Enabling Natural Zero-Shot Prompting on Encoder Models via Statement-Tuning

Ahmed Elshabrawy[1], **Yongxin Huang**[2], **Iryna Gurevych**[1,2], **Alham Fikri Aji**[1]
[1]Department of Natural Language Processing, MBZUAI
[2]Ubiquitous Knowledge Processing Lab, Technical University of Darmstadt
[1]{ahmed.elshabrawy,iryna.gurevych,alham.fikri}@mbzuai.ac.ae
[2] www.ukp.tu-darmstadt.de

Abstract

While Large Language Models (LLMs) exhibit remarkable capabilities in zero-shot and few-shot scenarios, they often require computationally prohibitive sizes. Conversely, smaller Masked Language Models (MLMs) like BERT and RoBERTa achieve state-of-the-art results through fine-tuning but struggle with extending to few-shot and zero-shot settings due to their architectural constraints. Hence, we propose Statement-Tuning, a technique that models discriminative tasks as a set of finite statements and trains an encoder model to discriminate between the potential statements to determine the label. We do Statement-Tuning on multiple tasks to enable cross-task generalization. Experimental results demonstrate that Statement-Tuning achieves competitive performance compared to state-of-the-art LLMs with *significantly* fewer parameters. Furthermore, we compare with previous encoder-based methodology and show that our method is more accurate and more robust to spurious patterns. Moreover, the study investigates the impact of several design choices on few-shot and zero-shot generalization, revealing that Statement-Tuning can achieve strong performance with modest training data and benefits from task and statement diversity for unseen task generalizability. We release all the code used to generate statement data, train and evaluate our Statement-Tuned models.[1]

1 Introduction

Large Language Models (LLMs) have shown great capabilities in zero-shot and few-shot settings (Radford et al., 2019; Brown et al., 2020; Artetxe et al., 2022). However, such capabilities are more difficult to observe in encoder-only models like BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) due to their architectural design. These models are typically pre-trained in an unsupervised manner on a large corpus with a Masked Language Modeling (Devlin et al., 2019) or Discriminative (Clark et al., 2020) objective and fine-tuned by adding task-specific layers to enable their usage on a particular task, such as binary/multi-label classification, token/sequence classification, multiple choice, etc. These task-specific layers, thus, can not be extended effectively to new tasks in a few-shot or zero-shot manner.

In this work, we explore the feasibility of utilizing encoder models that are usually specialized for a certain task to take on various, unseen Natural Language Understanding (NLU) tasks, akin to zero-shot prompting in decoder models. One benefit of using encoder models is that they are generally more compact. Yet, encoder models have achieved state-of-the-art results on many NLU tasks through task-specific fine-tuning. So it would be interesting if LLM-level zero-shot prompting could be achieved by encoder models to leverage their powerful NLU capabilities at more computationally feasible sizes.

To achieve this, some techniques try to reformulate various downstream tasks with a unified format resembling the pre-training objective, enabling few-shot transfer for encoder models (Schick and Schütze, 2021a,b; Xia et al., 2022). Without few-shot examples, the zero-shot generalization of these models relies mainly on the language modeling ability learned in the pre-training phase, not benefiting from further multitask training on diverse reformulated tasks. In this work, we take inspiration from multitask instruction tuning methods for decoder models (Wei et al., 2022; Sanh et al., 2022) and unified format fine-tuning methods for encoder models (Yin et al., 2019; Xu et al., 2023) to propose Statement-Tuning, a novel intuitive approach for encoder-only models to generalize to zero-shot and few-shot unseen tasks through universal multitask fine-tuning with data formatted as statements. Our approach thus has the generalization ability similar

[1]https://github.com/afz225/statement-tuning

Findings of the Association for Computational Linguistics:
NAACL 2025, pages 8317–8336

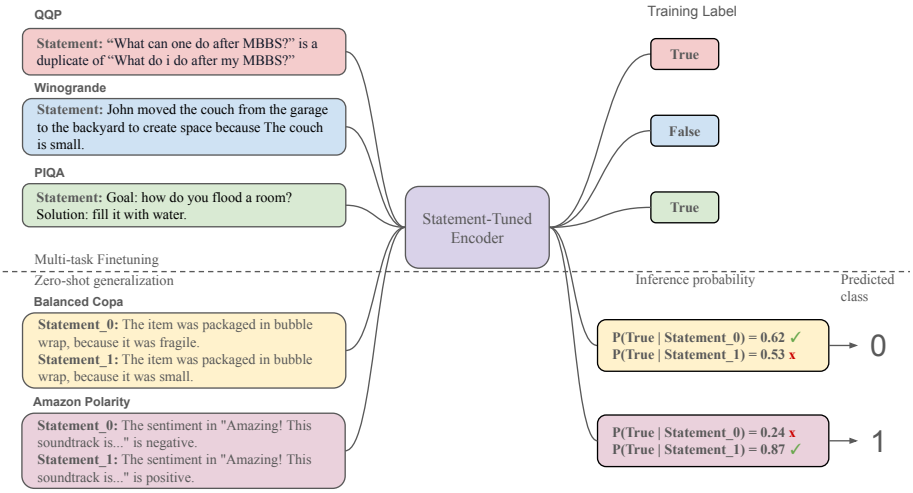

Figure 1: Overview of Statement-Tuning. We train an encoder to discriminate the truth value of statements from multiple tasks, then we apply it in the zero-shot setting by creating a statement for each possible target label and choosing the most likely one according to the encoder discriminator.

to decoder models with a fraction of the parameters and training data.

As seen in Figure 1, we verbalize a diverse set of NLU tasks into natural language statements, and then fine-tune an encoder-only MLM, RoBERTa, on a universal binary sequence classification task, which we call Statement-Tuning, to assign a truth value (True or False) to any given statement. By fine-tuning encoder models across diverse tasks and statements, we show zero-shot generalization capabilities to unseen tasks by similarly transforming them into statements. Moreover, we show few-shot capabilities by continually fine-tuning this model with a small amount of downstream data, also formatted into statements. Statement-Tuning is capable of matching or even outperforming (32-shot and) zero-shot performance of many state-of-the-art LLMs with a fraction of the parameters.

Our ablation study shows that depending on the task, we can achieve substantial few-shot and zero-shot generalizability with as few as 1,000 statements per training dataset or approximately 16,000 training statements in total, which correspond to even fewer original task examples since one example can be turned into multiple statements through different templates. Furthermore, we find that the statement and task diversity tend to have a beneficial effect on the performance and generalizability of Statement-Tuning. In summary, our primary contributions are:

1. To the best of our knowledge, we are the first to propose a combination of elaborate statement formulation and Masked Language Models as a simple and effective data/resource-efficient alternative for LLMs for zero-shot NLU task generalization.
2. Through extensive experimentation and comparison, we demonstrate that Statement-Tuning performs on par with and in many instances exceeds the performance of state-of-the-art supervised fine-tuned LLMs with 200 times fewer parameters. Moreover, we show that Statement-Tuning outperforms previous approaches on encoders and less reliant on superficial lexical clues.
3. We explore a large number of design choices to study how Statement-Tuning benefits from the number of statement examples, the statement template diversity and task diversity in multitask Statement-Tuning, and demonstrate the data/resource-efficiency of Statement-Tuning.

2 Related Work

Zero-Shot and Few-Shot Approaches Utilizing Label Semantics Various approaches have been proposed to reformulate zero-shot classification to leverage label semantics instead of indices, enabling the more generalized use of encoder models. In the reformulated task, textual labels are combined with the original input text, and the model should predict whether the label matches the text. TARS (Halder et al., 2020) utilizes the simple concatenation of input text and label text, with no explicit connection through natural language. Similarly, Xu et al. (2023) reformulate discriminative

tasks with minimal prompts, which are mostly simple concatenations of elements in the raw input. They find it effective for zero-shot generalization for ELECTRA-style encoder models but do not test other encoder-only models with normal MLM training.

Yin et al. (2019) propose an entailment-based approach where the input is a pair of texts: the original input text is the premise and label is converted into a hypothesis. However, the approach is limited by the fact that not every discriminative task can be formulated as an entailment task (in the form of a premise and a hypothesis). The entailment formulation also limits the training tasks to entailment datasets (MNLI, RTE, and FEVER), which often causes the model to over-rely on spurious lexical patterns, hindering generalization (Ma et al., 2021). Our approach Statement-Tuning serves as a more universal formulation of any discriminative task in the form of statements and demonstrates less reliance on superficial lexical clues.

Few-shot approaches utilizing cloze-style templates have also been proposed. PET (Schick and Schütze, 2021a) requires an ensemble of encoder models and iterative training, relies on additional unlabeled data; Improved PET variants (Schick and Schütze, 2021b; Tam et al., 2021) use complex losses to compute the probability of each token in multi-token labels. To automate verbalizer construction, Zhao et al. (2023) retrieve label words from the PLM's embedding space, but their prompt with a single masked token for label word prediction primarily supports simple classification tasks. Our method keeps the normal sequence classification training paradigm, while in the meantime effectively leverages the label semantics, enabling straightforward zero-shot generalization to a broader range of tasks.

Zero-Shot Prompting and Multitask Tuning LLMs excel at unseen-task/zero-shot generalization (Brown et al., 2020). Building on this, recent work explores multitask training with diverse prompts for improved zero-shot performance (Sanh et al., 2022; Wei et al., 2022; Chung et al., 2024). These methods fine-tune large models on constructed datasets with various task prompts, achieving strong zero-shot results on unseen tasks. However, effective instruction-tuning often requires billions of parameters (Zhang et al., 2024), limiting their application to smaller models. Ye et al. (2022) aim to distill this zero-shot ability in a smaller model like an LSTM through synthetic data creation using an LLM, but they create task-specific models rather than a single smaller model that is capable of generalizing. Our work demonstrates similar or superior generalization than LLMs using a single smaller MLM with less training data.

Task: MNLI

Premise: Conceptually cream skimming has two basic dimensions - product and geography.

Hypothesis: Product and geography are what make cream skimming work.

Options: ["entailment", "neutral", "contradiction"]

Statement Conversion:

S_1: "Conceptually cream skimming has two basic dimensions - product and geography" entails "Product and geography are what make cream skimming work".
S_2: "Conceptually cream skimming has two basic dimensions - product and geography" is neutral with regards to "Product and geography are what make cream skimming work".
S_3: "Conceptually cream skimming has two basic dimensions - product and geography" contradicts "Product and geography are what make cream skimming work".

Figure 2: Example conversion of the MNLI task to natural language statements.

3 Method: Statement-Tuning

In this section, we outline the steps involved in Statement-Tuning. First, tasks are verbalized into natural language statements. Then they are used to train the statement discriminator and derive the target label.

Task Verbalization Any discriminative task with a finite set of targets can be verbalized into a finite set of natural language statements. Figure 2 shows the example of converting the MNLI task into statements. Similar to prompting, each task has its own statement templates, based on each possible label. The truth label for training purposes on each statement depends on whether the statement contains the correct target label or not.

Statement Fine-Tuning To create the training data for multitask statement fine-tuning, we exhaustively generate statements across 16 diverse NLP datasets (categorized into 9 tasks, see Appendix E) using many varied statement templates (see Appendix A) per dataset: QQP (Sharma et al., 2019), Winogrande (Sakaguchi et al., 2020), PiQA (Bisk et al., 2020), MNLI (Williams et al., 2018), SNLI (Bowman et al., 2015), Mintaka (Sen et al., 2022), Yelp Polarity (Zhang et al., 2015), WikiLingua (Ladhak et al., 2020), SQuAD (Rajpurkar et al., 2016), TweetEval's Offensive task (Zampieri et al., 2019), Massive (FitzGerald et al., 2023; Bastianelli et al., 2020), Definite Pronoun Resolution (Rahman and Ng, 2012), QASC (Khot et al., 2020), SciQ (Welbl et al., 2017), RACE (Lai et al., 2017),

and SAMSum (Gliwa et al., 2019). We fine-tune RoBERTa (Liu et al., 2019) with a binary sequence classification head to predict the truth value of the statements. By fine-tuning the model across diverse tasks, templates, and domains, the model should be able to generalize across unseen templates and tasks, as long as it can be phrased as a true/false statement.

Zero-Shot and Few-Shot Inference To perform inference on statement-finetuned RoBERTa, we also need to transform the input into statements. We randomly choose a statement template for each dataset at inference time. In our experiments, we show that our statement-tuning is robust to different templates. We exhaustively generate a statement for each possible label, as shown in Figure 1. Then, for each statement corresponding to each label, we predict the probability of such a statement being true. The final label is the statement with the highest true probability. Zero-shot inference is done by directly performing the aforementioned inference regime on the statement-finetuned RoBERTa, while K-shot inference is done after continual fine-tuning on K examples of task-specific statements.

4 Experimental Setup

4.1 Evaluation Datasets

We measure our model's generalizability using another set of 7 diverse datasets representing a variety of unseen tasks or unseen domains: Balanced COPA (BCOPA; Kavumba et al., 2019; Roemmele et al., 2011), MRPC (Dolan and Brockett, 2005), Emotion (Saravia et al., 2018), Amazon Polarity (McAuley and Leskovec, 2013; Zhang et al., 2015), FigQA (Liu et al., 2022), StoryCloze (2016) (Mostafazadeh et al., 2017), and Yahoo Answers Topics (Zhang et al., 2015). Among the evaluation data, MRPC (paraphrase identification) and Amazon Polarity (sentiment analysis) represent tasks seen during training but in different domains and demonstrate *cross-domain* generalizability. The rest are unseen tasks and hence examine the *cross-task* generalizability.

4.2 Statement Finetuning Configurations

We statement-finetune both RoBERTa-base and RoBERTa-large across diverse NLP tasks outlined in Section 3. However, as statement fine-tuning expands the dataset with various templates over all possible labels, it is arguably unwise to fine-tune on all possible generated statements. Moreover, each task has a different data size, leading to unbalanced fine-tuning data. Therefore, we sample statements randomly for each task, uniformly across true and false statements. In true/false statements, we also balance original classes. We explore sample size from 1,000 statements to 50,000 statements per dataset. We encourage the invariance to phrasing in diverse statements by designing multiple statement templates per dataset (a list of all statement templates is shown in Appendix A). Furthermore, we run the training five times to account for randomness in training data creation. In the evaluation, we randomly pick a template for each dataset in a single evaluation run and also repeat the evaluation five times. We thus report the mean and standard deviation of 5×5 runs to show the general task accuracy and the (in)variance to phrasing. We also explore the effect of statement diversity during training in Section 5.6.

After multitask statement tuning is completed, we can further continue fine-tuning the model on the target downstream dataset. Specifically, we explore various n-shot configurations: Full/3,000-shot, 1,000-shot, 500-shot, 200-shot, and 32-shot, where we use limited data from the training sets of the corresponding dataset to fine-tune our statement-tuned models. For the Full/3,000-shot case, we cap the training set at 3,000 examples, otherwise, we use the entire set (this is the case for Amazon Polarity only). For StoryCloze, there is no training set, so we just carry out 32-shot (using 32 samples from the test set for fine-tuning and evaluating on the rest) and zero-shot experiments. As for Yahoo Answers Topic and Emotion, due to them being multi-class classification tasks, we cap the n-shot analysis at 200-shot due to the larger number of choices per example (and hence a larger number of statements per example).

4.3 Other Baselines

To assess the feasibility of our approach, we compare Statement-Tuned RoBERTa base/large models with 125 million parameters and 355 million parameters respectively with a range of competitive multitask fine-tuned encoder-decoder models and decoder-only LLMs spanning a parameter range from 60 million parameters to 70 billion parameters. We include the following open-source models: Meta-Llama-3-70B-Instruct (AI@Meta, 2024), Llama-2-13B-chat, Llama-2-7B-chat (Tou-

	#Parameters	BCOPA	MRPC	FigQA	Amazon Polarity	StoryCloze	YA Topic	Emotion	Avg
Meta-Llama-3-70B-Instruct	70B	89.0	71.3	42.0	94.7	82.7	61.9	51.8	70.5
Llama-2-13b-chat-hf	13B	89.6	60.8	40.9	93.7	82.4	53.2	51.6	67.5
Llama-2-7b-chat	7B	86.6	54.4	40.1	90.5	78.5	47.8	50.0	64.0
Mistral-7B-Instruct-v0.2	7B	89.4	73.0	41.4	88.9	82.3	57.7	55.3	69.7
Qwen1.5-7B-Chat	7B	87.0	75.5	42.1	95.3	79.7	59.1	57.8	70.9
Pythia-6.9B	6.9B	82.2	62.0	41.7	83.3	71.2	32.2	25.1	56.8
Pythia-2.8B	2.8B	79.6	68.4	41.2	77.7	69.7	12.1	35.4	54.9
Phi-2	2.7B	87.2	67.9	41.8	86.6	77.7	38.7	53.1	64.7
FlanT5-Large	770M	67.6	81.1	40.1	96.0	63.0	51.0	59.9	65.5
Qwen1.5-0.5B-Chat	500M	69.2	32.6	38.7	69.7	68.9	21.9	6.6	43.9
BART-large-mnli	406M	50.4	35.8	46.9	49.4	47.3	6.5	11.7	35.4
FlanT5-Small	60M	52.8	31.9	42.0	88.8	51.5	24.5	21.7	44.7
Other encoder-only Approaches:									
NPPrompt (RoBERTa-large) (Zhao et al., 2023)	355M	-	46.0	-	80.4	-	46.0	36.0	-
NLI (Yin et al., 2019)	355M	61.8	60.8	65.3	91.5	76.0	45.0	48.5	64.1
Our Approach:									
RoBERTa-base (Best)	**125M**	$75.3_{(0.5)}$	$72.3_{(1.5)}$	$61.4_{(0.6)}$	$92.9_{(1.3)}$	$79.1_{(1.1)}$	$40.2_{(3.8)}$	$48.5_{(5.1)}$	67.1
RoBERTa-base (4k)	**125M**	$72.4_{(0.5)}$	$69.6_{(1.1)}$	$60.7_{(0.9)}$	$92.3_{(0.8)}$	$78.5_{(2.7)}$	$37.9_{(2.7)}$	$46.6_{(4.3)}$	65.4
RoBERTa-large (Best)	**355M**	$85.1_{(0.7)}$	$71.8_{(0.8)}$	$74.2_{(1.4)}$	$95.4_{(0.4)}$	$92.1_{(0.7)}$	$49.9_{(2.1)}$	$50.7_{(1.4)}$	75.3
RoBERTa-large (10k)	**355M**	$85.1_{(0.7)}$	$71.5_{(0.8)}$	$73.0_{(2.4)}$	$95.4_{(0.4)}$	$91.1_{(0.8)}$	$48.4_{(0.7)}$	$49.1_{(3.2)}$	73.4
Full/3000-shot:									
RoBERTa-base (FT)	125M	74.2	87.0	88.1	94.3	-	71.0	82.2	-
RoBERTa-large (FT)	355M	86.0	87.6	92.0	96.5	-	68.5	78.2	-

Table 1: Comparison of our approach against many pre-trained open-source encoder-decoder and decoder-only Large Language Models as well as two other encoder-only models on 7 Natural Language Understanding tasks in zero-shot conditions. FT stands for Full Fine-tuning and is included as upper bounds for reference. For Statement-Tuning, we report the average across 5 training runs and 5 evaluation runs and include the average standard deviation in parenthesis. We highlight all scores in gray where our approach with RoBERTa-base (best) exceeds or is equal to the score given by the model.

vron et al., 2023), Mistral-7B-Instruct-v0.2 (Jiang et al., 2023), QWEN1.5-7B-chat and QWEN1.5-0.5B-chat (Bai et al., 2023), Pythia-6.9B and Pythia-2.8B (Biderman et al., 2023), Phi-2 (Li et al., 2023), FlanT5-Large and FlanT5-Small (Chung et al., 2024), and BART-large-mnli (Lewis et al., 2020).

We use the chat/instruction-tuned version of the models to allow for better instruction following. We try to select models that have not seen the evaluation data to the best of our knowledge, however, the training data of many of these models is not fully outlined and there can always be the possibility of contamination (Li and Flanigan, 2024). Although these models have already been trained on a large number of instruction fine-tuning datasets, to guarantee a fair comparison as much as possible, we additionally instruction-tune a subset of the models using LoRA (Hu et al., 2022) on the same training datasets used for Statement-Tuning.[2] Details regarding data formatting, hyper-parameters, and results are reported in Appendix H.

We train and evaluate all the models on a configuration of 5 AMD EPYC Rome CPU cores and at most 4 Nvidia Tesla A100 40GB GPUs (we only use 4 GPUs for inference of the largest LLMs, and 1 GPU for Statement-Tuning). The prompts and evaluation are derived from the Language Model Evaluation Harness library (Gao et al., 2024).

To compare with other zero-shot encoder-only methods, we evaluated against the NLI-based approach from Yin et al. (2019) and NPPrompt (Zhao et al., 2023). For the NLI method, we used a RoBERTa-large model fine-tuned on SNLI (Bowman et al., 2015), MNLI (Williams et al., 2018), FEVER-NLI (Thorne et al., 2018), and ANLI (R1, R2, R3) by Nie et al. (2020), with templates detailed in Appendix I. For NPPrompt, we used the authors' original code, evaluating only classification tasks due to adaptation challenges for other task types.

5 Results and Analysis

In this section, we dive deep into the results of our experimentation to derive insights about our approach.

5.1 Overall Result

Table 1 shows zero-shot performance of statement-tuned RoBERTa and baselines. Recall that we explore various statement-tuning sizes, hence here we report the best performance across all training sizes and performance for the 4,000 and 10,000 sample

[2]Due to limited computational resources, we are only able to perform this extended training and analysis on a subset of the models ranging from 500M to 13B parameters.

Model	Shuffled	BCOPA	MRPC	FigQA	Amazon Polarity	StoryCloze	Yahoo Topic	Emotion
NLI	No	61.8	60.8	65.3	91.5	76	45	48.5
	Yes	58.4	63.1	61.4	72.2	67.8	35	37.8
Ours (10k)	No	85.1	71.5	73.0	95.4	91.1	48.4	49.1
	Yes	72.6	61.7	62.0	84.0	72.0	42.5	44.0
NLI	% Drop from Shuffling:	5.5	-3.8	6.0	21.1	10.8	22.2	22.1
Ours (10k)	% Drop from Shuffling:	14.7	13.7	15.1	11.9	21.0	12.2	10.4

Table 2: Comparison of our method with Yin et al. (2019) on all the tasks. We also rerun our analysis with shuffled words in the input and compare the percentage drop in performance after shuffling.

sizes per dataset for the base and large models, respectively. The effect of statement tuning sample size is explored in Section 5.4.

Statement-Tuning Enables Effective Zero-Shot Generalization on Masked Language Models. The result shows that the multitask statement-tuned encoder model can achieve zero-shot generalization across unseen tasks and domains. On BCOPA (unseen task) and Amazon Polarity (unseen domain), our zero-shot statement-tuned models even achieve accuracies on par with models fine-tuned on the full datasets. We also see that the larger model (RoBERTa-large) achieved much better generalization than the base model in general.

5.2 Comparison Against Larger Zero-Shot Models

Our approach is also competitive against other pretrained open-source encoder-decoder and decoder-only Large Language Models under zero-shot prompting. Despite having significantly fewer parameters than all the models reported (except for FlanT5-small), our approach matches or exceeds many of them on the tasks reported. In average, our best Statement-Tuned RoBERTa-large model with only 355M parameters outperforms the best performing LLM (Qwen1.5-7B-Chat) by 4.4 and the largest LLM (Meta-Llama-3-70B-Instruct, with approximately 200 times the number of parameters) by 4.8. It is worth noting that our RoBERTa-base models with only 125M parameters almost completely outperforms all models under or equal to 6.9B parameters (except for FlanT5-Large) on all tasks (except for BCOPA). Our models are dominant on FigQA and StoryCloze, both of which are unrepresented in the training data, with the best performing RoBERTa-large model scoring an additional **32.2** and **9.4** points over Llama3-70B-Instruct on the accuracy respectively.

We observe similar results in the 32-shot setting (see Appendix C) and when the LLMs are additionally instruction-tuned on the same data (see Appendix H). These results demonstrate the capabilities of much smaller encoder models as being accurate and light alternatives (in terms of parameters; for speed comparison see Appendix G) to LLM zero-shot (and few-shot) prompting in natural language understanding.

5.3 Comparison with other Encoder Methods

Our approach consistently outperforms the NLI baseline proposed by Yin et al. (2019) across tasks when using RoBERTa-large and across most tasks when using RoBERTa-base (except FigQA and Yahoo Topic). The performance gap is more evident in multiple-choice tasks (BCOPA, FigQA, StoryCloze) than in simpler sentence classification tasks, suggesting NLI training is less effective for multiple-choice scenarios. Compared to NPPrompt (Zhao et al., 2023), our method significantly outperforms on all tasks with RoBERTa-large. Even when comparing Statement-Tuned RoBERTa-base to NPPrompt RoBERTa-large, we perform better on all tasks except Yahoo Topic. Despite requiring multitask fine-tuning, the performance gains, the versatility and the modest training set sizes justify our approach.

Robustness to Spurious Patterns. Regarding the concerns raised by Ma et al. (2021), we evaluated our method's reliance on shallow lexical patterns by measuring accuracy drops after randomly permuting input tokens (Table 2). Our multi-task Statement-Tuning model, trained on diverse tasks, exhibited a greater accuracy drop compared to the NLI baseline, indicating less reliance on spurious patterns. Notably, the NLI model's accuracy on MRPC improved by 3.8% post-perturbation, confirming its reliance on lexical clues.

The only tasks where our method did not show a larger accuracy drop than NLI were Sentiment Analysis, Topic Classification, and Emotion Classification, which are tasks typically less dependent on word order and reasoning. This lower drop may reflect enhanced robustness rather than reliance on

Statement Sample	Average accuracy	
	RoB-base	RoB-large
1,000	63.0	71.1
2,000	62.7	71.0
3,000	63.1	73.3
4,000	**65.4**	72.9
5,000	64.7	72.2
10,000	64.3	**73.4**
20,000	64.9	68.6
40,000	64.1	72.0
50,000	58.5	68.2

Table 3: Average accuracy over all evaluation tasks when trained with different statement sample size per dataset.

Dataset Size	BCOPA	MRPC	FIGQA	AP	S-Cloze	Yahoo Topic	Emotion	AVG
80k	$72.6_{(1.3)}$	$66.1_{(5.9)}$	$60.9_{(1.2)}$	$92.3_{(0.6)}$	$73.0_{(5.7)}$	$39.4_{(2.9)}$	$46.6_{(3.4)}$	64.4
70k	$71.3_{(0.9)}$	$63.5_{(7.4)}$	$60.3_{(1.0)}$	$89.9_{(2.4)}$	$74.3_{(6.6)}$	$28.6_{(2.3)}$	$46.3_{(4.2)}$	62.0
65k	$71.7_{(1.4)}$	$66.8_{(4.2)}$	$58.8_{(2.1)}$	$92.3_{(0.4)}$	$75.1_{(4.4)}$	$33.0_{(1.5)}$	$45.9_{(2.7)}$	63.4
60k	$71.1_{(0.6)}$	$69.5_{(3.0)}$	$59.2_{(2.3)}$	$92.6_{(0.5)}$	$67.9_{(8.3)}$	$29.2_{(2.8)}$	$44.3_{(4.7)}$	62.0
55k	$71.1_{(1.4)}$	$69.0_{(2.1)}$	$59.0_{(1.5)}$	$91.6_{(0.6)}$	$72.1_{(6.7)}$	$25.8_{(3.0)}$	$45.6_{(5.9)}$	62.0
50k	$64.9_{(1.8)}$	$69.4_{(2.8)}$	$55.6_{(3.3)}$	$58.2_{(6.6)}$	$55.5_{(6.4)}$	$24.4_{(2.0)}$	$38.6_{(6.6)}$	52.4
25k	$52.3_{(1.9)}$	$61.1_{(1.1)}$	$49.2_{(2.5)}$	$50.5_{(6.4)}$	$53.0_{(3.9)}$	$18.3_{(1.9)}$	$20.7_{(6.4)}$	43.6

Table 4: Effect of increasing both task diversity and dataset size on Statement-Tuned RoBERTa-base. AP denotes Amazon Polarity. The data size of each training task is fixed at 5000 and task diversity is increased the same way as in Section 5.7.

lexical patterns.

5.4 Statement Finetuning Sample Size

Recall that we only perform statement fine-tuning on a sample of all possible statements from the training dataset. Here, we explore the effect of sample size per dataset in the multitask statement fine-tuning on both zero-shot and few-shot performance.

Zero-Shot. As shown in Table 3, with only 1k samples per datasets, we can already reach 96% of the best performance, which is obtained with 4k samples on RoBERTa-base and 10k samples on RoBERTa-large, showing the sample-efficiency of statement-tuning. For RoBERTa-base, introducing more data after 4,000 samples does not further improve the accuracy on downstream tasks and even causes a decrease. RoBERTa-large benefits from a larger fine-tuning data size with the best average performance observed when the statement number per training set increases to 10,000, more than doubling the optimal sample size of RoBERTa-base. We hypothesize that this is due to a larger capacity to understand and discriminate between natural language statements which allows RoBERTa-large to benefit from more training data as opposed to RoBERTa-base, which has a more limited capacity to develop a general semantic understanding of the truthfulness of statements. Nonetheless, we also observe a decrease in RoBERTa-large's performance after 10k samples. The existence of a point of diminishing returns in both models when it comes to Statement-Tuning training data sizes indicates that too many samples may lead to overfitting to the training tasks, which affects the generalizability to unseen tasks. To address overfitting concerns, we increased both dataset size and task diversity, observing consistent performance gains (Table 4). This suggests prior issues were indeed likely due to overfitting. Hence, we recommend a fixed dataset size of 5,000 per training task and expanding training data by adding more tasks rather than increasing data size of individual tasks.

Few-Shot. While the statement-tuned model shows zero-shot generalization, we can further fine-tune the model on the target downstream task. As seen in Figure 3, we investigate the effect of both the multitask statement-tuning sample size and the number of shots from the target tasks on the n-shot performance on the 7 evaluation datasets.

When increasing the multitask statement-tuning sample size, we observe a trend in n-shot performance similar to the general zero-shot performance shown in Table 3. For example, the optimal data size is achieved at around 4k~5k on COPA, Emotion and Yahoo, and there is an apparent drop in accuracy across different shot numbers and tasks when increasing the sample size from 40k to 50k. There turns out to be a high degree of correlation among all n-shot and 0-shot performance (see Figure 6), indicating that observed trends in the 0-shot scenario can be informative for the few-shot cases.

However, the results seem to indicate a general trend of diminishing returns past using 200-shot fine-tuning. Nevertheless, it seems that a great deal of the potential performance is achieved with the zero-shot application of the approach, hence further supporting the utility of our approach when task-specific data is scarce.

5.5 Comparison with Standard Fine-Tuning

To observe the improvement over regular fine-tuning of RoBERTa-base, we also include Figure 4, where the y-axis, Delta, represents the improvement over regular fine-tuning for the particular n-shot. For zero-shot, we take random choice as the baseline. Generally, continually fine-tuning our model is better than fine-tuning vanilla RoBERTa under an extremely low N-shot setting. However, in some instances such as BCOPA and (to a certain

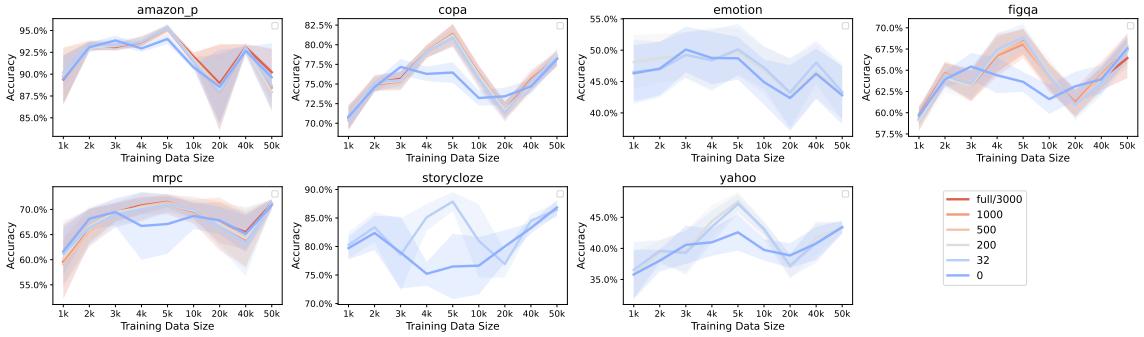

Figure 3: N-shot accuracy of Statement-Tuned RoBERTa-base models across training datasets of different sizes. The x-axis denotes the number of statements per Statement-Tuning training dataset, with the number of training datasets fixed.

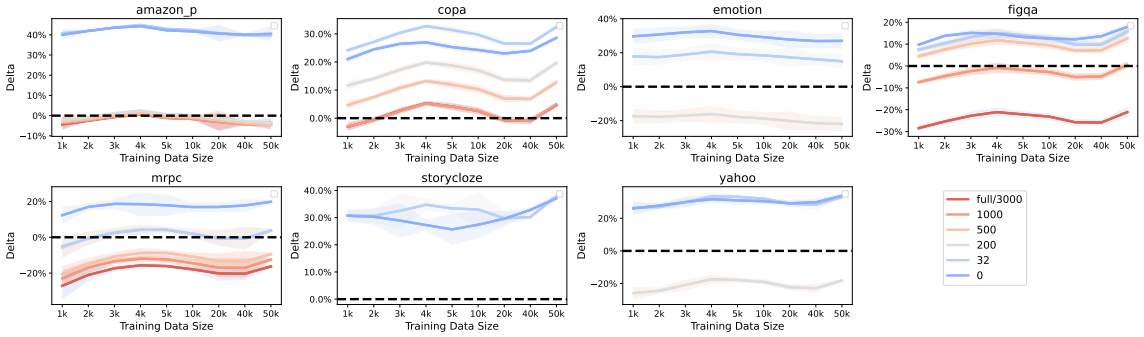

Figure 4: N-shot improvement of Statement-Tuned RoBERTa-base with varying training set sizes over standard fine-tuning. The y-axis, Delta, is the difference between the accuracy of the Statement-Tuned model and the accuracy achieved by regular fine-tuning of RoBERTa-base on the task. A positive Delta indicates improvement over the baseline approach.

SPC	BCOPA	MRPC	FigQA	AP	S-Cloze	YA Topic	Emotion	AVG
1	91.0$_{(1.7)}$	**74.8**$_{(2.2)}$	**49.8**$_{(4.8)}$	59.1$_{(0.5)}$	58.4$_{(17.1)}$	78.0$_{(3.4)}$	**41.2**$_{(2.1)}$	**62.6**$_{(6.9)}$
2	91.3$_{(1.0)}$	70.2$_{(1.4)}$	49.2$_{(3.1)}$	**61.6** $_{(1.1)}$	56.9$_{(8.3)}$	**79.9**$_{(2.2)}$	33.5$_{(\mathbf{1.4})}$	60.5$_{(3.6)}$
3	**93.0**$_{(\mathbf{0.3})}$	73.2$_{(\mathbf{0.7})}$	43.2$_{(\mathbf{1.0})}$	60.5$_{(1.1)}$	64.3$_{(6.7)}$	74.2$_{(3.0)}$	31.3$_{(2.4)}$	59.5$_{(\mathbf{3.0})}$
4	92.1$_{(\mathbf{0.3})}$	70.9$_{(1.6)}$	49.4$_{(\mathbf{1.0})}$	59.9$_{(\mathbf{0.4})}$	**68.1**$_{(\mathbf{6.4})}$	68.0$_{(7.5)}$	38.8$_{(2.1)}$	61.9$_{(3.9)}$
5	92.4$_{(0.5)}$	69.6$_{(1.0)}$	46.2$_{(3.5)}$	60.5$_{(1.1)}$	66.8$_{(6.7)}$	78.5$_{(\mathbf{2.1})}$	35.9$_{(3.0)}$	61.6$_{(3.2)}$

Table 5: The Zero-shot performance of the base model using various degrees of SPC, where a larger SPC value indicates greater statement diversity during training. We report the average as the geometric mean of the task performance to account for the differing accuracy ranges of each task. Each value is a mean over 5 evaluation runs and we include the standard deviation in the parentheses.

extent) FigQA, we tend to observe a benefit against regular fine-tuning even for a higher number of few-shot examples.

Our approach is recommended in extreme few-shot and zero-shot scenarios. When more data is available, directly fine-tuning RoBERTa-base is better. Our method's good performance with limited data can be attributed to improved generalizability from multitask statement tuning and data augmentation effect of statements generated from few-shot examples, which enhances data efficiency.

5.6 Effect of Statement Diversity

As part of our investigation of Statement-Tuning, we would like to explore the effect of template diversity during Statement-Tuning. We hypothesize that randomly applying a larger number of different statement templates per training corpus will allow for improved performance on unseen tasks, as it will make the model more robust to the phrasing of statement templates and prevent it from relying on superficial cues in certain templates.

In our main experiments, each dataset employs several templates (see Appendix A). In this experiment, we limit each corpus to only use the maximum of N different templates, which we call Statements per Category (SPC). We statement-tune RoBERTa base models with a fixed training set size of 4,000 statements per training corpus with a varying level of SPC.

Table 5 shows that though BCOPA and StoryCloze benefits from a larger SPC, increasing SPC doesn't always boost average task performance, with the highest being 62.6 at SPC 1. However,

Statement-Tuning Training									Evaluation							
PD	CR	NLI	QnA	SA	WSD	IC	OLI	SU	BCOPA	MRPC	FIGQA	AMAZON P.	StoryCloze	YA Topic	Emotion	AVG
x	x	x	x	x	x	x	x	x	$71.0_{(0.9)}$	$65.7_{(3.1)}$	$59.8_{(1.0)}$	$90.7_{(1.3)}$	$75.1_{(3.8)}$	$36.8_{(2.7)}$	$46.2_{(3.8)}$	$61.2_{(2.7)}$
x	x	x	x	x	x	x	x		$69.8_{(2.6)}$	$65.9_{(6.1)}$	$60.4_{(0.4)}$	$91.2_{(0.7)}$	$79.8_{(1.7)}$	$29.4_{(3.7)}$	$47.1_{(0.4)}$	$60.0_{(3.0)}$
x	x	x	x	x	x	x			$70.0_{(0.3)}$	$64.2_{(7.9)}$	$59.3_{(0.2)}$	$92.0_{(0.3)}$	$70.5_{(6.4)}$	$31.2_{(2.3)}$	$49.4_{(3.0)}$	$59.6_{(4.1)}$
x	x	x	x	x	x				$68.7_{(2.1)}$	$64.6_{(6.8)}$	$58.8_{(0.7)}$	$91.3_{(0.5)}$	$77.3_{(7.0)}$	$18.7_{(3.0)}$	$53.4_{(4.9)}$	$56.5_{(4.4)}$
x	x	x	x	x					$70.2_{(0.5)}$	$67.0_{(4.6)}$	$59.4_{(0.8)}$	$91.8_{(0.1)}$	$73.4_{(6.7)}$	$20.5_{(3.4)}$	$52.2_{(3.0)}$	$57.2_{(3.5)}$
x	x	x	x						$70.0_{(1.2)}$	$67.4_{(2.5)}$	$59.2_{(0.3)}$	$78.0_{(12.0)}$	$75.3_{(10.1)}$	$36.6_{(3.3)}$	$40.3_{(2.2)}$	$58.8_{(6.2)}$
x	x	x							$50.2_{(1.7)}$	$40.6_{(8.5)}$	$50.9_{(1.7)}$	$55.9_{(5.5)}$	$50.2_{(6.6)}$	$3.8_{(1.8)}$	$7.2_{(2.8)}$	$26.0_{(4.8)}$

Table 6: Comparison of the effect of reducing task diversity in the training of Statement-Tuning models on zero-shot accuracy on unseen datasets. The last column is the average using the geometric mean to account for the different accuracy ranges of the different evaluation sets. The total training set size remains constant at approximately 100,000 statements across all configurations.

average standard deviation drops significantly from 6.9% to 3.6% when SPC increases from 1 to 2, reaching a low of 3.0% at SPC 3. This suggests that more template diversity improves stability and consistency. Therefore, using at least 2 different templates is recommended for better robustness.

5.7 Effect of Task Diversity

We examine the importance of task variety in Statement-Tuning. Our Statement-Tuning datasets can be grouped into 9 task categories: Summarization (SU), Sentiment Analysis (SA), Question Answering (QA), Natural Language Inference (NLI), Commonsense Reasoning (CR), Paraphrase Detection (PD), Word Sense Disambiguation (WSD), Intent Classification (IC), and Offensive Language Identification (OLI). See Appendix E for the dataset breakdown. We perform statement tuning on RoBERTa-base with various task subsets, dynamically sampling data to maintain 100k total statements.

Table 6 shows the zero-shot performance of the statement tuning approach with a fixed training set size but varying task types. Average performance increases from 26.0 to 61.2 as the number of tasks is increased from the minimum of 3 to the maximum of 9. Robustness also improves, shown by a decrease in average standard deviation from 4.8% to 2.7%. This demonstrates that increasing training task diversity can boost performance and reduce variance. Unsurprisingly, the inclusion of the Sentiment Analysis task substantially improves the performance on Amazon Polarity from the same task category. Another related task, Emotion, also shows a large increase after adding the SA task. Although the inclusion of SA and WSD hurts the performance on a dissimilar task, Yahoo Answer Topic, the accuracy is recovered after adding the more related Intent Classification task, and reaches the highest 36.8 when training with all tasks. However, the enhancement of downstream tasks does not always come from similar training tasks. More interestingly, adding the QA task leads to a significant jump in the performance of all evaluation tasks. Though Paraphrase Identification is always included in the training, MRPC still benefits from the QA task, reflected by a great improvement of 26.8. Both related and unrelated training tasks can have a positive effect on the downstream tasks, highlighting the value of task diversity in multitask statement-tuning. Sometimes adding an unrelated task causes a performance drop on certain datasets, e.g. StoryCloze after adding IC, but including more tasks alleviates the problem, again confirming the advantage of task diversity.

6 Conclusion

As part of their emergent abilities, LLMs generalize to many unseen tasks/domains through few-shot and zero-shot prompting, but are prohibitively computationally expensive and difficult to adapt. To address this issue, we introduce Statement-Tuning, a novel technique for few-shot and zero-shot task generalization for encoder models. We find that this approach can match or outperform few-shot and zero-shot prompting of many much larger decoder-only or encoder-decoder models on many tasks at a fraction of the parameters. Additionally, our approach offers both performance and robustness gains over previous encoder-only approaches. Experimentation shows that the approach can be leveraged by training on as few as 16,000 statements. We find training task and statement template diversity to be generally helpful. We speculate that the benefits of this approach could extend beyond task generalization and could prove useful for cross-lingual task transfer, and would like to explore this in future work.

Limitations

While our approach offers advantages in computational efficiency compared to LLMs, the cost scales with the number of possible targets due to the requirement of one forward pass per label. That being said, it is still possible to apply our method in extreme multi-class classification because we do not have to use all possible statements with all possible labels for training, as our model is trained exactly to generalize to unseen labels by learning the relation of label semantics and the input text. Additionally, task-specific *full* fine-tuning can still achieve better performance in the presence of more training data. Therefore, we recommend the use of our approach in the low-resource/no-resource scenario. Furthermore, our method can be sensitive to the Statement-Tuning training set size and other hyperparameters, hence some exploration of ideal hyperparameters may be required before employing Statement-Tuned models. In addition, we limit our analysis only to English, it would be interesting to observe whether the technique enables cross-lingual transfer but we leave this to future work. Finally, our reliance on encoder-based models restricts its application to Natural Language Understanding tasks, excluding tasks like translation or abstractive summarization.

Ethics Statement

We affirm our commitment to more accessible and climate-aware NLP, and hope this work inspires more computationally efficient approaches to NLP. All data and models we use are publicly available. Furthermore, the success of Statement-Tuning relies on fine-tuning pretrained encoder models, which are pretrained on large datasets, and hence, Statement-Tuning is susceptible to inheriting and enforcing any harmful biases existing in the pretraining data.

Acknowledgements

Yongxin Huang is supported by HUAWEI Technologies (Ireland) Co., Ltd.

References

AI@Meta. 2024. Llama 3 model card.

Mikel Artetxe, Shruti Bhosale, Naman Goyal, Todor Mihaylov, Myle Ott, Sam Shleifer, Xi Victoria Lin, Jingfei Du, Srinivasan Iyer, Ramakanth Pasunuru, Giridharan Anantharaman, Xian Li, Shuohui Chen, Halil Akin, Mandeep Baines, Louis Martin, Xing Zhou, Punit Singh Koura, Brian O'Horo, Jeffrey Wang, Luke Zettlemoyer, Mona Diab, Zornitsa Kozareva, and Veselin Stoyanov. 2022. Efficient large scale language modeling with mixtures of experts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11699–11732, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *CoRR*, abs/2309.16609.

Emanuele Bastianelli, Andrea Vanzo, Pawel Swietojanski, and Verena Rieser. 2020. SLURP: A spoken language understanding resource package. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7252–7262, Online. Association for Computational Linguistics.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 2397–2430. PMLR.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. PIQA: reasoning about physical commonsense in natural language. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7432–7439. AAAI Press.

Samuel R. 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, Lisbon, Portugal. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind

Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2024. Scaling instruction-finetuned language models. *J. Mach. Learn. Res.*, 25:70:1–70:53.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing, IWP@IJCNLP 2005, Jeju Island, Korea, October 2005, 2005*. Asian Federation of Natural Language Processing.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2023. MASSIVE: A 1M-example multilingual natural language understanding dataset with 51 typologically-diverse languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4277–4302, Toronto, Canada. Association for Computational Linguistics.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. A framework for few-shot language model evaluation.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79, Hong Kong, China. Association for Computational Linguistics.

Kishaloy Halder, Alan Akbik, Josip Krapac, and Roland Vollgraf. 2020. Task-aware representation of sentences for generic text classification. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3202–3213, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Pride Kavumba, Naoya Inoue, Benjamin Heinzerling, Keshav Singh, Paul Reisert, and Kentaro Inui. 2019. When choosing plausible alternatives, clever hans can be clever. In *Proceedings of the First Workshop on Commonsense Inference in Natural Language Processing*, pages 33–42, Hong Kong, China. Association for Computational Linguistics.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. 2020. Qasc: A dataset for question answering via sentence composition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8082–8090.

Faisal Ladhak, Esin Durmus, Claire Cardie, and Kathleen McKeown. 2020. WikiLingua: A new benchmark dataset for cross-lingual abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4034–4048, Online. Association for Computational Linguistics.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In

Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 785–794, Copenhagen, Denmark. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Changmao Li and Jeffrey Flanigan. 2024. Task contamination: Language models may not be few-shot anymore. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 18471–18480. AAAI Press.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need II: phi-1.5 technical report. *CoRR*, abs/2309.05463.

Emmy Liu, Chenxuan Cui, Kenneth Zheng, and Graham Neubig. 2022. Testing the ability of language models to interpret figurative language. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4437–4452, Seattle, United States. Association for Computational Linguistics.

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. *CoRR*, abs/1907.11692.

Tingting Ma, Jin-Ge Yao, Chin-Yew Lin, and Tiejun Zhao. 2021. Issues with entailment-based zero-shot text classification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 786–796, Online. Association for Computational Linguistics.

Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, page 165–172, New York, NY, USA. Association for Computing Machinery.

Nasrin Mostafazadeh, Michael Roth, Annie Louis, Nathanael Chambers, and James Allen. 2017. LSDSem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 46–51, Valencia, Spain. Association for Computational Linguistics.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4885–4901. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Altaf Rahman and Vincent Ng. 2012. Resolving complex cases of definite pronouns: the winograd schema challenge. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 777–789. Association for Computational Linguistics.

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, Austin, Texas. Association for Computational Linguistics.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S. Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *Logical Formalizations of Commonsense Reasoning, Papers from the 2011 AAAI Spring Symposium, Technical Report SS-11-06, Stanford, California, USA, March 21-23, 2011*. AAAI.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8732–8740. AAAI Press.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush. 2022. Multitask prompted training enables zero-shot task generalization. In *The Tenth International Conference on*

Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022. OpenReview.net.

Elvis Saravia, Hsien-Chi Toby Liu, Yen-Hao Huang, Junlin Wu, and Yi-Shin Chen. 2018. CARER: Contextualized affect representations for emotion recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3687–3697, Brussels, Belgium. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

Priyanka Sen, Alham Fikri Aji, and Amir Saffari. 2022. Mintaka: A complex, natural, and multilingual dataset for end-to-end question answering. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1604–1619, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Lakshay Sharma, Laura Graesser, Nikita Nangia, and Utku Evci. 2019. Natural language understanding with the quora question pairs dataset. *CoRR*, abs/1907.01041.

Derek Tam, Rakesh R. Menon, Mohit Bansal, Shashank Srivastava, and Colin Raffel. 2021. Improving and simplifying pattern exploiting training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4980–4991, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. 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 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106, Copenhagen, Denmark. Association for Computational Linguistics.

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, New Orleans, Louisiana. Association for Computational Linguistics.

Mengzhou Xia, Mikel Artetxe, Jingfei Du, Danqi Chen, and Veselin Stoyanov. 2022. Prompting ELECTRA: Few-shot learning with discriminative pre-trained models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11351–11361, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Haike Xu, Zongyu Lin, Jing Zhou, Yanan Zheng, and Zhilin Yang. 2023. A universal discriminator for zero-shot generalization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10559–10575, Toronto, Canada. Association for Computational Linguistics.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022. ZeroGen: Efficient zero-shot learning via dataset generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11653–11669, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3914–3923, Hong Kong, China. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. 2024. Instruction tuning for large language models: A survey.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 649–657.

Xuandong Zhao, Siqi Ouyang, Zhiguo Yu, Ming Wu, and Lei Li. 2023. Pre-trained language models can be fully zero-shot learners. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15590–15606, Toronto, Canada. Association for Computational Linguistics.

A Statement Templates

A.1 QQP Templates

Task	Statement Template
QQP	"{{text1}}" is a duplicate of "{{text2}}" "{{text1}}" duplicates {{text2}} "{{text1}}" is not a duplicate of "{{text2}}" "{{text1}}" does not duplicate "{{text2}}"

A.2 Winogrande Templates

Task	Statement Template
Winogrande	In "{{sentence}}", _ is: {{option1/option2}} Q: "{{sentence}}", A: {{option1/option2}} The missing word in: "{{sentence}}" is {{option1/option2}} _ in: "{{sentence}}" is {{option1/option2}} "{{sentence}}", _ is: {{option1/option2}}

A.3 PiQA Templates

Task	Statement Template
PiQA	{{goal}} {{sol1/sol2}} Goal: {{goal}}, Solution: {{sol1/sol2}} If the goal is: {{goal}}, then the solution is: {{sol1/sol2}} Problem: {{goal}}, Solution: {{sol1/sol2}}

A.4 MNLI and SNLI Templates

Task	Statement Template
MNLI	"{{text1}}" entails "{{text2}}" {{text1}}? yes, {{text2}} Premise: {{text1}}, Hypothesis: {{text2}}, label: Entailment "{{text1}}" is neutral with regards to "{{text2}}" {{text1}}? maybe, {{text2}} Premise: {{text1}}, Hypothesis: {{text}}, label: Neutral "{{text1}}" contradicts "{{text2}}" {{text1}}? no, {{text2}} Premise: {{text1}}, Hypothesis: {{text}}, label: Contradiction

A.5 Mintaka Templates

Task	Statement Template
Mintaka	Q: {{question}}, A: {{answerText}} {{question}} {{answerText}} Question: {{question}}, Answer: {{answerText}} The answer of {{question}} is {{answerText}}

A.6 Yelp Polarity Templates

Task	Statement Template
Yelp Polarity	"Title: {{title}}, Content: {{content}}" has negative sentiment {{title}} {{content}} has negative sentiment "Title: {{title}}, Content: {{content}}", Sentiment: Negative {{title}} {{content}} It was terrible The sentiment in "{{title}} {{content}}" is negative "Title: {{title}}, Content: {{content}}" has positive sentiment {{title}} {{content}} has positive sentiment "Title: {{title}}, Content: {{content}}", Sentiment: Positive {{title}} {{content}} It was great The sentiment in "{{title}} {{content}}" is positive

A.7 WikiLingua Templates

Task	Statement Template
WikiLingua	Passage: {{source}}, Summary: {{target}} The summary of "{{source}}" is {{target}} Context: {{source}}, Summary: {{target}} Q: Summarize the following: {{source}}, A: {{target}} The answer of "Summarize the following {{source}}" is {{target}}

A.8 SQuAD Templates

Task	Statement Template
SQuAD	Context: {{context}}\n Question: {{question}}\n Answer: {{answers/random_span}} {{context}}\n According to the passage above, the answer of {{question}} is {{answers/random_span}} "Passage: {{context}}\n Question: {{question}}\n Answer: {{answers/random_span}} {{context}}\n Q: {{question}}\n A:{{answers/random_span}}

A.9 BCOPA Templates

Task	Statement Template
BCOPA	The cause of {{premise}} is that {{choice1/choice2}} {{premise}} because {{choice1/choice2}} {{premise}} due to {{choice1/choice2}} The effect of {{premise}} is that {{choice1/choice2}} {{premise}} therefore {{choice1/choice2}} {{premise}}, so {{choice1/choice2}}

A.10 MRPC Templates

Task	Statement Template
MRPC	"{{text1}}" is a paraphrase of "{{text2}}" "{{text1}}"\n In other words: "{{text2}}" {{text1}}? yes, {{text2}} "{{text1}}" can be stated as "{{text2}}" {{text1}}" is the same as saying "{{text2}}"

A.11 Amazon Polarity Templates

Task	Statement Template
Amazon Polarity	"Title: {{title}}, Content: {{content}}" has negative sentiment {{title}} {{content}} has negative sentiment "Title: {{title}}, Content: {{content}}", Sentiment: Negative {{title}} {{content}} It was terrible The sentiment in "{{title}} {{content}}" is negative The emotions conveyed in "{{title}} {{content}}" are negative "Title: {{title}}, Content: {{content}}" has positive sentiment {{title}} {{content}} has positive sentiment "Title: {{title}}, Content: {{content}}", Sentiment: Positive {{title}} {{content}} It was great The sentiment in "{{title}} {{content}}" is positive The emotions conveyed in "{{title}} {{content}}" are positive

A.12 FigQA Templates

Task	Statement Template
FigQA	{{startphrase}} {{ending1/ending2}} {{startphrase}} therefore {{ending1/ending2}} startphrase: {{startphrase}}, ending: {{ending1/ending2}} if {{startphrase}} then {{ending1/ending2}} {{startphrase}} means {{ending1/ending2}}

A.13 StoryCloze Templates

Task	Statement Template
StoryCloze	{{input_sentence_1}} {{input_sentence_2}} {{input_sentence_3}} {{input_sentence_4}} {{sentence_quiz1/sentence_quiz2}}

A.14 Yahoo Topics Answers Templates

Task	Statement Template
YA Topic	{{question_title}} {{question_content}} the topic is {{topic}}

A.15 Emotion Templates

Task	Statement Template
Emotion	{{question_title}} {{question_content}} the topic is {{topic}}

A.16 Offensive Templates

Task	Statement Template
Offensive	"{{text}}". The tweet is {{label}}. This tweet "{{text}}" is considered {{label}}. Tweet: "{{text}}". Label: {{label}}. "{{text}}". This text is {{label}}. The text "{{text}}" is {{label}}.

A.17 Massive Templates

Task	Statement Template
Massive	The utterance "{{utt}}" is under the {{scenario}} scenario. Utterance: "{{utt}}" Scenario: {{scenario}} User: "{{utt}}". The best scenario for the user query is {{scenario}}. The scenario of user's utterance "{{utt}}" is {{scenario}}.

A.18 Definite Pronoun Resolution Templates

Task	Statement Template
DPR	{{sentence_with_pronoun_replaced}} {{sentence}}. Based on the sentence, {{pronoun}} refers to {{candidates}}. The pronoun {{pronoun}} in "{{sentence}}" is referring to {{candidates}}. {{sentence}}. '{{pronoun}}' refers to {{candidates}}.

A.19 QASC Templates

Task	Statement Template
QASC	{{formatted_question}}. Answer: {{answer_key}} Q: "{{formatted_question}}." A: {{answer_key}} Question: "{{formatted_question}}." Answer: {{choices[answer_key]}} Context: {{combined_facts}} Question: {{question}} Answer: {{choices[answer_key]}} {{question}} Based on the passage "{{combined_facts}}", the answer if the question is "{{choices[answer_key]}}". {{combined_facts}} {{question}} {{choices[answer_key]}} Context: {{combined_facts}} Question: {{formatted_question}}. Answer: {{answer_key}} {{formatted_question}}. The answer is {{answer_key}}

A.20 SciQ Templates

Task	Statement Template
SciQ	{{question}} {{correct_answer}} Question: {{question}} Answer: {{correct_answer}} {{support}} Question: {{question}} Answer: {{correct_answer}} {{support}} According to the information, {{question}}. Answer: {{correct_answer}}. The answer to the question {{question}}, according to "{{support}}" is {{correct_answer}}.

A.21 RACE Templates

Task	Statement Template
RACE	{{article}} {{question_replaced_with_answer}}

B Fine-Tuning Setup

To fine-tune RoBERTa-base/RoBERTa-large on Statement-Tuning, we train for 15 epochs using an initial learning rate of 1e-06 and a weight decay of 0.01. We use a warm-up ratio of 0.1. We use 10% of the training data for validation. We use a training batch size of 16 for RoBERTa-base and a training batch size of 8 for RoBERTa-large.

C 32-shot Generalization

Table 7 shows the results of 32-shot fine-tuning on 7 target downstream datasets of our models and baselines. We observe similar trends to zero-shot setting as discussed in Section 5.2.

D Regular Classification of Statement-Tuned Models

In figure 5, we visualize the relative improvement of our Statement-Tuned RoBERTa-base models regularly fine-tuned on N-shot downstream data over the regularly fine-tuned RoBERTa-base. The results are not as good as fine-tuning using statements.

E Task Categories Breakdown

For the statement tuning task diversity, we group datasets based on task categories as follows (evaluation datasets are underlined):

1. Summarization (**SU**): WikiLingua, SAMSum
2. Sentiment Analysis (**SA**): Yelp Polarity, Amazon Polarity
3. Question Answering (**QA**): Mintaka, SQuAD, QASC, SciQ, RACE
4. Natural Language Inference (**NLI**): MNLI, SNLI
5. Commonsense Reasoning (**CR**): Winogrande, PiQA
6. Paraphrase Detection (**PD**): QQP, MRPC
7. Word Sense Disambiguation (**WSD**): Definite Pronoun Resolution
8. Intent Classification (**IC**): Massive
9. Offensive Language Identification (**OLI**): Tweet Eval's Offensive
10. Sentence Completion: BCOPA, StoryCloze
11. Emotion Recognition: Emotion
12. Topic Classification: Yahoo Answer Topic
13. Nonliteral Reasoning: FigQA

F N-Shot Correlation

Figure 6 shows the correlation of accuracies achieved in different N-shot settings with various shot numbers.

G Inference Speed Comparison

We report the average examples/sec processed for each of the datasets in Table 8. It is important to note that all models are run on a single GPU, except for Meta-Llama-3-70B-Instruct and Llama-2-13B-chat which were run on 4 and 2 GPUs, respectively.

H Further Training of Instruction-Tuned Decoders on Statement-Tuning Data

As seen in Table 10, we report the comparison of our approach with several decoders that were instruct-tuned using an instruction-tuning dataset created using the same training corpora used for Statement-Tuning. Training details are outlined in Table 9. Furthermore, the templates used to form instructions are based on those used for Flan (Wei et al., 2022) (we make the dataset available here: https://huggingface.co/datasets/ashabrawy/st_instruction_data).

The models tested are a subset of the ones reported in Table 1 due to the time and computational expense of instruction-tuning and hardware limitations. However, even when the LLM models are instruction-tuned on the same data as the Statement-Tuned RoBERTa models, we observe similar trends where performance of the RoBERTa-base model tends to match the performance of all models up to 6.9B parameters on all tasks except for BCOPA. Furthermore, the RoBERTa-large model approaches or exceeds performance on all tasks for models with 7B+ parameters. The same trend of dominating performance on FigQA and StoryCloze is observed.

I NLI Prompt Templates

We outline the NLI-style templates used for the tasks in Table 11.

	#Parameters	BCOPA	MRPC	FigQA	Amazon Polarity	StoryCloze	Yahoo Answers Topic	Emotion	Avg
Meta-Llama-3-70B-Instruct	70B	95.2	78.9	46.1	96.6	86.5	66.4	58.8	75.5
Llama-2-13b-chat-hf	13B	93.2	71.6	44.3	95.7	84.9	61.1	56.6	72.5
Llama-2-7b-chat	7B	91.0	67.9	42.8	95.2	82.1	61.9	54.3	70.7
Mistral-7B-Instruct-v0.2	7B	93.8	78.2	44.8	96.2	87.0	65.0	57.0	74.6
Qwen1.5-7B-Chat	7B	91.4	79.4	43.8	95.1	82.4	63.9	58.0	73.4
Pythia-6.7B	6.7B	84.6	66.9	39.2	91.6	74.0	38.3	52.0	63.8
Pythia-2.7B	2.7B	80.8	63.5	41.5	90.8	71.7	35.5	47.5	61.6
Phi-2	2.7B	90.8	74.0	44.7	93.8	81.6	58.4	58.4	71.7
FlanT5-Large	770M	66.2	78.7	39.7	75.3	59.9	38.0	34.6	56.1
Qwen1.5-0.5B-Chat	500M	73.4	56.1	38.5	84.2	68.8	36.1	31.4	55.5
BART-large-mnli	406M	52.2	32.4	42.0	50.6	51.1	7.1	10.0	35.0
FlanT5-Small	60M	52.0	32.6	41.4	75.8	50.0	9.1	9.8	38.7
Our Approach: RoBERTa-base (Best)	**125M**	75.0$_{(0.5)}$	70.0$_{(15.2)}$	61.1$_{(0.4)}$	92.8$_{(1.0)}$	79.7$_{(1.5)}$	39.2$_{(4.4)}$	48.1$_{(3.9)}$	66.6
Our Approach: RoBERTa-base (4k)	**125M**	75.0$_{(0.6)}$	70.0$_{(1.9)}$	60.3$_{(1.0)}$	92.4$_{(0.8)}$	79.7$_{(3.5)}$	38.2$_{(2.5)}$	45.4$_{(3.2)}$	65.9
Our Approach: RoBERTa-large (Best)	**355M**	85.1$_{(0.8)}$	71.5$_{(1.9)}$	74.7$_{(1.8)}$	95.3$_{(0.8)}$	91.9$_{(0.2)}$	50.2$_{(2.2)}$	49.8$_{(1.4)}$	74.1
Our Approach: RoBERTa-large (10k)	**355M**	85.1$_{(0.8)}$	71.5$_{(0.8)}$	72.6$_{(1.8)}$	95.3$_{(0.3)}$	91.0$_{(1.0)}$	48.2$_{(0.8)}$	48.4$_{(3.6)}$	73.1
Full-shot:									
RoBERTa-base (FT)	125M	74.2	87.0	88.1	94.3	-	71.0	82.2	-
RoBERTa-large (FT)	355M	86.0	87.6	92.0	96.5	-	68.5	78.2	-

Table 7: Comparison of our approach against many pretrained open-source encoder-Decoder and Decoder-only Pretrained Large Language Models on 7 Natural Language Understanding tasks in 32-shot conditions. We highlight all scores in gray where our approach with RoBERTa-base (best) exceeds or is equal to the score given by the model.

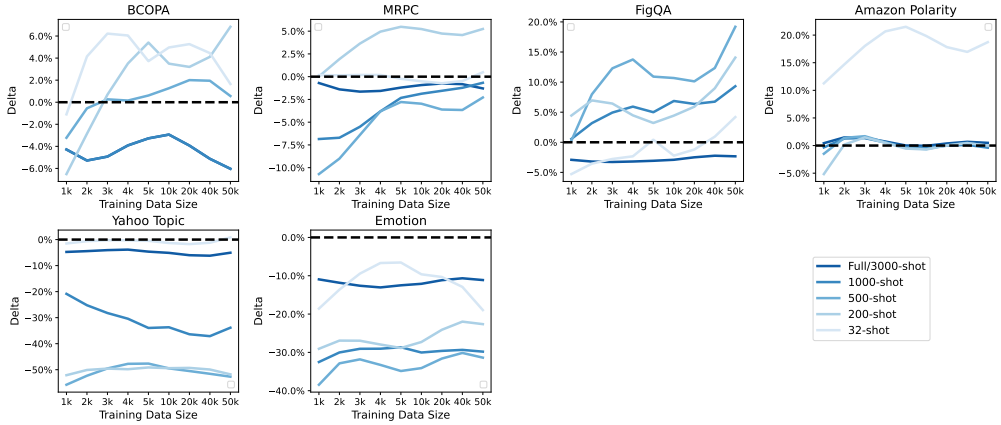

Figure 5: N-shot improvement of Statement-Tuned RoBERTa-base models used for regular fine-tuning. The y-axis, Delta, is the difference between the accuracy of the Statement-Tuned model fine-tuned for the task directly by discarding the Statement-Tuning classification head and the accuracy achieved by regular fine-tuning of RoBERTa-base on the task. A positive Delta indicates improvement over the baseline approach.

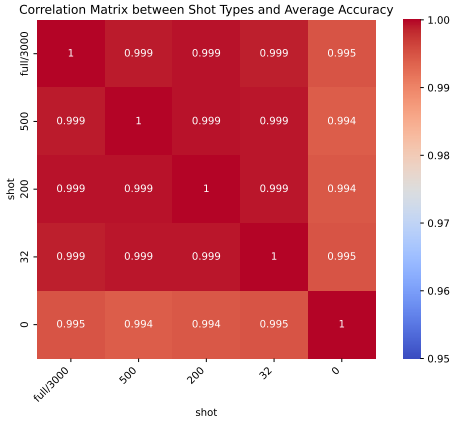

Figure 6: N-shot correlation using the average accuracy across all training set sizes and evaluation sets.

Model	BCOPA	MRPC	FigQA	Amazon Polarity	StoryCloze	YA Topic	Emotion	Avg
Qwen1.5-0.5B-Chat	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
phi-2	1.4	1.4	1.4	1.4	1.5	1.4	1.4	1.4
Meta-Llama-3-70B-Instruct*	2.9	1.3	3.2	0.9	4.9	1.1	2.1	2.3
flan-t5-large	8.2	13.2	13.2	13.2	13.2	13.2	13.2	12.5
Llama-2-13b-chat-hf*	8.7	5.7	12.8	4.3	15.7	4.4	6.9	8.3
Our Approach (roberta-large)	9.3	14.5	15.0	15.0	14.7	3.1	5.1	11.0
bart-large-mnli	9.7	14.1	14.0	14.2	14.1	13.7	13.8	13.4
pythia-6.9b	12.0	0.6	4.6	0.4	0.6	2.2	0.4	3.0
Llama-2-7b-chat-hf	12.5	0.6	4.6	0.4	0.6	2.3	0.5	3.1
Mistral-7B-Instruct-v0.2	12.8	0.5	2.7	0.3	0.5	1.7	0.4	2.7
pythia-2.8b	13.6	16.7	24.9	15.2	27.2	15.1	20.9	19.1
flan-t5-small	13.9	39.2	39.1	39.3	39.4	39.3	39.3	35.6
Our Approach (roberta-base)	17.9	49.8	50.0	49.8	49.9	10.3	17.0	34.9

Table 8: The average examples per second processed by each model on each task. * indicates that the model required the use of more than one GPU.

	Llama-2-13b	**Qwen1.5-7B**	**Pythia-6.9B**	**Pythia-2.9B**	**Phi-2**	**Qwen1.5-0.5B**
#Parameters	13B	7B	6.9B	2.9B	2.7B	500M
Quantization	8bit	4bit	4bit	4bit	4bit	4bit
Sequence Length	4096	2048	2048	2048	2048	2048
lora_r	32	32	32	32	32	32
lora_alpha	16	64	64	64	64	64
lora_dropout	0.05	0.05	0.05	0.05	0.05	0.05
adam_beta2	0.999	0.999	0.95	0.95	0.95	0.999
adam_epsilon	1e-8	1e-8	0.00001	0.00001	0.00001	1e-8
max_grad_norm	none	none	1.0	1.0	1.0	none
optimizer	adamw_bnb_8bit	adamw_torch	adamw_torch	adamw_torch	adamw_torch	adamw_torch
gradient acc.	4	4	4	4	4	4
micro batch size	2	1	1	1	1	1
lr_scheduler	cosine	cosine	cosine	cosine	cosine	cosine
learning_rate	0.0002	0.0002	0.00001	0.00001	0.000003	0.0002

Table 9: Instruction Tuning Hyperparameters

	#Parameters	BCOPA	MRPC	FigQA	Amazon Polarity	StoryCloze	YA Topic	Emotion	Avg
Llama-2-13b	13B	89.6	60.8	40.9	93.7	82.4	53.2	51.6	67.5
Qwen1.5-7B	7B	87.2	78.9	41.4	94.8	75.7	47.8	56.5	68.9
Pythia-6.9B	6.9B	82.8	68.1	40.0	71.7	71.5	16.4	27.5	54.0
Pythia-2.9B	2.9B	79.6	67.9	40.3	77.2	69.7	21.2	30.6	55.2
Phi-2	2.7B	87.2	68.1	41.7	85.6	77.8	38.4	53.5	64.6
Qwen1.5-0.5B	500M	72.4	68.4	39.4	49.8	67.6	33.2	72.4	57.6
Our Approach: RoBERTa-base (Best)	**125M**	$75.3_{(0.5)}$	$72.3_{(1.5)}$	$61.4_{(0.6)}$	$92.9_{(1.3)}$	$79.1_{(1.1)}$	$40.2_{(3.8)}$	$48.5_{(5.1)}$	67.1
Our Approach: RoBERTa-base (4k)	**125M**	$72.4_{(0.5)}$	$69.6_{(1.1)}$	$60.7_{(0.9)}$	$92.3_{(0.8)}$	$78.5_{(2.7)}$	$37.9_{(2.7)}$	$46.6_{(4.3)}$	65.4
Our Approach: RoBERTa-large (Best)	**355M**	$85.1_{(0.7)}$	$71.8_{(0.8)}$	$74.2_{(1.4)}$	$95.4_{(0.4)}$	$92.1_{(0.7)}$	$49.9_{(2.1)}$	$50.7_{(1.4)}$	75.3
Our Approach: RoBERTa-large (10k)	**355M**	$85.1_{(0.7)}$	$71.5_{(0.8)}$	$73.0_{(2.4)}$	$95.4_{(0.4)}$	$91.1_{(0.8)}$	$48.4_{(0.7)}$	$49.1_{(3.2)}$	73.4
Full/3000-shot:									
RoBERTa-base (FT)	125M	74.2	87.0	88.1	94.3	-	71.0	82.2	-
RoBERTa-large (FT)	355M	86.0	87.6	92.0	96.5	-	68.5	78.2	-

Table 10: Comparison of our approach against many pretrained open-source encoder-Decoder and Decoder-only Instruction-tuned Pretrained Large Language Models on 7 Natural Language Understanding tasks in Zero-shot conditions. FT stands for Full fine-tuning and is included for reference. For Statement-Tuning, we report the average across 5 training runs and 5 evaluation runs and include the average standard deviation in parenthesis. We highlight all scores in gray where our approach with RoBERTa-base (best) exceeds or is equal to the score given by the model.

Task	**Premise**	**Hypotheses**
BCOPA	{premise}	"The {question} is {choice1}" "The {question} is {choice2}"
MRPC	{text1} {text2}	"The two sentences are not equivalent" "The two sentences are equivalent"
FigQA	{startphrase}	{ending1} {ending2}
Amazon Sentiment	{text}	"The sentiment of the text is negative" "The sentiment of the text is positive"
StoryCloze (SC)	{input_sentence_1} {input_sentence_2} {input_sentence_3} {input_sentence_4}	{sentence_quiz1} {sentence_quiz2}
Yahoo Topic Classification	{question_title} {question_content}	"it is related with society or culture" "it is related with science or mathematics" "it is related with health" "it is related with education or reference" "it is related with computers or Internet" "it is related with sports" "it is related with business or finance" "it is related with entertainment or music" "it is related with family or relationships" "it is related with politics or government"
Emotion Classification	{text}	"this person feels sad" "the person feels joyful" "the person loves that" "the person feels angry" "the person is afraid of something" "the person feels surprised"

Table 11: Task-specific premise and hypotheses templates for converting tasks into NLI format.