

“Briefly discuss about full fine-tuning and adapter-based tuning.”

➤ **Full Fine-tuning**

- Transfer Learning에서 Source model로 사용하려는 pre-trained model의 전체 parameter에 대하여 학습을 활성화시키는 기법이다. 그렇게 함으로써 모델은 target task의 데이터에 맞게 전체 layer와 parameter를 업데이트하게 된다.
- 이는 target task의 데이터가 크거나 target task 자체가 복잡한 경우에 대하여 좋은 성능을 보일 수 있다는 장점이 있다.
- 반면, 모델의 전체 구조를 학습시키기 때문에 그에 비례하여 더 많은 자원 및 비용을 소비한다는 점과 target task의 데이터가 많지 않은 경우 memorization에만 치중하게 되어 과적합이 발생할 수 있다는 단점이 있다.
- 따라서 Full Fine-tuning 방식은 target task의 데이터 및 리소스가 충분할 때에 권장된다.

➤ **Adapter-based tuning**

- Source model로 사용하려는 pre-trained model의 특정 일부 layer에 대해서만 adapter module을 부착하고 학습하는 기법이다. 이 때 adapter module의 학습은 원본 모델의 parameter를 업데이트하는 것이 아닌 adapter 자신의 parameter를 학습하는 방식으로 이루어진다.
- 따라서 Full Fine-tuning 방식과 비교했을 때 전체 모델을 다시 학습시킬 필요가 없고, 그 결과 자원 및 비용 측면에서 이득을 얻을 수 있다는 장점이 있다.
- 그러나, 모델의 전체 구조를 target task에 맞게 fine-tuning하지 않는다는 점은 곧 full fine-tuning 방식에 비해 낮은 performance를 보일 수 있다는 단점으로 이어진다.
- 그러므로, adapter based-tuning 방식은 상대적으로 target task의 데이터나 리소스가 충분치 않을 때에 유용한 방법이 될 수 있다.

“Explain architectural differences between full fine-tuning and adapter-based tuning in this experiment.”

➤ **Full Fine-tuning**

- Pre-trained BERT 모델의 전체 parameter를 trainable하게 설정한다.
- 학습을 위한 “forward” 메소드 부분에 Adapter module을 적용하는 부분이 없고, BERT 모델이 가지고 있던 layer(pooling 포함)들을 사용하여 output을 도출한다.
- 결과적으로, pre-trained BERT 모델의 전체 parameter는 target task에 맞추어 추가적인 재학습을 거친다.

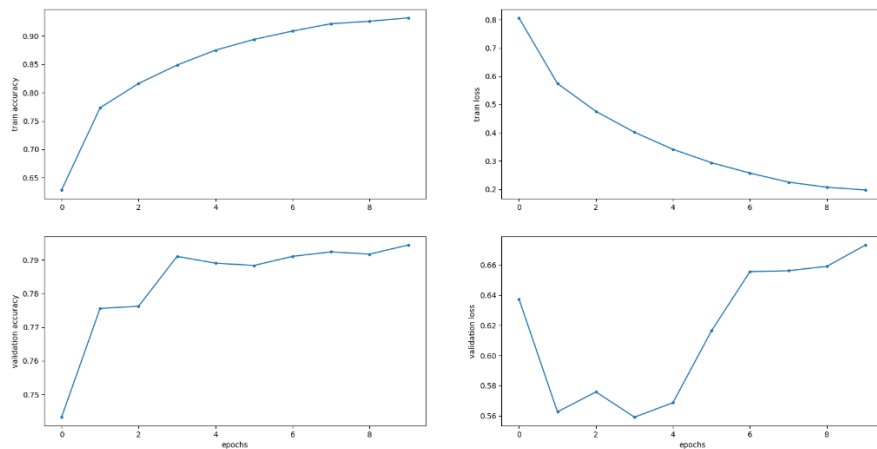
➤ **Adapter-based tuning**

- Pre-trained BERT 모델의 일부 layer에 별도의 adapter module을 부착하여 학습되도록 설정한다.
- 학습을 위한 “forward” 메소드 부분에 Adapter module을 적용하는 부분이 있다.
- 또한, “top_adapter”와 “layer_wise_adapter”라는 두 종류의 adapter module을 사용하고 있다.
 - ◆ Top adapter: pre-trained model의 last layer에 부착된다.
 - ◆ Layer wise adapter: pre-trained model의 각 encoder layer에 부착된다.
- 결과적으로, pre-trained BERT 모델의 특정한 부분만 target task에 맞추어 추가적인 재학습을 거치게 된다.

"Analyze and compare the results of 2 fine-tuning methods and each modified settings (Q1, Q2, Q3, Q4) on model performance."

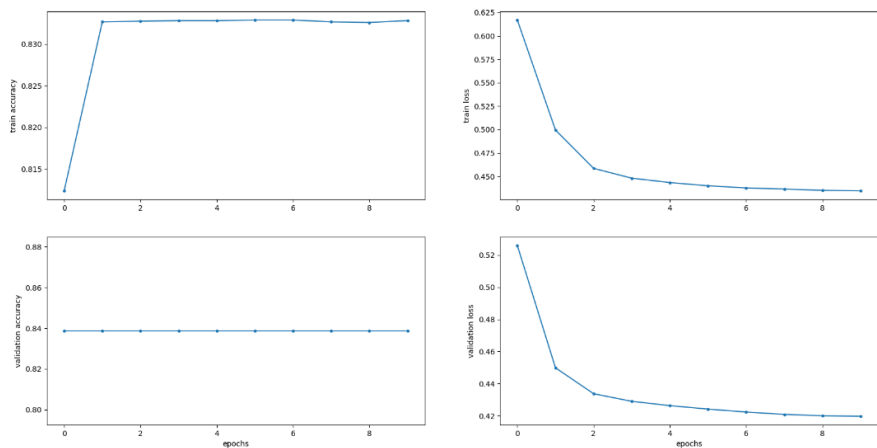
1. IMDB Dataset

➤ Full Fine-tuning (Q1)



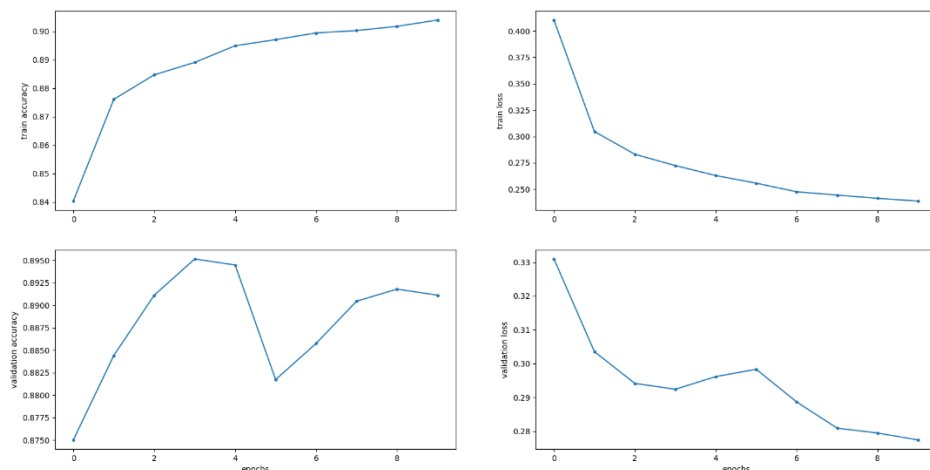
Val loss 0.2453 accuracy 0.9486
inference acc: 0.9692

➤ Adapter-based tuning (Q2) - TOP



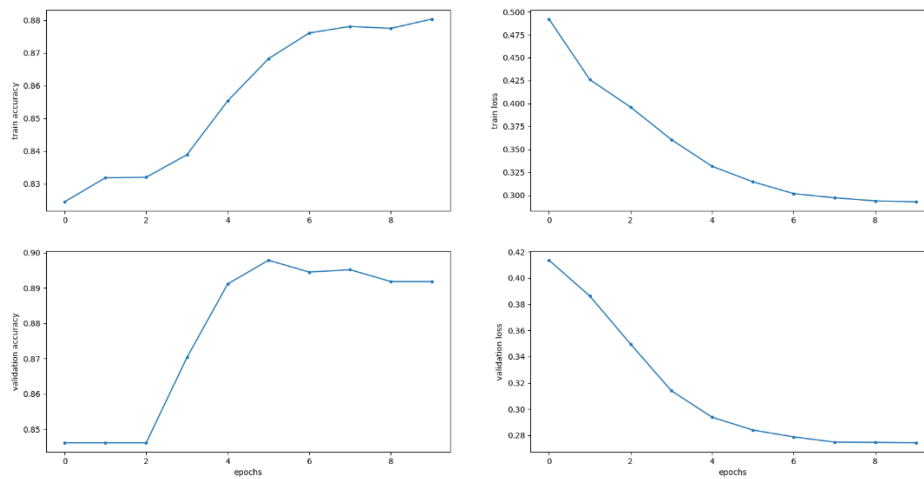
Val loss 0.4196 accuracy 0.8387
inference acc: 1.0

➤ Adapter-based tuning (Q2) - LAYER



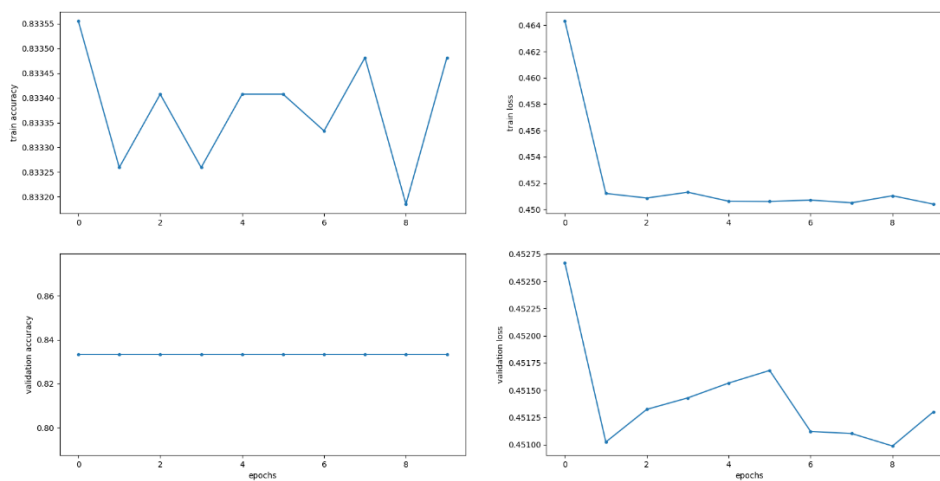
Val loss 0.2774 accuracy 0.8911
inference acc: 0.9468

➤ Adapter-based tuning (Q3) - TOP



Val loss 0.2742 accuracy 0.8918
inference acc: 0.9711

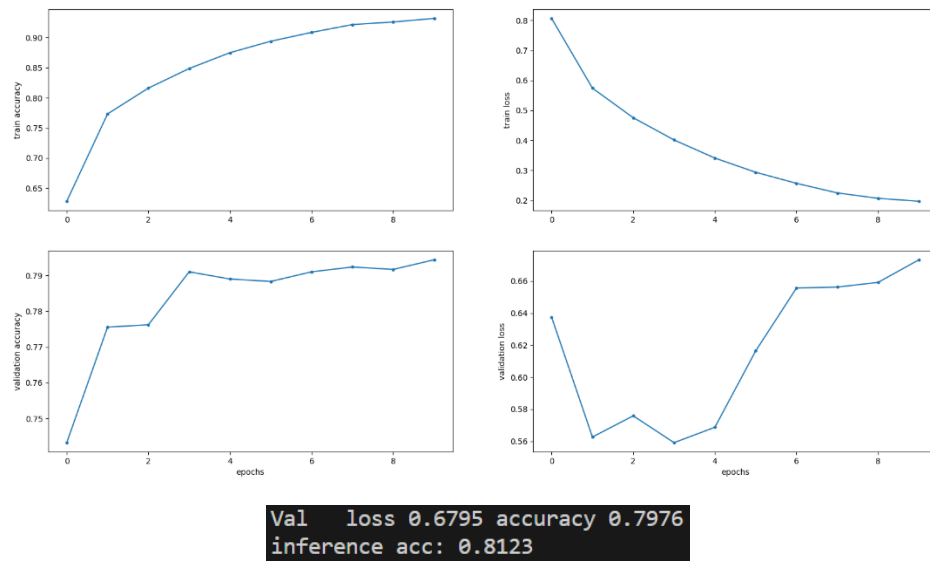
➤ Adapter-based tuning (Q3) - LAYER



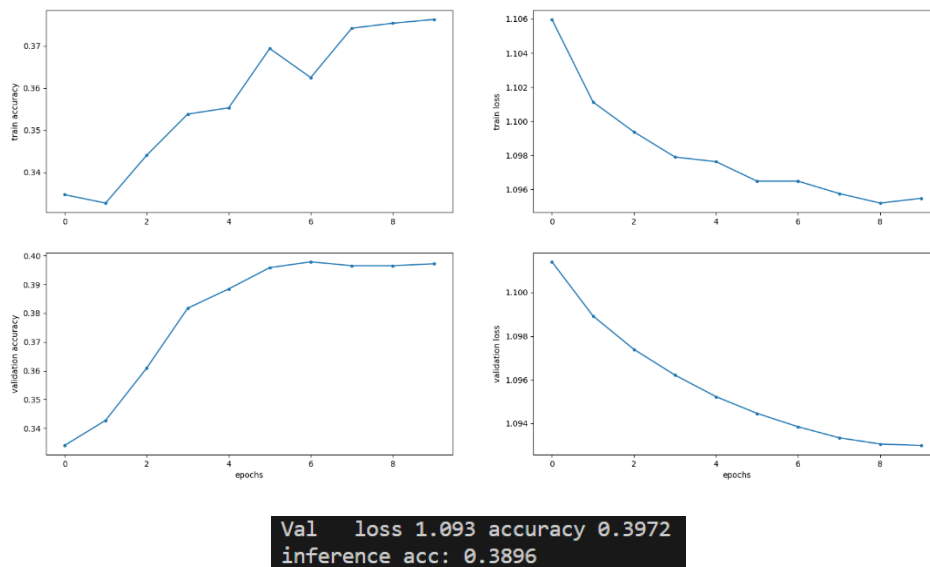
Val loss 0.4513 accuracy 0.8333
inference acc: 1.0

2. SNLI Dataset

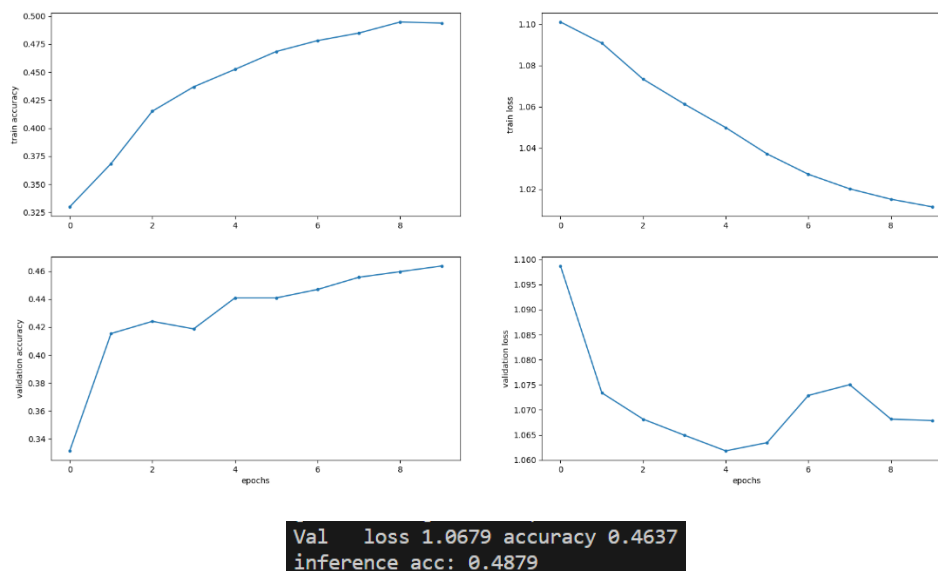
➤ Full Fine-tuning (Q1)



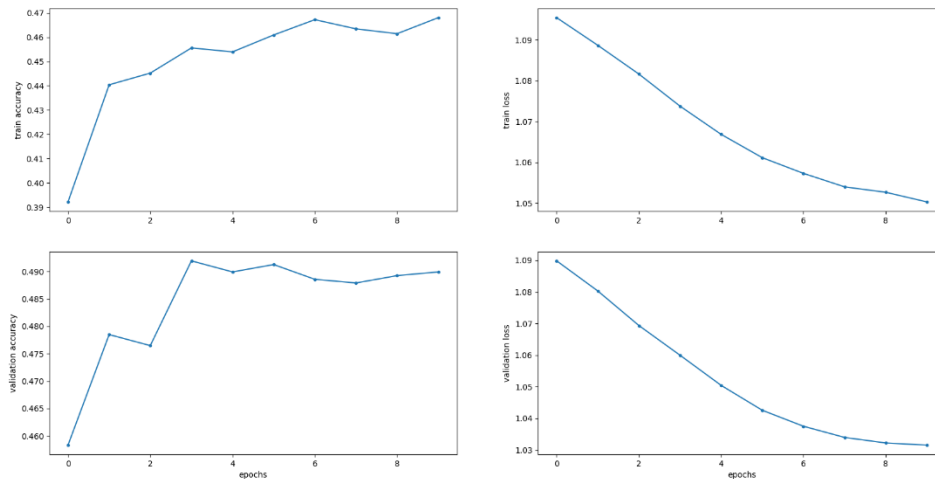
➤ Adapter-based tuning (Q2) - TOP



➤ Adapter-based tuning (Q2) - LAYER

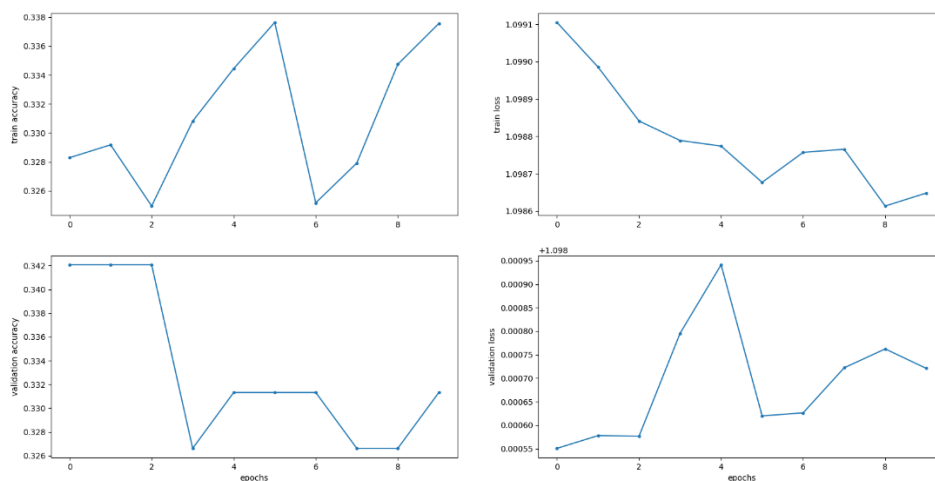


➤ Adapter-based tuning (Q3) - TOP



```
Val   loss 1.0315 accuracy 0.4899
inference acc: 0.4831
```

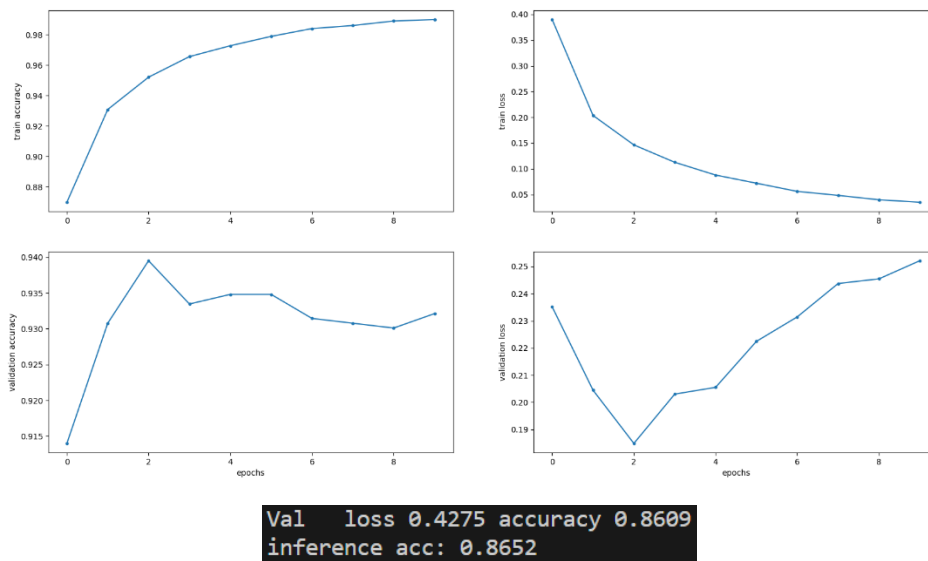
➤ Adapter-based tuning (Q3) - LAYER



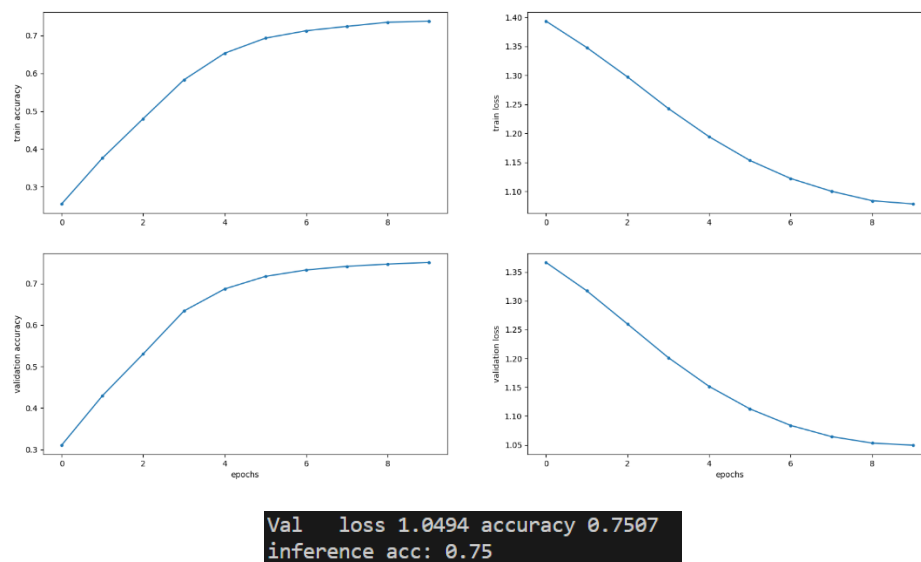
```
Val   loss 1.0987 accuracy 0.3313
inference acc: 0.3289
```

3. AG News Dataset

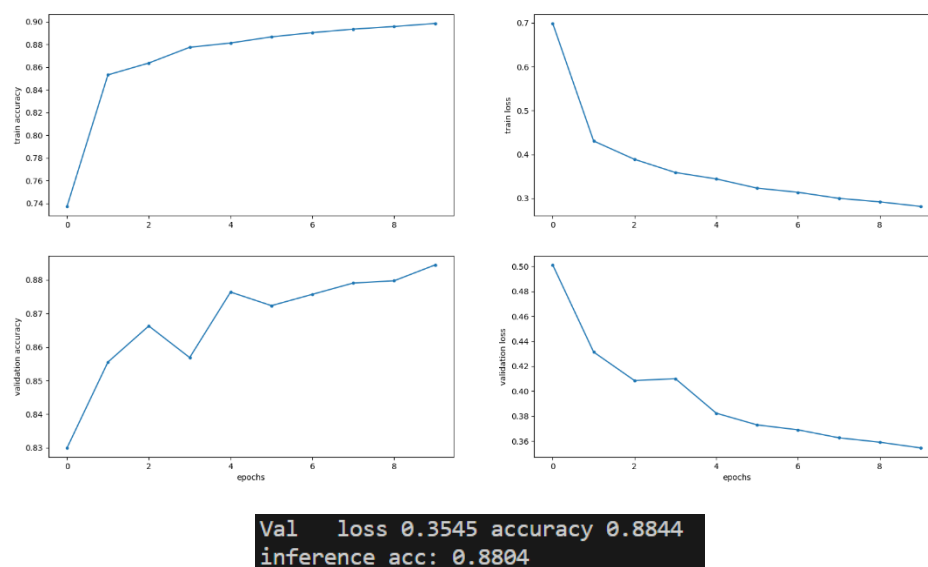
➤ Full Fine-tuning (Q1)



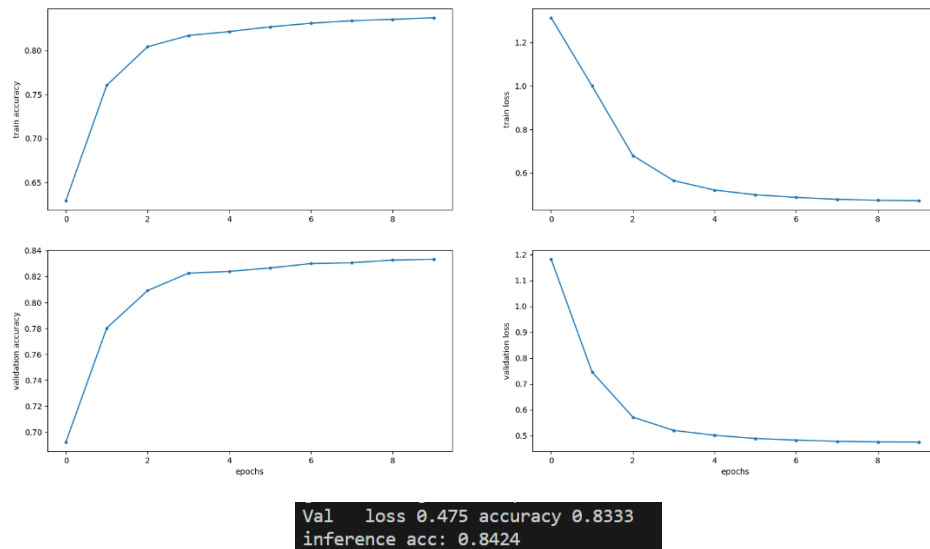
➤ Adapter-based tuning (Q2) - TOP



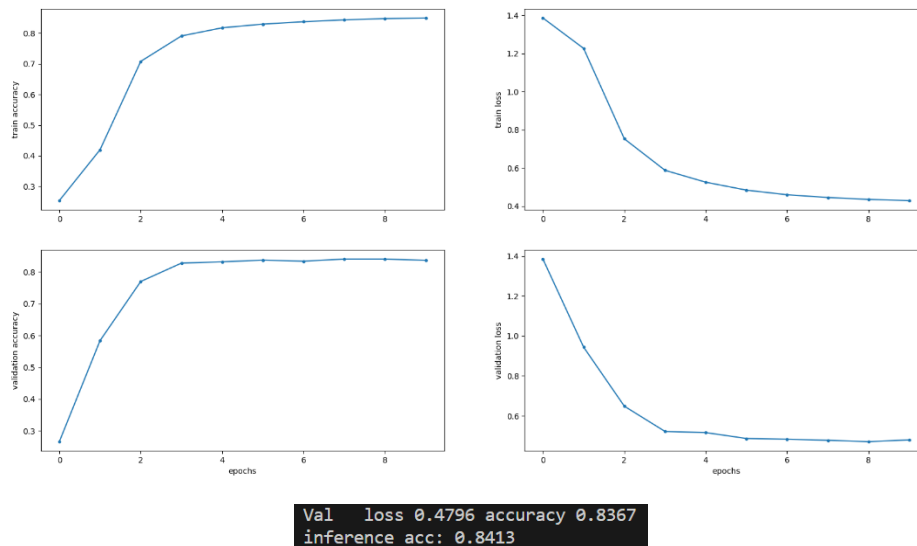
➤ Adapter-based tuning (Q2) - LAYER



➤ Adapter-based tuning (Q3) - TOP



➤ Adapter-based tuning (Q3) - LAYER



4. 전반적인 성능 비교

- Full Fine-tuning 방식이 Adapter-based tuning보다 학습에 더 오랜 시간이 걸렸다. 이는 모델의 전체 파라미터를 학습시키는 방향으로 fine-tuning이 이루어지기 때문이라고 본다.
- 반면 Adapter-based tuning은 학습 시간은 더 적게 소요되었으며, 초반의 epoch에서는 full fine-tuning의 동 epoch과 비교했을 때 낮은 성능을 보였다.
- Adapter-based tuning 내에서도, "Top" 방식이 "Layer" 방식보다 학습이 빨리 진행되었다. 이 역시 "Top" 방식은 전체 layer에 대하여(Q2, Q3 실험의 가정) adapter를 부착하는 "Layer"방식과 달리 모델의 Last layer에만 adapter를 적용하여 학습시키기 때문일 것이다.
- 또한 full fine-tuning과 adapter-based tuning의 성능을 비교하면, 각 Dataset에 대하여 full fine-tuning을 한 경우가 가장 높은 inference accuracy를 보였다. 이는 1번 문제에서 언급한 "full fine-tuning의 장점" 및 "adapter-based tuning의 단점"과 일맥상통한다.
- Top과 Layer 방식을 비교했을 때에는 대부분의 경우에서 Layer adapter를 적용한 경우가 더 높은 성능을 보였는데, 이는 Top보다 더 많은 layer(parameter)를 target task에 맞추어 fine-tuning시켰기 때문일 것이다.