HPML final project: Optimizing ESM, a protein large language model

Vinay Swamy

## Goal and Objectives

ESM is a protein large language model developed by [Meta](https://github.com/facebookresearch/esm) [1]. The protein embeddings generated by ESM are used in a variety of downstream tasks such as “protein-level” tasks such structure prediction, protein function prediction, and protein-protein interaction prediction [2], [3], [4]. ESM also has been used for modelling “proteome-level” tasks, such as bacterial genome modelling. In these settings, protein representations using a frozen ESM model are generated across the whole protein, aggregrated via pooling, and used as input to a downstream model. Directly finetuning ESM on the level of proteomes as not been reported before, and is a promising direction for future research. However, the existing implementation, though widely used, ESM has prohibitvely high memory and run-time requirement to be used for proteome-level tasks. The goal of this project is to optimize the existing implementation of ESM to improve the performance of the model, as well as make finetuning more accesible to the community. In particular, I’m interested in devleoping a framework for finetuning ESM directly on larger protein sets.

## Approach and Performance Optimization Techniques to be used

We propose the following optimizations:

* Implementation of FlashAttention to improve model performance and reduce memory usage [5]. FlashAttenion is an IO-aware implementation of the scaled dot-product attention mechanism in transformers
* Reduction of model memory footprint through quantization. Quantization invovles converting the model-weights to a lower precision such as int8 or int4
* Improved run time through TorchScript(s)
* Implementation of Quantization aware LoRA finetuning paradigm to the model for low-memory footprint finetuning [6] . Low Ranking Adaption, or LoRA, uses frozen model weights along trainable low rank weight matrices. This allows for a significant reduction in memory footprint for finetuning.
* Implementation of a framework for finetuning ESM on larger protein sets. In the setting where each observation is a set of proteins and a batch is multiple sets, iteratively run chunks of a batch through the model, so that a batch of size
* Optimization of dataloading and auxillary tasks to improve training speed

## Challenges

Optimizations like FlashAttention pose mainly implementation challenges, as it is fully equivalent to the standard scaled dot product attention. To ensure model fidelity after incorporating these changes, I will implement a series of rigorous tests to ensure that the model is still performing as expected. Other optimizations like TorchScript(tracing) and quantization and LoRA finetuning pose a more difficult challenge, as these changes can lead to large downstream changes in model performance. To ensure that the model maintains performance after these implementations,I propose two benchmarks: the TAPE protein function benchmark, and unchanged performance on in ESMfold [7] . To quantify the performance of the model on proteome level tasks, I will model drug sensitivties in cancer cell lines using the PRISM dataset (This task is one of the main parts of my thesis work) [8] .

## Implementation Details

The model will be implemented primarily in PyTorch. The main external libaries that will be used will be FlashAttention and LoRAlib, both available on github. The model will be trained on my lab’s compute cluster, which has multiple nodes with 8 x A6000 GPUs each with 48Gb of memory. I will leverage existing code for ESM and ESMfold. The TAPE benchmark and PRISM dataset are publicaly available.

## Demo

I plan to make the model fully available in a github repo, and will provide a google colab notebook on how to run and use the model.

## References

1. Rives A, Meier J, Sercu T, et al (2021) Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proceedings of the National Academy of Sciences 118(15):e2016239118. <https://doi.org/10.1073/pnas.2016239118>

2. Meier J, Rao R, Verkuil R, Liu J, Sercu T, Rives A (2021) [Language models enable zero-shot prediction of the effects of mutations on protein function](https://doi.org/10.1101/2021.07.09.450648)

3. Hwang Y, Cornman AL, Kellogg EH, Ovchinnikov S, Girguis PR (2023) [Genomic language model predicts protein co-regulation and function](https://doi.org/10.1101/2023.04.07.536042)

4. Lin P, Yan Y, Huang S-Y (2023) DeepHomo2.0: Improved protein-protein contact prediction of homodimers by transformer-enhanced deep learning. Brief Bioinform 24(1):bbac499. <https://doi.org/10.1093/bib/bbac499>

5. Dao T, Fu DY, Ermon S, Rudra A, Ré C (2022) [FlashAttention: Fast and memory-efficient exact attention with IO-awareness](http://arxiv.org/abs/2205.14135)

6. Hu EJ, Shen Y, Wallis P, et al (2021) [LoRA: Low-rank adaptation of large language models](http://arxiv.org/abs/2106.09685)

7. Rao R, Bhattacharya N, Thomas N, et al (2019) [Evaluating protein transfer learning with TAPE](http://arxiv.org/abs/1906.08230)

8. Corsello SM, Nagari RT, Spangler RD, et al (2020) Discovering the anti-cancer potential of non-oncology drugs by systematic viability profiling. Nat Cancer 1(2):235–248. <https://doi.org/10.1038/s43018-019-0018-6>