On Measuring the Intrinsic Few-Shot Hardness of Datasets

Xinran Zhao* **Shikhar Murty*** **Christopher D. Manning**
Computer Science Department, Stanford University
{xzhaoar,smurty,manning}@cs.stanford.edu

Abstract

While advances in pre-training have led to dramatic improvements in few-shot learning of NLP tasks, there is limited understanding of what drives successful few-shot adaptation in datasets. In particular, given a new dataset and a pre-trained model, what properties of the dataset make it *few-shot learnable* and are these properties independent of the specific adaptation techniques used? We consider an extensive set of recent few-shot learning methods, and show that their performance across a large number of datasets is highly correlated, showing that few-shot hardness may be intrinsic to datasets, for a given pre-trained model. To estimate intrinsic few-shot hardness, we then propose a simple and lightweight metric called `Spread` that captures the intuition that few-shot learning is made possible by exploiting feature-space invariances between training and test samples. Our metric better accounts for few-shot hardness compared to existing notions of hardness, and is ~8–100x faster to compute.

1 Introduction

A growing body of recent work has shown impressive advances in few-shot adaptation of pre-trained transformers (Radford et al., 2019; Schick and Schütze, 2020; Brown et al., 2020; Karimi Mahabadi et al., 2021; Liu et al., 2021a, among others). Despite this progress, there is no concrete understanding of when and why few-shot learning may be successful for a given pre-trained model. Indeed, *no free lunch* style arguments necessitate the existence of tasks that are not few-shot learnable by a given pre-trained model, regardless of the adaptation method, and in practice, very similar datasets exhibit varying levels of success when state-of-the-art few-shot adaptation methods are applied (Fig. 1a).

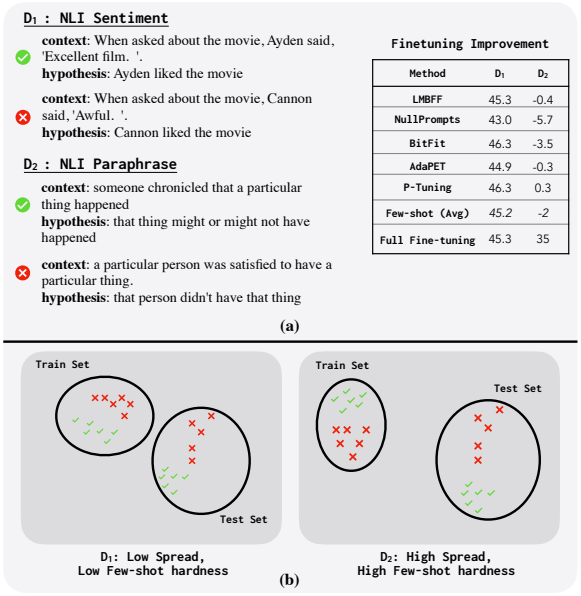

Figure 1: (a) Fine-tuning a model on D_1 (D_2) on the entire dataset leads to a +45.3 (+35) point improvement over a random baseline. However, most methods are successful at few-shot adaptation on D_1 with an average improvement of ~45 points over a random baseline, while few-shot adaptation is unsuccessful on D_2. (b) We observe that features of test inputs are closer to training set inputs for D_1 than D_2, motivating `Spread` as a metric for evaluating *few-shot hardness*.

This work advances understanding of *few-shot learnability* in two ways. First, we find that, given a dataset, few-shot performance of various adaptation methods is highly correlated. This suggests the existence of adaptation-method independent factors behind few-shot learnability of a dataset, for a given pre-trained model. Next, we propose a simple and lightweight metric to estimate this *intrinsic few-shot learnability*. Concretely, we consider an extensive set of recently proposed adaptation methods and find that on a wide range of datasets, these methods have highly correlated behaviors i.e., the degree to which a few-shot adaptation method succeeds on a dataset is correlated across methods.

* Equal Contribution

Next, we consider two recently proposed methods that may be used for assessing *intrinsic dataset hardness*—Rissanen Data Analysis (RDA, Perez et al. (2021)) and Sensitivity Analysis (SA, Hahn et al. (2021)). RDA computes dataset hardness as the area under the curve of the test loss as a function of number of training samples, while SA computes hardness by examining how perturbations in the input features cause a model to change its predicted label. From experiments, we show that SA is poorly correlated with few-shot hardness and RDA, while well correlated, is very expensive to compute. In response, we propose a new metric that measures the ability of a model to exploit feature-space invariances between the train and test set to make few-shot generalizations. For instance, consider the datasets in Fig. 1b, where D_1 has a test set that "looks similar" to the training set while D_2 has a test set that looks dissimilar to the training set. We capture this intuition into a lightweight metric called `Spread` and show that `Spread` correlates as well or better than existing methods (Spearman correlation of 0.467 vs 0.356) while being ~8–100x more computationally efficient.

2 Background

Consider a labeled dataset $\mathcal{D} = \{z^{(1)}, z^{(2)}, \ldots\}$, split into a training set $\mathcal{D}_{tr}$ and a test set $\mathcal{D}_{ts}$, where each example $z^{(k)}$ is a tuple consisting of an input $x^{(k)}$ and a label $y^{(k)}$. Typically, in k-way l-shot learning (l is typically less than 128), $\mathcal{D}_{tr}$ consists of l examples of each of the k labels. Given some pre-trained model f, a few-shot adaptation method m uses $\mathcal{D}_{tr}$ to modify f, outputting an "adapted model" that can make predictions on $\mathcal{D}_{ts}$.

State-of-the-art approaches for such adaptation typically involve either recasting the task into the pre-training objective of the model (Gao et al., 2021a; Tam et al., 2021), or using lightweight / parameter efficient finetuning (Houlsby et al., 2019; Logan IV et al., 2022; Li and Liang, 2021; Liu et al., 2021b; Ben Zaken et al., 2022). For this work, we experiment with an extensive set of recently proposed few-shot adaptation methods that we further categorize into *prompt-based* methods which includes LMBFF (Gao et al., 2021a), AdaPET (Tam et al., 2021), Null Prompts (Logan IV et al., 2022) and Prompt-Bitfit (Ben Zaken et al., 2022)[1], and *Light-weight* finetuning methods which includes Prefix Tuning (Li and Liang, 2021) and Compacter (Karimi Mahabadi et al., 2021).

[1] As demonstrated by (Logan IV et al., 2022), we use trigger prompts in BitFit to improve few-shot performance.

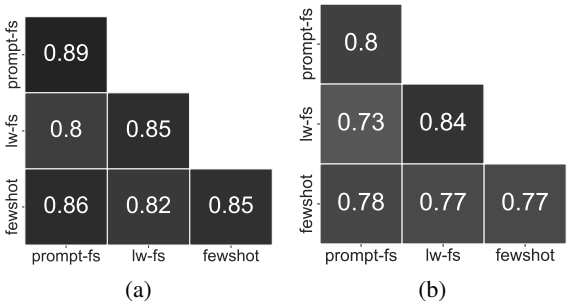

Figure 2: Average correlation between hardness measurements from various few-shot adaptation methods on (a) FS-GLUE and (b) FS-NLI. "prompt-fs" refers to prompt-based methods while "lw-fs" refers to lightweight finetuning methods. We note a large correlation of 0.85 and 0.77. We also observe a higher correlation among methods from the same category.

3 Intrinsic Few-Shot Hardness

For a fixed method and dataset, we define "method specific few-shot hardness" (MFH) as a function that takes a dataset $\mathcal{D}$ and some adaptation method m, and outputs a real number that captures few-shot hardness of $\mathcal{D}$ with respect to f and m. Concretely, for this work, we define $\mathrm{MFH}(\mathcal{D}, f, m)$ as the classification accuracy of the adapted model, normalized against the classification accuracy of the majority baseline, on $\mathcal{D}_{ts}$.

Datasets. In FS-GLUE, we consider 11 tasks (details in Appendix A.1) from the GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019) benchmarks, covering a wide range of task formats and objectives. Additionally, since these tasks have differing textual formats as well as performance metrics (F1 / accuracy / Pearson's correlation), we curate FS-NLI where every dataset is recast into 2 way NLI, and the performance metric is standardized as classification accuracy. To do this, we take datasets from White et al. (2017); Poliak et al. (2018); Richardson et al. (2019), giving us a collection of 28 datasets (more details in Appendix A.1), covering a wide range of tasks such as sentiment analysis, negation comprehension, name entity classification (NEC), paraphrase detection, event classification, logical entailment etc.

3.1 Experiments

We compute MFH values for all adaptation methods on FS-GLUE and FS-NLI. For each dataset, we sample a fixed training set consisting of 64 examples per label. Finally, we control for hyperparameter tuning across methods (details in Appendix A.2). We report the average spearman correlation between MFH values for all pairs of adaptation methods for every dataset in Fig. 2.

Results. We observe an extremely high average correlation between hardness measurements for different methods—0.85 and 0.77 for FS-GLUE and FS-NLI respectively. The average correlation is higher among methods within the same category than among methods from different categories e.g., on FS-GLUE, the average correlation between prompt-based and lightweight finetuning methods is 0.8, while the average correlation among all prompt-based methods is 0.89. We conclude that few-shot hardness may be intrinsic to datasets, i.e., for a given pre-trained model, a dataset may be "easy" or "hard" *regardless* of the adaptation method used.

4 Automatic Metrics For Intrinsic Few-Shot Hardness

Noting that the experiments of Section 3 suggest that few-shot hardness may be intrinsic to datasets, we define the "intrinsic few-shot hardness" (IFH) of a dataset as the mean MFH values across a collection of adaptation method $\mathcal{M} = \{m_1, m_2, \ldots\}$,

$$\text{IFH}(f, \mathcal{D}) \triangleq \frac{\sum_{m \in \mathcal{M}} \text{MFH}(\mathcal{D}, f, m)}{|\mathcal{M}|} \quad (1)$$

Clearly, computing IFH values for some dataset is computationally intensive since it requires running multiple few-shot adaptation methods on f. Thus, to automatically estimate IFH, an automatic hardness metric h takes f and $\mathcal{D}$ and outputs a (real-valued) score $h(f, \mathcal{D})$ that is well-correlated with $\text{IFH}(f, \mathcal{D})$ across some collection of datasets $\{\mathcal{D}_1, \mathcal{D}_2, \ldots\}$.

4.1 Existing Metrics

We consider two recently proposed dataset hardness metrics that have been applied in the context of pre-trained models. $h_{\text{sensitivity}}$ (Hahn et al., 2021) measures how many tokens of an input need to be perturbed for model predictions to change, on average. A low sensitivity implies that model predictions change only when a large subset of an input's tokens are perturbed, making the dataset "easy". h_{RDA} (Perez et al., 2021) approximates hardness as the area under the curve of the test loss obtained by finetuning f on successively larger slices of the training data—a large area implies that a large number of training samples are required to achieve low test set loss, implying that the dataset is hard for f.

Metric	Correlation	Compute Time (s)
$h_{\text{sensitivity}}$	0.06	80
h_{RDA}	0.36	1,020
$h_{\texttt{Spread}}$	**0.47**	10

Table 1: Compared to baseline hardness metrics, $h_{\texttt{Spread}}$ is able to better account for intrinsic few-shot hardness and is computationally lightweight. All experiments are run on a single 1080ti GPU for profiling.

4.2 Our Approach

In the few-shot regime, one of the factors affecting dataset hardness is the degree to which input features for a given label in the test set differ from the training set (Fig. 1b). Based on this intuition, we propose a simple *few-shot specific* hardness metric called `Spread` that computes the average euclidean distance of test input features to the closest training set input.

For some input $z = (x, y) \in \mathcal{D}$, let $f(z)$ denote the vector valued features of the input x produced by the model. We define the distance between some example $z^{(k)}$ to the training set $\mathcal{D}_{tr}$ as,

$$d(f, z^{(k)}, \mathcal{D}_{tr}) \triangleq \min_{z} \|f(z^{(k)}) - f(z)\| \quad (2)$$
$$\text{subject to: } z = (x, y) \in \mathcal{D}_{tr}$$
$$y = y^{(k)}.$$

Then, $h_{\texttt{Spread}}(f, \mathcal{D})$ can be computed as

$$h_{\texttt{Spread}}(f, \mathcal{D}) \triangleq \frac{1}{|\mathcal{D}_{ts}|} \sum_{z^{(k)} \in \mathcal{D}_{ts}} d(f, z^{(k)}, \mathcal{D}_{tr}) \quad (3)$$

4.3 Experiments

We experiment with RDA, SA and `Spread` as our few-shot hardness metrics. To obtain input features for computing `Spread`, we use SimCSE (Gao et al., 2021b) features of the input with a RoBERTa-large base. Given a collection of datasets and methods,

Method	Correlation
LMBFF	0.72
AdaPET	0.58
Null Prompt	0.67
Prompt-Bitfit	0.79

Table 2: Correlation between hardness measurements from using the same adaptation method with different pre-trained models. We observe a high correlation among hardness measurements from using different base pre-trained models.

Method	Ours	Control
LMBFF	17.1	**19.7**
AdaPET	**32.3**	8.5
Null Prompt	**3.3**	-7.7
Prompt-Bitfit	**2.9**	-9.0
Few-shot Avg.	**13.9**	2.9

Table 3: Comparing a `Spread`-inspired training set sampling strategy to random sampling. We observe an average boost of 13.9 accuracy points compared to the control where the ~3 point boost is due to ensembling.

we measure the correlation between metric outputs and IFH values for each dataset, and report the average across all datasets. We also report time taken for computing the metric on a single 1080ti GPU for each dataset, and report the average compute time across all datasets.

Results. We report results on FS-NLI to ensure uniformity of task formats and performance metrics. From Table 1, we observe that $h_{\text{sensitivity}}$ is poorly correlated with IFH. Next, we note that while h_{RDA} produces better hardness judgements, these come at the cost of increased computation time. Finally, $h_{\texttt{Spread}}$ is able to best account for intrinsic few-shot hardness while being ~100x computationally lightweight compared to h_{RDA}.

4.4 Measuring IFH across pre-trained models

While we define IFH of a dataset with respect to a fixed pre-trained model, certain datasets or tasks might be few-shot hard for a wider range of pre-trained models. To investigate this further, we experiment with Electra-large (Clark et al., 2020) as the base pre-trained model. We obtain hardness measurements on FS-NLI and compute the correlation between methods with different base pre-trained models (RoBERTa-large vs Electra-large). From average correlation across datasets in Table 2, we find that hardness measurements from the same method with different pre-trained models are well-correlated. We conclude that there may even be "pre-trained model independent" factors behind intrinsic few-shot hardness, and we leave further analysis of this to future work.

5 Decreasing Few-Shot hardness via dataset decomposition

We conclude experiments with a simple application of `Spread` by proposing a training set sampling strategy that minimizes `Spread` values for improved few-shot performance. Our strategy is based on performing a "clustering based decomposition" of a dataset, similar to Murty et al. (2021). In particular, we run k-means clustering on input features from the pre-trained model to create P clusters. Then, we train P *distinct* models on few-shot training sets sampled from each of the clusters. At test time, examples are classified into one of the P clusters, and the corresponding model is used to make predictions. To account for any effects due to ensembling, we compare with an approach where we randomly sample a model to make predictions, instead of using the model corresponding to the cluster.

Results. We experiment with the sampling strategy described above on a named entity classification based dataset from FS-NLI. From Table 3, we note an improvement of ~14 accuracy points, while the control increases by only a ~3 accuracy points.

6 Related Work and Discussion

Most notions of what makes datasets challenging are either model agnostic—example length (Spitkovsky et al., 2010), order sensitivity (Nie et al., 2019) etc, or indirect (Agarwal and Hooker, 2020). On the other hand, intrinsic hardness as defined in this work, is both model-centric and directly measures the ability of a *specific* model family to make good generalizations from a training set. While measuring hardness of datasets has seen some recent traction (Hahn et al., 2021; Perez et al., 2021), to the best of our knowledge, we are the first to study hardness in the context of few-shot adaptation. We show that few-shot hardness of datasets (as measured by test set performance normalized against a random baseline) among an extensive set of recently proposed methods is highly correlated, suggesting that few-shot hardness may be a property intrinsic to datasets. We then propose a simple hardness metric called `Spread`, based on the in-

tuition that a test set with input features close to the few-shot training set is easy, since it allows a model to exploit feature-space invariances. Compared to other metrics, `Spread` provides hardness judgements that are much better correlated with intrinsic few-shot hardness while being 8–100x faster to compute, compared to prior hardness metrics.

Metrics for predicting fewshot hardness have several applications. For the NLP practitioner, `Spread` could be used as a simple plug-and-play estimator of whether few-shot adaptation of a pre-trained model might be successful for a given use case. For the dataset developer, `Spread` could be used to adversarially curate harder few-shot benchmarks. Finally, a model developer can use `Spread` to inform better training set sampling strategies to improve test set performance in the few-shot training regime, as well as for model selection by selecting models with lower `Spread`.

7 Acknowledgements

SM was funded by a gift from Apple Inc. We are grateful to John Hewitt, Eric Mitchell, Roy Schwartz and the anonymous reviewers for helpful comments.

8 Reproduciblity

Our code is available at: https://github.com/colinzhaoust/intrinsic_fewshot_hardness.

9 Limitations

Instance-level Analysis. We focus on discovering and measuring intrinsic few-shot hardness at the dataset level and do not study *instance-level* hardness quantitatively or qualitatively. Understanding hardness at the instance level can further help understand the recent successes behind few-shot learning in NLP.

Base model selection. We compare few-shot performance between a variety of methods with two base models that have very different pre-training objectives, yet have correlated few-shot behaviors across tasks and adaptation methods. Of course, there exist a much wider range of pre-trained models with very diverse pre-training data and objectives. While beyond the scope of this work, we believe that studying factors such as the relationship between few-shot performance and pre-training data / objectives is worth further investigation.

References

Chirag Agarwal and Sara Hooker. 2020. Estimating example difficulty using variance of gradients. *CoRR*, abs/2008.11600.

Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. 2022. BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–9, Dublin, Ireland. 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. *CoRR*, abs/2005.14165.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Pre-training transformers as energy-based cloze models. In *EMNLP*.

Ido Dagan, Bill Dolan, Bernardo Magnini, and Dan Roth. 2010. Recognizing textual entailment: Rational, evaluation and approaches – erratum. *Natural Language Engineering*, 16(1):105–105.

Marie-Catherine de Marneffe, Mandy Simons, and Judith Tonhauser. 2019. The commitmentbank: Investigating projection in naturally occurring discourse. *Proceedings of Sinn und Bedeutung*, pages 107–124.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021a. Making pre-trained language models better few-shot learners. In *Association for Computational Linguistics (ACL)*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021b. SimCSE: Simple contrastive learning of sentence embeddings. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Michael Hahn, Dan Jurafsky, and Richard Futrell. 2021. Sensitivity as a Complexity Measure for Sequence Classification Tasks. *Transactions of the Association for Computational Linguistics*, 9:891–908.

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 *Proceedings of the 36th International Conference on Machine Learning*.

Rabeeh Karimi Mahabadi, Sebastian Ruder, Mostafa Dehghani, and James Henderson. 2021. Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks. In *Annual Meeting of the Association for Computational Linguistics*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ArXiv*, abs/2107.13586.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2021b. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *CoRR*, abs/2110.07602.

Robert Logan IV, Ivana Balazevic, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel. 2022. Cutting down on prompts and parameters: Simple few-shot learning with language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2824–2835, Dublin, Ireland. Association for Computational Linguistics.

Shikhar Murty, Tatsunori B. Hashimoto, and Christopher Manning. 2021. DReCa: A general task augmentation strategy for few-shot natural language inference. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1113–1125, Online. Association for Computational Linguistics.

Yixin Nie, Yicheng Wang, and Mohit Bansal. 2019. Analyzing compositionality-sensitivity of NLI models. In *AAAI*.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. Rissanen data analysis: Examining dataset characteristics via description length. In *ICML*.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. 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 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.

Adam Poliak, Aparajita Haldar, Rachel Rudinger, J. Edward Hu, Ellie Pavlick, Aaron Steven White, and Benjamin Van Durme. 2018. Collecting diverse natural language inference problems for sentence representation evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 67–81, Brussels, Belgium. Association for Computational Linguistics.

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

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.

Kyle Richardson, Hai Hu, Lawrence S. Moss, and Ashish Sabharwal. 2019. Probing natural language inference models through semantic fragments. *CoRR*, abs/1909.07521.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *AAAI spring symposium: logical formalizations of commonsense reasoning*, pages 90–95.

Timo Schick and Hinrich Schütze. 2020. It's not just size that matters: Small language models are also few-shot learners. *Computing Research Repository*, arXiv:2009.07118.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Valentin I. Spitkovsky, Hiyan Alshawi, and Daniel Jurafsky. 2010. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 751–759, Los Angeles, California. Association for Computational Linguistics.

Derek Tam, Rakesh R Menon, Mohit Bansal, Shashank Srivastava, and Colin Raffel. 2021. Improving and simplifying pattern exploiting training. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 4144–4150.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2018. Neural network acceptability judgments. *arXiv preprint arXiv:1805.12471*.

Aaron Steven White, Pushpendre Rastogi, Kevin Duh, and Benjamin Van Durme. 2017. Inference is everything: Recasting semantic resources into a unified evaluation framework. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 996–1005, Taipei, Taiwan. Asian Federation of Natural Language Processing.

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. Association for Computational Linguistics.

Dataset	#Test	Majority
event	4,342	0.50
ner	37,638	0.50
gender	464	0.50
puns	1,756	0.50
lexico_syntactic	15,236	0.51
relation_extraction	761	0.60
sentiment	600	0.50
semantic_role	1,821	0.59
paraphrase	2,109	0.55
anaphora	146	0.51
negation	1,000	0.67
boolean	1,000	0.73
quantifier	1,000	0.66
counting	1,000	0.66
conditional	1,000	0.66
comparative	1,000	0.65
monotonicity	2,000	0.67
monotonicity_simple	1,000	0.67
monotonicity_hard	1,000	0.68
rte	277	0.53
mnli	10,000	0.63
ner_merged	36,789	0.51
ner_person	9,032	0.55
ner_entity	5,086	0.58
ner_location	8,958	0.64
ner_event	96	0.5
ner_organization	7,851	0.5
ner_time	5,766	0.65

Table 4: Statistics of NLI datasets evaluated in this work, all the datasets have two label classes (entailed and not-entailed). The number of examples in support and validation set are jointly 64/128 per label class, except from *ner_merged*, where we use the support/test examples from all other *ner_x* tasks. *ner* denotes the named entity classification (NEC) task and *ner_x* denotes the NEC task with *x* as the label.

A Appendix

A.1 Dataset Details

We consider 11 tasks from the GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019) benchmarks, namely SST-2 (Socher et al., 2013), CoLA (Warstadt et al., 2018), MNLI (Williams et al., 2018), QNLI (Rajpurkar et al., 2016), RTE (Dagan et al., 2010), MRPC (Dolan and Brockett, 2005), QQP (Wang et al., 2017), BoolQ (Clark et al., 2019), CB (de Marneffe et al., 2019), COPA (Roemmele et al., 2011), and WiC (Pilehvar and Camacho-Collados, 2019). We exclude the datasets that contain passages longer than the prompts.

We also take NLI datasets from White et al. (2017); Poliak et al. (2018); Richardson et al. (2019), giving us a collection of 28 datasets. If the original dataset contains three labels (entailment, contradiction, neutral), we merge contradiction and neutral to form a binary classification with entailed and not-entailed as the examples. The statistics of the datasets are shown in Table 4.

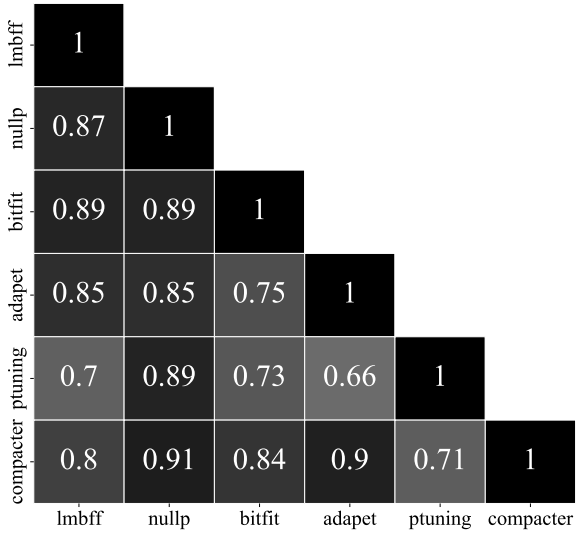

Figure 3: Average correlation between hardness measurements from various few-shot adaptation methods on FS-GLUE. We observe general good correlation among various few-shot methods.

A.2 Implementation Details

Hyper-parameter Details: For grid search, we choose a learning rate from the set 5e-6, 1e-5, 5e-5 and train for 30 epochs. For each method, we train for 1000 steps, evaluating every 100 steps. All adaptation methods use RoBERTa-large as the base model. All other hyperparameters are as used in the original works. We follow the original work to generate and select prompts for LMBFF. For AdaPET, we use the same prompts as we used in LMBFF.

A.3 Intrinsic few-shot hardness

In Fig. 2, we show averaged spearman correlations of methods clustered according to their type. In Fig. 3 and Fig. 4, we further show detailed correlations between every pair of methods. We generally observe high correlations across methods for both FS-GLUE and FS-NLI tasks. In Fig. 4, we further add $h_{\texttt{Spread}}$, h_{RDA}, and $h_{\text{sensitivity}}$ as references.

A.4 Decreasing Few-Shot hardness via dataset decomposition (Addendum)

We present the strategy of clustering based decomposition on named entity classification task from FS-NLI and demonstrates its effectiveness in Table 3. Since there naturally exist heuristics based cluster (e..g., questions about location, organiza-

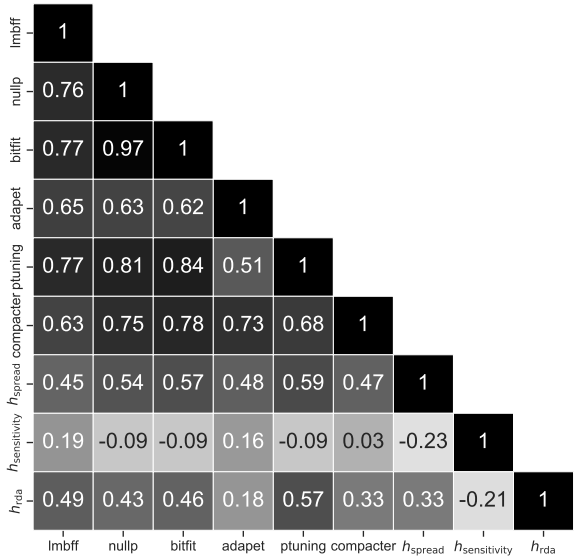

Figure 4: Average correlation between hardness measurements from various few-shot adaptation methods on FS-NLI. We observe general good correlation among various few-shot methods.

Clusters	1	2	3	4	5	6
person	0.21	0.05	0.06	0.11	0.05	0.19
entity	0.07	0.14	0.12	0.09	0.09	0.02
location	0.04	0.17	0.12	0.11	0.05	0.26
event	0.03	0.03	0.01	0.03	0.02	0.02
organization	0.06	0.12	0.11	0.07	0.05	0.26
time	0.20	0.07	0.05	0.15	0.05	0.02

Table 5: Jaccard index between each pair of clusters, where 1 to 6 denote the 6 clusters generated by k-means clustering over the representation space and person, entity, and etc denote the heuristics based clusters.

tion, and person) in NEC, we here show the relevance between model-generated clusters and these heuristics based cluster by computing the Jaccard index. From Table 5, we can observe that model-based clusters are not decomposing the dataset in a similar way as the heuristics.