

# Master thesis report 5

University of Tartu

March 13, 2023

## 1 Parameter efficient fine-tunings (PEF)

I started with soft prompt implementation(i.e. prompt tuning [Lester et al., 2021]). However, even though the implementation inside T5 transformer class was quite easy, nonetheless it wasn't working(i.e. nor loss, and nor accuracy improved over time). Then accidentally I found transformer extension library adapter transformer, that implement various PEFs algorithms like adapters, prefix-tuning, etc.

I tuned the model. Results presented in Figure 1. I run training for 5 days, and we can see that by the 10th epoch adapters, lora, and finetuning attained their best performance at around 60%. The best performance was achieved by adapters with 63% accuracy, the lowest with 33% by in-context learning(where I trained using only 10K factual examples, few-shot learning simulation). Prefix-tuning was the slowest, and I experimented with two prefix lengths 30 and 200. It is clear that prefix length is a hyperparameter, and I need to mine for good one. But I think small one should give better results, considering that for 200 I've obtained worse accuracy than for 30.

## 2 Data

[Neeman et al., 2022] still didn't provide their datasets. So, I started working on that to get it myself. It is currently in the progress.

## 3 Adversarial training

Maybe adversarial training could be also used as PEFs. Similarly as in [Welbl et al., 2020].

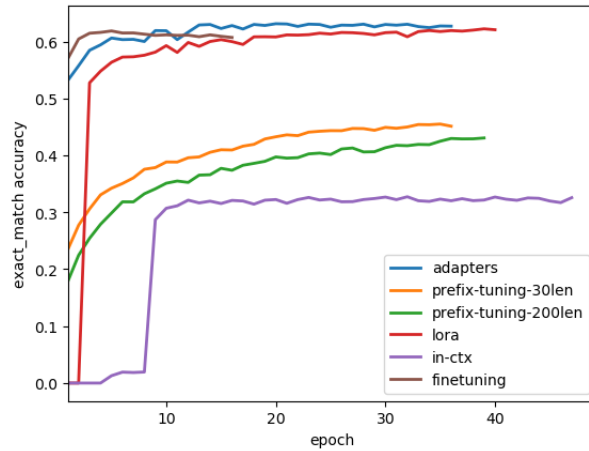


Figure 1: Different turning

## 4 Literature review

I wasn't able to find papers stating that PEF gives generalization. Thus I consider it as weak spot of my work. I won't be able to answer question why.

## 5 Agenda

1. soft-prompt library and obtained results.
2. In-context learning with following patter 'Using context: " $\langle$ context $\rangle$ ". Answer following question: " $\langle$ question $\rangle$ "  $\langle$ /s $\rangle$ ' didn't give promising results.
  - Looking for hard-prompt manually would be time consuming.
  - How do I approach this in-context learning? It should be few-shot learning?
3. Data set preparation. In the progress right now.
4. Lets talk about experiments. After I augment the data I would have everything to start training.
  - Are we going to follow the same structure of the experiments?
  - We need better results on data with counterfactual and answer-ability augmentations?

- We could also perform adversarial training like in undersensitivity paper([Welbl et al., 2020]).
- Experiments:
  - Data aug + finetuning
  - Data aug + adaptors
  - Data aug + soft-prompting(prefix-tuning, LoRA)
  - Data aug + in-context-learning
  - Data aug + Adversarial training
- (a) We also need a baseline with soft tuning, and adaptors, right?

## 6 This week plan:

1. Design experiments.
2. Perform experiments.
3. Fix in-context learning.
4. Fix prefix-tuning.
5. Prepare data.

## References

- [Lester et al., 2021] Lester, B., Al-Rfou, R., and Constant, N. (2021). The power of scale for parameter-efficient prompt tuning.
- [Neeman et al., 2022] Neeman, E., Aharoni, R., Honovich, O., Choshen, L., Szpektor, I., and Abend, O. (2022). Disentqa: Disentangling parametric and contextual knowledge with counterfactual question answering.
- [Welbl et al., 2020] Welbl, J., Minervini, P., Bartolo, M., Stenetorp, P., and Riedel, S. (2020). Undersensitivity in neural reading comprehension.