A.MODEL SUMMARY

A1. Background on my team

- Competition Name: LMSYS - Chatbot Arena Human Preference Predictions
- Team Name: **BlackPearl (no leak)**
- Private Leaderboard Score:0.96898
- Private Leaderboard Place: 1st

- team member

  Name: ZhouYang(Kaggle ID:**sayoulala**)

  Location:Shanghai,China

  Email:100972458@qq.com

A2. Background on my team

- Professional Background：MeiTuan LLM Algorithm Engineer
- Prior Experience:I have been continuously engaged in large model applications and have won 4 gold medals on Kaggle. I have extensive competition experience.
- What made me decide to enter this competition? Because this competition is very meaningful, addressing a challenging problem that requires breakthroughs. Additionally, it has good CV/LB (Cross-Validation/Leaderboard) performance, making it a valuable opportunity for growth and improvement.
- How much time did me spend on the competition? On and off for about a month.

A3. Solution Summary

My base models include Llama3 70b, Qwen2 72b, and Gemma2-9b, all leveraging the AutoModelForSequenceClassification architecture with LoRA and QLoRA techniques. Initial

post-pretraining involved one epoch on the UT dataset, followed by training on 5-fold splits and obtaining logits distributions (llama3,qwen2). The fine-tuning incorporated distillation loss using the 9b model with learning rates set at 5e-5. For model ensembling, the LoRA layers across 5 folds were averaged. We then quantized the models to 8-bit using GPTQ and applied TTA during the final submission. The most important part is using multiple models for distillation and performing multi-fold averaged LoRA fusion.

A4.Solution Detail

Dataset

In this competition, I mainly used three datasets: the official training set combined with the deduplicated 33k data, and the UT dataset for post-pretraining.

- Kaggle train data
- UT data (https://www.kaggle.com/competitions/lmsys-chatbot-arena/discussion/499756)
- 33k data (https://www.kaggle.com/competitions/lmsys-chatbot-arena/discussion/500973)

Base Models

- Llama3 70b (https://huggingface.co/meta-llama/Meta-Llama-3-70B-Instruct)
- Qwen2 72b （https://huggingface.co/Qwen/Qwen2-72B-Instruct）
- Gemma2-9b (https://huggingface.co/google/gemma-2-9b-it)

Base Model Architecture

- AutoModelForSequenceClassification
- LoRA (9b)
- QLoRA (Llama3 and Qwen2)
- All linear for LoRA
- Parameters: r=64, α=128
- Max length: 1024

- Epochs: 2
- Global batch size: 64
- device: A100*80G*8

Data Input

Follow this format during the fine-tuning process.

Swap the positions of answers A and B before training, so the training data is double the original quantity.

```
 prompt = f"""User question:
\"""{query}\"""
Answer A:
\"""{answer_a}\"""
Answer B:
\"""{answer_b}\"""
"""
```

Post-Pretrain

- To begin with, train one epoch on three models using the UT dataset (lr=1e-5).

Get the Logits Distribution

- Load the weight from post-pretrain, split the dataset into 5 folds for training (e.g., train on 4/5 Kaggle train data + 33k data, dev on 1/5 Kaggle train data) to train Llama3 70b and Qwen2 72b.
- Then infer the probability distribution of the training set.

Distill to the 9b Model with Logits

- After obtaining the logits distribution, load the 9b model for fine-tuning and incorporate the distillation loss during the fine-tuning process (at least three losses for training, lr=5e-5).

Model Ensemble

- Directly average the LoRA layers of the 5 folds.

Get 8-bit Model

- Quantize to 8-bit using GPTQ and use TTA (length 2000) during submission.

CV/LB Results

- (Here, I will only provide my final results. There were too many experiments before, but these results are the most important.)
  - Qwen72b: CV of 5 folds: 0.875, 0.881, 0.869, 0.880, 0.875
  - Llama3 70b: CV of 5 folds: 0.874, 0.877, 0.877, 0.873, 0.873
  - **Distilled Gemma 9b: CV of 5 folds: 0.862, 0.876, 0.858, 0.872, 0.868**
  - Merge LoRA and quantize to 8-bit: LB: 0.882 (With TTA: 0.876); final PB: 0.96898
  - (In the final submission, I had one sub that also failed to run because I deleted another model that I had uploaded.)

A5.Interesting findings

I noticed the following characteristics in this task:

1. The data is very noisy; even with a 70B model, the training loss can only be reduced to around 0.7x.
2. The larger the model, the better the performance. In high-noise environments, larger LLM parameters yield better results.
3. When distilling two models into a 9B model, it can achieve a similar effect to merging the two models. Should we consider trying to distill more models?

A6. Model Execution Time

In an environment with 8 NVIDIA A100 80GB GPUs, training one fold of the 70B model and then inferring the training set probabilities takes 1 day. Therefore, 5 folds would take 5 days. Post-pretraining on the UT dataset requires 1 day, amounting to a total of 12 days. Fine-tuning the Gemma2-9b model takes half a day, requiring an additional 2.5 days. Therefore, the entire process would take approximately 15 days, given an environment with 8 NVIDIA A100 80GB GPUs.