Co-Training Improves Prompt-Based Learning for Large Language Models

Hunter Lang, Monica Agrawal, Yoon Kim, David Sontag

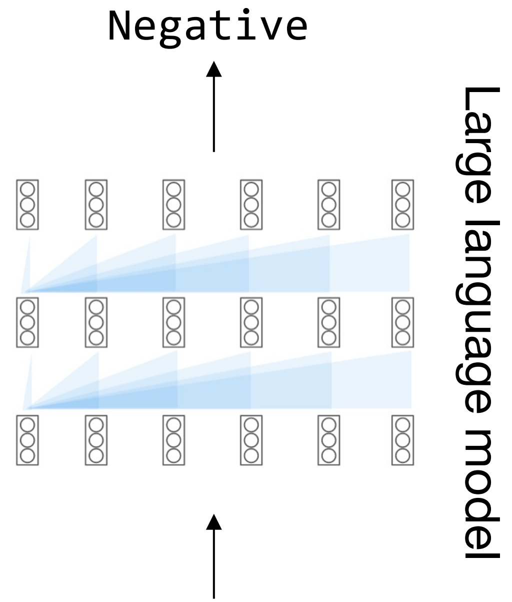
Prompt-Based Learning

- *In-context learning*
- Format `example` with `template`

```
Review: the acting was subpar.
Positive or Negative?
```

Prompt-Based Learning

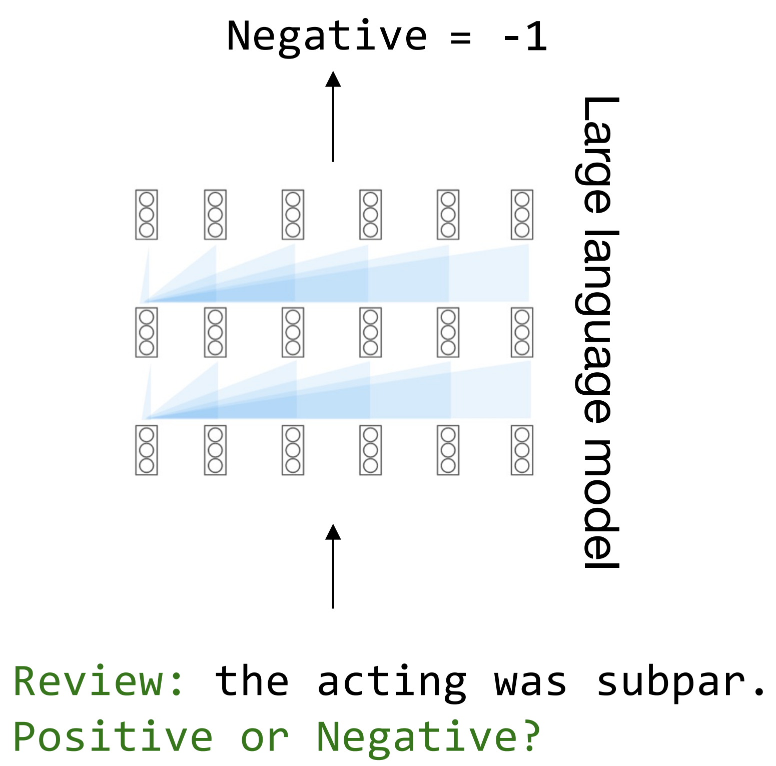
- *In-context learning*
- Format example with template
- Predict the next word



Review: the acting was subpar.
Positive or Negative?

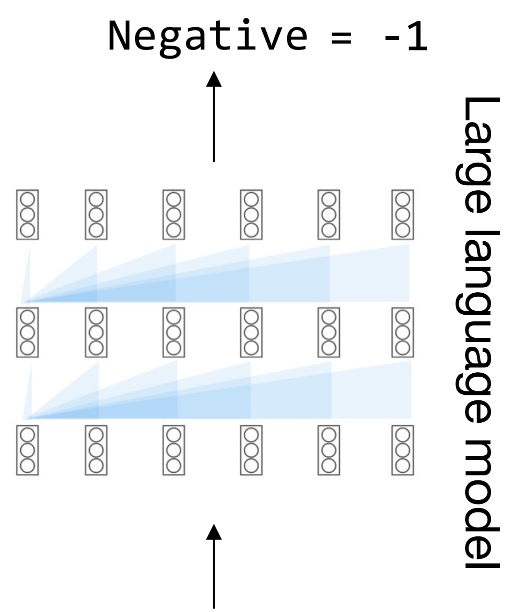
Prompt-Based Learning

- *In-context learning*
- Format example with template
- Predict the next word
- Map to a label (Negative = -1)



Prompt-Based Learning

- *In-context learning*
- Format example with `template`
- Predict the next word
- Map to a label (`Negative` = -1)
- Optional: also give a `labeled example`




```
Review: this movie was great.
Positive or Negative? Positive

Review: the acting was subpar.
Positive or Negative?
```

Prompt-Based Learning: **Problems**

- Hard to deploy (expensive APIs, data restrictions)
- Can significantly underperform supervised learning
- Sensitive to `labeled examples` and their ordering.

Prior Work: **Calibrate Before Use (CBU)**

- Improve performance by **renormalizing** output probabilities.
- Estimate the rescaling in a **data-free** manner using null inputs (“N/A”, “”, etc.)
- Makes GPT-3 less sensitive to `example ordering` and **improves accuracy**.

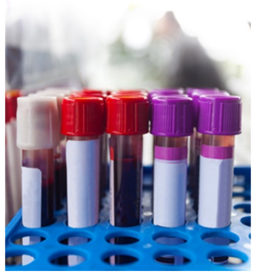
Prior Work: **Calibrate Before Use (CBU)**

- Improve performance by **renormalizing** output probabilities.
- Estim ”, “”, etc.)
- Makes GPT-3 less sensitive to `example ordering` and **improves accuracy**.

Can we do better if we have a large amount of **unlabeled** data?

Background: Co-Training [Blum and Mitchell '98]

- A semi-supervised approach for leveraging unlabeled data.
- Pair of models are trained over different "views" of the same underlying data.

View	$\phi_0(X)$	$\phi_1(X)$
Model	h_0	h_1
		
	Lab tests	X-ray

Background: Co-Training [Blum and Mitchell '98]

- A semi-supervised approach for leveraging unlabeled data.
- Pair of models are trained over different "views" of the same underlying data.

View	$\phi_0(X)$	$\phi_1(X)$
Model	h_0	h_1

- The two models $h_0(\phi_0(X))$ and $h_1(\phi_1(X))$ are **iteratively trained** on confidently-labeled data points from the **other model.**

Background: Co-Training [Blum and Mitchell ‘98]

Catch: need a good initial model to start the co-training process.

(Most) Prior work: use a small amount of labeled data to train initial model.

Our work: use a zero- or few-shot LLM as the initial model

- The two models $h_0(\phi_0(X))$ and $h_1(\phi_1(X))$ are **iteratively trained** on confidently-labeled data points from the **other model.**

Our Work: **Co-Training + Prompting**

- Combine *co-training* (Blum and Mitchell, 1998) with prompt-based learning

- Few-shot or zero-shot LLM is the initial model

Our Work: **Co-Training + Prompting**

- Combine *co-training* (Blum and Mitchell, 1998) with prompt-based learning

- Few-shot or zero-shot LLM is the initial model

- **Key idea:** refine the Large Language Model (GPT-3 / T0) together with a much smaller model (BERT, DeBERTa).

Our Work: **Co-Training + Prompting**

- Combine *co-training* (Blum and Mitchell, 1998) with prompt-based learning

- Few-shot or zero-shot LLM is the initial model

- **Key idea:** refine the Large Language Model (GPT-3 / T0) together with a much smaller model (BERT, DeBERTa).

- **Improves** few-shot and zero-shot **performance** for GPT-3 and T0.

Our Work: **Co-Training + Prompting**

- Combine *co-training* (Blum and Mitchell, 1998) with prompt-based learning
- Few-shot or zero-shot LLM is the initial model
- **Key idea:** refine the Large Language Model (GPT-3 / T0) together with a much smaller model (BERT, DeBERTa).
- **Improves** few-shot and zero-shot **performance** for GPT-3 and T0.
- **Distills** the large language model into a smaller task-specific model

Our Work: **Co-Training + Prompting**

- Combine *co-training* (Blum and Mitchell, 1998) with prompt-based

Key challenge: how do we fine-tune the LLM?
Answer: depends on the model!

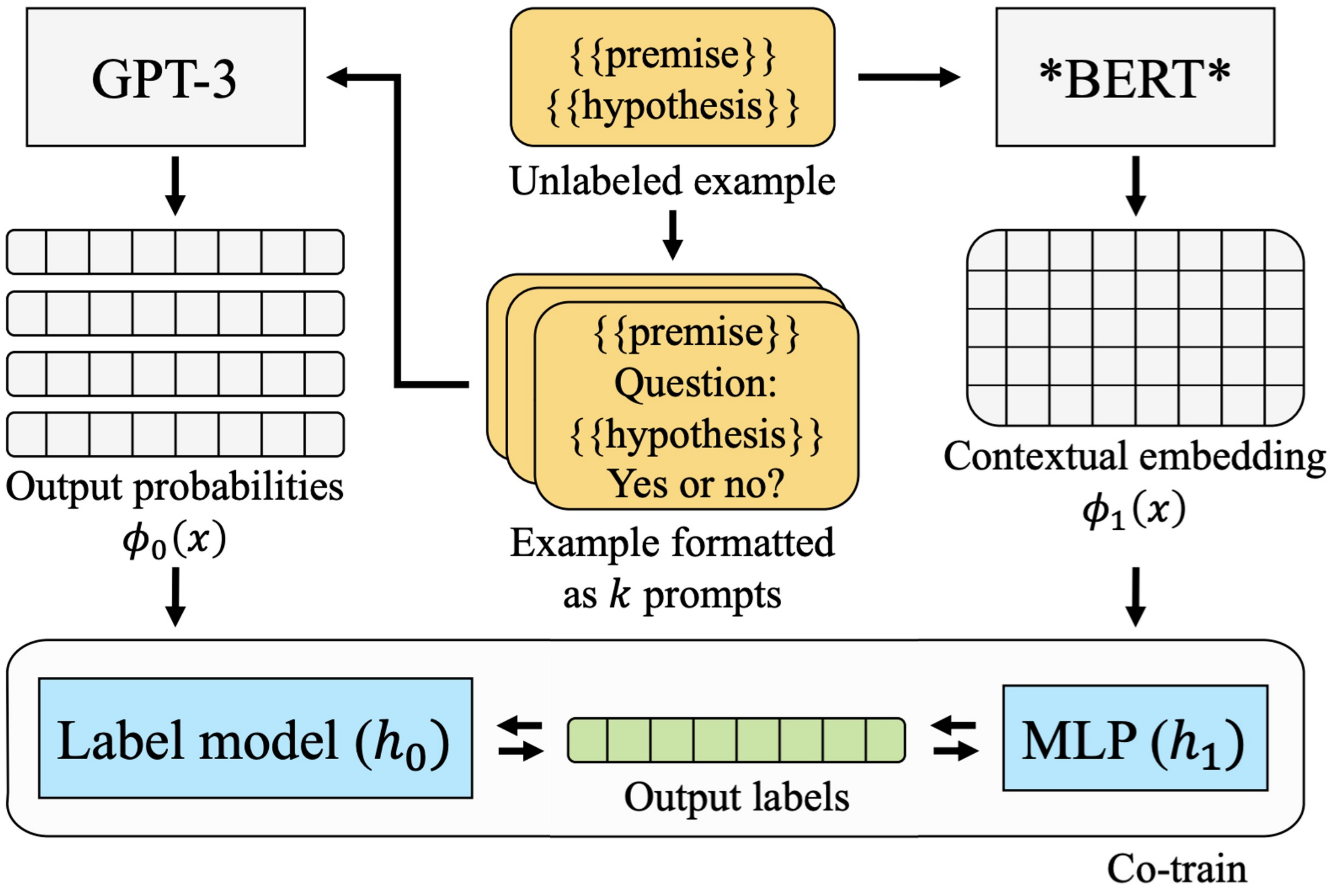
Setting #1 (GPT-3): no gradient access, output probabilities only

Setting #2 (T0): full model access, can compute gradients, but full-fine-tuning is too inefficient

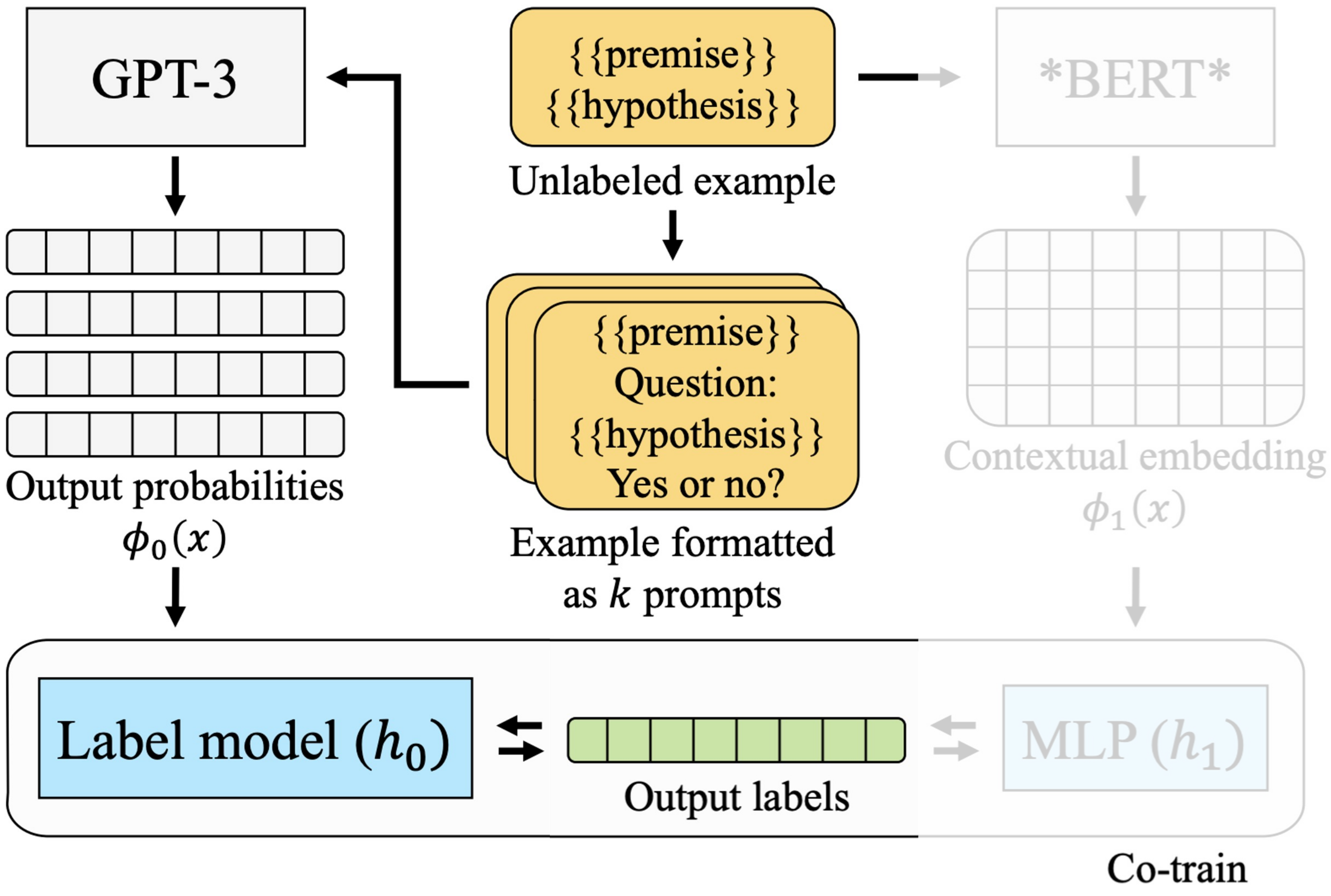
-

- **Distills** the large language model into a smaller task-specific model

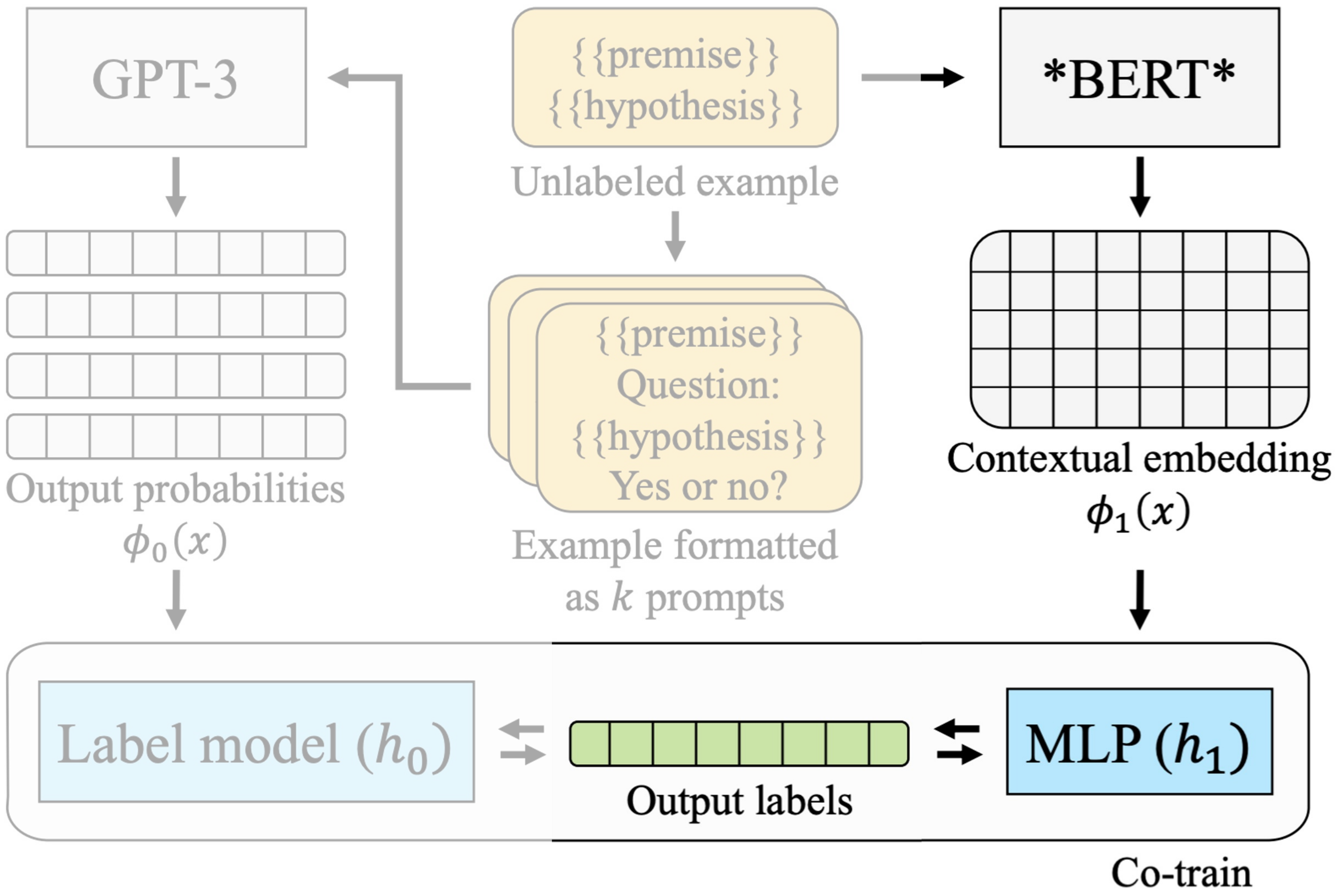
Setting #1: Few-shot GPT-3



Setting #1: Label model details



Setting #1: Other model



Setting #1: Few-shot results

Model	View	RTE (2-class)	CB (3-class)	TREC (6-class)
GPT-3 4-shot (from Zhao et al. (2021))	*	58.7 (11.9)	45.2 (19.4)	60.2 (7.6)
Calibrate Before Use (CBU) (Zhao et al., 2021)	*	60.4 (8.1)	60.7 (6.7)	69.7 (1.4)
Prompt-based FT (Gao et al., 2021)	*	52.8 (0.9)	84.4 (3.2)	54.8 (2.9)
Label Model (no co-training)	ϕ_0	62.8	76.8	77.2
Label Model $\rightarrow$ DeBERTa distillation	ϕ_1	67.2 (0.5)	81.6 (2.2)	63.3 (0.4)
Label Model + *co-training*	ϕ_0	64.9 (1.1)	83.5 (2.3)	78.3 (1.2)
DeBERTa-large + *co-training*	ϕ_1	**67.4** (2.3)	**86.2** (3.2)	**80.6** (1.1)

GPT-3 with the same # of labeled examples

LM-BFF with same # of labeled examples

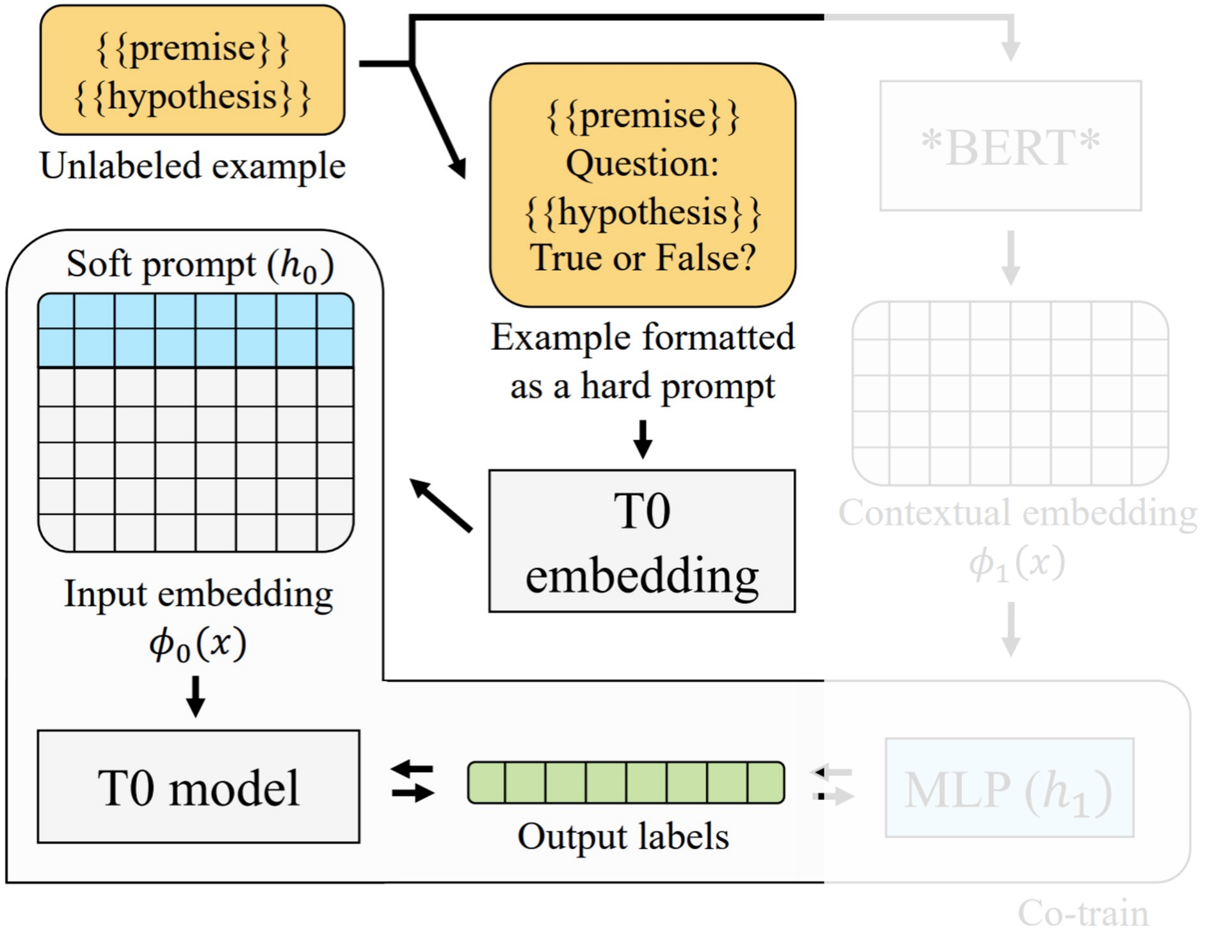
Cotrained LLM (ϕ_0) and DeBERTa (ϕ_1)

Setting #2: Co-Training with Zero-shot Learning

T0 [Sanh et al. '21]: trained on tasks converted as natural instructions ⇒ meaningful zero-shot learning performance.

$$h_0(\phi_0(X))$$

Soft prompt vectors appended to T0 word embeddings.

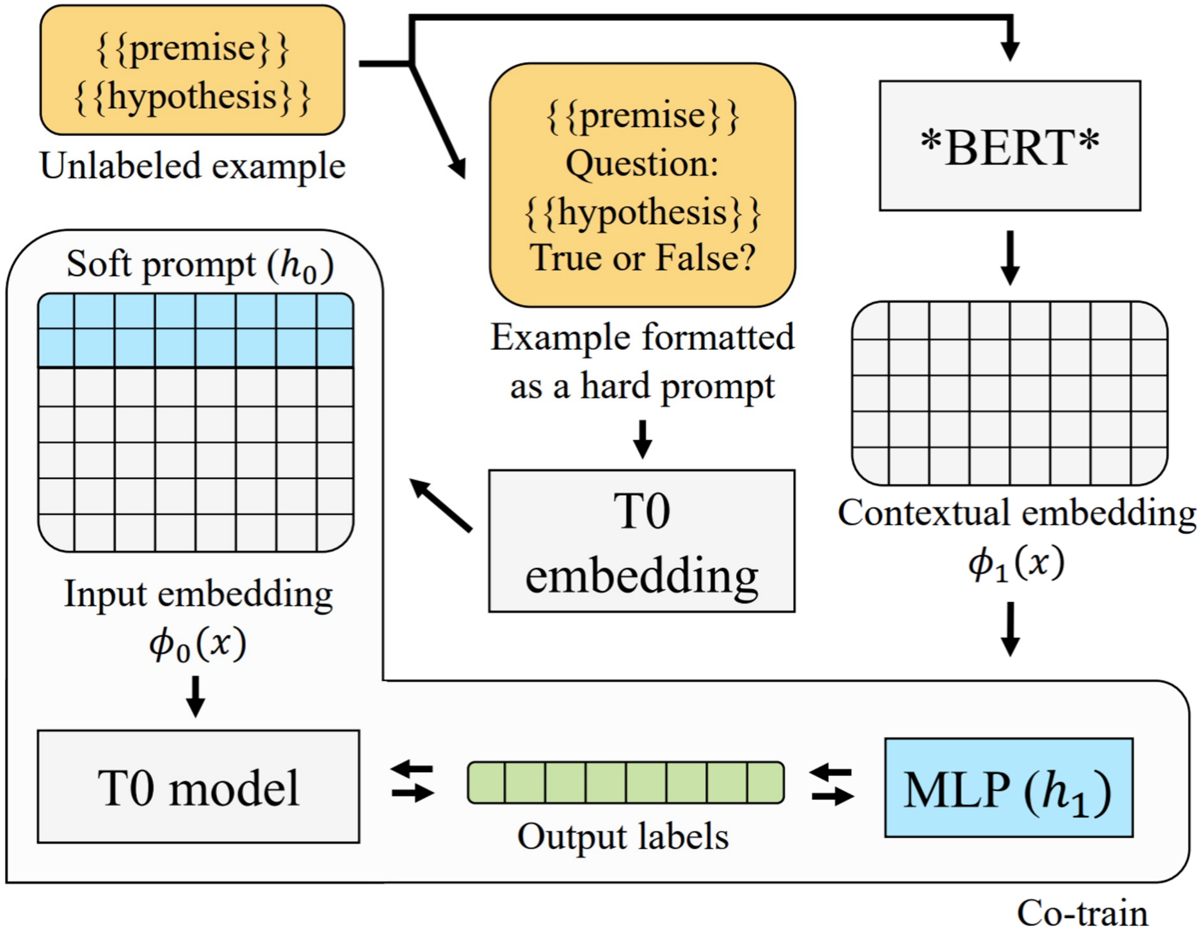


Setting #2: Co-Training with Zero-shot Learning

T0 [Sanh et al. '21]: trained on tasks converted as natural instructions ⇒ meaningful zero-shot learning performance.

$h_0(\phi_0(X))$ Soft prompt vectors appended to T0 word embeddings.

$h_1(\phi_1(X))$ DeBERTa + MLP classifier (same as before).



Setting #2: Co-Training with Zero-shot Learning

Model/Algorithm	View	RTE	CB	BoolQ
T0-3B (best) (Sanh et al., 2022)	ϕ_0	68.9	66.1	59.1
T0-3B zero-shot (no co-training)	ϕ_0	68.9	58.9	56.4
T0-3B soft prompt + *co-training*	ϕ_0	87.0	67.9	49.1
DeBERTa-large + *co-training*	ϕ_1	86.3	67.9	48.9
T0-3B soft prompt on full train	ϕ_0	90.6	80.4	86.9
DeBERTa-large on full train	ϕ_1	93.3	95.2	86.1

Best-performing T0 prompt

Cotrained LLM (ϕ_0) and DeBERTa (ϕ_1)

Summary

- *Co-Training can:*
  - **Improve** prompt-based learning by fine-tuning the LLM with another model
  - **Distill** the LLM to a smaller, task-specific model
- *Future Directions:*
  - Co-Training + Prompting with structured output spaces
  - Explore other efficient fine-tuning methods

Hall E, Poster 124