CS 224N Project

Prototypical Pre-Training for Robust Multi-Task Learning

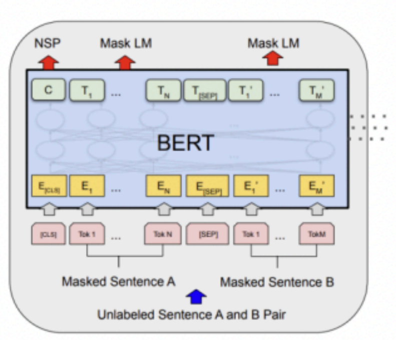
Rohan Sikand and Andre Yeung

Background & Introduction

- (Snell et al., 2017) propose a meta learning technique called **Prototypical Networks (ProtoNets)** for few-shot learning to learn classification tasks with only a few examples per class.
- We extend the idea of ProtoNets for the purpose of pre-training. Specifically, we propose a novel variant, "prototypical pre-training," that learns a feature space using BERT, which acts as an embedding function to map input vectors into this learned feature space.
- We demonstrate Improvements in performance with our architecture over a traditional supervised multi-task baseline and other methods tested.

Problem Setup

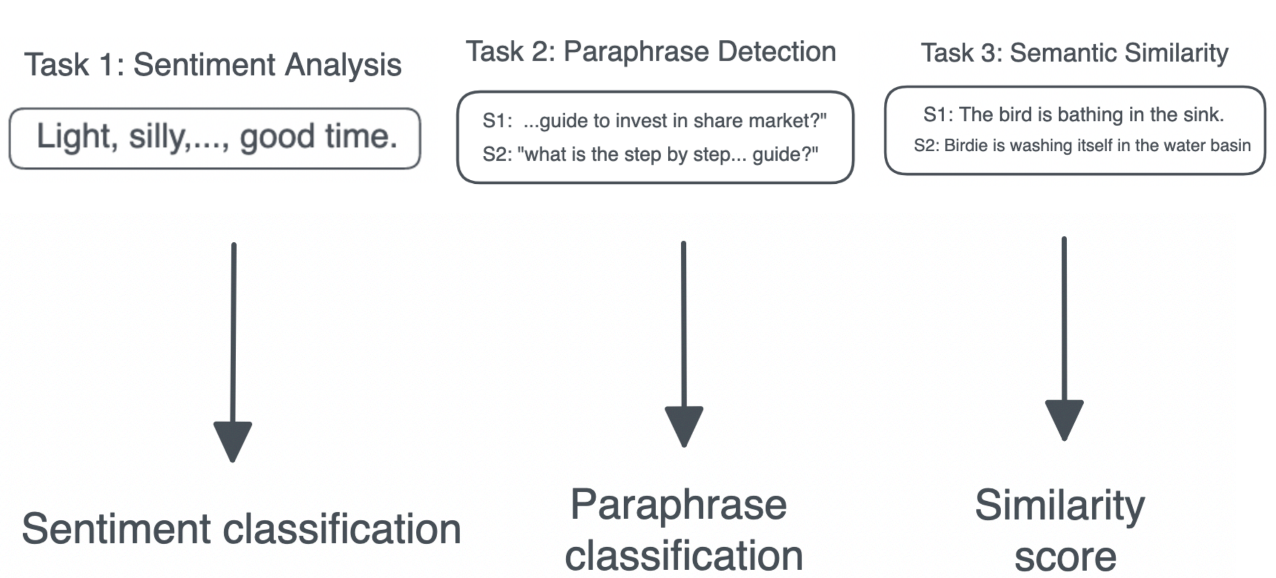
- **BERT:** bidirectional transformer language model



Pre-training

+

- **Multi-task learning (MTL):** an efficient approach for jointly training models to perform multiple related tasks all at once
- trained on NLP tasks of sentiment analysis (SST), paraphrase prediction (PARA), and semantic textual similarity analysis (STS)



Methods

Prototypical Learning:

Step 1: encode vectors in batch using BERT

$$q = f_\phi(x_i)$$

Step 2: average latent vectors of same class to form "prototype" for each class k

$$c_k = \frac{1}{|S_k|} \sum_{x_i \in S_k} f_\phi(x_i)$$

Step 3: calculate distance to each prototype for each sample in the batch

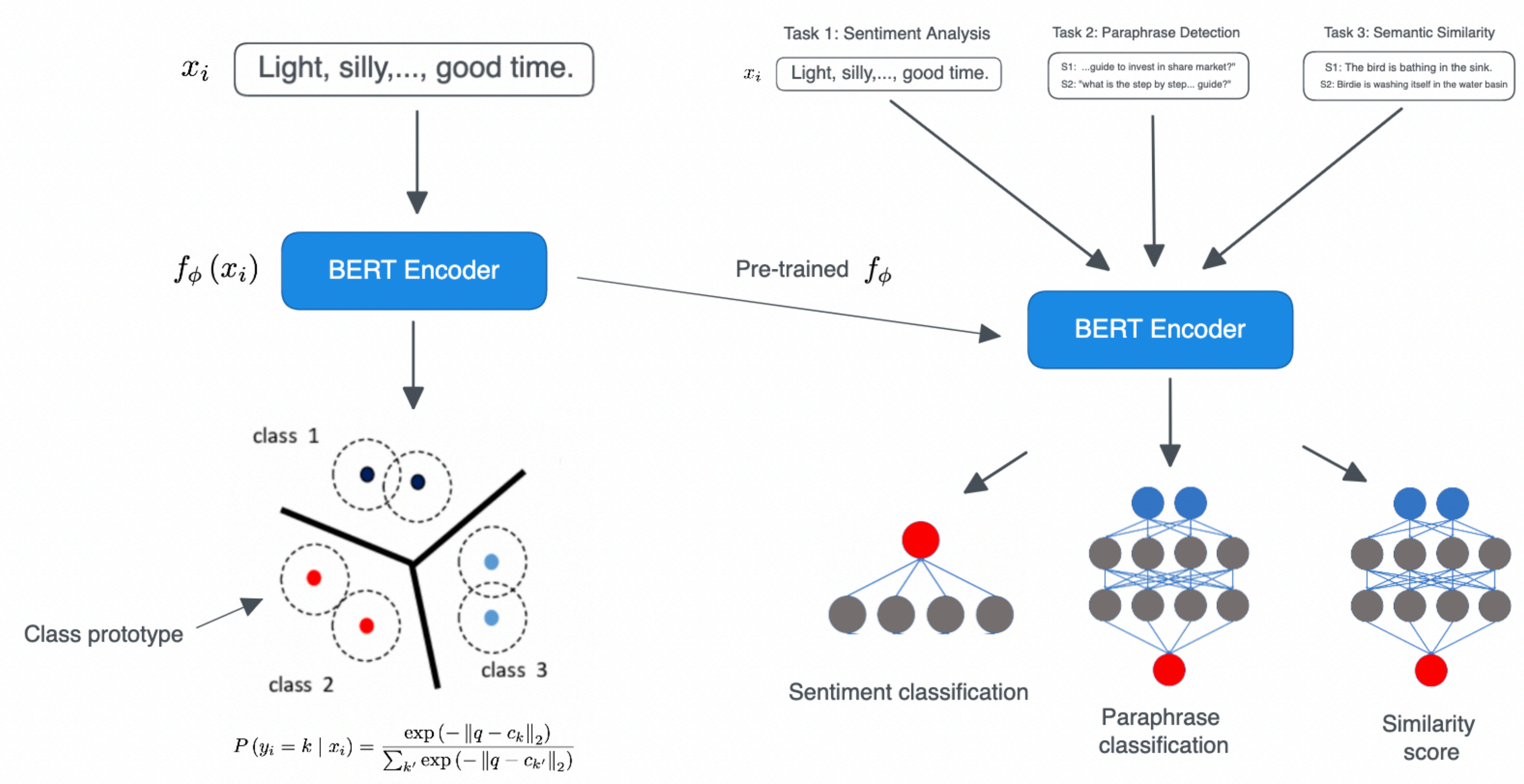
$$-\|q - c_k\|_2$$

Step 4: apply softmax to form probability distribution

$$P(y_i = k \mid x_i) = \frac{\exp(-\|q - c_k\|_2)}{\sum_{k'} \exp(-\|q - c_{k'}\|_2)}$$

Phase 1: Prototypical Pre-training

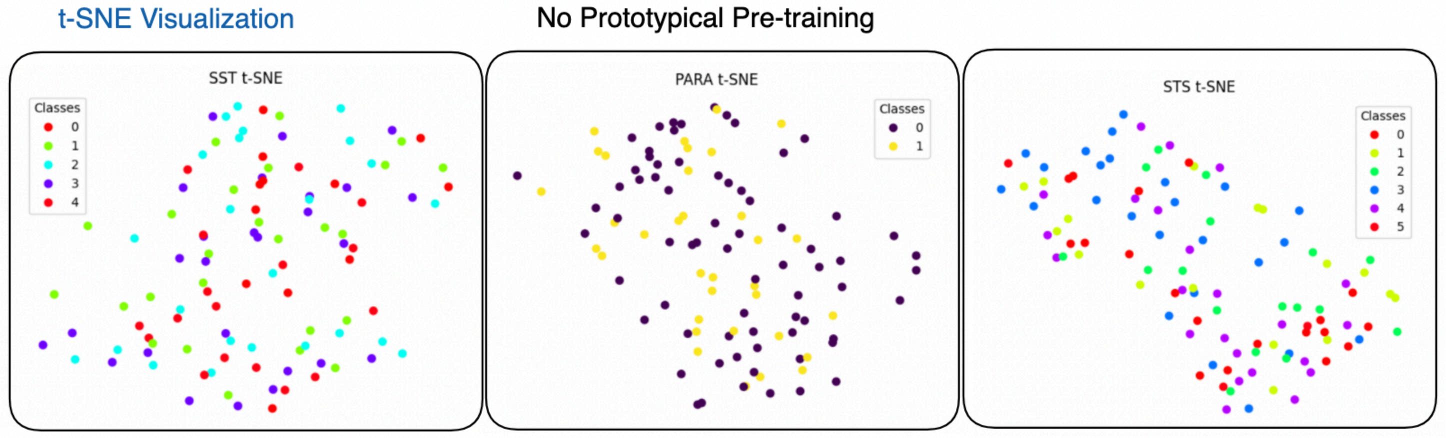
Phase 2: Fine-Tuning



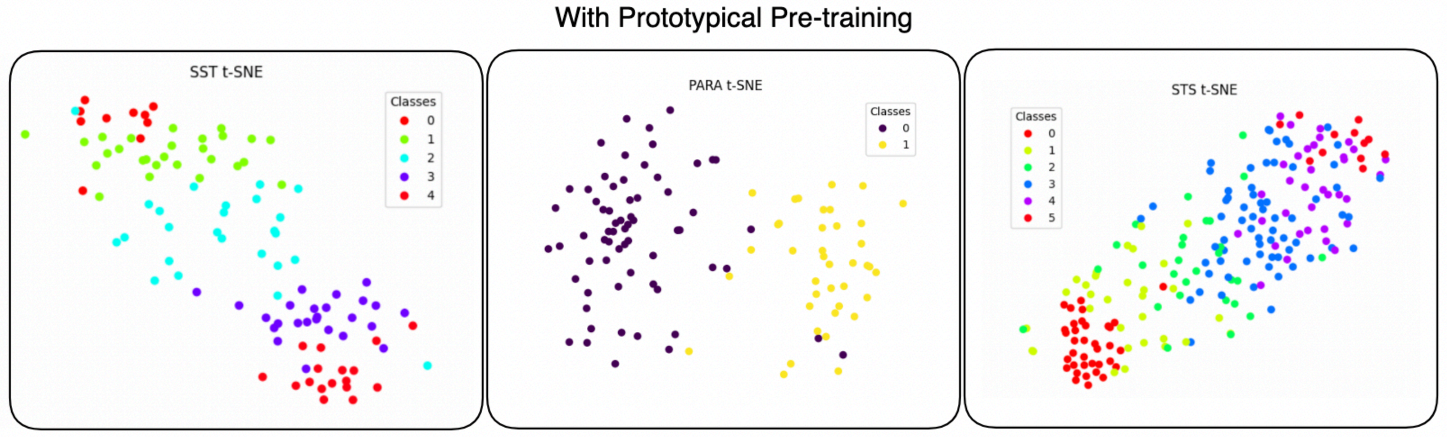
Analysis

t-SNE Visualization

No Prototypical Pre-training



With Prototypical Pre-training



Results

Table 1: Dev set evaluations for all experiments and methods for each task and overall (average) [2].

Method	SST Acc (%)	PARA Acc (%)	STS Corr	Overall
Baseline	32.7	64.8	0.264	0.413
Unfrozen Baseline	49.0	78.2	0.383	0.552
Add Combination *	35.1	62.5	0.290	0.422
Mult Combination *	48.7	69.2	0.367	0.515
Concat Combination *	49.0	66.4	0.289	0.481
Cosine Similarity STS	49.0	75.9	0.702	0.650
SST Weighted *	50.2	62.9	0.465	0.532
PARA Weighted *	32.1	72.7	0.369	0.472
STS Weighted *	29.6	63.9	0.571	0.502
Shared Layers *	47.9	69.4	0.259	0.477
Prototypical Pre-trained	**49.8**	**77.9**	**0.735**	**0.670**

Table 2: Test set evaluations for prototypical pre-trained method.

Method	SST Acc (%)	PARA Acc (%)	STS Corr	Overall
Prototypical Pre-trained	**52.4**	**78.0**	**0.714**	**0.672**

Conclusion

- We introduce the concept of prototypical pre-training for producing a learned feature space that is robust and performant across multiple tasks in downstream fine-tuning.
- The proposed method (0.670 avg) performed the best over all other methods tested, including the supervised baseline (0.552 avg).
- Future work includes theoretical analysis, different variants of prototypical learning, and weighted ensembling during pre-training and fine-tuning.

References

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. Journal of machine learning research, 9(11).

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. Advances in neural information processing systems, 30.

Mike Wu, Noah Goodman, Chris Piech, and Chelsea Finn. 2021. Prototransformer: A meta-learning approach to providing student feedback. arXiv preprint arXiv:2107.14035.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020. Gradient surgery for multi-task learning. Advances in Neural Information Processing Systems, 33:5824–5836.