Our artifact contains our code on fine-tuning, merging and evaluating models with or w/o our approach, Medusa. It can be found in our open repository on github: PKU-ASE-RISE/medusa.

Setup

Prepare Python3.12 and newest pip, install dependencies by

```
 pip install  -r ./requirements.txt
```

Train

You can do a vanilla finetuning process on RTE of GLUE dataset with T5-base by

```
python T5mask.py --mask=normal --out_dir=ckpts/normal/rte/ --device=cuda:0 --
dataset=cola --model=google/t5-v1_1-base --modularized
```

Our training method is written in 'T5_mutual_mask.py'. For example, command finetuning on dataset COLA and RTE when we have full parameter control is:

```
python T5_mutual_mask.py --top_k=80 --mask=soft_magnitude_mutual_mask --
out_dir=ckpts/mutual/cola_rte/ --device=cuda:0 --datasets cola rte --
model=google/t5-v1_1-base --modularized
```

, while finetuning on dataset COLA to be merged with a model already trained on RTE, e.g. the vanilla model finetuned above, which corresponds to the situation we only hold partial parameter control ownership, can be done through:

```
python T5_mutual_mask.py --top_k=80 --mask=soft_magnitude_mutual_mask --
out_dir=ckpts/single/cola_with_rte/ --device=cuda:0 --datasets cola --ref_models
ckpts/normal/rte_best --model=google/t5-v1_1-base --modularized
```

Besides, we can add '--mixed' to enable mixed precision finetuning with bf16 if your device supports; '--peft ia3' or '--peft lora' can apply PEFT finetuning instead of full finetuning. If you encounter CUDA memory problems, try '--smaller_batch=k' to use k times smaller batch and less CUDA memory.

Merge & Evaluation

The merging process and evaluation process is done simultaneously.

```
python T5merge.py --method=_MEDUSA\
    --out_file=logs/merege_cola_rte_test_cola.txt\
    --models ckpts/mutual/cola_rte/cola_best ckpts/mutual/cola_rte/rte_best\
    --dataset=cola\
    --base_model=google/t5-v1_1-base\
    --modularized\
    --device=cuda:0
```

out_file contains models' file paths and the merging result. The merging method in '--method' parameter can be chosen from:

```
'''
    _SUM            simple averaging
    _TA             task arithmetic with lambda=4
    TIES_           TIES with k=20% lambda=1
    DARE_SUM        DARE with k=10% and simple averaging
    DARE_TA         DARE with k=10% and task arithmetic
    DARE_TIES       DARE with k=10% and TIES
    _MEDUSA         MEDUSA merging (after masked finetuning),
                    which is similar to TIES without keeping top-k%
'''
```