Prompting ELECTRA: Few-Shot Learning with Discriminative Pre-Trained Models

Mengzhou Xia[1] **Mikel Artetxe**[2] **Jingfei Du**[2]
Danqi Chen[1] **Ves Stoyanov**[2]
[1]Princeton University [2]Meta AI
{mengzhou, danqic}@cs.princeton.edu
{artetxe, jingfeidu, ves}@meta.com

Abstract

Pre-trained masked language models successfully perform few-shot learning by formulating downstream tasks as text infilling. However, as a strong alternative in full-shot settings, discriminative pre-trained models like ELECTRA do not fit into the paradigm. In this work, we adapt prompt-based few-shot learning to ELECTRA and show that it outperforms masked language models in a wide range of tasks. ELECTRA is pre-trained to distinguish if a token is generated or original. We naturally extend that to prompt-based few-shot learning by training to score the originality of the target options without introducing new parameters. Our method can be easily adapted to tasks involving multi-token predictions without extra computation overhead. Analysis shows that ELECTRA learns distributions that align better with downstream tasks.[1]

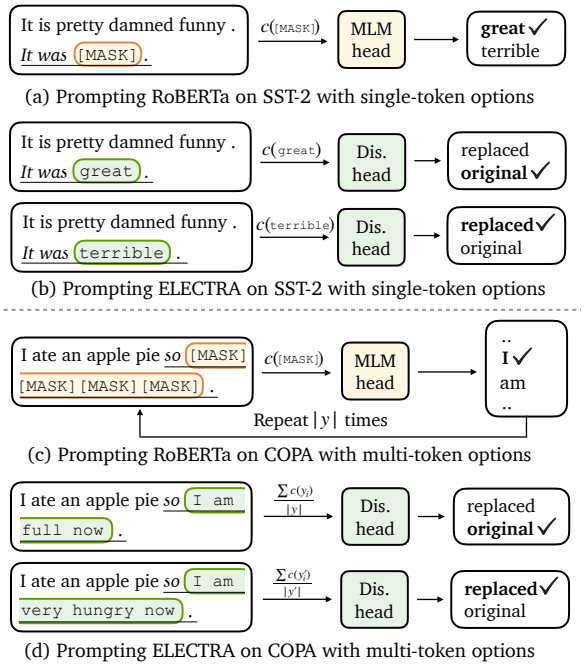

Figure 1: Prompt-based fine-tuning with RoBERTa and ELECTRA for two downstream tasks: SST-2 (Socher et al., 2013) and COPA (Roemmele et al., 2011). The underlined text is the task-specific template. $c(\cdot)$: contextualized embedding; y and y': a correct and an incorrect option, respectively.

1 Introduction

Large pre-trained language models are known to be effective zero-shot and few-shot learners when scaled (Brown et al., 2020; Artetxe et al., 2021; Rae et al., 2021). Much smaller masked language models (MLMs), like BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), can be fined-tuned with only a few examples by utilizing prompt-based fine-tuning, which updates the model to select the correct target word or option (Schick and Schütze, 2021a; Gao et al., 2021).

In this paper, we hypothesize that discriminative pre-trained models like ELECTRA (Clark et al., 2020) will make even stronger few-shot learners as alternatives to MLMs as they are pre-trained to distinguish between challenging alternatives. To test this hypothesis, we explore prompt-based learning with ELECTRA by aligning its pre-training objective—distinguishing if a single token is generated or original—with prompt-based predictions for downstream tasks. We reuse ELECTRA's discriminative head to classify the correct target word as original tokens. As an additional benefit, we can naturally adapt the approach to multi-token spans by aggregating either hidden representations or output probabilities. In contrast, MLMs require autoregressive decoding to adapt to multi-token options (Schick and Schütze, 2021b).

We propose an approach to prompting ELECTRA, as shown in Figure 1. Though trained with the same or even less computation than BERT and RoBERTa, ELECTRA turns out to be a more effective few-shot learner. It outperforms BERT and

[1]Code is available at https://github.com/facebookresearch/ELECTRA-Fewshot-Learning.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 11351 - 11361
December 7-11, 2022

RoBERTa by 10.2 and 3.1 points on average across 9 tasks with single-token options for base-sized models in the few-shot setting, and the trend prevails for large-sized models. ELECTRA also outperforms RoBERTa on 4 tasks with multi-token options. Our analysis suggests that the failing predictions from ELECTRA's generator could actually feed negatives with opposite meanings from the correct tokens to the discriminator, which strengthens ELECTRA's ability to distinguish concepts with opposite meanings for zero-shot predictions.

2 Background

2.1 Prompting Masked Language Models

MLMs such as BERT and RoBERTa are trained by masking words in inputs and maximizing the probability of original tokens that are replaced by [MASK] tokens. Given a sequence $x_1, x_2, \cdots, x_n$ with the i-th token masked, the objective is:

$$-\log \frac{\exp\left(c([\texttt{MASK}]) \cdot \mathbf{e}_{x_i}\right)}{\sum_{v \in \mathcal{V}} \exp\left(c([\texttt{MASK}]) \cdot \mathbf{e}_v\right)},$$

where $\mathbf{e}_v$ denotes the embedding of the word $v \in \mathcal{V}$. We use $c(\cdot)$ to denote the contextualized representation for simplicity. Prompt-based learning turns the objective into a softmax distribution over all the target words of a prompt template (Schick and Schütze, 2021a; Gao et al., 2021). For example, in binary sentiment analysis, given an input sentence x, its associated label $y \in \{\text{positive}, \text{negative}\}$ and a template $\mathcal{T}$, we formulate the prompt as:

$$\mathcal{T}(x) = x \text{ It was } [\texttt{MASK}] \ .$$

By defining a mapping $\mathcal{M} : \mathcal{Y} \to \mathcal{V}$ from the task label space to words in the vocabulary, the task is transformed into predicting the target word $\mathcal{M}(y)$:

$$-\log \frac{\exp\left(\mathrm{c}([\texttt{MASK}]) \cdot \mathbf{e}_{\mathcal{M}(y)}\right)}{\sum_{y' \in \mathcal{Y}} \exp\left(\mathrm{c}([\texttt{MASK}]) \cdot \mathbf{e}_{\mathcal{M}(y')}\right)}.$$

This formulation can be used for prompt-based zero-shot evaluation and few-shot fine-tuning to perform gradient updates. For tasks involving multi-token options, such as multiple-choice tasks, prompt-based fine-tuning with MLMs is less intuitive. For example, Schick and Schütze (2021b) adopt a multi-class hinge loss for training and devise a heuristic decoding method to estimate probabilities for target options during inference. The disadvantages are (1) such usage of MLMs deviates from the pre-training objective; (2) the pseudo-autoregressive decoding approach cannot forward in batches during inference, which is computationally inefficient.

2.2 Discriminative Pre-trained Models

Discriminative pre-trained models such as ELECTRA (Clark et al., 2020) cast the word prediction problem into a binary classification problem. In ELECTRA, a discriminator and a smaller generator are jointly trained with the goal to distinguish if the tokens are sampled from the generator or from the original data:

$$-\sum_i \big(\mathbb{1}(x'_i = x_i) \log \mathcal{H}(c(x_i)) + \mathbb{1}(x'_i \neq x_i) \log(1 - \mathcal{H}(c(x'_i)))\big) \,,$$

where $\{x_i\}$ are tokens from the original sentence, $\{x'_i\}$ are tokens from the corrupted sentence, and $\mathcal{H}$ denotes the discriminator head. We refer readers to Clark et al. (2020) for more details.

3 Method: Prompting ELECTRA

Discriminative models like ELECTRA are strong alternatives to MLMs, so they have the potential to be effective few-shot learners even though they do not fit the current paradigm. Furthermore, ELECTRA could be more amenable to solving tasks involving multi-token options by reusing the discriminative head. In this section, we propose adapting ELECTRA to accommodate a wide range of tasks involving single-token or multi-token options for prompt-based learning.[2]

3.1 Tasks with Single-token Target Words

The prompts for ELECTRA models are formulated with an input sentence x, a label $y \in \mathcal{Y}$, and a template $\mathcal{T}$ with the mapping function $\mathcal{M}$. An example of sentiment classification is as follows:

$$\mathcal{T}(x, y) = x \text{ It was } \mathcal{M}(y) \ .$$

For each input sentence, we create $|\mathcal{Y}|$ prompts and forward them for gradient updates such that the model predicts the correct target word as an original token and incorrect ones as generated tokens:

[2]Two concurrent works explore similar ideas to prompt discriminative pre-trained models (Yao et al., 2022; Li et al., 2022). Both approaches concatenate the labels in one forward pass while we forward the input with different target words/options. We also demonstrate the effectiveness on multiple-choice tasks.

	SST-2			SST-5			MR		
	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA
Zero-shot (✓)	61.6	77.8	**82.8**	26.0	30.3	**31.1**	55.8	77.7	**81.5**
Few-shot	72.8 (6.4)	**84.5 (2.3)**	78.2 (7.6)	34.9 (2.0)	37.9 (1.3)	**41.7 (1.8)**	70.8 (5.2)	**76.8 (3.7)**	76.3 (2.9)
Few-shot (✓)	84.6 (1.0)	89.9 (0.6)	**91.2 (0.7)**	37.9 (1.4)	43.3 (1.2)	**49.3 (1.5)**	78.2 (1.1)	85.0 (0.9)	**88.0 (0.5)**
Full-shot	93.2 (0.3)	94.8 (0.3)	**95.5 (0.1)**	53.4 (0.1)	**55.8 (0.1)**	54.8 (0.2)	86.8 (0.3)	88.7 (0.2)	**90.3 (0.0)**
	MNLI			**RTE**			**QNLI**		
	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA
Zero-shot (✓)	43.5	48.1	**51.9**	48.7	53.4	**57.8**	49.5	50.5	**54.5**
Few-shot	41.3 (1.7)	42.2 (2.8)	**44.7 (3.1)**	52.8 (4.1)	54.2 (2.8)	**59.1 (1.7)**	68.4 (4.8)	65.1 (5.1)	**69.7 (3.7)**
Few-shot (✓)	47.9 (0.7)	59.1 (2.1)	**60.8 (2.3)**	57.5 (2.6)	62.7 (2.2)	**67.0 (1.4)**	56.0 (0.7)	67.4 (2.8)	**70.6 (4.0)**
Full-shot	84.7 (0.3)	87.4 (0.0)	**88.6 (0.0)**	69.1 (1.6)	74.2 (0.2)	**78.3 (1.1)**	91.6 (0.1)	92.6 (0.1)	**93.2 (0.0)**
	SNLI			**AGNews**			**BoolQ**		
	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA
Zero-shot (✓)	38.7	48.8	**56.6**	60.6	**73.2**	72.2	47.7	55.9	**59.1**
Few-shot	50.4 (2.8)	44.8 (3.9)	**50.5 (3.3)**	84.9 (0.6)	**85.5 (0.8)**	81.4 (1.4)	54.7 (2.5)	56.8 (3.9)	**57.2 (2.1)**
Few-shot (✓)	51.0 (2.6)	66.3 (3.0)	**72.4 (2.0)**	84.6 (1.2)	**87.1 (0.6)**	86.9 (1.0)	57.4 (2.9)	57.8 (2.4)	**60.8 (4.2)**
Full-shot	91.2 (0.0)	91.9 (0.1)	**92.4 (0.1)**	94.9 (0.0)	**95.4 (0.1)**	94.9 (0.0)	75.3 (1.9)	78.6 (0.3)	**81.1 (1.1)**

Table 1: Zero-shot , few-shot (16 examples per label) and full-shot results of BERT, RoBERTa and ELECTRA base models. ✓ denotes whether a prompt is used or not (Appendix H); otherwise, it adopts standard fine-tuning using the `[CLS]` token. We report average accuracy across 3 runs with standard deviations in parenthesis. We highlight the best number for each setting in bold.

$$- \log \mathcal{H}(c(\mathcal{M}(y))) \\ - \sum_{y' \in \mathcal{Y}/\{y\}} \log(1 - \mathcal{H}(c(\mathcal{M}(y')))).$$

During inference, the model predicts how likely it is for each target option to fit into the sentence and outputs the most likely one. This approach allows us to perform prompt-based zero-shot prediction and few-shot fine-tuning analogously to the MLM paradigm[3]. Note that this approach requires forwarding the input with different target words $|\mathcal{Y}|$ times, which is less efficient than MLMs.

3.2 Tasks with Multi-token Target Options

We handily adapt ELECTRA's discriminative objective to accommodate tasks with multi-token options for prompt-based fine-tuning. The mapping $\mathcal{M} : \mathcal{Y} \to \mathcal{V}^*$ is an identity function for tasks where the target spans are the options themselves. Consider the multiple-choice task COPA (Roemmele et al., 2011); given a premise x, a template $\mathcal{T}$, and an option $y \in \mathcal{Y}$, we formulate the prompt as:

$$\mathcal{T}(x, y) = x \text{ so/because } \mathcal{M}(y) \text{ .}$$

As an option $\mathcal{M}(y)$ contains multiple tokens, we either average the hidden representations of all tokens in $\mathcal{M}(y)$ (equivalent to y):

$$\mathcal{H}\left(\frac{1}{|y|}\sum_j c(y_j)\right);$$

where y_j denotes the j-th token of an option y, or use the average probability of all tokens in y as the final prediction:

$$\frac{1}{|y|}\sum_j \mathcal{H}(c(y_j));$$

or simply take `[CLS]` token's probability: $\mathcal{H}(c(\texttt{[CLS]}))$. These approaches fully reuse pre-trained weights of ELECTRA, including the discriminator head, and refrain from autoregressive-style decoding. Similar to PET, we only use them for few-shot fine-tuning due to the discrepancy from pre-training.

4 Experimental Results

4.1 Setup

We run experiments with released checkpoints of BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019) and ELECTRA (Clark et al., 2020) from the *transformers* (Wolf et al., 2019) library. We use base-sized models unless otherwise specified[4].

[3] We also experimented with a variation to adapt the discriminative objective for contrastive learning, but the results were not as competitive. Please see Appendix F for details.

[4] More details on pre-trained models are in Appendix A.

	COPA	StoryCloze	HellaSwag	PIQA
RoBERTa-base				
[CLS]	69.3 (2.1)	81.4 (1.2)	32.1 (2.3)	54.6 (1.0)
PET	**78.1 (2.6)**	69.8 (3.9)	30.1 (2.1)	61.9 (1.2)
ELECTRA-base				
[CLS]	76.0 (0.8)	84.5 (1.1)	**58.8 (2.0)**	64.9 (1.1)
prob	76.7 (1.7)	85.7 (1.5)	54.5 (1.0)	65.8 (0.3)
rep	76.7 (1.7)	**86.6 (1.1)**	58.0 (0.7)	**66.2 (0.4)**
RoBERTa-large				
[CLS]	76.0 (1.4)	86.5 (4.9)	44.1 (5.1)	55.5 (1.6)
PET	85.4 (2.9)	85.4 (5.3)	46.9 (5.0)	64.6 (3.9)
ELECTRA-large				
[CLS]	**96.0 (0.8)**	**95.6 (0.7)**	**79.9 (2.6)**	71.2 (3.8)
prob	89.0 (1.6)	90.2 (0.5)	77.9 (0.6)	72.0 (0.7)
rep	88.7 (0.9)	90.4 (0.6)	78.1 (0.5)	**72.7 (1.3)**

Table 2: Multiple-choice task results for prompt-based fine-tuning on RoBERTa and ELECTRA with 32 examples across three runs. *CLS*, *prob* and *rep* denote that we take the [CLS] representation, the average probability or the average representations for prediction.

For tasks with single-token target words, we conduct prompt-based zero-shot evaluations, as well as standard[5] and prompt-based few-shot training for each checkpoint. We evaluate on 9 tasks including SST-2, SST-5, MR, MNLI, RTE, QNLI, SNLI, AGNews, and BoolQ[6]. For tasks with multi-token options, we evaluate the few-shot setting. These tasks include COPA, StoryCloze, HellaSwag, and PIQA. Details of datasets (including references) and prompts are in Appendix B and Appendix H.

For our default experiments, we use 16 examples per label for single-token tasks and 32 examples for multiple-choice tasks. We follow Gao et al. (2021) to create a development set with the same size as the training set for model selection and conduct three runs of experiments to mitigate instability issues (Dodge et al., 2020) for all experiments.[7]

4.2 Tasks with Single-token Target Words

Table 1 reports zero-shot and few-shot fine-tuning results on base-sized models.[8] ELECTRA shows a clear advantage compared to BERT and RoBERTa, with an average margin of 7.9 and 3.5 points on zero-shot prediction, respectively, and an average margin of 10.2 and 3.1 on prompt-based few-shot fine-tuning. The difference is much smaller on standard few-shot fine-tuning (3.1 and 1.1, respectively),[9] suggesting that ELECTRA is inherently better at prompt-based learning, in addition to being a better model in general. On that note, we find that prompt-based fine-tuning consistently outperforms standard fine-tuning in line with prior work (Gao et al., 2021; Schick and Schütze, 2021b), which reinforces the importance of using prompts in the few-shot learning setting.

4.3 Tasks with Multi-token Target Options

For tasks involving multi-token options, we focus on the few-shot fine-tuning setting and we use task-specific templates to encode data in all experiments. For both models, we experiment with the few-shot fine-tuning setting where we map the [CLS] representations to scalars. For RoBERTa, we train a head from scratch and for ELECTRA, we reuse the discriminator head. Additionally, we test the PET approach (Schick and Schütze, 2021b) on RoBERTa models as illustrated in Figure 1.

As shown in Table 2, ELECTRA generally presents better and stabler performance than RoBERTa. PET (Schick and Schütze, 2021b), which uses a heuristic autoregressive decoding approach, in most cases outperforms RoBERTa with [CLS] fine-tuning, but still falls behind ELECTRA models. For ELECTRA, using average token representations is comparable or outperforms [CLS] representations for prediction on the base-sized model but [CLS] fine-tuning leads to the best performance on the large-sized model.

These results demonstrate the potential of discriminative models on a broader range of tasks under the few-shot setting.[10]

5 Analysis

5.1 Number of Examples

Figure 3 shows the standard and prompt-based few-shot fine-tuning performance as the number of instances (K) increases for RoBERTa and ELECTRA on four datasets.[11] ELECTRA outperforms RoBERTa with a small K, and the two converge when $K \geq 256$. The performance gap increases as the number of examples decreases, demonstrating that ELECTRA's discriminative pre-training objective is well-suited for few-shot applications.

[5] We use the [CLS] token for prediction in standard fine-tuning, known as head fine-tuning in Le Scao and Rush (2021).

[6] BoolQ is licensed under CC-BY-SA 3.0.

[7] More training details are in Appendix C.

[8] Results on large-sized models are in Appendix D.

[9] The gains of ELECTRA over RoBERTa and BERT on full dataset fine-tuning are similar, 3.3 and 1.2, respectively.

[10] While we focus on MLMs for their direct comparability, ELECTRA also outperforms GPT-3 results reported in Brown et al. (2020) for equivalent model sizes.

[11] See Appendix E for results on the rest of the datasets.

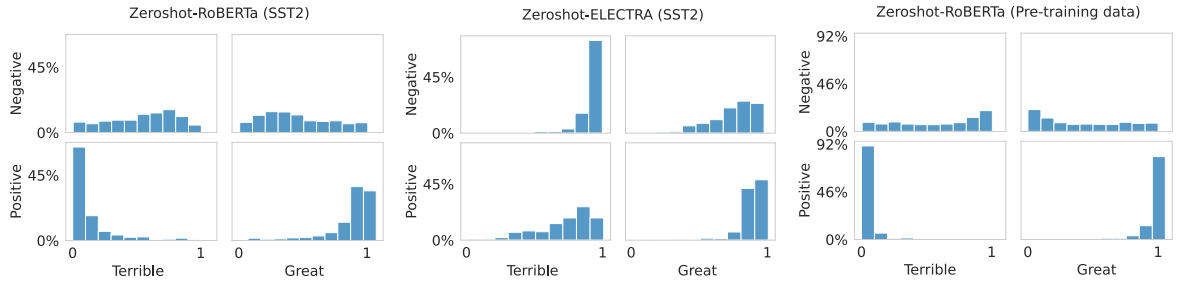

Figure 2: Zero-shot prediction distributions on SST-2 with RoBERTa (left) and ELECTRA (middle). Zero-shot prediction distributions on pre-training data that contain target words (right). Each sub-graph shows the output distribution for inputs associated with a label $y \in \{\text{negative}, \text{positive}\}$ when prompted with the target words $\{\text{great}, \text{terrible}\}$. The y-axis shows the percentage of values in each subgraph. For RoBERTa, the values are normalized across target words, while for ELECTRA, the scores are the raw outputs from its discriminator.

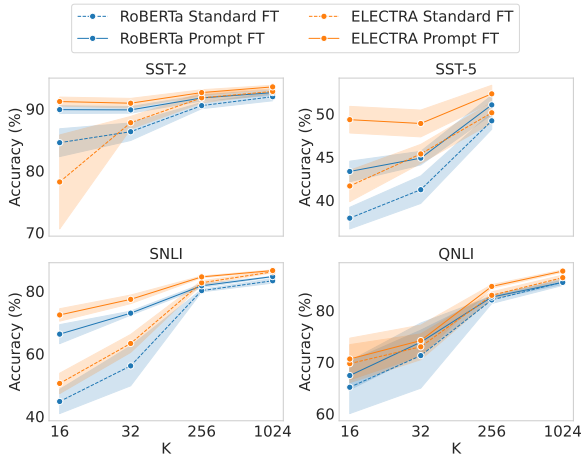

Figure 3: Few-shot performance of RoBERTa v.s. ELECTRA with standard and prompt-based fine-tuning as K (# examples per label) increases. FT: fine-tuning.

5.2 Prediction Analysis

Figure 2 presents the output distributions of zero-shot predictions of RoBERTa and ELECTRA on SST-2.[12] We normalize the RoBERTa output across target words (*great*, *terrible*) and keep the ELECTRA output as it is. For negative examples, the predictions from RoBERTa are only slightly skewed towards *terrible*, indicating that RoBERTa likely assigns a similar probability to the antonym *great* when masking the word *terrible*. This finding sheds light on why ELECTRA outperforms RoBERTa, as it has likely seen the closely-related alternative words during training and learned to suppress the probability of these words being original.

We analyze RoBERTa's output distribution on its pre-training corpus to verify that the analysis does not spuriously correlate with the task template. We randomly sample sentences that either contain the word *great* or *terrible* and forward the sentences through the model after masking these two words. We visualize the normalized output distribution over *great* and *terrible* in Figure 2 and observe a similar pattern as RoBERTa's zero-shot prediction distribution on SST-2. It corroborates our hypothesis that masked language models fail to predict the correct word but instead output the antonym in some cases, e.g., when the ground truth is *terrible*, which enables ELECTRA to distinguish semantically opposite words and further strengthens its prompt-based prediction ability.

[12] In Appendix G, we show that the output distribution shifts to a polarized shape with few-shot fine-tuning.

6 Conclusion

We explore discriminative pre-trained models for prompt-based zero-shot and few-shot learning. We find that these models consistently outperform masked language models that are trained with equivalent or even less computation, suggesting that discriminative pre-trained models are more effective zero-shot and few-shot learners. Analysis shows that the ELECTRA's generator could very likely feeds negatives like antonyms to the discriminator, which serves as a direct contrast during pre-training. We also speculate that discriminative models are less vulnerable to the surface form competition (Holtzman et al., 2021), and we would like to dig deeper into this hypothesis in future work.

Acknowledgements

The authors thank Myle Ott, Dan Friedman, Sadhika Malladi, Zexuan Zhong, Tianyu Gao and the anonymous reviewers for their valuable feedback on our paper.

Limitations

One limitation of this work is that we limit our exploration within the scope of discriminative tasks. It is prohibitively expensive to apply the prompting approach of ELECTRA to tasks without a limited set of candidates. The prompting approach we propose for ELECTRA requires one forward pass for each option in one example. In contrast, masked language models only require one forward pass for each example.

Another limitation is that we only include a limited set of continuation-based multiple-choice tasks for evaluation due to space constraints. We leave evaluating on a more diverse set of multiple-option tasks as future work.

References

Mikel Artetxe, Shruti Bhosale, Naman Goyal, Todor Mihaylov, Myle Ott, Sam Shleifer, Xi Victoria Lin, Jingfei Du, Srinivasan Iyer, Ramakanth Pasunuru, et al. 2021. Efficient large scale language modeling with mixtures of experts. *arXiv preprint arXiv:2112.10684*.

Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. 2009. The fifth PASCAL recognizing textual entailment challenge. In *TAC*.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The PASCAL recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, pages 177–190. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305*.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and William B Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9.

R Bar Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second PASCAL recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, volume 7.

Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. 2021. Surface form competition: Why the highest probability answer isn't always right. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7038–7051.

Teven Le Scao and Alexander M Rush. 2021. How many data points is a prompt worth? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2627–2636.

Zicheng Li, Shoushan Li, and Guodong Zhou. 2022. Pre-trained token-replaced detection model as few-shot learner. *arXiv preprint arXiv:2203.03235*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. pages 839–849.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI Spring Symposium Series*.

Timo Schick and Hinrich Schütze. 2021a. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269.

Timo Schick and Hinrich Schütze. 2021b. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Yuan Yao, Bowen Dong, Ao Zhang, Zhengyan Zhang, Ruobing Xie, Zhiyuan Liu, Leyu Lin, Maosong Sun, and Jianyong Wang. 2022. Prompt tuning for discriminative pre-trained language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3468–3473, Dublin, Ireland. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? pages 4791–4800.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

A Model Details

We list the details of the pre-trained models, including training corpora, vocabulary size, training steps, and GLUE development set results in Table 3. ELECTRA, which is trained on the same set of corpora as BERT, outperforms BERT on GLUE datasets by 3 to 5 points. It slightly underperforms RoBERTa on the base size but is comparable to RoBERTa on the large size.

B Datasets

We experiment on 1) sentence classification tasks, including 3 sentiment analysis datasets: SST-2, SST-5 (Socher et al., 2013), MR (Pang and Lee, 2005); 4 natural language inference tasks: MNLI (Williams et al., 2018), RTE (Dagan et al., 2005; Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009), QNLI (Rajpurkar et al., 2016), SNLI (Bowman et al., 2015); AGNews (Zhang et al., 2015), which is a news classification dataset, BoolQ (Clark et al., 2019), which is a dataset of boolean questions; 2) multiple-choice tasks, which involve multi-token options, including COPA (Roemmele et al., 2011), StoryCloze (Mostafazadeh et al., 2016), Hellaswag (Zellers et al., 2019), PIQA (Bisk et al., 2020). We construct a validation set the same size as the training set in few-shot settings and report results on the full validation set for all datasets.

C Training Details

Following Gao et al. (2021), we conduct a grid search for all few-shot experiments and take learning rates from {1e-5, 2e-5, 3e-5} and batch sizes from $\{2, 4, 8\}$. For each trial, we perform gradients updates for 1000 steps, evaluate the model every 100 steps and select the model with the best validation accuracy. For full-shot experiments, we conduct a grid search with learning rates from {1e-5, 2e-5, 3e-5} and use a batch size of 16.

D Results on Large-sized Models

We present prompt-based zero-shot and few-shot results on large-sized models in Table 4 to show that the trend prevails when the model scales up. Except for SNLI, the average gain from prompt-based fine-tuning for ELECTRA is significantly larger than BERT and RoBERTa. Notably, ELECTRA also significantly outperforms BERT and RoBERTa on zero-shot prediction.

E Number of Examples

We show the few-shot results as a function of K on BoolQ, RTE, AGNews and MR in Figure 4. ELECTRA significantly outperforms RoBERTa on BoolQ and RTE across all settings, suggesting that ELECTRA is an overall stronger model for these datasets. On MR, we observe a similar pattern where the gap between ELECTRA and RoBERTa gets smaller, showing that ELECTRA benefits from prompt training more than RoBERTa. On AGNews, ELECTRA underperforms RoBERTa on standard fine-tuning but closes the gap on prompt-based fine-tuning, backing up the argument that ELECTRA benefits more from the prompt.

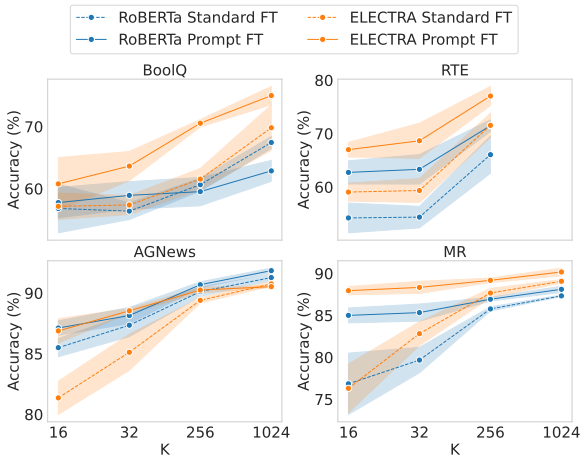

Figure 4: Few-shot performance of RoBERTa v.s. ELECTRA with standard and prompt-based fine-tuning as K (the number of instances per label) increases on more tasks.

F An Alternative Contrastive Objective

We also explored another contrastive objective with ELECTRA's logits for prompt-based few-shot fine-tuning. For all the prompts of an input x with the label set $\mathcal{Y}$, we define the loss as

$$-\log \frac{\exp(\phi(c(y)))}{\sum_{y' \in \mathcal{Y}} \exp(\phi(c(y')))}$$

where $\mathcal{H}(x) = \frac{1}{1+e^{-\phi(x)}}$ and $\phi(x)$ denote the logits from the discriminator. We directly contrast the correct target option with the incorrect ones with this objective. We show results on SST-2 and AGNews in Table 5. Prompt-based fine-tuning with the original ELECTRA objective outperforms the contrastive objective. We hypothesize that the downside of the contrastive objective is that it forces one input with different target options to be packed

Models	Pretrain Corpora	Corpora Size	# Vocab	Steps	GLUE
$\text{BERT}_{\text{base}}$	Wikipedia, BooksCorpus	16GB	30K	1M	82.2
$\text{RoBERTa}_{\text{base}}$	Wikipedia, BooksCorpus, CC-News, OpenWebText, Stores	160GB	50K	500K	86.4
$\text{ELECTRA}_{\text{base}}$	Wikipedia, BooksCorpus	16GB	30K	1M	87.1
$\text{BERT}_{\text{large}}$	Wikipedia, BooksCorpus	16GB	30K	464K	84.0
$\text{RoBERTA}_{\text{large}}$	Wikipedia, BooksCorpus, CC-News, OpenWebText, Stores	160GB	50K	500K	88.9
$\text{ELECTRA}_{\text{large}}$	Wikipedia, BooksCorpus, ClueWeb, CommonCrawl, Gigaword	33GB	30K	400K	89.0

Table 3: Pre-training details of BERT, RoBERTa and ELECTRA. The GLUE results are taken from Clark et al. (2020) and Liu et al. (2019) on the development set.

	SST-2			**SST-5**		
	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA
Zero-shot (✓)	61.2	83.6	**86.0**	25.7	**34.7**	32.1
Few-shot	82.4 (3.0)	**85.4 (2.9)**	75.8 (5.2)	40.1 (2.4)	41.3 (1.2)	**42.8 (0.9)**
Few-shot (✓)	87.9 (0.8)	93.0 (0.6)	**93.6 (0.4)**	42.4 (1.5)	47.1 (0.9)	**50.3 (1.8)**
Full-shot	94.3	96.6	**97.1**	53.3	56.8	**58.9**
	SNLI			**BoolQ**		
	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA
Zero-shot (✓)	41.5	49.8	**59.4**	49.3	53.4	**71.1**
Few-shot	51.2 (3.3)	51.4 (3.1)	**66.7 (2.7)**	56.0 (2.3)	59.5 (3.0)	**61.3 (1.5)**
Few-shot (✓)	60.6 (2.8)	**79.4 (1.4)**	79.1 (2.0)	56.9 (0.3)	70.3 (2.6)	**75.2 (1.2)**
Full-shot	91.6	92.1	**92.2**	73.1	**85.2**	85.0

Table 4: Zero-shot and few-shot (16 examples per label) and full-shot results of large-sized BERT, RoBERTa and ELECTRA. ✓: denotes whether prompts are used or not.

into the same batch instead of shuffling the whole dataset randomly, and it affects the optimization. We also experiment on the original discriminative objective with the same batch restriction and observe a performance drop to verify the hypothesis.

G Few-shot Output Distribution

We show the few-shot output distribution of RoBERTa and ELECTRA on SST-2 in Figure 5. The output distributions are polarized after few-shot training.

H Prompts

We largely follow previous works to construct our prompts. For sentiment classification tasks and natural language inference tasks, we use prompts from Gao et al. (2021). For AGNews, we use the prompt from Holtzman et al. (2021) and for BoolQ, we use the prompt from Schick and Schütze (2021b). For tasks involving multi-token options, we simply concatenate the context and options, which largely follows Holtzman et al. (2021). The prompt details can be found in Table 6 and Table 7.

To verify that the prompts does not affect our major conclusion, we conduct prompt-based few-shot finetuning experiments with different prompts for four tasks. The prompts we use are in Table 8. Results in Table 9 show that ELECTRA outperforms RoBERTa with different prompts.

Task	K	Original	Original w/o shuffling	Contrastive
SST-2	16	**91.2** (0.7)	**91.2** (0.8)	91.0 (0.4)
	32	**90.9** (0.8)	90.5 (0.8)	90.6 (0.7)
	256	**92.6** (0.5)	92.2 (0.4)	92.2 (0.7)
	1024	**93.6** (0.3)	92.9 (0.5)	93.1 (0.3)
AGNews	16	**86.5** (1.1)	85.4 (1.3)	85.4 (0.8)
	32	**88.4** (0.3)	86.5 (0.6)	86.7 (0.7)
	256	**90.3** (0.2)	89.8 (0.2)	89.3 (0.2)
	1024	**90.5** (0.1)	90.1 (0.2)	89.5 (0.3)

Table 5: Few-shot prompt-based fine-tuning results on different objectives with ELECTRA$_{\text{base}}$. Original w/o shuffling denotes that we load the batches without data shuffling to mimic the data loading restriction when training with the contrastive objective).

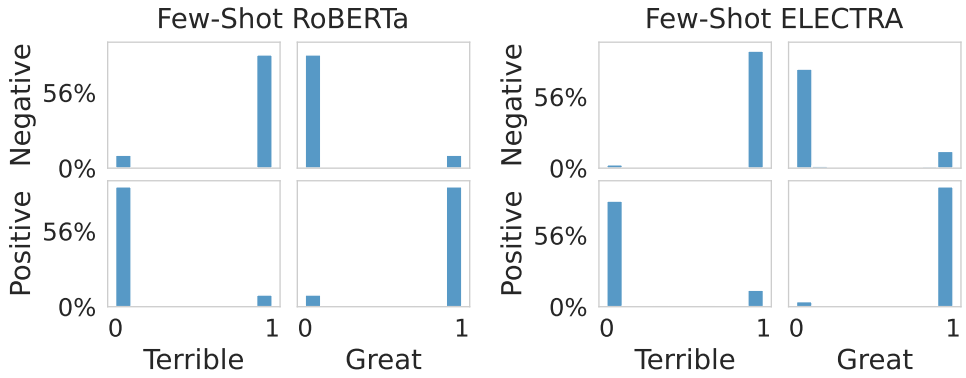

Figure 5: Few-shot prediction distributions on SST-2 with RoBERTa$_{\text{base}}$ and ELECTRA$_{\text{base}}$. Each sub-graph shows the output distribution for inputs with a label $y \in \{\text{negative}, \text{positive}\}$ when prompted with the corresponding target option $\mathcal{M}(y)$.

Task	Template	Label Words
SST-2	<sentence> It was `[MASK]` .	positive: great, negative: terrible
SST-5	<sentence> It was `[MASK]` .	v.positive: great, positive: good, neutral: okay, negative: bad, v.negative: terrible
MR	<sentence> It was `[MASK]` .	positive: great, negative: terrible
MNLI	<premise>? `[MASK]` , <hypothesis>	entailment: Yes, neutral: Maybe, contradiction: No
SNLI	<premise>? `[MASK]` , <hypothesis>	entailment: Yes, neutral: Maybe, contradiction: No
RTE	<premise>? `[MASK]` , <hypothesis>	entailment: Yes, not entailment: No
QNLI	<premise>? `[MASK]` , <hypothesis>	entailment: Yes, not entailment: No
AGNews	`[MASK]` News: <sentence>	World: World, Sports: Sports, Business: Business, Sci/Tech: Tech
BoolQ	<passage> Question: <question> ? Answer: `[MASK]` .	No: No, Yes: Yes

Table 6: Task templates for tasks with single-token verbalizers.

Task	Template
COPA	<sentence> so/because `[OPTION]`
StoryCloze	<sentence1> <sentence2> <sentence3> <sentence4> `[OPTION]`
Hellaswag	<context> `[OPTION]`
PIQA	<sentence> `[OPTION]`

Table 7: Task templates for tasks with multi-token verbalizers.

Text	$\mathcal{T}$	Template
MNLI	$\mathcal{T}_1$	<premise> ? `[MASK]` , <hypothesis>
	$\mathcal{T}_2$	<premise> ? `[MASK]` . <hypothesis>
	$\mathcal{T}_3$	"<premise>" ? `[MASK]` , "<hypothesis>"
RTE	$\mathcal{T}_1$	<premise> ? `[MASK]` , <hypothesis>
	$\mathcal{T}_2$	<premise> ? `[MASK]` . <hypothesis>
	$\mathcal{T}_3$	"<premise>" ? `[MASK]` , "<hypothesis>"
COPA	$\mathcal{T}_1$	<sentence> so/because `[OPTION]`
	$\mathcal{T}_2$	`[OPTION_1]` or `[OPTION_2]` ? <sentence>so/because `[OPTION]`
StoryCloze	$\mathcal{T}_1$	<sentence1> < sentence2> < sentence3> < sentence4> `[OPTION]`
	$\mathcal{T}_2$	`[OPTION_1]` or `[OPTION_2]` ? <sentence1> <sentence2> <sentence3> <sentence4> `[OPTION]`

Table 8: Task templates for task sensitivity test.

		MNLI	RTE	COPA	SC
$\mathcal{T}_1$	RoBERTa	59.1	62.7	72.7	71.0
	ELECTRA	**60.8**	**67.0**	**75.0**	**86.9**
$\mathcal{T}_2$	RoBERTa	55.3	63.2	69.7	71.7
	ELECTRA	**61.0**	**64.9**	**74.7**	**86.4**
$\mathcal{T}_3$	RoBERTa	57.3	63.9	-	-
	ELECTRA	**60.9**	**67.2**	-	-

Table 9: Few-shot results with different templates with base-sized models. ELECTRA still outperforms RoBERTa with different templates (provided in Table 8).