1.b ii

The performance of both med and tiny bert models increases with more support examples. Bert-med model outperforms the bert-tiny model when the number of support examples are limited.

1.c

With 11.2 million parameters and 4-bytes per parameter, the total memory space required is

$11.2 * 10^6 * 4 / 1024^2 = 43MB$

1.d

With 540B parameters and 4-byte per parameter, the total memory size is

$540 * 10^9 * 4 / 1024^3 = 2TB$

2.b ii

The full-size GPT2 model outperforms the medium-size in zero-shot and few-shot scenarios, and for both model sizes, providing more than 1 support example do not lead to significant performance improvements.

2.c ii

The TL;DR prompt outperform the no prompt formatting. In my custom prompt format, I added "article:" before the article/input and "summary:" before the summary/target. For the medium GPT-2, my custom prompt and TL;DR prompt achieved relatively similar rouge scores. For full-size GPT-2, my custom prompt performs better in 1-shot scenario and worse in 0-shot and few-shot scenarios compared to TL;DR prompt.

3.b

The pre-trained weight matrix W^0 have d_1*d_2 parameters and AB^T have $(d_1 + d_2)*p$ parameters. The ratio of parameters fine-tuned by LoRA to the number of parameters in W^0 is $(d_1 + d_2)*p/ d_1*d_2$.

When $p << d_1*d_2/(d_1 + d_2)$, LoRA will provide the greatest savings in newly-created parameters.

3.d ii

When evaluated with xsum dataset, 0-shot and 1-shot scenarios show little difference across all fine-tuning methods, while in few-shot scenarios, fine-tuning with LoRA outperforms fine-tuning on the last or the first model parameter layer.

When evaluated with babi dataset, 0-shot scenario show little difference across all fine-tuning methods, while in 1-shot and few-shot scenarios, fine-tuning with LoRA achieves highly competitive performance with fine-tuning on the middle layer, and much better than fine-tuning on the last or the first layer.

4.a

When there are no or just a few support examples, in-context learning may be a better choice for 0-shot and 1-shot learning. When there are more support examples, fine-tuning in general yields a better

performance. In-context learning tends to have a higher variance in its results and the more examples do not guarantee better results.

4.b

The evaluation performance of in-context few-shot performance for 5 random orderings of the prompt is 0.704. In-context learning has a higher standard deviation than fine-tuning.