Grad2Task: Improved Few-shot Text Classification Using Gradients for Task Representation

Jixuan Wang, Kuan-Chieh Wang, Frank Rudzicz, Michael Brudno

VECTOR INSTITUTE

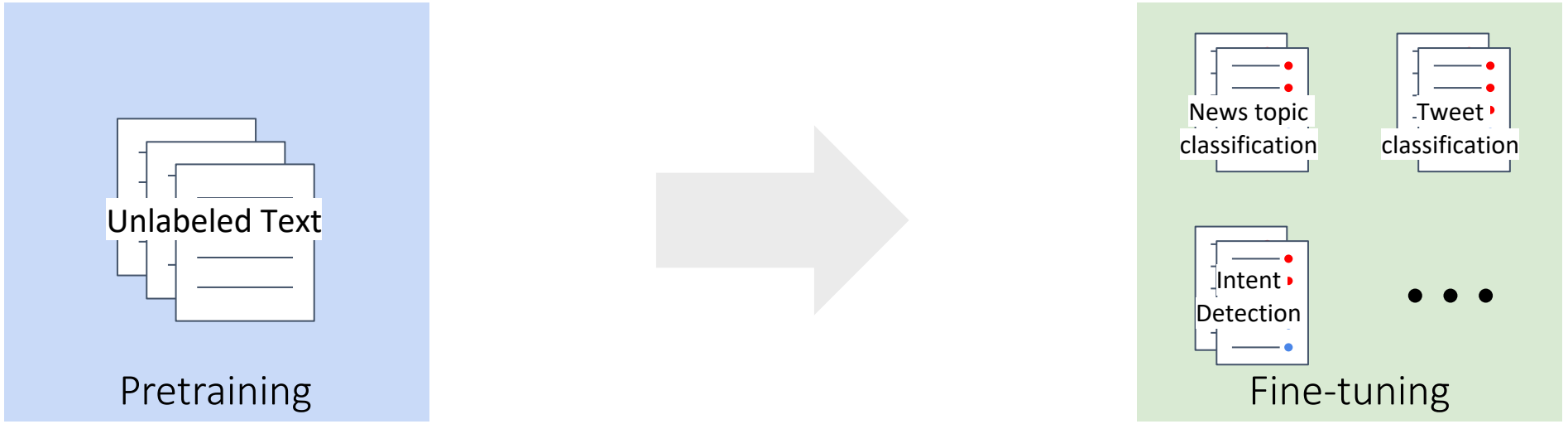
UNIVERSITY OF TORONTO

Few-Shot Text classification

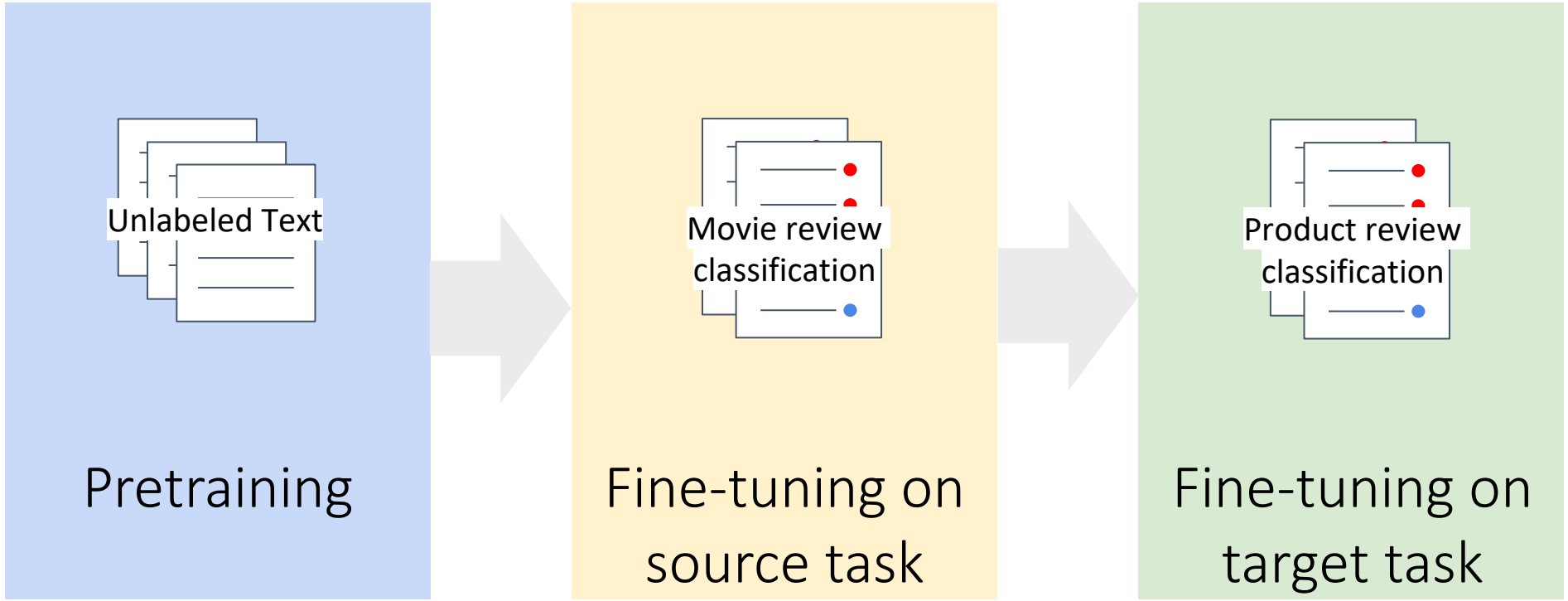
- Text classification: Predicting the label given a sentence or a pair of sentences
- For example:
  - Predicting whether a movie review is positive or negative
  - Predicting the topic of news headlines
  - Predicting whether two sentences are paraphrases of each other
  - …
- Few-shot: Only a handful of labeled examples are given for each class, i.e., 5-shot classification.

Fine-tuning Pretrained Language Models

- Transformer-based pretrained large-scale language models (LMs) have achieved tremendous success on many NLP tasks.
- Fine-tuning still requires a large amount of labeled data.

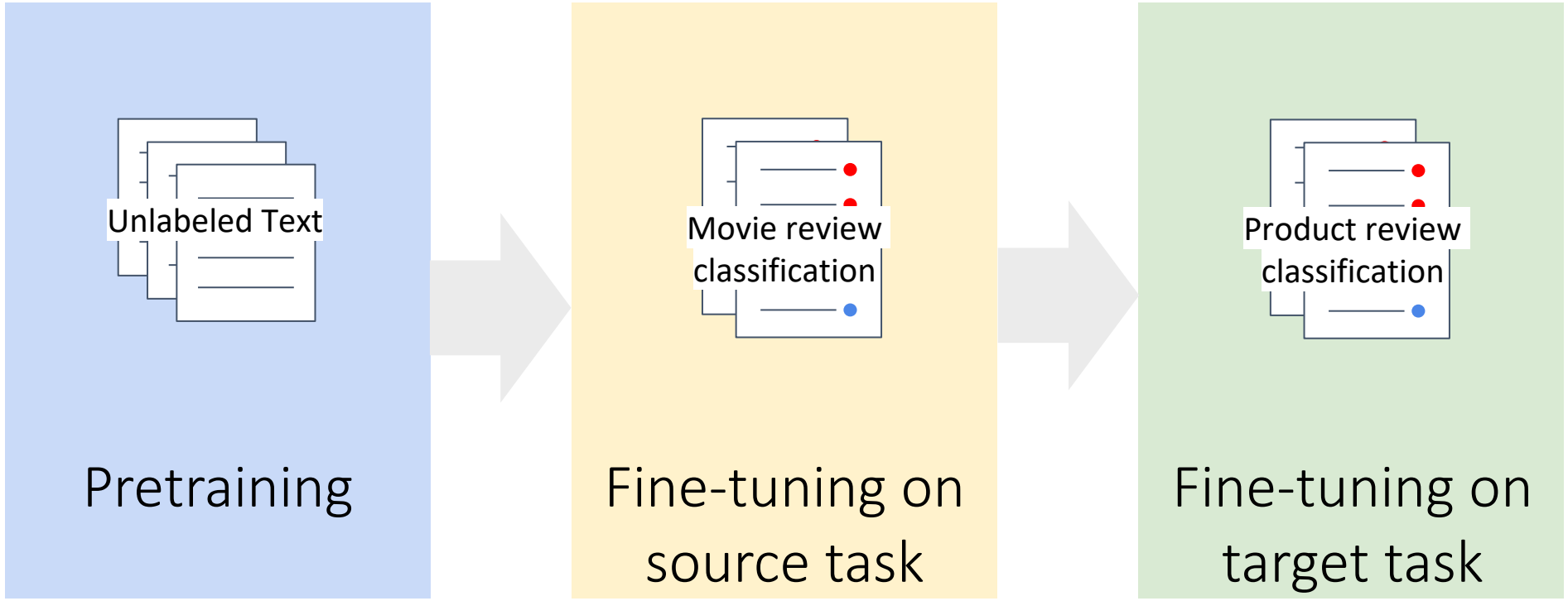


Transfer Learning

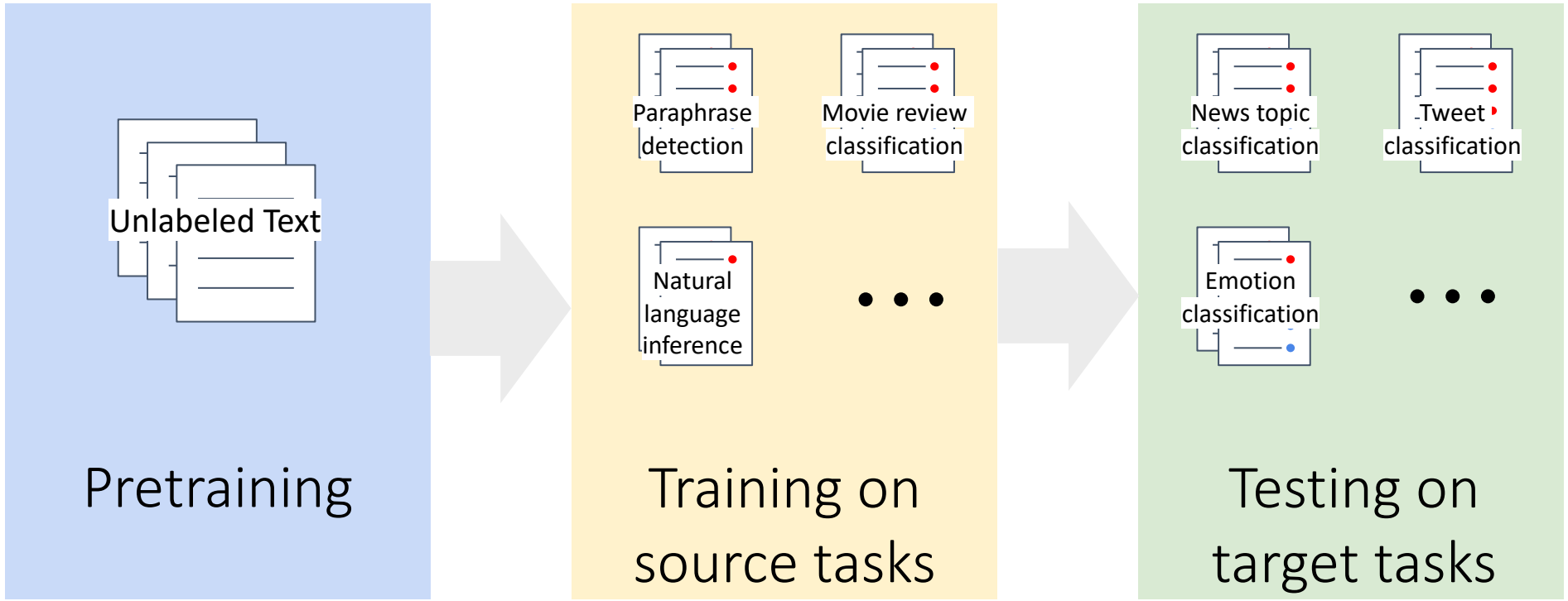


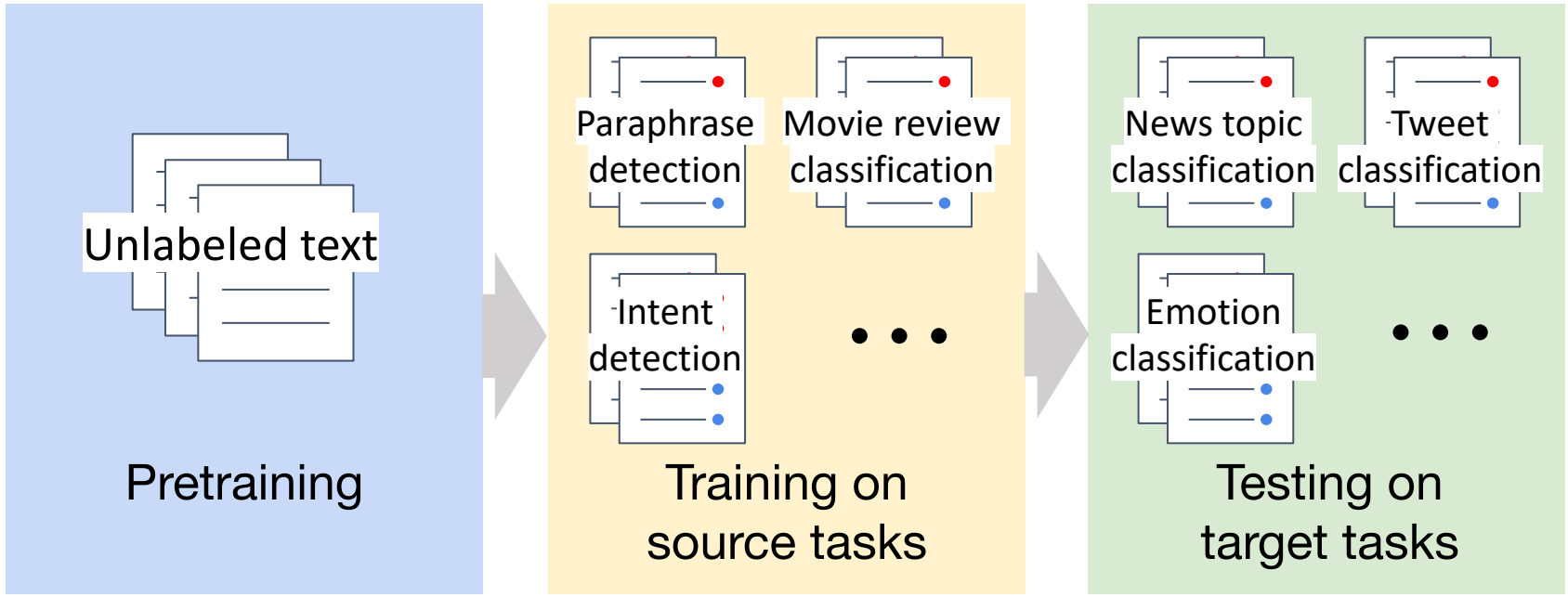
Transfer Learning

- Transfer learning typically considers moderate or many shot learning.
- Can we utilize a set of source tasks instead of just one?



Few-Shot Text Classification




Unlabeled text
Pretraining
Paraphrase detection
Movie review classification
Intent detection
Training on source tasks
News topic classification
Tweet classification
Emotion classification
Testing on target tasks

Few-Shot Text Classification by Meta-Learning

- Meta-learning: Using the experience of solving a distribution of related tasks, or episodes, to better solve new tasks.
- Challenge 1: Tasks may have different structure, e.g., different number of labels, different label semantics, etc.
  - Popular approaches, like Prototypical Network (ProtoNet), were designed to learn from a distribution of tasks with similar structure.
  - We need approaches that can learn from heterogeneous task distributions by performing more task-specific adaptation.
- Challenge 2: Tradeoff between flexibility and robustness.
  - Approaches like MAML have high flexibility but are also prone to overfitting.
  - We need a better balance between flexibility and robustness.

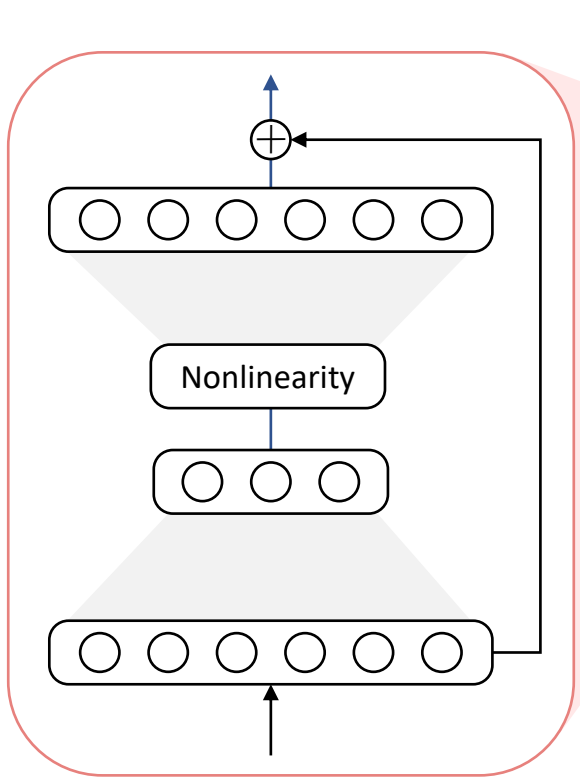
Contributions

- A novel model-based meta-learning approach based on CNAP
  - We use BERT with bottleneck adapters as the base model
  - Using gradients as task representation instead of average input representation
  - Training an adaptation network to generate parameters that modulate the base model conditioned on the task representation
  - Fixed-cost computation during testing; better balance between flexibility and robustness

- Outperform other transfer learning and meta-learning approaches

Requeima, James, et al. "Fast and flexible multi-task classification using conditional neural adaptive processes." *Advances in Neural Information Processing Systems* 32 (2019): 7959-7970.

Base Model: BERT with bottleneck adapters



Output Sequence

Linear Layer

Bottleneck Adapter

Fully-Connected Layer

Bottleneck Adapter

Self-Attention Layer

Transformer Layer

Bottleneck Adapter

Fully-Connected Layer

Bottleneck Adapter

Self-Attention Layer

Transformer Layer

Base Model
(BERT + Adapters + Linear)

Input Sequence

Learnable Parameters

Frozen Parameters

Houlsby, Neil, et al. "Parameter-efficient transfer learning for NLP." *International Conference on Machine Learning*. PMLR, 2019.

Approach Overview

- Key idea: Use a separate model to generate adaptation parameters based on the task representation and modulate the base model with the generated parameters

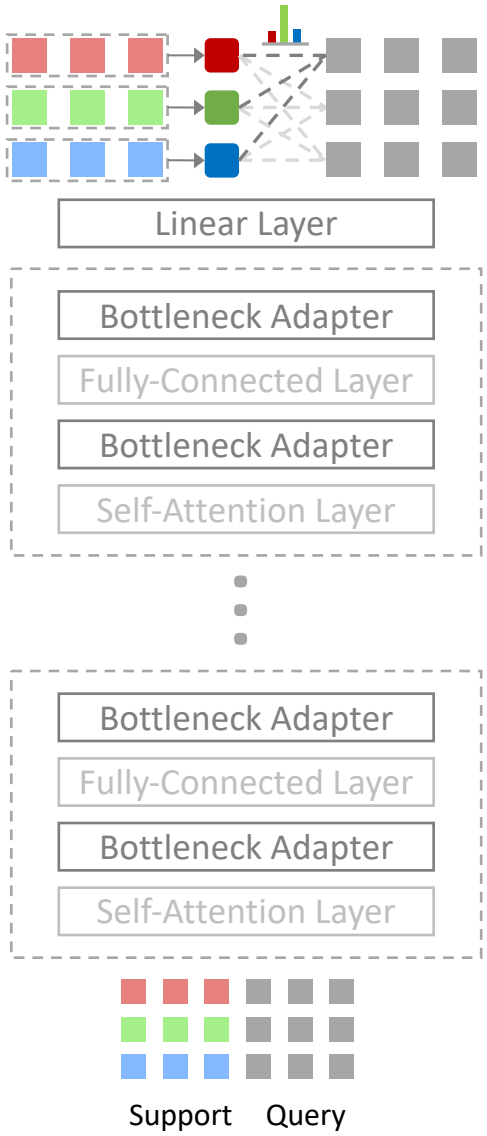
- Stage 1: Train the base model as a prototypical network (ProtoNet)

- Stage 2: Train an adaptation network to modulate the base model according to the task representation

- Both stages use episodic training.
  - Support set for adaptation; query set for loss calculation.

Prototypical Network

Stage 1



1. Build centroids:

$$\mu^c = \frac{1}{|S^c|} \sum_{x_i \in \mathcal{S}^c} f(x_i; \theta, \alpha, \omega)$$

$$S^c = \{(x_i, y_i) \,|\, (x_i, y_i) \in \mathcal{S}, y_i = c\}$$

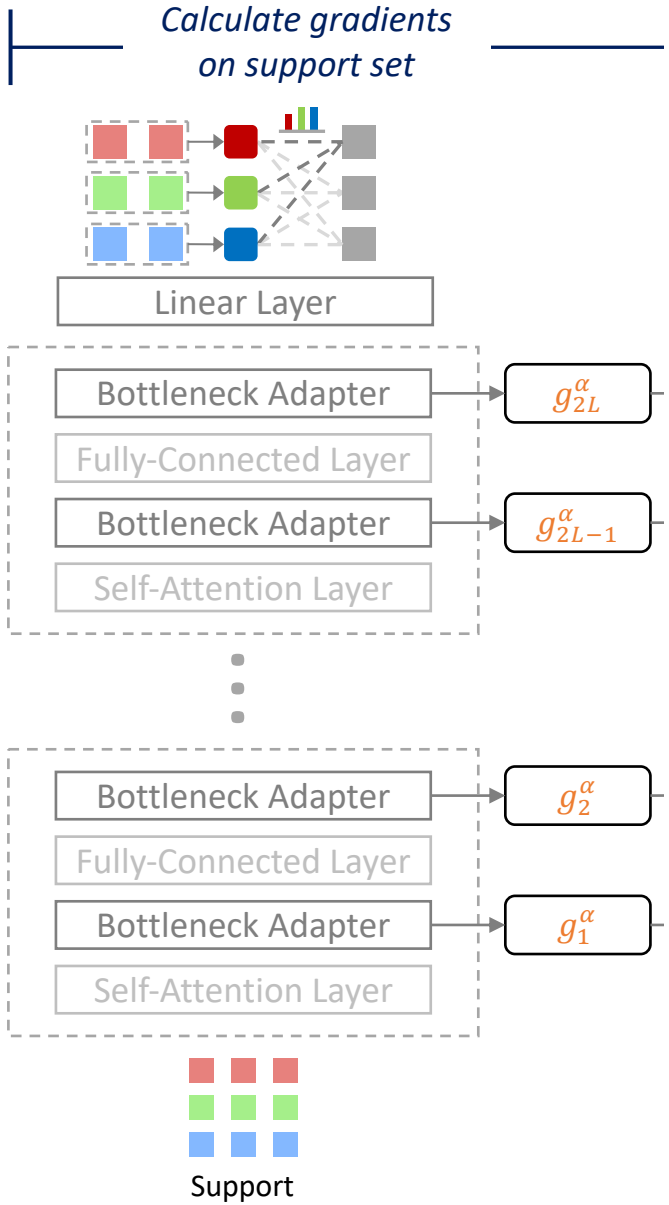
2. Apply Softmax over the Euclidean distances:

$$p^{\text{base}}(y = c \,|\, x) = \text{softmax}(\text{euc}(f(x; \theta, \alpha, \omega), \mu^c))$$

3. Calculate cross entropy loss on the query set (ProtoNet loss):

$$\ell_{pn}(t) = \frac{1}{|\mathcal{Q}|} \sum_{x_i \in \mathcal{Q}} -\log p^{\text{base}}(y_i \,|\, x_i)$$

Stage 2



Fisher Information Matrix (FIM):

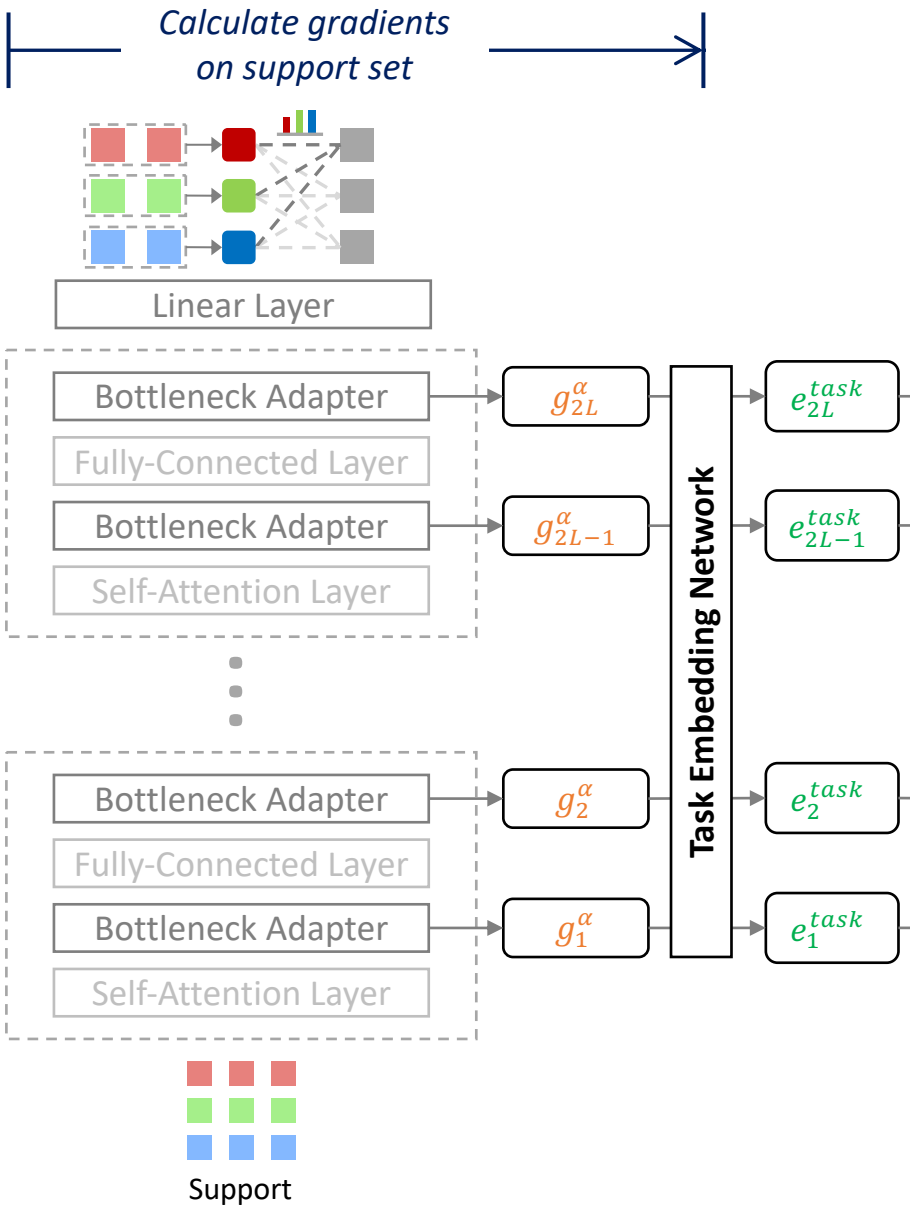
$$F_{\Theta} = \mathbb{E}_{x,y\sim\hat{p}(x)p^{\text{base}}(y\,|\,x)} \nabla_{\Theta} \log p^{base}(y\,|\,x) \nabla_{\Theta} \log p^{base}(y\,|\,x)^{T}$$

Only use the diagonal values and those values corresponding to the adapter parameters.

- Inspired by Task2Vec, a task embedding approach based on the FIM.
- Gradients can capture information from both input and output.

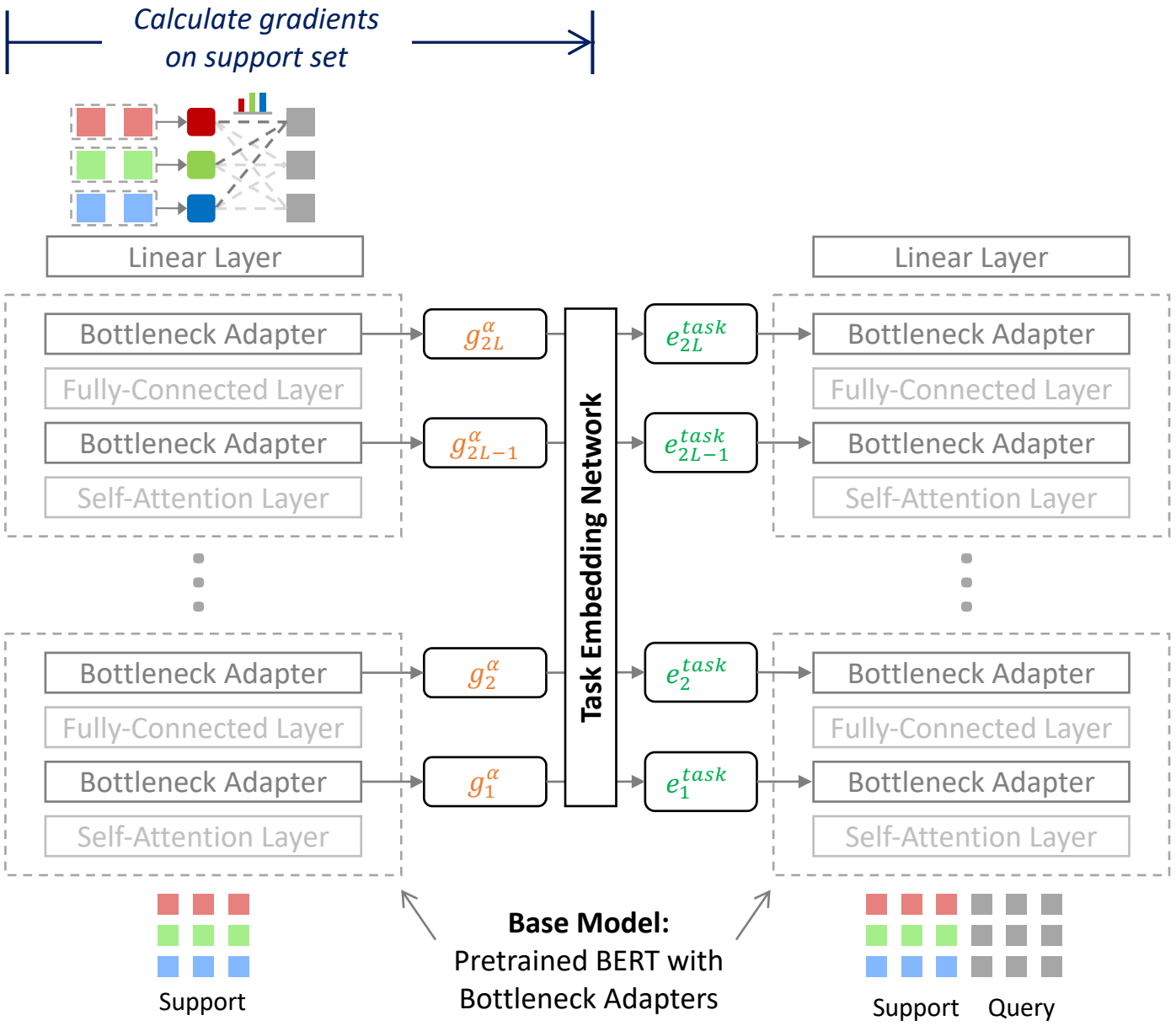
Achille, Alessandro, et al. "Task2vec: Task embedding for meta-learning." *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019.

Stage 2

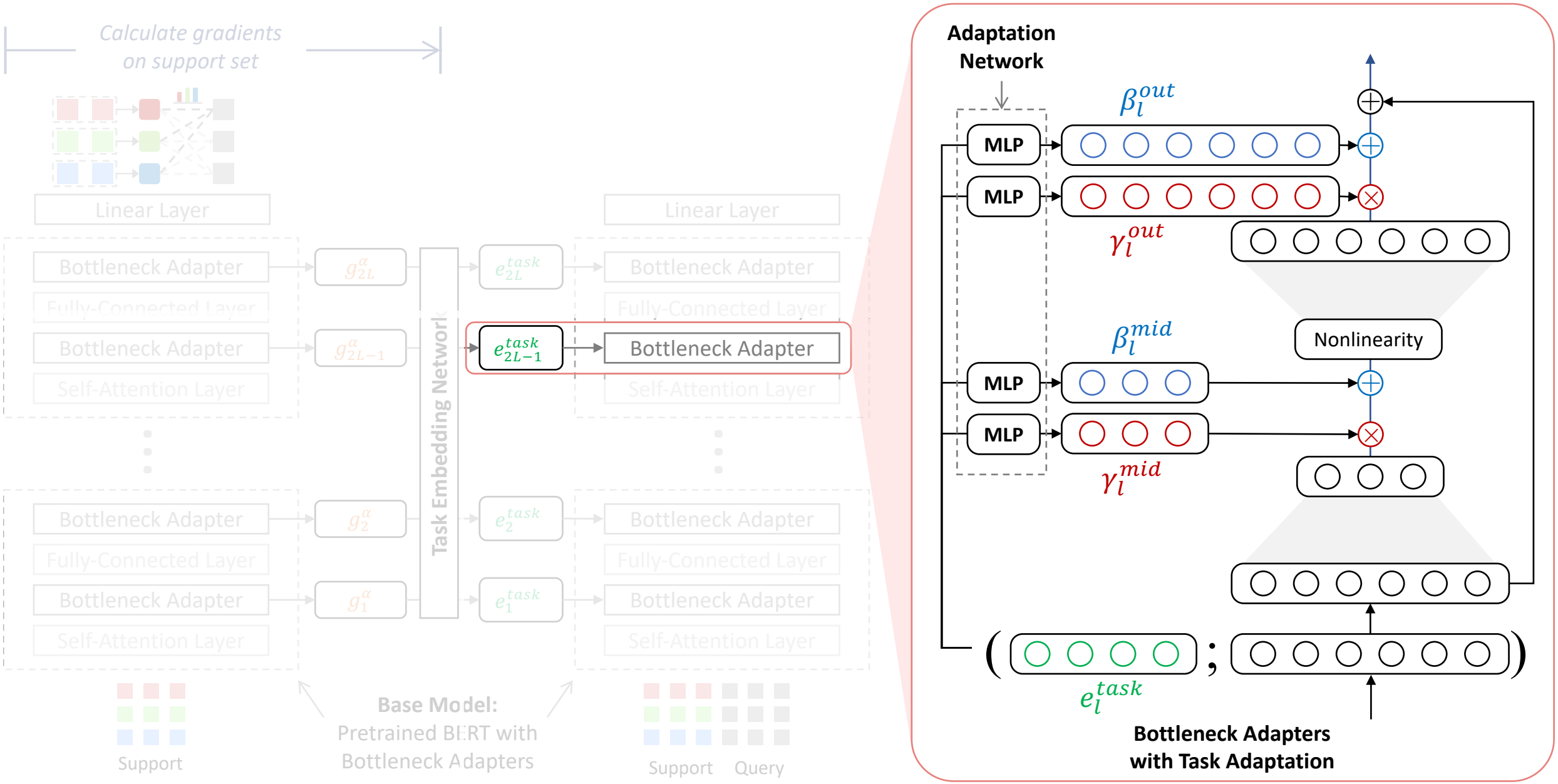


Use a RNN as the task embedding network

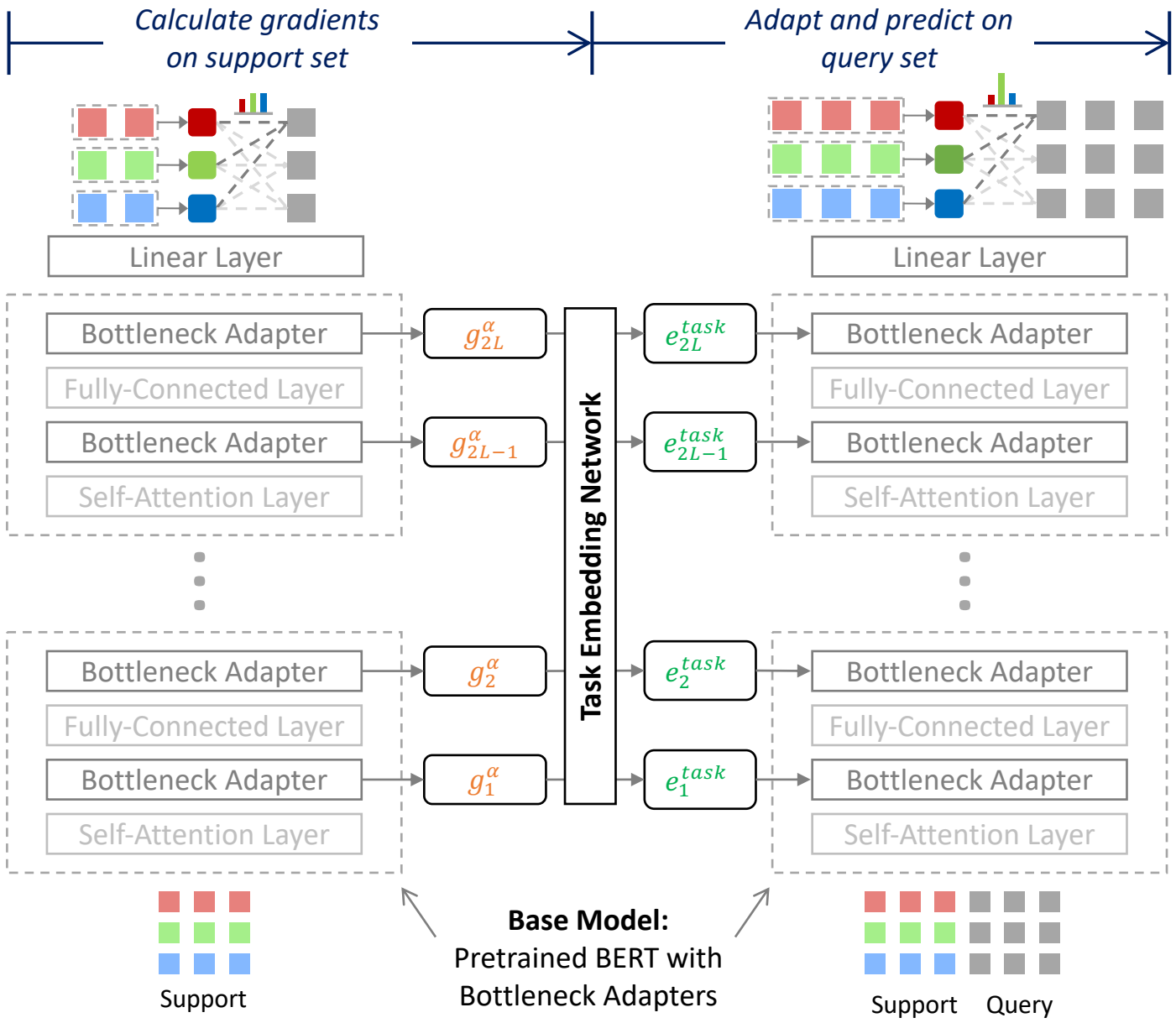
Stage 2



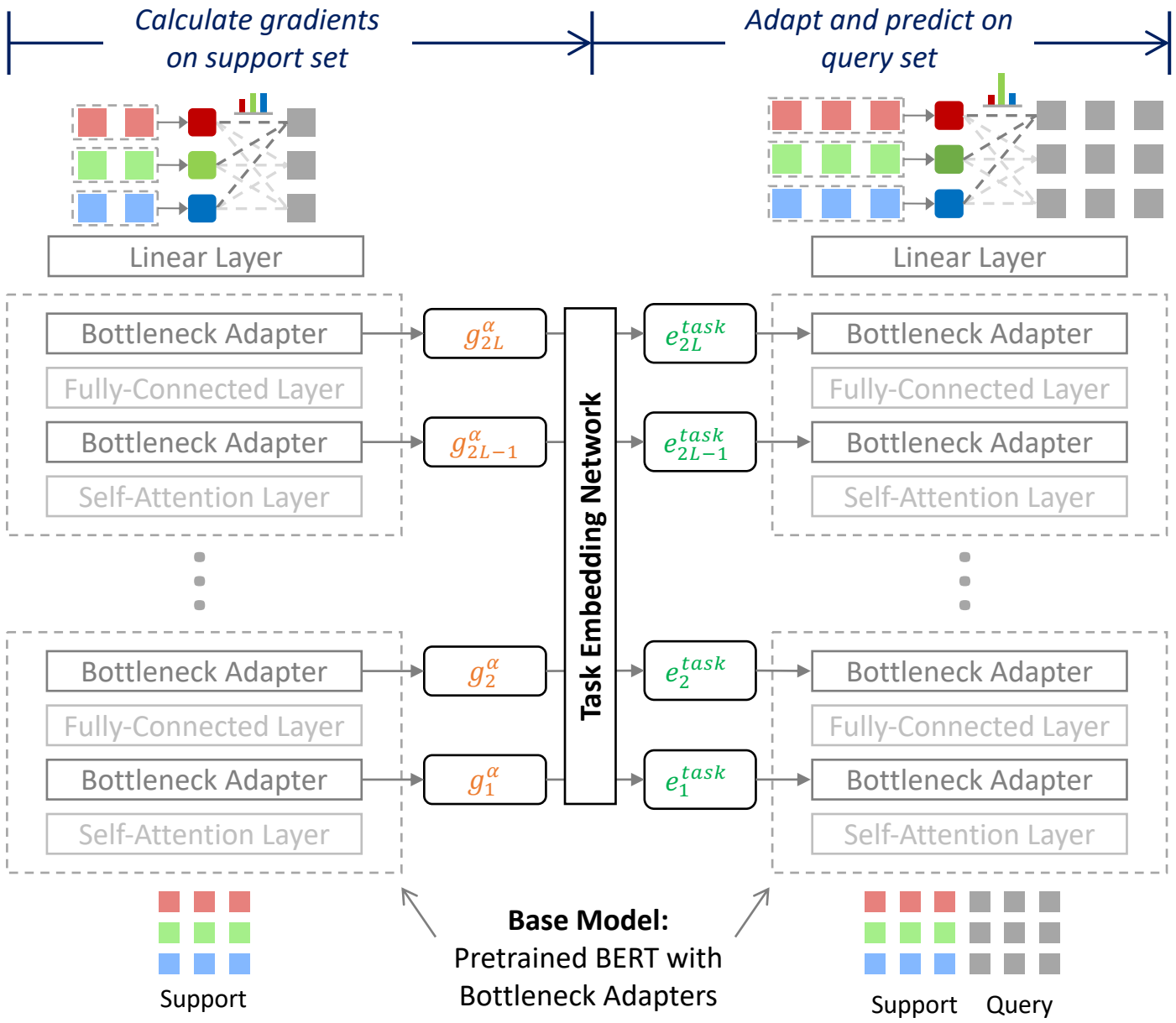
Task embeddings are used as input to generate adaptation parameters, which are applied on each adapter.


Calculate gradients on support set
Linear Layer
Bottleneck Adapter
Fully-Connected Layer
Bottleneck Adapter
Self-Attention Layer
Task Embedding Network
Base Model:
Pretrained BERT with
Bottleneck Adapters
Support
Support
Query
Adaptation Network
MLP
MLP
MLP
MLP
Nonlinearity
Bottleneck Adapters with Task Adaptation

Stage 2



After adaptation, calculate the ProtoNet loss on the query set and do backpropagation to update the task embedding network and adaptation network.



Compared with fine-tuning based approach: A shared model for all tasks; no need to tune hyperparameters for each task.

Compared with MAML: Fixed-cost adaptation during testing; no need to tune hyperparameter during testing.

Compared with ProtoNet: Task-specific adaptation.

Experiments: Meta-Training Datasets

We follow the evaluation process of Leopard.

Dataset	Task	Labels	#Training	#Validation
MRPC [15]	Paraphase detection	"paraphase", "not paraphase"	3668	409
QQP [21]		"paraphase", "not paraphase"	363846	40430
QNLI [50]	NLI	"entailment", "not entailment"	104743	5463
RTE [13]		"entailment", "not entailment"	2490	277
SNLI [10]		"contradiction", "entailment", "neutral"	549367	9824
MNLI [51]		"contradiction", "entailment", "neutral"	392702	19647
SST-2 [41]	Movie review classification	"negative", "positive"	67349	872

Trapit Bansal, Rishikesh Jha, and Andrew McCallum. "Learning to Few-Shot Learn Across Diverse Natural Language Classification Tasks." *Proceedings of the 28th International Conference on Computational Linguistics*. 2020.

Experiments: Meta-Testing Datasets

Dataset	Task	#Test Size	Labels
airline	Sentiment classification on tweets about airline	7319	"neutral", "negative", "positive"
rating_kitchen	Product rating classification on Amazon	7379	"4", "2", "5"
disaster	Classifying whether tweets are relavant to disasters	5430	"not relevant", "relevant"
emotion	Emotion classification	20000	"enthusiasm", "love", "hate", "neutral", "worry", "anger", "fun", "happiness", "boredom", "sadness", "surprise", "empty", "relief"
political_audience	Classifying the audience/bias/message of social media messages from politicians	996	"national", "constituency"
political_bias		1287	"partisan", "neutral"
political_message		428	"personal", "policy", "support", "media", "attack", "other", "information", "constituency", "mobilization"
snips*	Intent detection	700	"play music", "add to playlist", "rate book", "search screening event", "book restaurant", "get weather", "search creative work"
huffpost_10*	Category classification on news headlines from HuffPost	4000	"politics", "entertainment", "travel", "wellness", etc.
yelp*	Business rating classification on Yelp	10000	"1", "2", "3", "4", "5"

Few-Shot Text Classification Results

Simple fine-tuning | Multi-task fine-tuning | MAML-based | ProtoNet w/ BERT | PN-BERT + bottleneck adapters | Ours

	Model	BERT*		MT-BERT*		Leopard		PN-BERT		PN-BN		Grad2Task	
#		Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
	airline	42.76	13.50	46.29	12.26	54.95	11.81	65.39	12.73	65.33	7.95	**70.64**	3.95
	disaster	**55.73**	10.29	50.61	8.33	51.45	4.25	54.01	2.90	53.48	4.76	55.43	5.89
	emotion	9.20	3.22	9.84	2.14	11.71	2.16	11.69	1.87	12.52	1.32	**12.76**	1.35
	political_audience	51.89	1.72	51.53	1.80	52.60	3.51	**52.77**	5.86	51.88	6.37	51.28	5.74
	political_bias	54.57	5.02	54.66	3.74	60.49	6.66	58.26	10.42	**61.72**	5.65	58.74	9.43
4	political_message	15.64	2.73	14.49	1.75	15.69	1.57	17.82	1.33	20.98	1.69	**21.13**	1.97
	rating_kitchen	34.76	11.20	36.77	10.62	50.21	9.63	**58.47**	11.12	55.99	9.85	57.09	9.74
	huffpost_10	-	-	-	-	11.8	1.41	14.97	1.69	16.81	2.52	**18.5**	2
	snips	-	-	-	-	21.36	2.7	28.99	3.93	46.29	3.91	**52.51**	2.68
	yelp	-	-	-	-	36.95	2.98	42.84	2.66	42.64	2.93	**43**	3.55
	Average	-	-	-	-	36.72	4.67	40.52	5.45	42.76	4.70	**44.11**	4.63

“Same/Different” Experiment Results

Classifying whether two few-shot tasks are sampled from the same dataset or not.

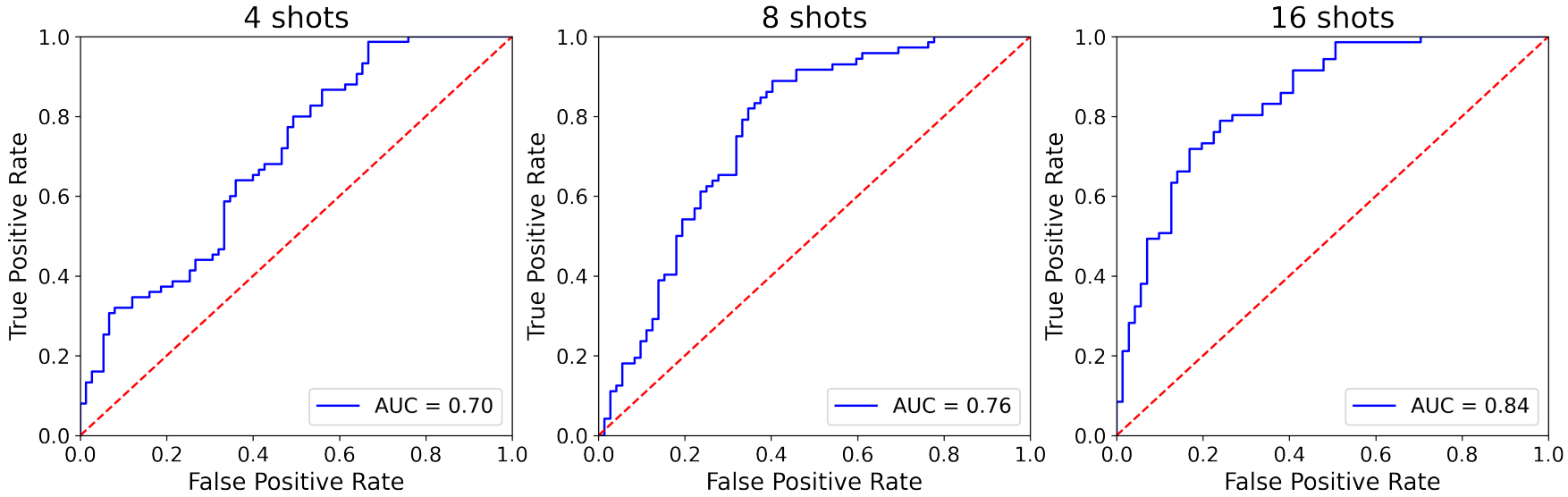


Figure 2: Results on the same/different experiments. Each column shows the results of classifying pairs of dataset with a certain number of shots.

Ablation Study

Table 2: Ablation results.

Model	Mean Acc.
Grad2Task w/ Gradients	**45.99**
ProtoNet Longer Training	45.10
Grad2Task w/ X	45.66
Grad2Task w/ X&Y	45.16
Grad2Task Adapt All	44.57
Grad2Task w/ Pretrained TaskEmb	45.68
Hypernetwork	44.79

Future Work

- Better ways to utilize gradient-based task representations for meta-learning
- Extend our framework to other kinds of NLP tasks
- Moderate-shot learning?

Grad2Task: Improved Few-shot Text Classification Using Gradients for Task Representation

Jixuan Wang, Kuan-Chieh Wang, Frank Rudzicz, Michael Brudno
jixuan@cs.toronto.edu

VECTOR INSTITUTE

UNIVERSITY OF TORONTO