ADL HW2 Report

NTU CSIE, R12922051
資工碩一 陳韋傑

Q1: Model

I used pre-trained google/mt5-small and finetune it on the given dataset.

T5 is a transformer-based NLP model pre-trained on Colossal Clean Crawled Corpus (C4) dataset. T5 frames all the NLP tasks as text generation tasks, by adding a prefix like “summarize” or “translate English to German”, T5 will know which task we want to perform and generate the corresponding results.*

mT5 is a multilingual variant of T5 that was pre-trained on multilingual C4 datasets (mC4) covering 101 languages, thus make it better at handling multilingual input or output. mT5 also use some techniques to prevent “accidental translation” in the zero-shot setting, where the model (partially) translate its prediction into the wrong language.**

For preprocessing, I use the pre-trained mT5 tokenizer on hugging face. Since T5 and mT5 needs a prefix to know which task to perform, I prepend “summarize: ” to the input text.

* Colin Raffel et al. Google. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer.
** Linting Xue et al. Google Research. mT5: A massively multilingual pre-trained text-to-text transformer.

Q2: Training

The hyperparameters I used are:

- Model name: google/mt5-small
- Epochs: 20
- Learning Rate: 3e-4
- Batch Size: 8
- Warmup Steps: 500
- Max Source Length: 1024
- Max Target Length: 128
- Optimizer: AdamW
- Scheduler: Linear

Besides from epochs, learning rate and warmup steps, I didn't adjust default parameters provided by sample code. (Ref. 1)

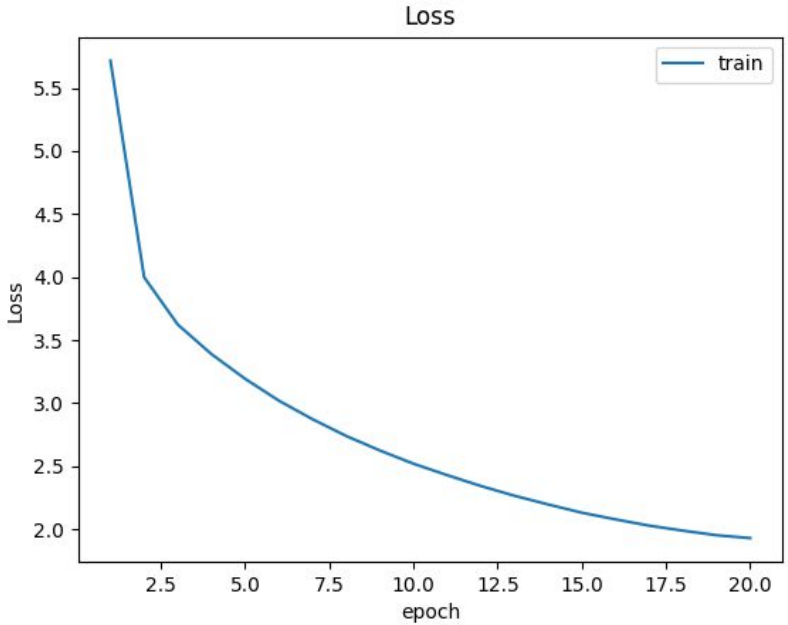
Judging by loss plot and rouge score plot, I believe the epoch should be increased instead of using the default value (which is 3).

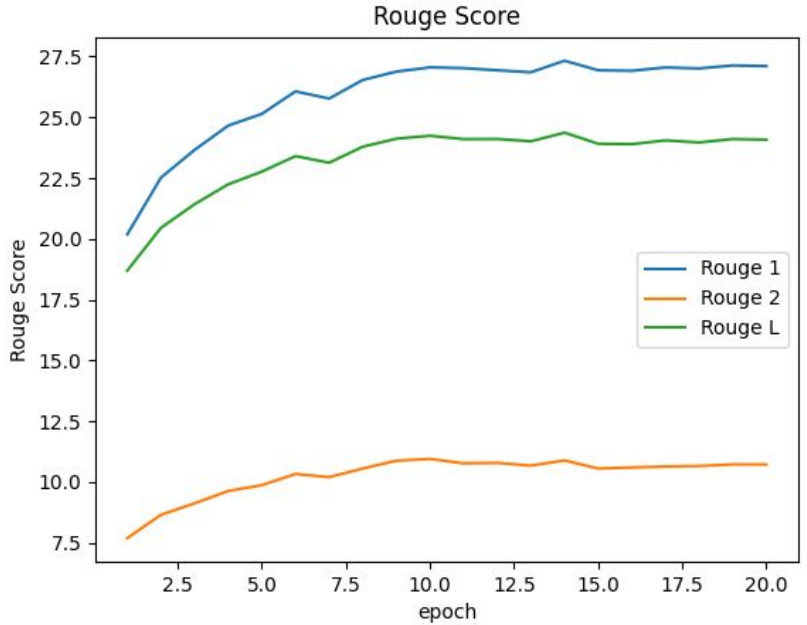
Also, I read an article says that t5 needs a larger learning rate (e.g. 1e-4 or 3e-4), so I adjust it accordingly (Ref 3.)

Finally, it's common to warmup learning rate, to increase training performance.

Q2: Training

The following figures are training loss and rouge score.
Rouge score are calculated using beam search with beam = 3.





Q3: Generation Strategies

- Greedy: Choose the most possible output word (argmax, equals to beam search with beam size = 1)
- Beam Search: Choose k most possible output sequences at each step (beam size = k)
- Sampling: Choose output word by sampling from the probability distribution of model output
- Top-k Sampling: Same as sampling, but only consider k most possible output (k=1 equals to greedy, k=V equals pure sampling)
- Top-p Sampling: Same as sampling, but only consider output with total probability greater than p
- Temperature: Appling a temperature parameter to the softmax layer, thus controlling the shape of output distribution (higher → more diverse, lower → more stable)

Please see next page for experiment results.

Strategy/Rouge Score (F)	Rouge 1	Rouge 2	Rouge L
Greedy	25.87	9.63	22.94
Beam Search (beam = 5)	27.33	10.89	24.24
Beam Search (beam = 10)	27.36	11.02	24.24
Beam Search (beam = 20)	**27.43**	**11.11**	**24.33**
Sampling(beam = 1, p = 0, k = 50, t = 1)	22.10	7.60	19.46
Sampling(beam = 3, p = 0, k = 50, t = 1)	26.66	10.31	23.64
Sampling(beam = 3, p = 0, k = 25, t = 1)	26.63	10.31	23.58
Sampling(beam = 3, p = 0, k = 100, t = 1)	26.66	10.29	23.63
Sampling(beam = 3, p = 0.7 , k = 50, t = 1)	26.58	10.26	23.59
Sampling(beam = 3, p = 0, k = 50, t = 3)	12.28	1.92	10.47
Sampling(beam = 3, p = 0, k = 50, t = 0.7)	25.87	9.87	23.04
Sampling(beam = 20, p = 0, k = 50, t = 1)	27.06	10.72	24.01

Q3: Generation Strategies

Discussion:

- My final strategy is just use beam search with beam size = 20
- Since summariztion task has ground truth answer, sampling methods will not perform well
- Rouge score increases as beam size increases, but inference time also increases
- Beam-search multinomial sampling performs better than simple multinomial sampling
- Top-k and Top-p threshold does not have a strong influence on performance when using beam-search multinomial sampling

References

1. https://github.com/huggingface/transformers/tree/main/examples/pytorch/summarization
2. https://huggingface.co/docs/transformers/main_classes/text_generation
3. https://huggingface.co/docs/transformers/model_doc/t5
4. https://blog.csdn.net/muyao987/article/details/125917234