLAMA consists in optimizing LLaMA 2 (7B Chat) by reducing computational and memory costs through **layer skipping** and **repetition**, effectively performing "surgery" on the model.



Configurations

A total of **25 configurations** of skipped and repeated layers were tested by completing the phrase "Once upon a time" using a maximum of 50 tokens. Most of them produced gibberish text.

Configurations

After careful examination, only **6 configurations** were ultimately chosen for more thorough testing,

- `'0-23_27-31'`, skips layers from 24 to 26
- `'15_single_skip'`, skips only layer 15
- `'mid_expansion_with_repeats'`, skips layers [6, 7, 8, 9, 25, 26, 27, 28], repeats layers from 14 to 19 twice
- `'2_3rd_of_llama'`, skips layers from 11 to 20
- `'2_3rds_plus_llama'`, skips only odd indexed layers from 11 to 20
- `'skip_near_end_keep_last_two'`, skips layer from 27 to 29

The least computationally costly configuration is `'2_3rd_of_llama'` ($\sim$ 6.2M MACs)

- even though it doesn't involve layer repetition, it is also the **lightest memory-wise**.
- however, it also produced the **lowest quality** outputs.

Experimental Setup

The 6 configuration tested on the **HellaSwag** dataset and were compared to a **baseline** (i.e. the full Llama1-7B-chat model).

- The HellaSwag dataset consists in a set of questions with 4 possible answers, and only one of them is the correct one. Each model has to predict the right answer. It allows to test the **logical capabilities** of a model.
- The models were evaluated on 50 samples of this dataset.
- Each model has also undergone a preliminary qualitative test by generating answers to 50 samples of the Natural Questions dataset.

Results on HellaSwag

Configuration	HellaSwag	Avg. exec. time
`baseline`	0.34	91.1 s
`0-23_27-31`	0.38	81.2 s
`15_single_skip`	0.38	95.8 s
`mid_expansion_with_repeats`	0.22	**68.8 s**
`2_3rd_of_llama`	0.26	95.4 s
`2_3rds_plus_llama`	0.30	102.7 s
`skip_near_end_keep_last_two`	**0.42**	79.0 s

Table: HellaSwag scores (i.e. percentage of correct answers)

In conclusion

1. Skipping and repeating layers can be a **viable solution** to make light-weight LLMs.
2. Generally, repetition can cause the **quality of the result** to **drop** significantly.
3. Best results are achieved when skipping layers in the **middle** of the feature extractor.

Future work:

- Testing with more samples/configurations.
- kV cache compression to reduce memory footprint even further.

Thank you!

You can reach out at angelo.galavotti@studio.unibo.it for more questions!

Appendix: output examples

Configuration	Generated text
`0-7`	Once upon a timezetempreasacondarichte?? trickster goddess pue moonkennecticut [..] Reserveikaiwitzter PetersburgovPortail [..]
`all_except_last_two`	Once upon a time year0 **stadt [..] Death it Yearwaltapk Progress R?f?rencePU. ??? [..]
`only_even_layers`	Once upon a time S??. R S I d d? S S S S S [..]
`0r2-14_17-23_25-31r2`	Once upon a time, in the midst of a busy schedulesomeone's attention was caught.?You the world and its of the, and and and [..]
`first_last_8r2`	Once upon a time in?ceycofortia-inaymskoe Bridge—Monlinaiticky'830 [..]
`15r3_23r3_31r3`	Once upon a timepus pri rosgemeingemeingemeinwach junigemeingemei [..]