赛道二-作品报告

一、微调算法介绍

- 采用Lora算法
- 微调数据集
  - 规模：10w
  - 预处理方式：
    i. 使用大模型为所有 `train.json` 中的训练样本作难度打标，分为0，1，2，3，4，5六级，数值越大，难度越高。
    ii. 按难度1：难度2：难度3：难度4 = 4:3:2:1的比例从train.json数据集中分层抽取共10万个样本作为训练数据。
    iii. 使用正则表达式将所有训练数据的 `problem` 与 `solution` 中的数值部分均作**格式化处理：小数部分多于两位的数值，仅保留小数点后两位**；小数部分不多于两位的数值，保持不变。
    iv. 运行 `data_converter.py` 脚本，将问答对形式的json数据集转变成人机对话形式的json数据集。
    v. 运行 `llama_preprocess.py` 脚本将人机对话形式的json数据集转变成MindRecord格式的数据集。

二、微调超参配置

lora配置

- lora_rank: 8
- lora_alpha: 16
- lora_dropout: 0.0
- target_modules: '.*wq|.*wv'

```
seed: 0
output_dir: './output' # path to save checkpoint/strategy # last try: 730_float_formatted_10w_r8a16
load_checkpoint: '/home/ma-user/work/llama3-8B.ckpt'
src_strategy_path_or_dir: ''
```

```yaml
auto_trans_ckpt: False  # If true, auto transform load_checkpoint to load in distributed model
only_save_strategy: False
resume_training: False
run_mode: 'finetune'

# trainer config
trainer:
  type: CausalLanguageModelingTrainer
  model_name: 'llama3_8b'

# runner config
runner_config:
  epochs: 3
  batch_size: 32
  sink_mode: True
  sink_size: 2

# optimizer
optimizer:
  type: FP32StateAdamWeightDecay
  beta1: 0.9
  beta2: 0.95
  eps: 1.e-8

# lr sechdule
lr_schedule:
  type: CosineWithWarmUpLR
  learning_rate: 1.e-5
  lr_end: 0.0
  warmup_ratio: 0.03
  total_steps: -1 # -1 means it will load the total steps of the dataset

# dataset
train_dataset: &train_dataset
  data_loader:
    type: MindDataset
    dataset_dir: "/home/ma-user/work/train-fastchat256_ranked.mindrecord"
    shuffle: True
  input_columns: ["input_ids", "labels"]  # "input_ids", "labels" , labels are used in instruction finetune.
  num_parallel_workers: 8
  python_multiprocessing: False
  drop_remainder: True
  batch_size: 32
  repeat: 1
  numa_enable: False
```

```yaml
  prefetch_size: 1
train_dataset_task:
  type: CausalLanguageModelDataset
  dataset_config: *train_dataset
# if True, do evaluate during the training process. if false, do nothing.
# note that the task trainer should support _evaluate_in_training
function.
do_eval: False

# eval dataset
eval_dataset: &eval_dataset
  data_loader:
    type: MindDataset
    dataset_dir: ""
    shuffle: False
  input_columns: ["input_ids"]
  num_parallel_workers: 8
  python_multiprocessing: False
  drop_remainder: False
  repeat: 1
  numa_enable: False
  prefetch_size: 1
eval_dataset_task:
  type: CausalLanguageModelDataset
  dataset_config: *eval_dataset

use_parallel: True
# parallel context config
parallel:
  parallel_mode: 1 # 0-data parallel, 1-semi-auto parallel, 2-auto
parallel, 3-hybrid parallel
  gradients_mean: False
  enable_alltoall: False
  full_batch: True
  search_mode: "sharding_propagation"
  enable_parallel_optimizer: True
  strategy_ckpt_config:
    save_file: "./ckpt_strategy.ckpt"
    only_trainable_params: False
  parallel_optimizer_config:
    gradient_accumulation_shard: False
    parallel_optimizer_threshold: 64
# default parallel of device num = 8 for Atlas 800T A2
parallel_config:
  data_parallel: 1
  model_parallel: 4
  pipeline_stage: 1
```

```yaml
  use_seq_parallel: False
  micro_batch_num: 1
  vocab_emb_dp: True
  gradient_aggregation_group: 4
# when model parallel is greater than 1, we can set micro_batch_interleave_num=2, that may accelerate the train process.
micro_batch_interleave_num: 1

# recompute config
recompute_config:
  recompute: True
  select_recompute: False
  parallel_optimizer_comm_recompute: False
  mp_comm_recompute: True
  recompute_slice_activation: True

# callbacks
callbacks:
  - type: MFLossMonitor
  - type: CheckpointMointor
    prefix: "llama3_8b"
    save_checkpoint_steps: 1400
    integrated_save: False
    async_save: False
  - type: ObsMonitor

# mindspore context init config
context:
  mode: 0 #0--Graph Mode; 1--Pynative Mode
  device_target: "Ascend"
  enable_graph_kernel: False
  graph_kernel_flags: "--disable_expand_ops=Softmax,Dropout --enable_parallel_fusion=true --reduce_fuse_depth=8 --enable_auto_tensor_inplace=true"
  max_call_depth: 10000
  max_device_memory: "26GB"
  save_graphs: False
  save_graphs_path: "./graph"
  device_id: 0
  runtime_num_threads: 1

# model config
model:
  model_config:
    type: LlamaConfig
    batch_size: 32 # add for increase predict
    seq_length: 256
```

```yaml
    hidden_size: 4096
    num_layers: 32
    num_heads: 32
    n_kv_heads: 8
    vocab_size: 128256
    intermediate_size: 14336
    rms_norm_eps: 1.0e-5
    bos_token_id: 128000
    eos_token_id: 128001
    pad_token_id: 128002
    ignore_token_id: -100
    compute_dtype: "bfloat16"
    layernorm_compute_type: "float32"
    softmax_compute_type: "float32"
    rotary_dtype: "float32"
    param_init_type: "bfloat16"
    use_past: False
    scaling_factor: 1.0
    theta: 500000
    extend_method: "None" # support "None", "PI", "NTK"
    use_flash_attention: True # FA can accelerate training or finetune
    offset: 0
    fine_grain_interleave: 1
    checkpoint_name_or_path: "/home/ma-user/work/ms_ckpt/llama3-8B.ckpt"
    repetition_penalty: 1
    max_decode_length: 512
    top_k: 3
    top_p: 1
    do_sample: False
    pet_config:
      pet_type: lora
      # configuration of lora
      lora_rank: 8
      lora_alpha: 16
      lora_dropout: 0.0
      target_modules: '.*wq|.*wv'
  arch:
    type: LlamaForCausalLM

# metric
metric:
  type: PerplexityMetric

# wrapper cell config
runner_wrapper:
  type: MFTrainOneStepCell
  scale_sense: 1.0
```

```yaml
  use_clip_grad: True

eval_callbacks:
  - type: ObsMonitor

auto_tune: False
filepath_prefix: './autotune'
autotune_per_step: 10

profile: False
profile_start_step: 4
profile_stop_step: 8
init_start_profile: False
profile_communication: False
profile_memory: True
layer_scale: False
layer_decay: 0.65
lr_scale_factor: 256

# aicc
remote_save_url: "Please input obs url on AICC platform."

```

三、微调各阶段权重文件链接(obs桶)

包含模型微调过程中五个阶段（迭代step数分别为：2100，2800，3500，4200，4687）的四个rank_x合并权重checkpoint0.ckpt以及与lora合并后的merged_lora.ckpt，均上传至obs桶，以下为桶链接：

https://dian-stage1-checkpoint0.obs.cn-southwest-2.myhuaweicloud.com/checkpoint0.ckpt

https://dian-stage2-checkpoint0.obs.cn-southwest-2.myhuaweicloud.com/checkpoint0.ckpt

https://dian-stage3-checkpoint0.obs.cn-southwest-2.myhuaweicloud.com/checkpoint0.ckpt

https://dian-stage4-checkpoint0.obs.cn-southwest-2.myhuaweicloud.com/checkpoint0.ckpt

https://dian-stage5-checkpoint0.obs.cn-southwest-2.myhuaweicloud.com/checkpoint0.ckpt

https://dian-stage1-lora-merged.obs.cn-southwest-2.myhuaweicloud.com/merged_lora.ckpt

https://dian-stage2-lora-merged.obs.cn-southwest-2.myhuaweicloud.com/merged_lora.ckpt

https://dian-stage3-lora-merged.obs.cn-southwest-2.myhuaweicloud.com/merged_lora.ckpt

https://dian-stage4-lora-merged.obs.cn-southwest-2.myhuaweicloud.com/merged_lora.ckpt

https://dian-stage5-lora-merged.obs.cn-southwest-2.myhuaweicloud.com/merged_lora.ckpt

四、运行环境说明

无额外配置

五、微调后原有能力评分

- 此部分使用的yaml配置文件为：predict_llama3_8b_800T_A2_64G.yaml
- 测评结果：F1 score: 61.86649462896725, Em score: 47.26656990807934, total_count: 2067
- 微调参数比例：3407872/8030000000=0.000424392528019925280199252801992 53

```
(MindSpore) [ma-user msrun_log]$cat worker_0.log |grep "Network Parameters"
2024-07-30 23:49:14,207 - mindformers[mindformers/trainer/base_trainer.py:543] - INFO - Network Parameters: 3407872.
```

六、模型推理部分修改

1. 我们对 `run_llama3_test.py` 文件进行了部分修改，具体修改部分为

```
with open(input_dir, 'r', encoding='utf-8') as file:
        # print(file)
        for line in file:
            line = json.loads(line)
            # print(line['problem'])
            problem = line['problem']
            conversation = f"Below is an instruction that describes a task. Write a response that appropppriately completes the request.\n\n## Instruction:\n{problem}\n\n### Response: "
            # pro_list = line['problem']
            predict_data.append(conversation)
```

相当于让模型推理时提供一个模板，让大模型的回复更标准更规范，经测试，这对提升模型回答数学题的质量有帮助。

2. 此部分使用的yaml配置文件为：run_llama3_8b_8k_800T_A2_64G_lora_256_eval.yaml