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

Anonymous ACL submission

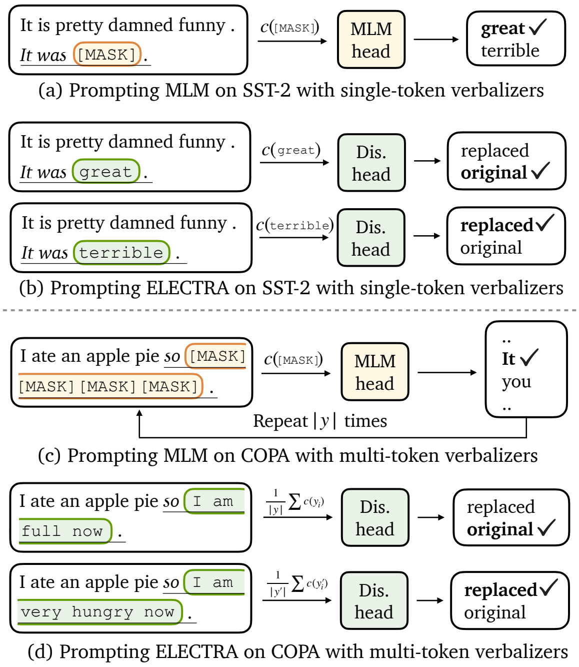
Abstract

Pre-trained masked language models have been successfully used for few-shot learning by formulating downstream tasks as text infilling. However, discriminative pre-trained models like ELECTRA, as a strong alternative in full-shot settings, does 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 verbalizers without introducing new parameters. Our method can be easily adapted to tasks involving multi-token verbalizers without extra computation overhead. Analysis shows that the distributions learned by ELECTRA align better with downstream tasks.

1 Introduction

Large pre-trained language models, which encode rich language properties, are known to be effective zero- and few-shot learners (Brown et al., 2020; Artetxe et al., 2021; Rae et al., 2021). Even relatively small masked language models (MLMs), like BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), demonstrate competitive few-shot performance through prompt-based fine-tuning, which updates the model to select the correct verbalizers (Schick and Schütze, 2021a; Gao et al., 2021).

Discriminative pre-trained models like ELECTRA (Clark et al., 2020) are strong alternatives to MLMs in full-shot settings, but their properties as zero- and few-shot learners remain unexplored. We hypothesize that models like ELECTRA would make strong zero- and few-shot learners 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, which distinguishes if a single token is generated or from the training data, with the downstream prompt-based prediction by reusing the discriminative head to classify correct verbalizers as original tokens. As an additional benefit, we can naturally adapt the approach to verbalizers spanning multiple tokens by aggregating either representations or probabilities. In contrast, MLMs require auto-regressive decoding to adapt to multi-token verbalizers (Schick and Schütze, 2021b).


It is pretty damned funny .
It was [MASK] .
c([MASK])
MLM head
great ✓
terrible
It is pretty damned funny .
It was great .
c(great)
Dis. head
replaced
original ✓
It is pretty damned funny .
It was terrible .
c(terrible)
Dis. head
replaced ✓
original
I ate an apple pie so [MASK] [MASK] [MASK] [MASK] .
c([MASK])
MLM head
..
It ✓
you
..
Repeat |y| times
I ate an apple pie so I am full now .
$\frac{1}{|y|}\sum c(y_i)$
Dis. head
replaced
original ✓
I ate an apple pie so I am very hungry now .
$\frac{1}{|y|}\sum c(y'_i)$
Dis. head
replaced ✓
original

(a) Prompting MLM on SST-2 with single-token verbalizers

(b) Prompting ELECTRA on SST-2 with single-token verbalizers

(c) Prompting MLM on COPA with multi-token verbalizers

(d) Prompting ELECTRA on COPA with multi-token verbalizers

Figure 1: Prompt-based fine-tuning with MLMs and **dis**criminative models with a SST-2 and COPA example. The underlined text is the task-specific *template*. $c(\cdot)$ denotes the contextualized embedding; y and y' denotes a correct and an incorrect option respectively.

Our approach to prompting ELECTRA outperforms BERT and RoBERTa by 10.2 and 3.1 points on average across 13 tasks (text classification, NLI, multiple-choice tasks) for base-sized models in the few-shot setting, and the trend prevails for large-sized models. Analysis shows that the output distributions of ELECTRA's pre-training task are close to downstream task distributions.

2 Background

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

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

where $\mathbf{e}_v$ denotes the output embedding of 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 verbalizers of a prompt template (Gao et al., 2021; Schick and Schütze, 2021a). 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:

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

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

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

This formulation can be used for both prompt-based zero-shot evaluation and few-shot fine-tuning to perform gradient updates.

For tasks involving multi-token verbalizers such as multiple-choice tasks, prompt-based fine-tuning with MLMs is less intuitive. Schick and Schütze (2021b) propose PET, which adopts a multiclass hinge loss for training and devise a heuristic decoding method to estimate probabilities for multi-token verbalizers during inference. The disadvantages are (1) such usage of MLMs deviates from the pre-training objective; (2) the auto-regressive decoding cannot forward in batches during inference, which is computationally inefficient.

Discriminative Pre-trained Models Discriminative models such as ELECTRA (Clark et al., 2020) cast the word prediction problem into a binary classification problem. In ELECTRA, a discriminator $\mathcal{D}$ and a smaller generator $\mathcal{G}$ are jointly trained with the goal to distinguish if the tokens are sampled from $\mathcal{G}$ or 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 verbalizers, as it does not require auto-regressive decoding. In this section, we propose to adapt ELECTRA to accommodate a wide range of tasks involving either single-token or multi-token verbalizers for prompt-based learning.

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

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

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

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

During inference, the model predicts how likely it is for each verbalizer to fit into the sentence and outputs the most likely one.[1] This approach allows us to perform prompt-based zero-shot prediction and few-shot fine-tuning analogously to the MLM paradigm.

Tasks with Multi-Token Verbalizers We handily adapt ELECTRA's discriminative objective to accommodate tasks with multi-token verbalizers for prompt-based fine-tuning. The mapping $\mathcal{M}$:

[1] One disadvantage is that this approach requires forwarding the input $|\mathcal{Y}|$ times, which is less efficient than MLMs.

	SST-2			SST-5			MR		
	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA
prompt zero-shot	61.6	77.8	**82.8**	26.0	30.3	**31.1**	55.8	77.7	**81.5**
standard few-shot FT	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)
prompt few-shot FT	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)**
standard full-shot FT	*93.6*	*95.1*	***95.6***	*53.3*	***55.9***	*55.0*	*87.1*	*88.9*	***90.4***
	MNLI			RTE			QNLI		
	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA
prompt zero-shot	43.5	48.1	**51.9**	48.7	53.4	**57.8**	49.5	50.5	**54.5**
standard few-shot FT	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)**
prompt few-shot FT	47.9 (0.7)	59.1 (2.1)	**60.8 (2.3)**	57.5 (2.6)	62.7 (2.17)	**67.0 (1.4)**	56.0 (0.7)	67.4 (2.8)	**70.6 (4.0)**
standard full-shot FT	*84.9*	*88.1*	***89.0***	*70.8*	*74.4*	***79.4***	*91.7*	*92.7*	***93.2***
	SNLI			AGNews			BoolQ		
	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA
prompt zero-shot	38.7	48.8	56.6	60.6	73.2	72.2	47.7	55.9	59.1
standard few-shot FT	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)**
prompt few-shot FT	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)**
standard full-shot FT	*92.3*	*94.1*	***94.6***	*94.9*	***95.5***	*95.0*	*77.1*	*78.8*	***82.0***

Table 1: Zero-shot and few-shot ($K = 16$) results of BERT, RoBERTa and ELECTRA base models. In the parenthesis are standard deviations of 5 runs. We highlight the best number for each setting.

$\mathcal{Y} \rightarrow \mathcal{V}^*$ is an identity function for such tasks where the verbalizers 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:

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

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

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

or use the average probability of all tokens in v as the final prediction:

$$\frac{1}{|y|}\sum_i \mathcal{H}(c(y_i))$$

Both methods [2] fully reuse all pre-trained weights of ELECTRA and refrain from autoregressive decoding. Similar to PET, we only use this approach for few-shot fine-tuning due to its discrepancy from pre-training.

4 Experimental 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. More pre-training details of the models are in Appendix A. We conduct prompt-based zero-shot evaluations as well as standard [3] and prompt-based few-shot training for each checkpoint. For few-shot experiments, we follow Gao et al. (2021) to create a development set the same size as the training set for model selection and conduct multiple runs of experiments to mitigate instability issues (Dodge et al., 2020). More training details are in Appendix C.

We evaluate on sequence classification tasks including SST-2, SST-5, MR, MNLI, RTE, QNLI, SNLI, AGNews and BoolQ; and multiple-choice tasks including COPA, StoryCloze, Hellaswag, PIQA. Dataset and template details are in Appendix B and Appendix H.

5 Results and Analysis

Tasks with Single-token Verbalizers Table 1 reports zero-shot and few-shot fine-tuning results on base-sized models[4]. 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

[2] We also experimented with another approach to adapt the discriminative objective for contrastive learning but the results were not as competitive. Please see Equation F for details.

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

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

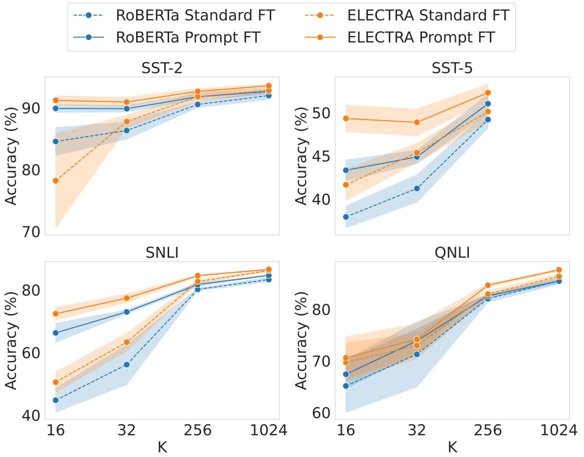


Figure 2: Few-shot performance of RoBERTa v.s. ELECTRA with standard and prompt-based fine-tuning as K (number of instances per label) increases.

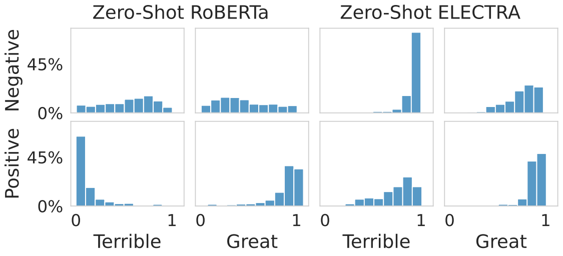


Figure 3: Zero-shot prediction distributions on SST-2 with RoBERTa and ELECTRA. Each sub-graph shows the output distribution for inputs associated with a label $y \in \{\text{negative, positive}\}$ when prompted with the verbalizers $\{\text{great, terrible}\}$. The y-axis shows the percentage of values in each subgraph.

few-shot fine-tuning (3.3 and 1.3, respectively),[5] suggesting that ELECTRA is inherently better at prompt 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 prompts for few-shot learning.

Tasks with Multi-token Verbalizers Table 2 shows results on multiple-choice tasks, the verbalizers of which are multi-token options. ELECTRA outperforms RoBERTa with PET (Schick and Schütze, 2021b), which uses a heuristic autoregressive decoding approach. ELECTRA$_{\text{base}}$ and ELECTRA$_{\text{large}}$ outperform their counterparts of RoBERTa fine-tuned with PET. This result demonstrates the potential of discriminative models on a broader range of tasks under the few-shot setting [6].

Model	Size	CP	SC	HS	PI
RoBERTa (PET)	125M	72.7	71.0	31.3	61.8
ELECTRA (prob)	109M	73.7	85.3	52.6	66.2
ELECTRA (rep)	109M	**75.0**	**86.9**	**56.0**	**67.4**
RoBERTa (PET)	335M	77.7	73.2	46.9	61.9
ELECTRA (prob)	335M	85.0	88.9	**77.7**	70.9
ELECTRA (rep)	335M	**90.7**	**90.4**	77.6	**71.7**

Table 2: Multi-choice task results for prompt-based fine-tuning with RoBERTa and ELECTRA with 32 randomly selected examples. We run each model three times and the standard deviations are around 1-2 points. *prob* and *rep* denote average probability and representations. CP: COPA, SC: StoryCloze, HS: Hellaswag, PI: PIQA.

Number of Examples Figure 2 shows standard and prompt-based few-shot fine-tuning performance as the number of instances (K) increases for RoBERTa and ELECTRA on four datasets[7]. ELECTRA outperforms RoBERTa as K increases, 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 zero- and few-shot applications.

Prediction Analysis We show the output distributions of zero-shot predictions from RoBERTa and ELECTRA on SST-2 in Figure 3. RoBERTa failed mostly on negative examples, and ELECTRA's outputs align with the task distribution better. 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 and find that they consistently outperform masked language models, suggesting that discriminative pre-trained models are effective zero-shot and few-shot learners. Analysis shows that the output distributions of discriminative models align with the downstream task distribution better. We speculate that this could be due to discriminative models being less vulnerable to the surface form competition (Holtzman et al., 2021), and we would like to dig deeper into this hypothesis in future work.

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

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

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

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. 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 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

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. *arXiv preprint arXiv:2104.08315*.

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.

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. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 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.

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*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 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 datastes 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 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 and evaluate the model every 100 steps and select the model with the best validation accuracy. For full-shot experiments, we conduct grid search with learning rates from $\{1e-5, 2e-5, 3e-5\}$ and use a batch size of 16.

D Large 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 SNLI, the average gain from prompt-based fine-tuning for ELECTRA$_{\text{large}}$ is significantly larger than BERT$_{\text{large}}$ and RoBERTa$_{\text{large}}$. Notably, ELECTRA$_{\text{large}}$ also significantly outperforms BERT$_{\text{large}}$ and RoBERTa$_{\text{large}}$ on zero-shot prediction.

E Number of Examples

We show the few-shot results as a function of K on the rest of the single-token tasks in Figure 4. ELECTRA significantly outperforms RoBERTa on

Models	Pretrain Corpora	Corpora Size	# Vocab	Steps	GLUE
BERT$_{\text{base}}$	Wikipedia, BooksCorpus	16GB	30K	1M	82.2
RoBERTabase	Wikipedia, BooksCorpus, CC-News, OpenWebText, Stores	160GB	50K	500K	86.4
ELECTRAbase	Wikipedia, BooksCorpus	16GB	30K	766K	85.1
BERTlarge	Wikipedia, BooksCorpus	16GB	30K	464K	84.0
RoBERTAalarge	Wikipedia, BooksCorpus, CC-News, OpenWebText, Stores	160GB	50K	500K	88.9
ELECTRAlarge	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
prompt zero-shot	61.2	83.6	**86.0**	25.7	**34.7**	32.1
standard few-shot FT	82.4 (3.0)	**85.4 (2.9)**	75.8 (5.2)	40.1 (2.4)	41.3 (1.2)	**42.8 (0.9)**
prompt few-shot FT	87.9 (0.8)	93.0 (0.6)	**93.6 (0.4)**	42.4 (1.5)	47.1 (0.9)	**50.3 (1.8)**
standard full-shot FT	*94.3*	*96.6*	***97.1***	*53.3*	*56.8*	***58.9***
	SNLI			BoolQ		
	BERT	RoBERTa	ELECTRA	BERT	RoBERTa	ELECTRA
prompt zero-shot	41.5	49.8	**59.4**	49.3	53.4	**71.1**
standard full-shot FT	51.2 (3.3)	51.4 (3.1)	**66.7 (2.7)**	56.0 (2.3)	59.5 (3.0)	**61.3 (1.5)**
prompt few-shot FT	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 standard FT	*91.6*	*92.1*	***92.2***	*73.1*	***85.2***	*85.0*

Table 4: Zero-shot and few-shot ($K = 16$) results of BERT, RoBERTa and ELECTRA large models.

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.

F Contrastive Objective

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

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

With this objective, we directly contrast the correct verbalizer with the incorrect ones and show results on SST-2 and AGNews in Table 5. Prompt-based fine-tuning with the original ELECTRA objective

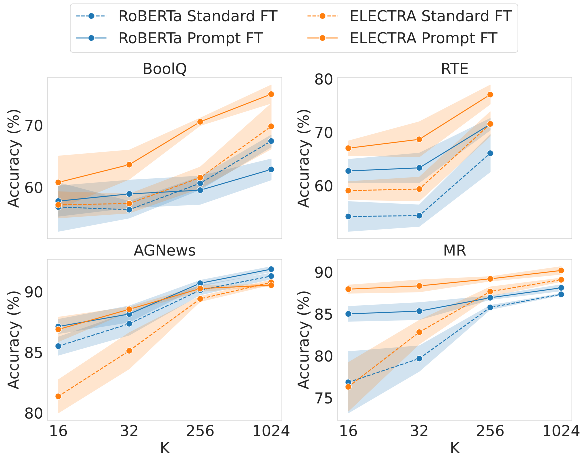


Figure 4: Fewshot performance of RoBERTa v.s. ELECTRA with standard and prompt-based fine-tuning as K (number of instances per label) increases on more tasks.

Task	K	Original	Original w/ Cons.	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$_{\text{base}}$.

outperforms the contrastive objective. We hypothesize that the downside of the contrastive objective is that one input with different verbalizers will be packed into the same batch, which affects the optimization. To verify the hypothesis, we also experiment on the original discriminative objective with the same batch restriction and observe a performance drop.

G Few-shot Plots

We show the few-shot output distribution of RoBERTa and ELECTRA on SST-2 in Figure 5.

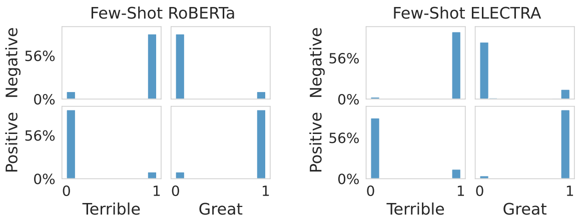


Figure 5: Few-shot prediction distributions on SST-2 with RoBERTa$_{\text{base}}$ and ELECTRA$_{\text{base}}$. Each sub-graph shows the output distribution for inputs with a label $y \in \{\text{negative}, \text{positive}\}$ when prompted with the corresponding verbalizer $\mathcal{M}(y)$.

H Template

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

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

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, netural: Maybe, contradiction: No
SNLI	<premise>? [MASK] , <hypothesis>	entailment: Yes, netural: 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.