Stanford | ENGINEERING
Computer Science

Parameter Efficient Fine-tuning for Multi-task Learning

Chih-Ying Liu [1] Jeffery Shen [2]
[1]Electrical Engineering, Stanford [2]Computer Science, Stanford

Introduction

Though BERT was able to obtain new state-of-the-art results on NLP tasks at release time, there are still areas to improve upon. One point of interest is parameter efficiency – BERT's state-of-the-art results used transfer from unsupervised pre-training, with a separate BERT model fine-tuned for each individual task. Our goal is to build robust embeddings that perform well across a large range of different tasks, without having to finetune individual models for individual tasks.

To do so, we implement Projected Attention Layers (PALs), adapters, and prefix tuning to achieve optimal performance over multi-tasks while being parameter-efficient. We also experiment with changes to the BERT model architecture by implementing Sentence-BERT and modifying the downstream classifier head architecture.

Background

In our project, we implement and explore the following concepts

- **Sentence-BERT** *Reimers and Gurevych 2019*: A modification of pretrained BERT network that use siamese network structure, mean pooling, and absolute element wise difference for sentence pair classification to obtain semantically meaningful sentence embeddings.
- **Projected Attention Layers (PALs)** *Stickland, Murray 2019*: A low-dimensional multi-head attention layer that is added in parallel to normal BERT layers. PALs involve a task-specific function $TS(h) = V^D g(V^E h)$ where V^D and V^E are some projection layer shared across layers, and $TS(.)$ is self-attention layer.
- **Prefix-Tuning** *Li and Liang*: A trainable continuous task-specific vectors prepended to the input of transformer layers.
- **Adapter** *Houlsby et al.*: A module added sequentially in transformer layers. Adapter contains a down project layer, an activation function, and a up project layer.

Data

All Data Sources:

Name	Task?	Size (Total)	Size (Train)
Stanford Sentiment Treebank (SST)	SA	11,855	8,544
CFIMDB	SA	2,438	1,705
Quora (QQP)	Paraphrase	202,151	141,506
SemEval STS	STS	8,628	6,040
SemEval SICK 2014	STS	10,000	4,500
Amazon Kindle Reviews	SA	982,619	*variable*
Rotten Tomatos	SA	634,251	*variable*

Final Data Sources:

Task	Final Train Set	Size
SA	SST Train + Rotten Tomatos (15k)	23,544
Paraphrase Detection	Quora Train	141,506
STS	SemEval SST Train + SICK2014 Train	10,540

Training Pipeline

Sampling Type	Distribution
Round Robin	$P_i \propto$ #Tasks
Proportional	$P_i \propto N_i$
Square-Root	$P_i \propto (N_i)^\alpha, \alpha = .5$
Annealed	$P_i \propto (N_i)^\alpha, \alpha = 1 - .8\frac{e-1}{E-1}$

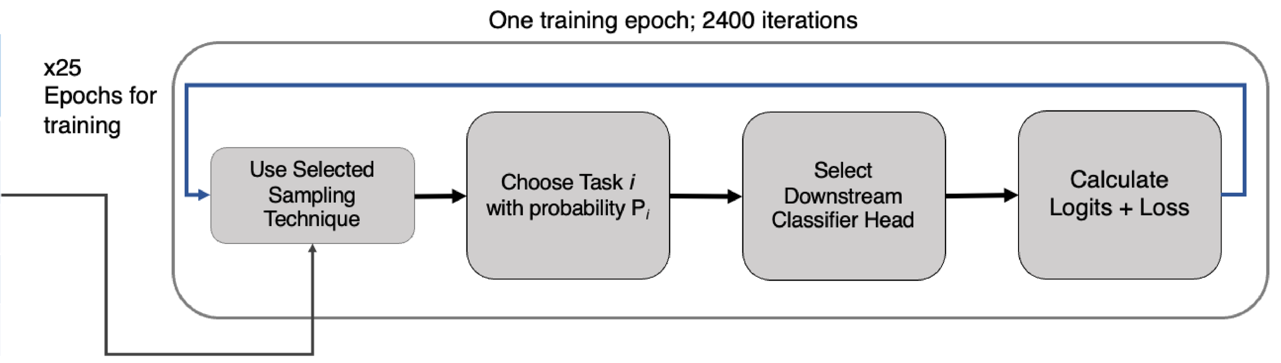

Figure 1. Training Pipeline

Model Architecture

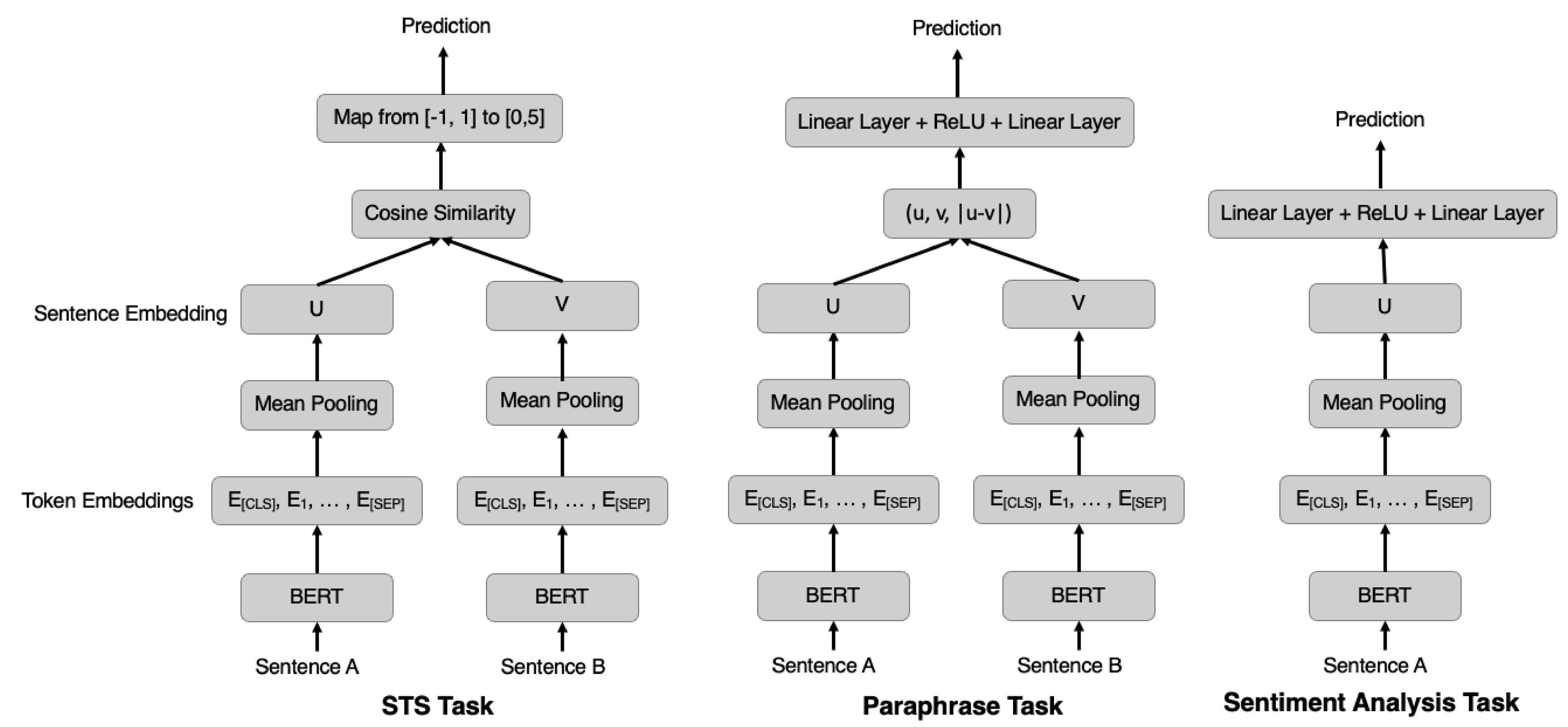

Figure 2. Model Architecture

SentenceBERT: Siamese network structure, mean pooling, and absolute element wise difference for sentence pair classification

Adaptation Modules

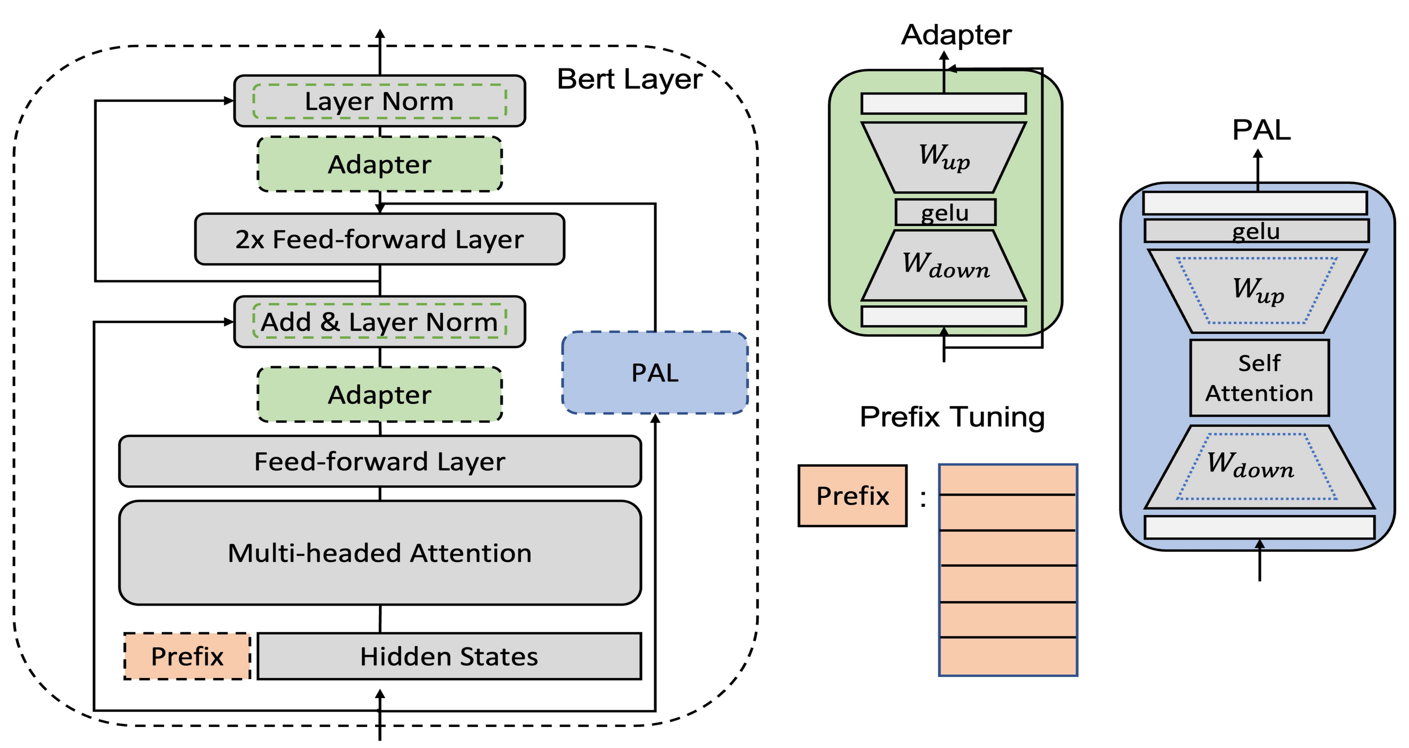

Figure 3. Adaptation modules

Insert task-specific adaptation modules, PAL, prefix, and adapter, into BERT layers.

Results/Conclusion

Performance of different methods

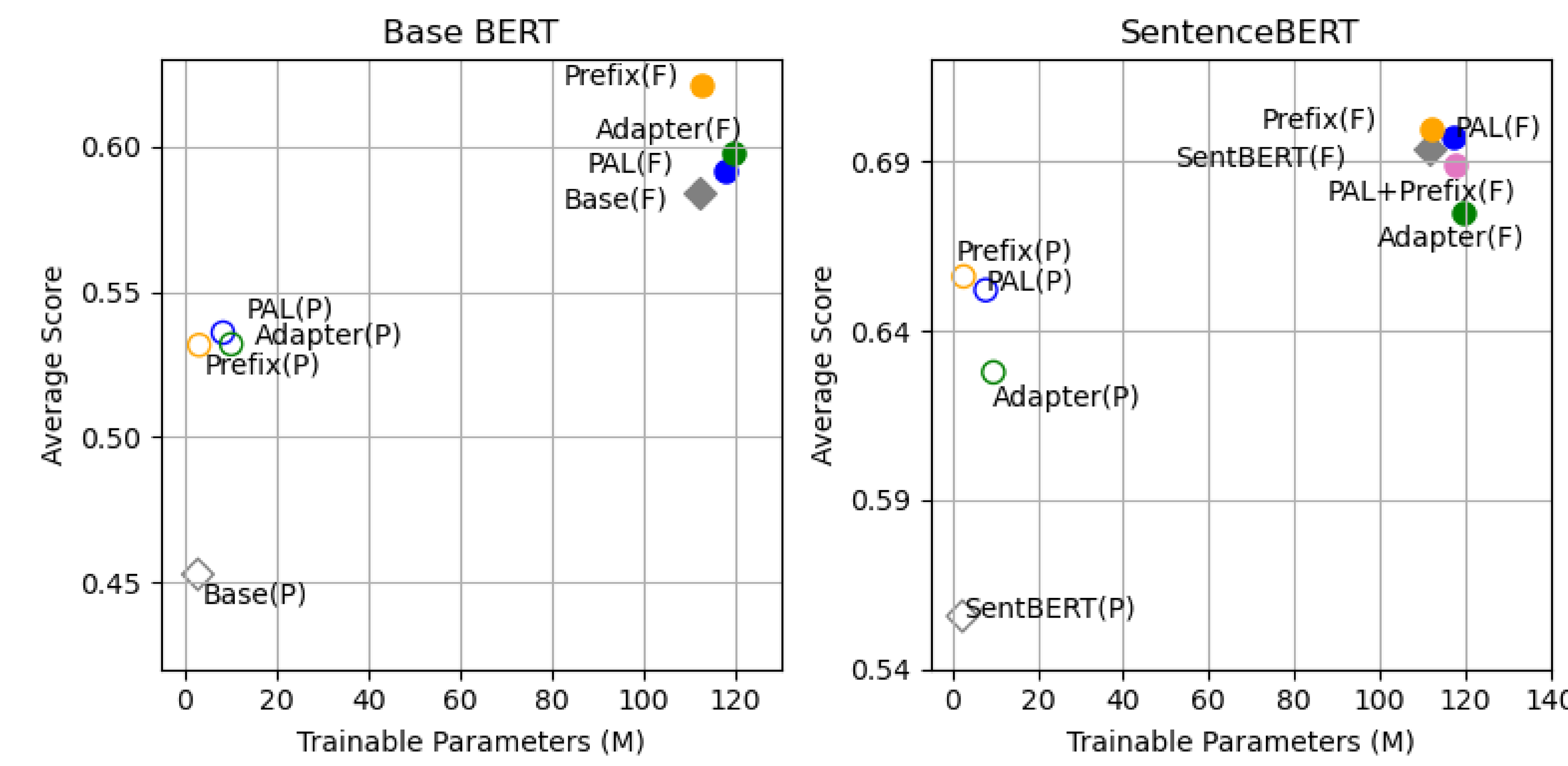

Figure 4. Performance v.s. trainable parameters of different methods. P: Pretrain (Fix BERT backbone), F: Finetune.

Backbone	Adaptation	SST Acc	Quora Acc	STS Coorelation	Avg Score
B	-	50.0	81.5	43.7	58.4
B	PAL	50.6	79.2	47.6	59.2
B	Prefix	49.9	82.9	53.6	62.1
B	Adapter	51.5	81.6	46.2	59.8
S	-	49.9	82.3	75.9	69.4
S	PAL	51.2	83.4	74.6	69.7
S	Prefix	50.5	81.9	77.5	70.0
S	Adapter	50.5	79.3	72.7	67.5
Ensemble x3		**53.2**	**83.5**	**78.5**	**71.7**

Table 1. Finetuning result for different backbone and adaptation modules. B: Base BERT, S: SentenceBERT.

- Sentence BERT, especially mean pooling, gives great improvement in similarity task.
- PAL, prefix, and adapter gives great improvement in pretaining mode, and is comparable to fine-tuning with less than 3% to 9% trainable parameters.

Prefix Length

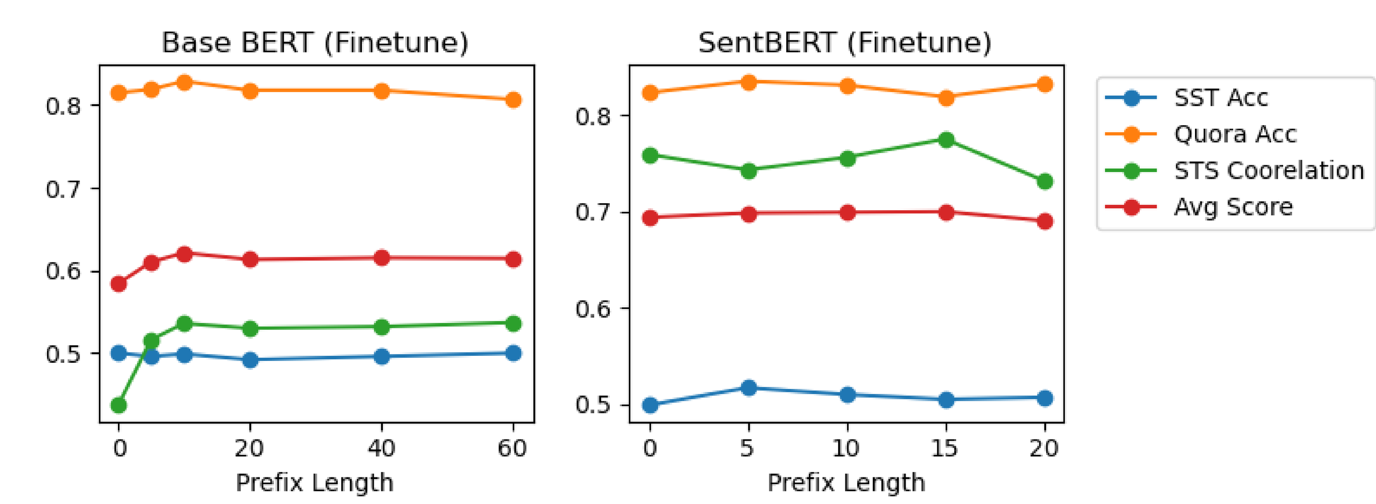

Figure 5. Prefix length v.s. performance.

- With Base Bert backbone, the performance increases as the prefix length increases up to 15.