Investigating the Generalizability of Pretrained Language Models across Multiple Dimensions: A Case Study of NLI and MRC

Ritam Dutt[1*], Sagnik Ray Choudhury[2*†], Varun Venkat Rao[3], Carolyn Rose[1], V.G.Vinod Vydiswaran[3]
[1]Carnegie Mellon University, [2]University of North Texas, [3]University of Michigan
rdutt@andrew.cmu.edu, sagnik.raychoudhury@unt.edu, varu@umich.edu, cprose@cmu.edu, vgvinodv@umich.edu

Abstract

Generalization refers to the ability of machine learning models to perform well on dataset distributions different from the one it was trained on. While several pre-existing works have characterized the generalizability of NLP models across different dimensions, such as domain shift, adversarial perturbations, or compositional variations, most studies were carried out in a stand-alone setting, emphasizing a single dimension of interest. We bridge this gap by systematically investigating the generalizability of pre-trained language models across different architectures, sizes, and training strategies, over multiple dimensions for the task of natural language inference and question answering. Our results indicate that model instances typically exhibit consistent generalization trends, i.e., they generalize equally well (or poorly) across *most* scenarios, and this ability is correlated with model architecture, base dataset performance, size, and training mechanism. We hope this research motivates further work in a) developing a multi-dimensional generalization benchmark for systematic evaluation and b) examining the reasons behind models' generalization abilities. [1]

1 Introduction

A machine learning model's generalization capability is defined as its capacity to apply encoded knowledge and strategies from previous experience to new situations. This is a key desideratum of all machine learning models, but NLP models are particularly interesting as the generalization scenario in NLP goes beyond the simple train-test split.

We present a comprehensive study of the generalization abilities of common models used in NLP.

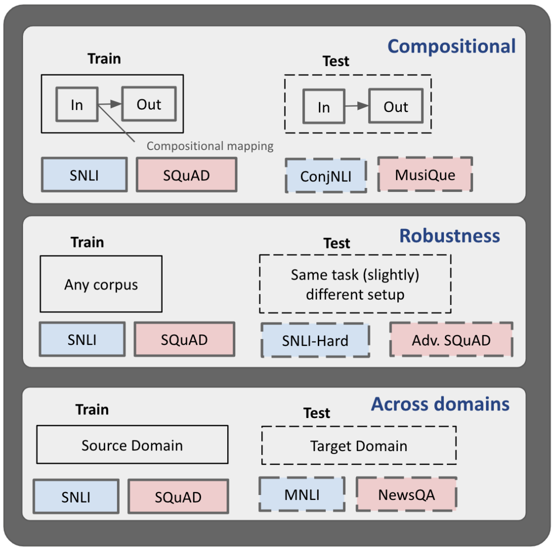


Figure 1: Hupkes et al. (2023) categorizes the generalization scenarios in NLP into *six* types. We chose *three* that cover many important scenarios. We trained models on SNLI and SQuAD, and tested them on various datasets corresponding to these dimensions. The datasets were chosen so as not to confound the dimensions. For example, the compositional test dataset for MRC (MusiQue) is a derivative of the source dataset SQuAD – there is no domain shift, and the dataset does not contain robustness testing perturbations.

Following Hupkes et al. (2023), we consider three types of generalization: 1. Domain; 2. Robustness; and 3. Compositional. These three multi-faceted aspects cover many scenarios with practical significance (Figure 1).

The most common type of generalization is **domain** generalization, where the model is trained on one domain and tested on another. Generally, domains in NLP are associated with sources as text from different sources have different linguistic styles (Lee, 2001).

Many standard NLP datasets have data points that can be solved by superficial cues, i.e., reasoning strategies unrelated to the expected causal mechanism of the task at hand. For example, in SNLI, Gururangan et al. (2018) shows that a nega-

*The first two authors contributed equally.
†Work done at the University of Michigan.
[1] code + data for analysis: https://github.com/sagnik/md-gen-nlp, 🤗 Trained models: https://huggingface.co/varun-v-rao.

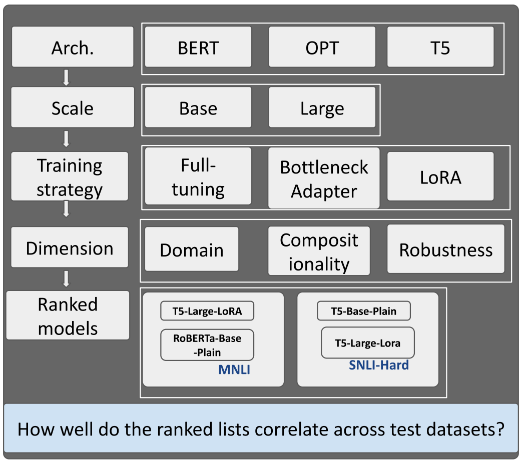


Figure 2: Our framework: we train 72 models on 2 base datasets, test them on 15 datasets corresponding to different dimensions of generalization, and analyze the results.

tion operator in the premise is a strong predictor of the "contradiction" class, or in many cases, the models can use the hypothesis alone to predict the class label. Likewise, Sen and Saffari (2020) observed that the answer phrase could be found in the first sentence of the context for several instances in popular extractive machine reading comprehension (MRC) datasets such as SQuAD (Rajpurkar et al., 2016) or HotpotQA(Yang et al., 2018). Zhang et al. (2020) and Ribeiro et al. (2020) also show that models are sometimes thrown off by semantics preserving perturbations that do not fool humans. Models also need to generalize to these instances – we refer to this as **robustness** generalization.

The final type of generalization we explore is **compositional**. A model demonstrates compositional generalization when it can methodically combine previously learned components to correctly solve new inputs composed of these components. Lake and Baroni (2018) presents a classic example – if a model understands that "doxing" refers to jumping up and "daxing" refers to moving left, would it realize that "dox then dax" refers to jumping up and moving left?

We train 3 instances each of the base and large versions of 4 models from 3 architecture families: encoder-only (EO), decoder-only (DO), and encoder-decoder (ED) on two representative datasets of two NLU tasks: SNLI for natural language inference (NLI) and SQuAD for machine reading comprehension (MRC) using full-training and parameter efficient fine-tuning or PEFT (Ding et al., 2023) in §2. Subsequently, we test them on 15 datasets from these tasks that correspond to different types of generalizations in §3. With this extensive setup, we ask the following questions:

- **RQ1**: Do certain model instances [2] generalize well across all types? Our goal is to see if the generalization ability of a model instance is generalization type-independent, i.e., it generalizes well across all scenarios. This question is asked at the instance level because McCoy et al. (2020a) has shown that model instances with similar test performances show wide differences when tested on different datasets.
- **RQ2**: We answer RQ1 affirmatively (§3.1) and find that the model instances from different seeds do not show large variances. This leads to a follow-up question (§3.2): are certain model configurations (architecture-size-training strategy) better at generalization than others?
- **RQ3**: How does model architecture (EO vs. DO vs. ED), size, or training strategy correlate with generalization? Is it type-dependent? We can expect over-parameterized models to generalize better (Belkin et al., 2019), as well as the PEFT models, as they have lower parameter changes than fully trained models and, consequently, less forgetting. While the first hypothesis holds, the second one does not.
- **RQ4**: Finally, we investigate whether certain generalization types are more challenging than the others. How is the target performance correlated with generalization dimensions (§3.4)?

Previous work has studied generalization in stand-alone cases, e.g., the datasets we have used here. Methods have been proposed to improve the generalization ability of both fully tuned and PEFT models by meta-learning (Lake and Baroni, 2023) or multi-task learning (Pfeiffer et al., 2021). Benchmarks such as Unified QA (Khashabi et al., 2020) have also been developed to test generalization.

Despite this rich history, less effort has been spent on developing a *systematic* categorization of generalization and studying how models generalize across such categories. Models need to generalize across *all* scenarios, and not just be robust against domain shift or compositional variations.

[2] 1. **model instance**: a particular instance of a trained model, e.g., a `T5`$_{\texttt{base}}$ model with LoRA trained on SNLI with a seed of 42. 2. **architecture**: model architecture, e.g., `RoBERTa`, `T5`. 3. **model configuration**: a combination of architecture-size-training strategy (`T5`$_{\texttt{base}}$ fully fine-tuned). 4. **architecture family**: types of architectures – encoder only (`BERT`, `RoBERTa`)/decoder-only (`OPT`).

This work is a step in this direction. Our comprehensive analysis highlights that model instances exhibit consistent generalization prowess across the board and that models from certain architectures or sizes are more generalizable than others. This is certainly not comprehensive, questions remain open about the choice and size of the base dataset, new model architectures, and most importantly, the reason behind a model's generalization ability which we defer for future work.

2 Tasks, Datasets & Models

We consider two representative NLU tasks: NLI and MRC. The NLI task involves determining if the meaning of one text fragment (hypothesis) can be inferred from another (premise). Independent of any specific application, this task is designed to encapsulate the essential inferences about the variability of semantic expression frequently required for various settings (Dagan et al., 2006). MRC is another common task – many NLU tasks have been formulated as MRC (He et al., 2015) or models trained on MRC format data have shown good performance on NLU tasks (McCann et al., 2018). We use the extractive version of MRC, where the input consists of a context (passage) and a question, and the answer has to be extracted from the context.

2.1 NLI Datasets

We consider SNLI (Bowman et al., 2015) as the source dataset, which is annotated with the labels corresponding to whether the hypothesis entails, is neutral, or contradicts the premise.

- **Domain:** We use both the matched and mismatched splits of the Multi-Genre NLI (MNLI) dataset (Williams et al., 2018) to test the generalization of an SNLI-trained model to different domains. We also use the TaxiNLI dataset (Joshi et al., 2020) that provides a hierarchical taxonomy of a subset of the MNLI dataset and categorizes the data points based on whether they require linguistic, logical, or world knowledge.
- **Robustness:** We cover the robustness scenarios by testing the models on four datasets. SNLI-H (Gururangan et al., 2018) is a set of SNLI test instances that common heuristics can not classify. The SNLI-CF dataset (Kaushik et al., 2019) comprises of "counter-factual" perturbations, where the annotators are asked to make minimal changes to an instance such that the label changes – a model can only classify these instances correctly if it understands the reasoning behind the NLI task. SNLI-BT is generated by back-translating the original SNLI test instances from En->Pt->En using a pre-trained multi-lingual BART model – this tests the models' ability to generalize against adversarial perturbations. Finally, HANS (McCoy et al., 2020b) is built from templates constituting different syntactic heuristics in NLI, such as lexical overlap or common subsequences between the premise and hypothesis.
- **Compositionality:** It is non-trivial to meaningfully combine SNLI instances, but in a compositional NLI dataset such as MoNLI (Geiger et al., 2020) all words or phrases of a composed instance come from SNLI. Consider a sentence from SNLI "The children are holding plants". Assume the phrase "flowers", which is a hyponym (per Wordnet) to the phrase "plants", appears in SNLI. Now the pair (premise: "The children are holding flowers", and hypothesis: "The children are holding plants") will have an entailment relation as every flower is a plant. Consequently, the label would change to neutral when the premise and hypothesis are reversed. Since the phrase that determines this relation exists in SNLI, the new dataset is merely a composition of the known constituents.[3] CONJNLI (Saha et al., 2020) focuses on conjunctive sentences – premises and hypotheses vary through the addition, removal, or substitution of conjuncts such as "and," "or", "but", and "nor" alongside elements like quantifiers and negations. This also presents a challenge in compositional generalization.

2.2 MRC Datasets

We train the MRC models on a popular extractive dataset SQuAD (Rajpurkar et al., 2016).

- **Domain:** NewsQA is a crowd-sourced dataset of approximately 100K human-generated QA pairs, where the context comes from 10K news articles from CNN. In SQuAD contexts are paragraphs from Wikipedia articles, therefore NewsQA presents a significant domain shift.
- **Robustness:** Adversarial Squad (Adv-SQuAD) is a robustness challenge set built on SQuAD insofar it adds a sentence that contains a phrase

[3] This is the *PMoNLI* part of the dataset. Negations would change the direction of the monotone operator: *not* holding plants $\Rightarrow$ *not* holding flower, but not the other way around. These instances comprise the *NMoNLI* dataset, which we do not use.

that a shortcut-dependent model (eg., one that chooses a phrase that is proximal to a key phrase from the question) would select (Jia and Liang, 2017). The HotpotQA dataset (Yang et al., 2018) was designed to test the multi-hop reasoning abilities of MRC models, i.e., a model should only be successful if it understands relations between entities that span multiple sentences. Similar to Jia and Liang (2017), Jiang and Bansal (2019) built a challenge set (Adv-HotpotQA) by adding a new passage to the context with a fake answer. The modifications in both Adv-HotpotQA and Adv-SQuAD do not change the original answer. Therefore, a model using the expected reasoning strategies would still be able to answer correctly, but a model dependent on shortcuts would fail.

- **Compositionality:** MusiQue (Trivedi et al., 2022) is designed to test compositionality in reading comprehension. The dataset is built on multiple MRC datasets (SQuAD, HotpotQA and three others) in a "bottom-up" approach. Pairs of *connected* single-hop questions are combined to create 2-hop questions first and are subsequently combined to produce k-hop questions recursively. We only choose the questions that are produced by combining SQuAD questions.

We use the validation or test (when available) split of the generalization datasets. In NLI, most datasets for compositional and robustness generalization are derivatives of the SNLI dataset itself, except for HANS and CONJNLI. They come from non-SNLI sources, but the distribution is not significantly different. *This allows us to not confound different dimensions of generalizability*. This is true for MRC as well, Adv-SQuAD and MusiQue (the portion we use) come from the base dataset SQuAD, and both Adv-HotpotQA and SQuAD come from the same domain. HANS has 2 labels (as opposed to 3 for SNLI), so the predicted labels of neutral and contradiction are merged. For consistency, we only use instances with a max tokenized sequence length of 512 (see the appendix for details).

2.3 Models & Training

We explore three popular families of transformer-based neural architectures, i.e., encoder-only (EO), decoder-only (DO), and encoder-decoder (ED) models. As the most popular/powerful representative for each architecture, we include `RoBERTa` (Liu et al., 2019) and `BERT` (Devlin et al., 2019) for EO, `OPT` (Zhang et al., 2022) for DO, and `T5` (Raffel et al., 2020) for (ED).

NLI is modeled as a sequence classification problem, and a linear layer is used as the classifier over the base encoders. MRC is modeled as a token classification problem with a linear layer, and the models are trained to predict a token's probability for being the start and end of an answer phrase (Devlin et al., 2019). We use the base and large versions for each model, and specifically for `BERT` these are the `cased` ones.

The models are trained by changing the full parameters as well as a fraction of them using two PEFT methods: Bottleneck adapters (Houlsby et al., 2019) and LoRA (Hu et al., 2021). Adapters introduce bottleneck feed-forward layers in each layer of a transformer model as the only trainable parameters. These adapter layers consist of a down-projection matrix $W_{\text{down}} : (d_{\text{hidden}}, d_{\text{bottleneck}})$, a RELU non-linearity ($f$) and an up-projection matrix $W_{\text{up}} : (d_{\text{bottleneck}}, d_{\text{hidden}})$, with the final equation: $h \leftarrow W_{\text{up}} \cdot f(W_{\text{down}} \cdot h)$. We use a reduction factor ($\frac{d_{\text{hidden}}}{d_{\text{bottleneck}}}$) of 16 for all models. Similar to Bottleneck adapters, LoRA injects trainable low-rank decomposition matrices into the layers of a pre-trained model. Any linear layer of the form ($h = W_0 x$) is re-parameterized as: $h = W_0 x + \frac{\alpha}{r} BAx$ where ($A \in R^{r \times k}$) and ($B \in R^{d \times r}$) are the trainable decomposition matrices and r is the low-dimensional rank of the decomposition. We set the rank at 16 and α at 32.

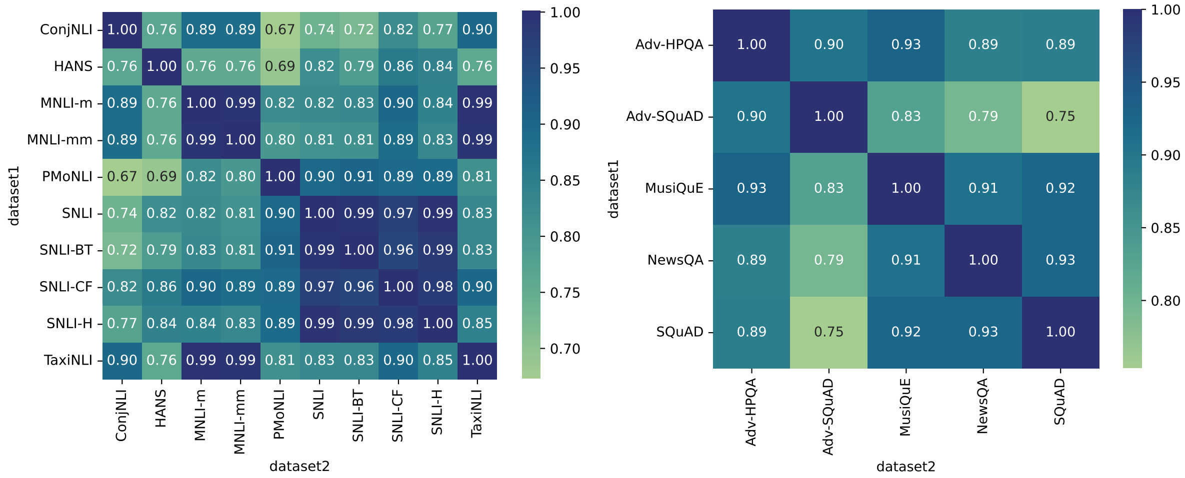
Each model is initialized with three seeds, and the training data sequence is shuffled. The models are trained with AdamW (Loshchilov and Hutter, 2019) optimizer, batch sizes varying between 32 and 64, and a learning rate of 2e-5 with a stepwise learning rate decay (Howard and Ruder, 2018) using the HuggingFace Transformers library (Wolf et al., 2019) (see the Appendix for details).

3 Results

3.1 RQ1: Does one model instance generalize well across generalization dimensions?

Our first hypothesis is a model instance generalizes well across different types. We test this by investigating whether the rankings of model instances are consistent, i.e. are well-correlated, across datasets that characterize different types of generalization.

We evaluate 72 model instances on each dataset corresponding to a task. Subsequently, for a given dataset pair in a task, we compute Spearman's



(a) NLI Datasets

(b) MRC Datasets

Figure 3: Spearmann's Rank Correlation ρ between the source and the target datasets for NLI and MRC on a per-instance basis.

rank correlation coefficient (ρ) of the corresponding model instances' scores (accuracy for NLI and F1-Score for MRC) for the two datasets. We are more interested in the rankings (relative performance) of model instances than the absolute scores since the datasets are not well calibrated amongst themselves. We present a heatmap of the correlation scores between pairs of datasets for NLI and MRC in Figures 3a and 3b, respectively.

We observe a strong to very-strong correlation ($\rho \geq 0.6$) [4] for all dataset pairs for both NLI and MRC tasks. For each of these comparisons, the correlation was statistically significant with a p-value lower than 0.05, **implying that we can reject the null hypothesis that the performances of model instances are not monotonically correlated**.

For NLI, the datasets derived from the same source, e.g., SNLI-CF, SNLI-BT, and PMoNLI from SNLI, or datasets that are created in a similar fashion like matched and mismatched splits of MNLI exhibit very strong correlation ($\rho \geq 0.90$). On the other hand, datasets derived from a different source like Wikipedia for CONJNLI or constructed in a templatized fashion like HANS demonstrate a more uniform correlation. We thus infer that the rankings of model instances depend more on the source than the type of generalization for NLI. For example, although PMoNLI and CONJNLI both test compositionality, the instances have the lowest correlation score ($\rho = 0.67$).

However, this observation is not as pronounced for MRC, where the model rankings correlate more with the generalization type than the dataset source. For example, we observe a higher correlation between Adv-HotpotQA and Adv-SQuAD ($\rho = 0.90$) than between Adv-SQuAD and SQuAD ($\rho = 0.75$). We also note a higher correlation across domains for MRC ($\rho = 0.92$ between SQuAD and NewsQA) than for NLI ($\rho \approx 0.8$ between MNLI and SNLI).

Having ranked the model instances in decreasing order of performance for each of the 10 NLI datasets, we can obtain a global (or unified) ranked list by aggregating these individual rankings. We employ the MC4 algorithm of Dwork et al. (2001) that constructs the ranking preferences based on a simple majority vote across the individual rankings to obtain the aggregated ranked list of instances. We do the same for the 5 datasets to create an aggregate ranked list for MRC. Spearmann's rank correlation coefficient between these two aggregated ranked lists for MRC and NLI is 0.93, which implies that the model instances also exhibit high correlation across tasks.

3.2 RQ2: Do model configurations generalize well across scenarios?

We extend our previous hypothesis to investigate whether certain model configurations (a combination of model architectures, scale, and training strategies) generalize well across different scenarios. We start by averaging the performance of a model configuration (architecture-size-training strategy combination) across three seeds and report the results in Tables 1 and 2 for NLI and MRC, respectively. Interestingly, we do not see a significant variation across instances from different seeds (as evidenced by low standard deviations) – a finding

[4] https://www.statstutor.ac.uk/resources/uploaded/spearmans.pdf

Table 1: Performance of NLI models when trained on the SNLI and evaluated on different datasets in terms of accuracy. We report the mean and standard deviation across three seeds. The best model is highlighted in bold, the second-best model is underlined, and the worst model is highlighted in red. `Adap` and `LoRA` refers to the adapter and LoRA training strategies.

	ID	**OOD**			**Robustness**				**Compositionality**	
Model	SNLI	MNLI-m	MNLI-mm	TaxiNLI	SNLI-BT	SNLI-CF	SNLI-H	HANS	ConjNLI	PMoNLI
$\texttt{BERT}_{\texttt{base}}$+ Adap	85.1±0.1	65.1±0.1	68.0±0.1	64.7±0.1	80.0±0.1	68.5±0.1	71.0±0.2	50.0±0.0	52.2±0.7	91.7±1.1
$\texttt{BERT}_{\texttt{base}}$+ LoRA	81.3±0.2	59.2±0.5	61.1±0.1	54.6±0.6	76.6±0.2	64.2±0.3	65.7±0.5	50.0±0.0	49.1±1.4	85.9±0.5
$\texttt{BERT}_{\texttt{base}}$	90.6±0.1	73.5±0.4	73.6±0.2	73.4±0.1	84.3±0.2	76.1±0.2	80.2±0.1	58.1±1.2	58.6±0.6	95.1±0.3
$\texttt{BERT}_{\texttt{large}}$+ Adap	88.8±0.2	72.8±0.8	73.2±0.8	72.8±1.0	83.1±0.2	73.3±0.2	77.3±0.3	50.3±0.4	56.7±1.1	96.1±0.3
$\texttt{BERT}_{\texttt{large}}$+ LoRA	86.2±0.4	68.3±0.5	69.2±0.7	67.7±1.5	80.9±0.1	69.3±0.4	73.1±0.6	50.1±0.2	54.3±1.5	94.7±1.2
$\texttt{BERT}_{\texttt{large}}$	91.1±0.1	76.6±0.1	76.2±0.3	76.5±0.4	84.7±0.1	77.4±0.3	81.7±0.2	58.1±1.2	61.1±0.8	97.6±0.4
$\texttt{RoBERTa}_{\texttt{base}}$+ Adap	88.3±0.1	75.8±0.6	75.9±0.3	74.4±0.2	83.0±0.0	72.9±0.2	76.1±0.2	50.3±0.1	54.8±0.6	95.1±0.1
$\texttt{RoBERTa}_{\texttt{base}}$+ LoRA	87.1±0.0	73.6±0.0	74.9±0.1	72.3±0.3	81.8±0.0	71.8±0.2	75.2±0.1	50.1±0.0	52.2±0.9	94.4±0.2
$\texttt{RoBERTa}_{\texttt{base}}$	91.4±0.0	80.2±0.2	79.9±0.2	80.1±0.2	85.2±0.1	77.9±0.1	82.1±0.1	65.9±2.0	60.8±0.4	96.6±0.2
$\texttt{RoBERTa}_{\texttt{large}}$+ Adap	91.7±0.0	83.8±0.4	83.0±0.4	83.9±0.1	85.4±0.0	79.9±0.5	82.4±0.1	67.8±1.4	61.4±0.2	**98.5±0.1**
$\texttt{RoBERTa}_{\texttt{large}}$+ LoRA	90.8±0.1	81.7±0.4	81.8±0.2	81.1±0.5	84.5±0.1	78.8±0.2	81.0±0.1	65.3±0.8	58.5±0.9	98.0±0.2
$\texttt{RoBERTa}_{\texttt{large}}$	**92.6±0.0**	85.0±0.0	84.3±0.1	85.0±0.1	**85.7±0.0**	**81.3±0.2**	**84.7±0.0**	**73.7±1.0**	65.5±0.3	**98.5±0.1**
$\texttt{OPT}_{\texttt{base}}$+ Adap	82.8±3.0	56.7±1.8	57.5±1.9	55.2±3.7	77.5±2.8	66.7±2.4	68.6±3.1	52.3±3.3	49.2±4.3	88.4±2.3
$\texttt{OPT}_{\texttt{base}}$+ LoRA	78.1±3.7	53.8±1.5	55.7±2.3	52.8±1.1	72.4±4.0	63.2±2.3	65.0±2.9	50.4±0.6	47.4±1.9	86.6±3.1
$\texttt{OPT}_{\texttt{base}}$	89.6±0.1	71.3±0.7	72.9±0.9	71.3±0.9	83.7±0.2	74.5±0.3	78.8±0.1	59.1±4.2	57.5±0.3	95.6±0.8
$\texttt{OPT}_{\texttt{large}}$+ Adap	88.6±0.2	66.6±1.3	69.2±0.8	66.0±2.1	81.9±0.5	73.4±0.3	77.5±0.2	61.7±6.8	55.4±1.0	90.9±1.5
$\texttt{OPT}_{\texttt{large}}$+ LoRA	83.6±2.2	63.5±3.6	65.0±3.4	60.7±4.7	78.0±2.5	69.5±1.2	71.4±2.1	60.1±2.3	56.7±3.1	91.9±3.0
$\texttt{OPT}_{\texttt{large}}$	90.4±0.4	75.5±0.4	77.3±0.3	75.4±0.3	84.1±0.3	76.5±0.8	80.7±0.5	65.8±0.6	60.7±1.3	95.2±2.0
$\texttt{T5}_{\texttt{base}}$+ Adap	88.6±0.0	80.1±0.1	80.3±0.1	80.3±0.3	82.9±0.0	74.8±0.2	77.7±0.1	60.2±0.1	64.0±0.9	94.6±0.4
$\texttt{T5}_{\texttt{base}}$+ LoRA	85.8±0.0	80.6±0.4	80.9±0.3	80.6±0.5	80.7±0.2	72.8±0.2	74.1±0.3	57.2±0.7	65.2±0.7	92.1±0.8
$\texttt{T5}_{\texttt{base}}$	89.7±0.1	81.4±0.1	80.9±0.2	81.2±0.1	83.7±0.1	75.9±0.2	79.5±0.1	63.3±0.3	65.2±0.9	95.3±0.3
$\texttt{T5}_{\texttt{large}}$+ Adap	91.8±0.0	86.2±0.1	85.5±0.3	86.6±0.4	85.4±0.1	80.3±0.3	82.7±0.1	68.2±1.1	66.0±0.1	98.1±0.2
$\texttt{T5}_{\texttt{large}}$+ LoRA	90.5±0.0	**87.5±0.1**	**87.5±0.3**	87.8±0.3	84.2±0.0	79.4±0.1	81.0±0.1	64.7±0.1	66.3±0.5	98.1±0.1
$\texttt{T5}_{\texttt{large}}$	92.1±0.1	87.3±0.1	86.8±0.2	**87.9±0.2**	85.5±0.0	81.0±0.2	83.3±0.1	71.6±0.6	**67.2±0.3**	98.0±0.1

different from prior work of McCoy et al. (2020a).

We also compute the Spearman's rank correlation coefficient between two dataset pairs for NLI and MRC in Figures 13a and 13b (appendix), respectively. The heatmaps indicate a strong positive correlation ($\rho \geq 0.7$) between all dataset pairs and inform us that the relative performance of these model configurations remains consistent across the target datasets and domains.

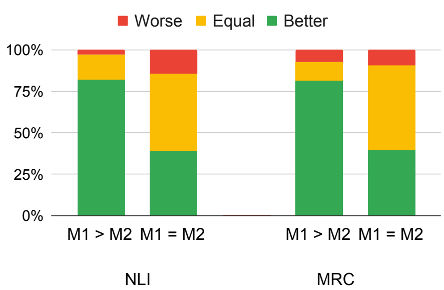
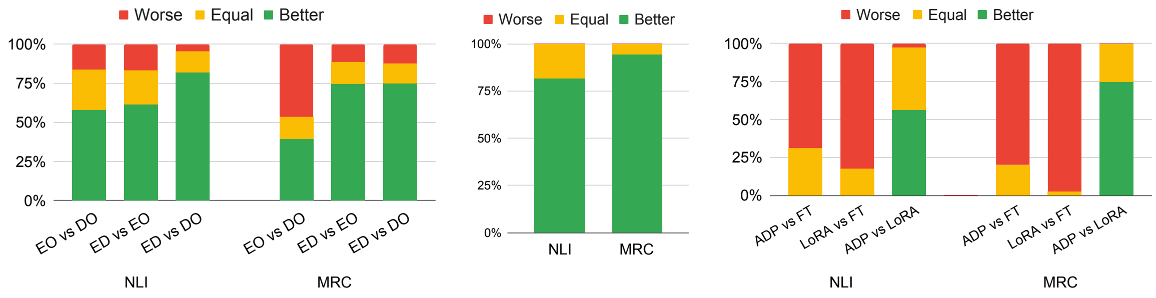


Figure 4: Fraction of cases where one model is significantly better, worse, or as good as the other on different target datasets. We consider two scenarios, (i) where one of the models was already significantly better on the source dataset ($M_1 > M_2$) and (ii) where the models had similar source performance ($M_1 = M_2$).

We further carry out a pair-wise comparison of model configurations to investigate whether the relative performance of a model pair on the source dataset (SNLI and SQuAD for NLI and MRC, respectively) persists across different target datasets. Simply put, if the performance of a model M_1 is significantly better than M_2 on the source dataset, does the situation remain the same across other targets? We adopt the non-parametric paired bootstrap test of Berg-Kirkpatrick et al. (2012) to check for statistical significance (p-value ≤ 0.05) in line with prior work (Dror et al., 2018). We note that M_1 has a similar performance with M_2 if we cannot reject the null hypothesis that one has a significantly higher performance than the other.

Figure 4 illustrates the fraction of cases where the relative performance of a model architecture pair is better, worse, or the same on the target datasets compared to the original source conditions. We observe that the models retain their relative performance for a majority of cases for both NLI and MRC, i.e. if M_1 is significantly better than M_2 on the base dataset, it will follow a similar trend across targets and vice versa. The notable exceptions are the PEFT-tuned versions of `T5` model which exhibit significantly higher performance than other models (such as `BERT` or `OPT` variants) on the tar-



(a) Architecture Types **(b)** Model Scale **(c)** PEFT

Figure 5: Fraction of times the given architecture configuration or training strategy is statistically better, equal, or worse for the two tasks of NLI and MRC.

get datasets for NLI despite a significantly worse performance on the SNLI source dataset. A similar finding holds for the fully-tuned `OPT` models that significantly outperform others (such as `BERT` and `T5`-PEFT variants) on MRC datasets.

3.3 RQ3: Architecture, Scale, and PEFT

Model Architecture: From Tables 1 and 2, we see that when controlled for the model size (base v large) and training strategy (full vs PEFT), certain models almost always perform better than the others, e.g., in NLI, the base versions of `T5` models (ED) are better than `RoBERTa` (EO) models in 7 out of 9 datasets, and `RoBERTa` is better than `OPT` (DO) in 8 out of 9. To formalize this, we compare the performance of a pair of models from different architectures (e.g., $\texttt{T5}_{\texttt{base}}$ vs. $\texttt{OPT}_{\texttt{large}}$) for a given dataset. Each architecture has instances from all sizes and training strategies, so we do not have to control for them explicitly.

We adopt the paired bootstrap test to compute the fraction of datasets where models corresponding to one family (say EO) are significantly better, worse, or equal compared to models of another family (say ED). Overall, we observe (Figure 5a) that ED models outperform both the EO and DO significantly on both tasks. On the other hand, models corresponding to the EO fare better for NLI as opposed to DO and vice-versa for MRC.

Scale: We compute the fraction of cases where the large variant of a model architecture is significantly better, worse, or equal to the corresponding base variant for a given dataset and task while controlling for the training strategy. Figure 5b shows that for both tasks, the large variants of models are significantly better than their corresponding base variants in a huge majority of cases. In fact, the base variant is never significantly better, although there are a few ties. This performance gain is also significantly higher in the generalization datasets compared to the base ones.

Parameter efficient fine-tuning (PEFT): We also explore whether PEFT models (i.e., Adapters and LoRA) are more adept at generalization than the corresponding fully fine-tuned (FT) variants. For each model pair, we compute the fraction of cases where the PEFT variant, i.e., Adapter vs. FT or LoRA vs. FT, was significantly better, equal, or worse than the corresponding fine-tuned variant. Figure 5c shows that PEFT models are indeed significantly worse. Moreover, this poorer performance is more pronounced for the LoRA models than for Adapters, such that adapter models are significantly better than LoRA models for both tasks.

3.4 RQ4: Difficult types of generalization

We inspect the absolute generalization performance of models on different datasets to investigate whether certain generalization categories or dimensions are more challenging than others. We characterize a dataset to be challenging for a given model based on the relative drop in performance of the model on the dataset compared to its' source performance (e.g., the performance of a model on SNLI and SQuAD respectively). We coin this performance difference as normalized source drop or NSD (Calderon et al., 2023) defined below, where M_s and M_t correspond to the performance of the model on the source and the target, respectively.

$$NSD = \frac{M_t - M_s}{M_s}$$

We carry out a two-way ANOVA analysis with NSD as the dependent variable with the generalization category (OOD, robustness, compositionality, or in-domain), architecture type (EO, ED, or DO), scale (large or base), and training strategy (FT, LoRA, or Adapter) as the independent covari-

Table 2: Performance of MRC models when trained on the SQuAD (ID) and evaluated on different datasets. We report the mean F1 score across three seeds (the stds vary between 0.0 and 3.2). The best model is highlighted in bold, the second-best is underlined, and the worst is highlighted in red. `OOD`, `Rob`, and `Comp` imply generalization across domains, robustness, and compositionality, respectively. `Adap` and `LoRA` refers to the adapter and LoRA training strategies.

Model	OOD	Rob		Comp	ID
	NQA	AHQ	ASQ	MsQ	SQ
$\texttt{BERT}_{\texttt{base}}$+ Adap	52.7	22.9	45.5	41.1	77.8
$\texttt{BERT}_{\texttt{base}}$+ LoRA	12.8	9.7	17.9	12.8	24.7
$\texttt{BERT}_{\texttt{base}}$	62.2	34.7	61.8	50.2	87.6
$\texttt{BERT}_{\texttt{large}}$+ Adap	60.1	25.0	64.3	50.2	85.6
$\texttt{BERT}_{\texttt{large}}$+ LoRA	42.2	17.0	46.3	37.0	67.1
$\texttt{BERT}_{\texttt{large}}$	65.2	39.4	72.5	62.2	90.7
$\texttt{RoBERTa}_{\texttt{base}}$+ Adap	55.0	26.3	63.5	51.5	85.8
$\texttt{RoBERTa}_{\texttt{base}}$+ LoRA	43.6	22.2	50.2	47.3	78.7
$\texttt{RoBERTa}_{\texttt{base}}$	63.3	39.0	73.0	61.4	92.0
$\texttt{RoBERTa}_{\texttt{large}}$+ Adap	66.8	46.6	<u>82.5</u>	65.5	93.4
$\texttt{RoBERTa}_{\texttt{large}}$+ LoRA	54.3	34.8	70.7	57.8	88.7
$\texttt{RoBERTa}_{\texttt{large}}$	**70.0**	**51.4**	**84.1**	**74.6**	**94.6**
$\texttt{OPT}_{\texttt{base}}$+ Adap	48.4	31.0	64.5	40.9	75.2
$\texttt{OPT}_{\texttt{base}}$+ LoRA	47.5	25.9	61.8	41.0	71.9
$\texttt{OPT}_{\texttt{base}}$	58.9	37.7	78.6	59.0	83.6
$\texttt{OPT}_{\texttt{large}}$+ Adap	55.1	34.7	79.0	47.0	83.5
$\texttt{OPT}_{\texttt{large}}$+ LoRA	57.9	33.5	79.0	45.6	83.3
$\texttt{OPT}_{\texttt{large}}$	62.4	42.0	81.6	68.7	85.9
$\texttt{T5}_{\texttt{base}}$+ Adap	67.2	37.8	74.2	61.1	90.3
$\texttt{T5}_{\texttt{base}}$+ LoRA	64.8	33.6	69.8	57.8	87.5
$\texttt{T5}_{\texttt{base}}$	67.5	38.6	74.8	64.0	90.9
$\texttt{T5}_{\texttt{large}}$+ Adap	69.7	46.5	82.3	69.9	93.7
$\texttt{T5}_{\texttt{large}}$+ LoRA	69.5	42.8	79.6	68.4	92.8
$\texttt{T5}_{\texttt{large}}$	<u>69.9</u>	<u>47.9</u>	**84.1**	<u>73.6</u>	<u>93.9</u>

ates. We observe a significant association for all the covariates (p-value ≤ 0.05), with the generalization category exhibiting the greatest significance, followed by the architecture type, training strategy, and scale for MRC. NLI exhibits a similar trend, with the only difference being that the scale is more significant than the training strategy.

Considering the in-domain category (i.e., performance on the base dataset) as the baseline, we observe a negative correlation for all the other generalization categories. The robustness category is the most challenging (with a larger negative coefficient), followed by compositionality and OOD for MRC. For NLI, the robustness category again incurs the highest negative correlation, followed by OOD and compositionality. We hypothesize that the general prowess of models on the PMoNLI dataset, surpassing even the ID performance, is responsible for the skewed trend. We also observe positive coefficients for the larger model variant, the ED model family, and the fully fine-tuned (FT) training strategy which is consistent from our past observations. We present the intercept values of our analysis in Table 3.

Category	NLI	MRC
Intercept	-0.052	-0.015
Gen-type: Comp	-0.132	-0.354
Gen-type: ROB	-0.170	-0.388
Gen-type: OOD	-0.158	-0.313
Arch-family: ED	0.073	0.023
Arch-family: EO	0.024	-0.047
Fine-tuning: FT	0.023	0.047
Fine-tuning: LoRA	-0.00	-0.020
Scale: Large	0.028	0.047

Table 3: Coefficients for the ANOVA analysis for NLI and MRC.

4 Related Work

Previous work has examined the generalization ability of NLP models in different scenarios, and developed strategies for improving their capabilities. Hupkes et al. (2023) provides a categorization of generalization types, of which we have discussed three that cover most datasets, but other types exist. *Cross-task (CT) generalization* measures a model's ability to generalize to new tasks. Instruction-tuned LLMs trained on massive crowd-sourced instruction datasets that contain task descriptions have shown strong CT generalization (Zhang et al., 2023). Recent LLMs such as GPT-3 (Brown et al., 2020) or LLama2 (Touvron et al., 2023) are zero-shot cross-task models, but possible data contamination raises concerns about their true generalization abilities (Li and Flanigan, 2024). *Syntactic generalization* involves generalization to new syntactic structures or unknown elements in known syntactic structures (Jumelet et al., 2021).

Among the categories of generalization we have considered, Ramponi and Plank (2020); Naik et al. (2022) presents a survey of neural models for *domain generalization*. For *robustness generalization*, many papers have proposed adversarial attacks to perturb the input to fool the model. These attacks can be white-box (Ebrahimi et al., 2018), i.e., the attacker has access to the model parameters or not (black-box (Jin et al., 2020), see Goyal et al. (2023) for a survey). However, not all of these attacks produce meaningful sentences, and more importantly, they do not test for a model's propensity toward shortcut learning (Geirhos et al., 2020), which our datasets do. Compositional generalization has been studied in machine translation (Dankers et al., 2022), semantic parsing (Kim and Linzen, 2020), and question answering over databases (Keysers et al., 2020). However, there hasn't been a systematic attempt to create new datasets by composing existing datasets with exceptions such as MusiQue (MRC) and SETI (Fu and Frank, 2023) (NLI).

Common strategies for improving a model's domain adaptation ability include: a) gradual fine-tuning with a mixture of data from different domains (Xu et al., 2021) – an approach motivated by curriculum learning, and b) domain adversarial training (Wright and Augenstein, 2020). To improve robustness generalization, researchers have trained on augmented data (Li et al., 2019), added a regularizer in the loss function (Goodfellow et al., 2015), and used a generator-discriminator setup (Kang et al., 2018). Neuro-symbolic methods (Gupta et al., 2020) and meta-learning (Lake, 2019) have been traditionally used to improve compositional generalization, and newer methods include better prompting strategies for in-context learning (Press et al., 2023). In contrast to previous work, our goal is not to provide a better algorithm/model for generalization but to examine existing models across different axes.

5 Conclusion & Future Work

We present a systematic study on the multi-dimensional (domain, robustness, and compositional) generalization abilities of common models used in NLP. Our main conclusions are: 1. Generalizability is a model instance characteristic and not generalization type-dependent – an instance typically does not generalize well in one dimension and poorly in others. 2. It is well correlated with model size, and certain architectures and training strategies generalize better than others. 3. Certain dimensions of generalization is harder to achieve compared to the others. We hope to inspire future work that looks further into the multi-dimensional aspect of generalizability and tries to understand why certain models generalize better than others.

Limitations

The conclusions of this study are dependent on the base datasets, models, and training methods used. There are many potential choices for these aspects, and while both the appropriateness and popularity inform our selections of the datasets or algorithms, we admit the conclusions might differ if we use alternatives. More base datasets and/or models would certainly improve the robustness of the conclusions, but these would exponentially increase the scale of the study. Other potential directions include investigating the amount of data needed for generalization, i.e., few-shot models, and cross-lingual generalization, but both are beyond the scope of the study. We have made empirical observations about generalization but have not investigated the theoretical reasons behind it. While that is beyond the scope of the study, we recognize this limitation.

Ethical Concerns

In this work, we train 72 models on the two datasets and further evaluate them on 15 datasets, which suffer from a combinatorial problem in terms of the necessary computing infrastructure. Our work consumed roughly two-thirds a month of GPU time ($\approx$ 500 hours). Combined with the size of the models, this limits the accessibility of this vein of research, especially if we were to expand to other datasets, model architectures, and few-shot training scenarios. More effort in understanding how to narrow down the choice of datasets before studying transfer would go a long way towards alleviating this issue. While we find that models generalize well across different scenarios, this should not be taken as an indication of their deployment eligibility in real-life scenarios. These models have not been tested for their propensity to generate toxic, biased, and offensive content.

Acknowledgements

We thank the anonymous reviewers for their helpful suggestions.

References

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. 2019. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854.

Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 2012. An empirical investigation of statistical significance in NLP. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 995–1005, Jeju Island, Korea. Association for Computational Linguistics.

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 B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared 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 M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher 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 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Nitay Calderon, Naveh Porat, Eyal Ben-David, Zorik Gekhman, Nadav Oved, and Roi Reichart. 2023. Measuring the robustness of natural language processing models to domain shifts. *arXiv preprint arXiv:2306.00168*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The pascal recognising textual entailment challenge. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 177–190, Berlin, Heidelberg. Springer Berlin Heidelberg.

Verna Dankers, Elia Bruni, and Dieuwke Hupkes. 2022. The paradox of the compositionality of natural language: A neural machine translation case study. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 4154–4175. Association for Computational Linguistics.

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.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. 2023. Parameter-efficient fine-tuning of large-scale pretrained language models. *Nature Machine Intelligence*, 5(3):220–235.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.

Cynthia Dwork, Ravi Kumar, Moni Naor, and Dandapani Sivakumar. 2001. Rank aggregation methods for the web. In *Proceedings of the 10th international conference on World Wide Web*, pages 613–622.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. HotFlip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 31–36, Melbourne, Australia. Association for Computational Linguistics.

Xiyan Fu and Anette Frank. 2023. SETI: Systematicity evaluation of textual inference. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4101–4114, Toronto, Canada. Association for Computational Linguistics.

Atticus Geiger, Kyle Richardson, and Christopher Potts. 2020. Neural natural language inference models partially embed theories of lexical entailment and negation. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 163–173, Online. Association for Computational Linguistics.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard S. Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. 2020. Shortcut learning in deep neural networks. *CoRR*, abs/2004.07780.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Shreya Goyal, Sumanth Doddapaneni, Mitesh M. Khapra, and Balaraman Ravindran. 2023. A survey of adversarial defenses and robustness in nlp. *ACM Comput. Surv.*, 55(14s).

Nitish Gupta, Kevin Lin, Dan Roth, Sameer Singh, and Matt Gardner. 2020. Neural module networks for reasoning over text. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112.

Luheng He, Mike Lewis, and Luke Zettlemoyer. 2015. Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 643–653, Lisbon, Portugal. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Lee. 2023. LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5254–5276, Singapore. Association for Computational Linguistics.

Dieuwke Hupkes, Mario Giulianelli, Verna Dankers, Mikel Artetxe, Yanai Elazar, Tiago Pimentel, Christos Christodoulopoulos, Karim Lasri, Naomi Saphra, Arabella Sinclair, et al. 2023. A taxonomy and review of generalization research in nlp. *Nature Machine Intelligence*, 5(10):1161–1174.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Yichen Jiang and Mohit Bansal. 2019. Avoiding reasoning shortcuts: Adversarial evaluation, training, and model development for multi-hop QA. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2726–2736, Florence, Italy. Association for Computational Linguistics.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is BERT really robust? A strong baseline for natural language attack on text classification and entailment. 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 8018–8025. AAAI Press.

Pratik Joshi, Somak Aditya, Aalok Sathe, and Monojit Choudhury. 2020. TaxiNLI: Taking a ride up the NLU hill. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 41–55, Online. Association for Computational Linguistics.

Jaap Jumelet, Milica Denic, Jakub Szymanik, Dieuwke Hupkes, and Shane Steinert-Threlkeld. 2021. Language models use monotonicity to assess NPI licensing. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 4958–4969. Association for Computational Linguistics.

Dongyeop Kang, Tushar Khot, Ashish Sabharwal, and Eduard H. Hovy. 2018. Adventure: Adversarial training for textual entailment with knowledge-guided examples. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2418–2428. Association for Computational Linguistics.

Divyansh Kaushik, Eduard Hovy, and Zachary Lipton. 2019. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations*.

Daniel Keysers, Nathanael Schärli, Nathan Scales, Hylke Buisman, Daniel Furrer, Sergii Kashubin, Nikola Momchev, Danila Sinopalnikov, Lukasz Stafiniak, Tibor Tihon, Dmitry Tsarkov, Xiao Wang, Marc van Zee, and Olivier Bousquet. 2020. Measuring compositional generalization: A comprehensive method on realistic data. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. UNIFIEDQA: Crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1896–1907, Online. Association for Computational Linguistics.

Najoung Kim and Tal Linzen. 2020. COGS: A compositional generalization challenge based on semantic interpretation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9087–9105, Online. Association for Computational Linguistics.

Brenden M. Lake. 2019. Compositional generalization through meta sequence-to-sequence learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 9788–9798.

Brenden M. Lake and Marco Baroni. 2018. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of

Proceedings of Machine Learning Research, pages 2879–2888. PMLR.

Brenden M Lake and Marco Baroni. 2023. Human-like systematic generalization through a meta-learning neural network. *Nature*, 623(7985):115–121.

David Yong Wey Lee. 2001. Genres, registers, text types, domains and styles: Clarifying the concepts and navigating a path through the bnc jungle. *Language Learning & Technology*, 5:37–72.

Changmao Li and Jeffrey Flanigan. 2024. Task contamination: Language models may not be few-shot anymore. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):18471–18480.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2019. Textbugger: Generating adversarial text against real-world applications. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*. The Internet Society.

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*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *CoRR*, abs/1806.08730.

R. Thomas McCoy, Junghyun Min, and Tal Linzen. 2020a. BERTs of a feather do not generalize together: Large variability in generalization across models with similar test set performance. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 217–227, Online. Association for Computational Linguistics.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2020b. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *57th Annual Meeting of the Association for Computational Linguistics, ACL 2019*, pages 3428–3448. Association for Computational Linguistics (ACL).

Aakanksha Naik, Jill Lehman, and Carolyn Rosé. 2022. Adapting to the long tail: A meta-analysis of transfer learning research for language understanding tasks. *Transactions of the Association for Computational Linguistics*, 10:956–980.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah Smith, and Mike Lewis. 2023. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5687–5711, Singapore. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

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.

Alan Ramponi and Barbara Plank. 2020. Neural unsupervised domain adaptation in NLP—A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6838–6855, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Swarnadeep Saha, Yixin Nie, and Mohit Bansal. 2020. ConjNLI: Natural language inference over conjunctive sentences. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8240–8252, Online. Association for Computational Linguistics.

Priyanka Sen and Amir Saffari. 2020. What do models learn from question answering datasets? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2429–2438. 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, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. MuSiQue: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

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.

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*.

Dustin Wright and Isabelle Augenstein. 2020. Transformer based multi-source domain adaptation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7963–7974, Online. Association for Computational Linguistics.

Haoran Xu, Seth Ebner, Mahsa Yarmohammadi, Aaron Steven White, Benjamin Van Durme, and Kenton Murray. 2021. Gradual fine-tuning for low-resource domain adaptation. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 214–221, Kyiv, Ukraine. Association for Computational Linguistics.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *EMNLP*, pages 2369–2380. Association for Computational Linguistics.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. 2023. Instruction tuning for large language models: A survey. *CoRR*, abs/2308.10792.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Wei Emma Zhang, Quan Z. Sheng, Ahoud Alhazmi, and Chenliang Li. 2020. Adversarial Attacks on Deep-learning Models in Natural Language Processing: A Survey. *ACM Transactions on Intelligent Systems and Technology*, 11(3):24:1–24:41.

Premise: The little boy in jean shorts kicks the soccer ball.
Hypothesis: A little boy is playing soccer outside.
Label: Neutral

Premise: The little boy in jean shorts kicks the soccer ball in the house.
Hypothesis: A little boy is playing soccer outside.
Label: Contradiction

Figure 6: A sample instance for robustness in NLI from SNLI-CF. The addition (in red) causes the label to change.

Premise: An Asian woman cutting the stems of a green leafy cabbage at a market.
Hypothesis: An Asian woman cutting the stems of a green leafy vegetable at a market.
Label: Entailment

Figure 7: A sample instance for compositionality in NLI. The label is entailment because every cabbage is a vegetable. Both "cabbage" and "vegetable" tokens appear in SNLI, but not in the same instance – this is a composed instance of these "constituents".

Premise: They're made from a secret recipe handed down to the present-day villagers by their Mallorcan ancestors, who came here in the early 17th century as part of an official repopulation scheme.
Hypothesis: The recipe passed down from Mallorcan ancestors is known to everyone.
Label: Contradiction

Figure 8: A sample instance for testing domain generalization in NLI from MNLI-matched.

Context: Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager.
Question: What is the name of the quarterback who was 38 in Super Bowl XXXIII?
Answer: John Elway

Context: Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.
Question: What is the name of the quarterback who was 38 in Super Bowl XXXIII?
Answer: John Elway

Figure 9: A sample instance for testing robustness generalization in MRC (Adv-SQuAD). Models are often fooled by the addition (red) and predict a different answer.

Context: One of Africa's brightest young writers, 31-year-old Chimamanda Adichie has already been recognised for her talent; her debut novel was shortlisted for the Orange Fiction Prize in 2004. The Nigerian novelist talks to CNN about her craft, her country and identity.
Question: What award has the novelist been nominated for?
Answer: Orange Fiction Prize

Figure 10: A sample instance for testing domain generalization in MRC (NewsQA)

Appendix

Datasets, models, hyperparameters, and training

We use publicly available datasets and modify them as needed. We present the dataset details in Table 4. Some instances are shown in Figures 6 to 11.

See Table 5 for the number of parameters in the used models.

For fully-tuned models, we use the HuggingFace Transformers library [5]. For EO models, we tokenize both NLI and MRC instances as pairs. For ED and DO models, we concatenate the premise and hypothesis as `premise: <> hypothesis: <>` for NLI instances. Similarly, for MRC instances, we concatenate the question and context as `question: <> context: <>`.

For LoRA models, we use the implementation from the HuggingFace PEFT library [6]. The hyperparameters are:

- $r = 16$
- $\alpha = 32$
- dropout = 0.05
- bias = None.

For Bottleneck adapters, we use the implementation from the adapters library in Adapter-hub [7] for all models except the `OPT` ones. The hyperparameters are:

- reduction_factor = 16

[5] https://github.com/huggingface/transformers

[6] https://github.com/huggingface/peft

[7] https://github.com/adapter-hub/adapters

Table 4: Details of the dataset used. We provide HuggingFace datasets public uris when available. For the datasets we created/modified, we provide a local copy.

dataset name	hf datasets link	split	size
SNLI	snli	train, validation, test	train: 550152, validation: 1000, test: 10000
MNLI-matched	multi_nli	validation_matched	9815
MNLI-mismatched	multi_nli	validation_mismatched	9832
HANS	hans	validation	30000
SNLI-CF	local	test	2000
SNLI-BT	local	test	18044
SNLI-H	au123/snli-hard	test	3261
CONJNLI	local	dev	624
TaxiNLI	local	dev	7728
SQuAD	rajpurkar/squad	train, validation	train: 87285, validation: 10485
Adv-SQuAD	local	validation footnote	3560
NewsQA	local	validation	1070
Adv-HotpotQA	local	validation	2828
MusiQue	local	validation	868

Context: During his bid to be elected president in 2004, Kerry frequently criticized President George W. Bush for the Iraq War. While Kerry had initially voted in support of authorizing President Bush to use force in dealing with Saddam Hussein, he voted against an $87 billion supplemental appropriations bill to pay for the subsequent war. His statement on March 16, 2004, "I actually did vote for the $87 billion before I voted against it," helped the Bush campaign to paint him as a flip-flopper and has been cited as contributing to Kerry's defeat.
Question: Why did Kerry criticize Bush during the 2004 campaign?
Answer: for the Iraq War

Context: In the lead up to the Iraq War, Kerry said on October 9, 2002; "I will be voting to give the President of the United States the authority to use force, if necessary, to disarm Saddam Hussein because I believe that a deadly arsenal of weapons of mass destruction in his hands is a real and grave threat to our security." Bush relied on that resolution in ordering the 2003 invasion of Iraq. Kerry also gave a January 23, 2003 speech to Georgetown University saying "Without question, we need to disarm Saddam Hussein. He is a brutal, murderous dictator; leading an oppressive regime he presents a particularly grievous threat because he is so consistently prone to miscalculation. So the threat of Saddam Hussein with weapons of mass destruction is real." Kerry did, however, warn that the administration should exhaust its diplomatic avenues before launching war: "Mr. President, do not rush to war, take the time to build the coalition, because itś not winning the war that's hard, it's winning the peace that's hard."
Question: When did Bush declare the Iraq War?
Answer: 2003

Context: During his bid to be elected president in 2004, Kerry frequently criticized President George W. Bush for the Iraq War. While Kerry had initially voted in support of authorizing President Bush to use force in dealing with Saddam Hussein, he voted against an $87 billion supplemental appropriations bill to pay for the subsequent war. His statement on March 16, 2004, "I actually did vote for the $87 billion before I voted against it," helped the Bush campaign to paint him as a flip-flopper and has been cited as contributing to Kerryś defeat. In the lead up to the Iraq War, Kerry said on October 9, 2002; "I will be voting to give the President of the United States the authority to use force, if necessary, to disarm Saddam Hussein because I believe that a deadly arsenal of weapons of mass destruction in his hands is a real and grave threat to our security." Bush relied on that resolution in ordering the 2003 invasion of Iraq. Kerry also gave a January 23, 2003 speech to Georgetown University saying "Without question, we need to disarm Saddam Hussein. He is a brutal, murderous dictator; leading an oppressive regime he presents a particularly grievous threat because he is so consistently prone to miscalculation. So the threat of Saddam Hussein with weapons of mass destruction is real." Kerry did, however, warn that the administration should exhaust its diplomatic avenues before launching war: "Mr. President, do not rush to war, take the time to build the coalition, because it's not winning the war that's hard, it's winning the peace that's hard."
Question: When did Bush declare the war causing Kerry to criticize him during the 2004 campaign?
Answer: 2003

Figure 11: A sample instance for testing compositionality in MRC (MusiQue) – The last question is a **composition** of the two questions above.

Table 5: Number of parameters in the used models.

model name	#params	
	base	large
BERT	110M	345M
RoBERTa	110M	345M
OPT	350M	1.3B
T5	220M	770M

- non_linearity = relu

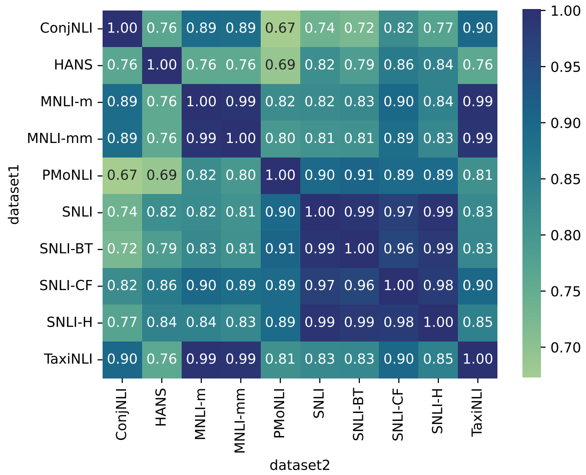
We do not use residual connections. For the OPT ones we implemented our own following (Hu et al., 2023). The hyper-parameters are kept the same.

We use the HuggingFace Transformers library for training the models, and the hyper-parameters are as follows:

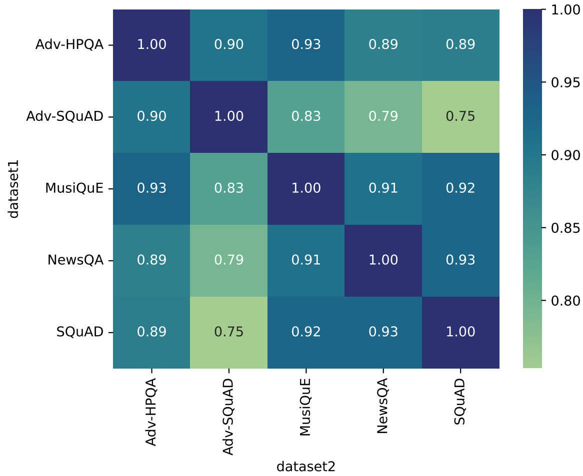
- Number of epochs: 3
- learning rate: 2e−5
- weight decay: 0.01

Results

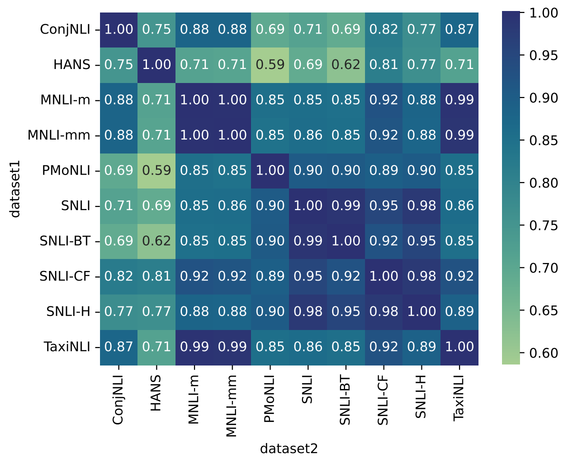
Spearman's rank correlation coefficient between two dataset pairs for NLI and MRC –Figures 13a and 13b.



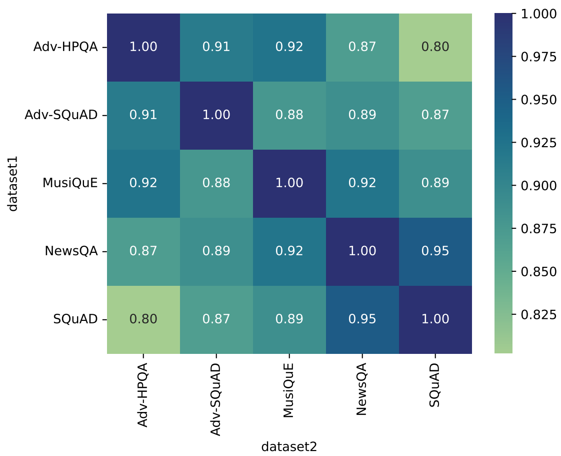
(a) NLI Datasets: Spearman's ρ



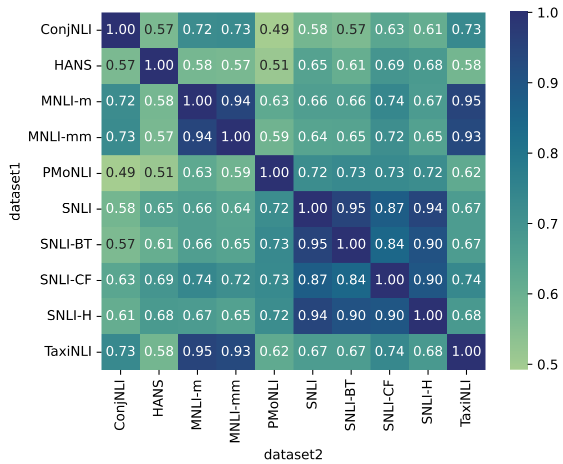
(b) MRC Datasets: Spearman's ρ



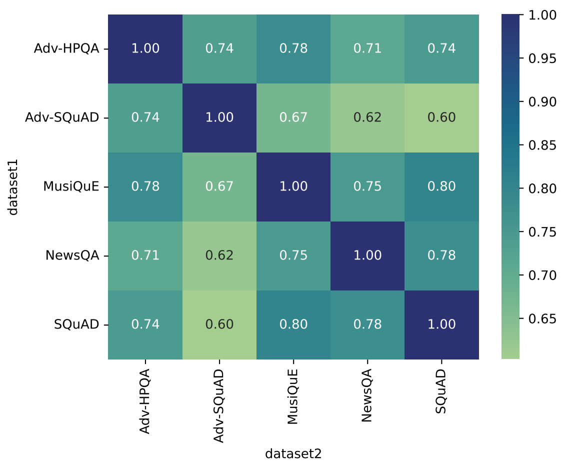
(c) NLI Datasets: Pearson's r



(d) MRC Datasets: Pearson's r

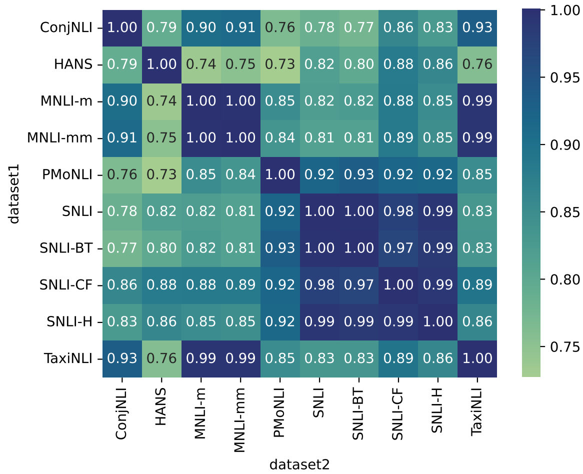


(e) NLI Datasets: Kendall's τ

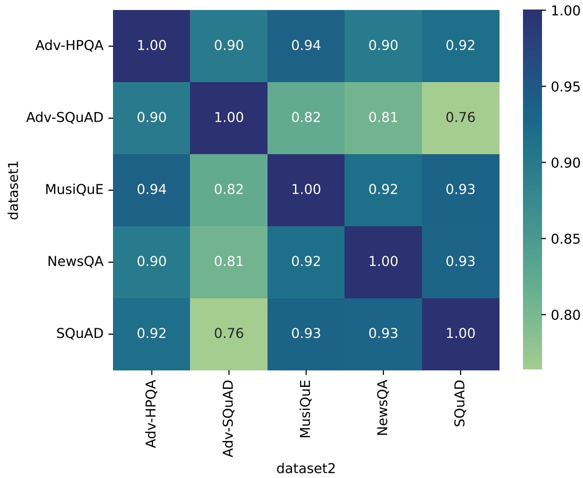


(f) MRC Datasets: Kendall's τ

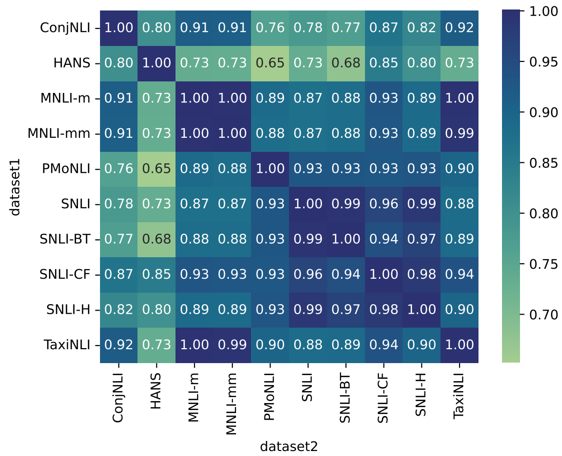
Figure 12: Correlation between the source and the target datasets for NLI and MRC on a per-instance basis for different kinds of correlation.



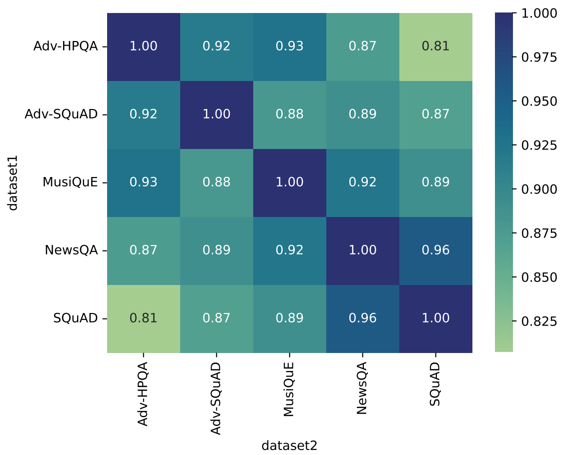
(a) NLI Datasets: Spearman's ρ



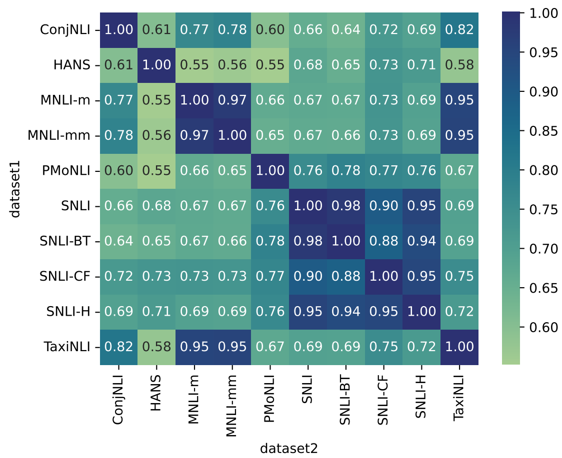
(b) MRC Datasets: Spearman's ρ



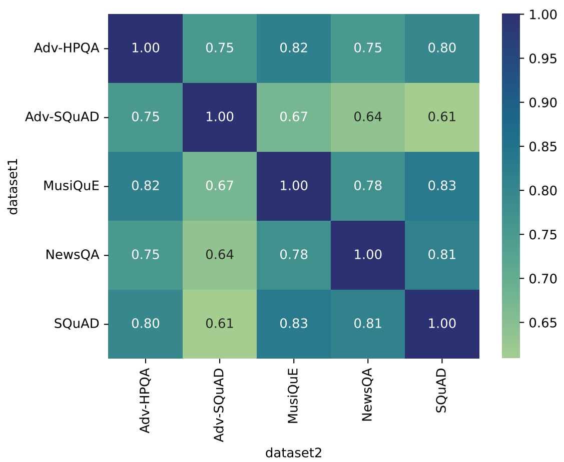
(c) NLI Datasets: Pearson's r



(d) MRC Datasets: Pearson's r



(e) NLI Datasets: Kendall's τ



(f) MRC Datasets: Kendall's τ

Figure 13: Correlation between the source and the target datasets for NLI and MRC on a per-architecture basis for different kinds of correlation.