Measuring the Robustness of NLP Models to Domain Shifts

Anonymous ACL submission

Abstract

Existing research on Domain Robustness (DR) suffers from disparate setups, limited task variety, and scarce research on recent capabilities such as in-context learning. Furthermore, the common practice of measuring DR might not be fully accurate. Current research focuses on challenge sets and relies solely on the Source Drop (SD): Using the source in-domain performance as a reference point for degradation. However, we argue that the Target Drop (TD), which measures degradation from the target in-domain performance, should be used as a complementary point of view. To address these issues, we first curated a DR benchmark comprised of 7 diverse NLP tasks, which enabled us to measure both the SD and the TD. We then conducted a comprehensive large-scale DR study involving over 14,000 domain shifts across 21 fine-tuned models and few-shot LLMs. We found that both model types suffer from drops upon domain shifts. While fine-tuned models excel in-domain, few-shot LLMs often surpass them cross-domain, showing better robustness. In addition, we found that a large SD can often be explained by shifting to a harder domain rather than by a genuine DR challenge, and this highlights the importance of TD as a complementary metric. We hope our study will shed light on the current DR state of NLP models and promote improved evaluation practices toward more robust models. [1]

1 Introduction

Modern transformer-based NLP models, and particularly *Large Language Models (LLMs)* have proven effective on various tasks and evaluation setups, including fine-tuning (Devlin et al., 2018; Raffel et al., 2020) and in-context learning (Brown et al., 2020; Chowdhery et al., 2022). Following that, there has been an improvement in the models' ability to perform tasks while transferring to domains with no labeled data available (Hendrycks et al., 2020; Ben-David et al., 2022a; Wang et al., 2022a). Despite these improvements, the performance upon domain shift can still be inferior to the model's performance on the source domains, a problem we refer to as the *Domain Robustness (DR) challenge* (Ramponi and Plank, 2020; Wang et al., 2022b; Hupkes et al., 2023; Yang et al., 2023b).

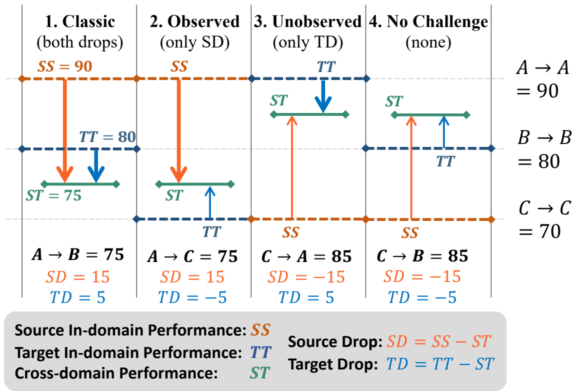

Figure 1: Illustration of the four domain shift scenarios. In the *Classic* and *Observed* scenarios, we observe a 15-point drop between the *Source In-domain Performance* (SS) and the *Cross-domain Performance* (ST). Conversely, in the *Unobserved* and *No Challenge* scenarios, SS $= 70$ and ST $= 85$, meaning the model gains 15 points upon domain shift. We would typically conclude that there is a DR challenge only in the first two scenarios. However, we argue that this commonly adopted perspective is inaccurate since it overlooks the *Target In-domain Performance* (TT). Our work provides a fresh perspective by considering both degradation metrics: The *Source Drop* (SD) and the *Target Drop* (TD).

Research of DR is quite disparate: A wide variety of setups, models, training procedures, and different dataset sizes are used. There is also a severe lack of variety in evaluation tasks for DR: Most papers use classification tasks, omitting important tasks such as sequence tagging, question answering, and text generation (Hendrycks et al., 2020; Koh et al., 2021). Moreover, many past works use challenge sets to measure the DR challenge. These

[1] Our benchmark will be released upon acceptance.

are highly curated datasets that select synthetic (Belinkov and Bisk, 2018; Rychalska et al., 2019) or particularly hard samples for models to process under domain shifts (McCoy et al., 2019; Yuan et al., 2023). All this makes it hard to compare different works and map out the extent of the DR challenge in a *natural domain shift setting*.

Moreover, prior works focused solely on fine-tuned models, disregarding few-shot setups that have become prominent.[2] In those setups, the DR challenge manifests itself more moderately: No training data can potentially anchor the model to the source distribution, but only a few demonstrations from the source domain are used in the prompt (Min et al., 2022; Weber et al., 2023).

Adding to the above, we observe a fundamental problem with how we examine the DR challenge. Let us conduct a thought experiment, illustrated in Figure 1: A model is trained and tested on data from domain A ($A \rightarrow A$), achieving a score of 90, but when tested on domains B and C, it scores 75. The observed 15-point drop typically leads to the conclusion the model lacks robustness, a common assertion in DR papers. But what if we were told that "had the model been trained and tested on data from B, it would have achieved a score of 80", would we still consider it as facing a severe DR challenge, given only a 5-point drop from B's in-domain performance ($B \rightarrow B$), rather than 15? Furthermore, if the model attains a score of 70 when trained and tested on domain C ($C \rightarrow C$), can we still assert a DR challenge exists even when it performs better cross-domain ($A \rightarrow C$)?

Building on the insights from the thought experiment, our paper introduces a novel perspective on the DR challenge. Traditional approaches typically focus on the **Source Drop (SD)**, assessing how model performance degrades compared to its source in-domain performance. However, this view overlooks the degradation compared to the setup where the model had been trained and tested on the target domain, which we define as the **Target Drop (TD)**. We study these variables and build various metrics upon them in §3.

Importantly, most works focus solely on the SD and overlook the TD, resulting in a partial depiction of the DR challenge. For instance, in studies involving challenge sets that report a large SD, the drop may be primarily attributed to shifting to a harder domain (TT < SS, see §3.1), and not by a genuine DR challenge, e.g., the *Classic* and *Observed* scenarios in Figure 1. By incorporating both metrics, we aim to provide a more holistic and accurate understanding of the DR challenge.

To overcome deficiencies in the current body of research, in §4 we introduce a novel DR benchmark. Unlike existing benchmarks, which largely rely on synthetic, adversarial, or challenge sets that may not adequately represent natural settings, our benchmark is unique and possesses four key properties: (i) It focuses on shifts (such as topical shifts) that naturally occur in real-life scenarios; (ii) It covers a wide variety of NLP tasks, more than other studies, including sequence and token level classification, QA, and generation tasks; (iii) Each task consists of several domains; and (iv) Each domain has a sufficient amount of labeled data, enabling its use as a source and as a target domain.

Following that, we conduct an extensive study by benchmarking many fine-tuned models and few-shot LLMs, detailed in §5. We examine factors such as the model size, dataset size, number of few-shot demonstrations, and more. Our findings, reported in §6, incorporate results of more than 14,000 domain shifts of 21 models and various training and testing setups. Our main findings are:

1. Fine-tuned models suffer from drops upon domain shifts. While the extent of the drop varies, challenging shifts are prevalent in every task;
2. Increasing the size of fine-tuned models enhances both in-domain and cross-domain performance while reducing performance drops, particularly in classification tasks;
3. Few-shot models also face a DR challenge as the domain of the demonstrations impacts their performance. However, the domain shift effect for few-shot models is weaker and more nuanced;
4. Increasing the fine-tuning dataset size as well as the number of few-shot demonstrations enhances in-domain and cross-domain performance but can also mildly increase the drop due to stronger "source domain anchoring";
5. While fine-tuned models excel in-domain, few-shot LLMs often surpass them cross-domain, showing better robustness and smaller drops;
6. Considering only one metric can lead to wrong conclusions since many domain shifts are not *Classic*, and only one drop metric (SD or TD) is positive while the other is negative;
7. We found that a large SD can often be explained

[2] We use *few-shot models* to denote LLMs in an in-context learning setting, where the prompt contains demonstrations.

by shifting to a harder domain, and not by a genuine DR challenge;

8. Our focus on many natural domain shifts reveals that while challenge sets are helpful diagnostic tools, they tend to overestimate the severity of DR, which is generally milder;

In conclusion, we show that thoroughly assessing DR in NLP models requires evaluating multiple domain shifts and incorporating both drop metrics (SD and TD). We manifest that while nuanced, the DR challenge is still prevalent. In §7, we delve into the implications of our findings for the NLP community. In Appendix §A, we present a theorem that elucidates some of our findings regarding the relationship between the DR metrics. We hope this work will provide a fresh perspective on model robustness and facilitate further research.

2 Related Work

The term DR generally refers to the extent to which the performance of a model does not degrade when applied to newly collected samples from other domains. In some cases, robustness refers to consistency (low variance) (Yu et al., 2022). Literature on robustness in NLP can be categorized by the type of distribution shift examined: Synthetic and Natural (Wang et al., 2022b; Hupkes et al., 2023).

Synthetic shift works include adversarial attacks (Jin et al., 2020), input perturbations (Belinkov and Bisk, 2018), counterfactual (Kaushik et al., 2020), diagnostic (Wang et al., 2019) and challenge (or contrast) sets (McCoy et al., 2019). These works assess robustness using datasets designed to challenge NLP models rather than represent a natural language distribution. While the synthetic shifts are helpful diagnostic tools (Goel et al., 2021), they do not accurately depict the actual state of DR "in the wild". Hence, we focus on natural domain shifts.

Natural shift study focuses on organic scenarios where a discrepancy exists between the training and deployment data. These studies encompass various setups, including medium shift (Miller et al., 2020), temporal shift (Cvejoski et al., 2022), and domain shift (e.g., to medical (Miller et al., 2021) and legal (Chalkidis et al., 2020) domains).

Researchers proposed various benchmarks to evaluate the robustness of NLP models and the quality of solutions, including domain shifts in a single NLP task (Budzianowski et al., 2018; Reid et al., 2022; Miller et al., 2020; Yu et al., 2021; Zhong et al., 2021; Chronopoulou et al., 2022; Gekhman et al., 2023b; Yu et al., 2023), with challenge sets (Rychalska et al., 2019; Mosbach et al., 2023; Weber et al., 2023; Yuan et al., 2023) or only with fine-tuned models (Hendrycks et al., 2020; Tu et al., 2020; Koh et al., 2021). Our study addresses a broad range of domain shifts in many more NLP tasks than previous work, including sequence and token-level classification, QA, and generation. In addition, we examine both small fine-tuned models and few-shot LLMs. Importantly, unlike other works, which focused on the source drop, we also consider the target drop, providing a more holistic perspective on DR. To the best of our knowledge, this is the most comprehensive DR study in NLP.

3 Domain Robustness

Domain is a widely used term in NLP that typically refers to a cohesive corpus or dataset, which may be characterized by factors such as topic, style, genre, syntax, linguistic register, and medium. Although 'domain' lacks a clear and consistent definition (Ramponi and Plank, 2020), we formally describe a *domain* $\mathcal{D}$ by a joint distribution $P_{\mathcal{D}}(X, Y)$ over $\mathcal{X}$ (the input space) and $\mathcal{Y}$ (the outcome space). In a *domain shift*, the source domain $\mathcal{S}$, and the target domain $\mathcal{T}$ differ in their underlying joint distribution $P_{\mathcal{S}}(X, Y) \neq P_{\mathcal{T}}(X, Y)$.

Given a training set of examples from the source domain $S \sim \mathcal{S}$, the goal of the NLP model is to learn $P_{\mathcal{S}}(X, Y)$ (or $P_{\mathcal{S}}(Y|X)$), and to the generalize to the (potentially unknown) target domain distribution(s) in which it will be deployed, $P_{\mathcal{T}}(X, Y)$. To evaluate the performance on the target domain, we use a test set $T \sim \mathcal{T}$, which is *unobserved during training*. We use the term *Domain Robustness (DR)* to describe **the inherent (in)ability of an NLP model to generalize from the source domain to the target domains**.

For fine-tuned models, the DR challenge arises when the test data comes from a domain that is different from the labeled training data. Meanwhile, few-shot models face the DR challenge when the domain of the demonstrations used in the prompt differs from that of the target data.

3.1 Measuring Domain Robustness

This subsection proposes concepts and metrics for characterizing the DR challenge, summarized in Table 1. Given a source domain $\mathcal{S}$ and a target domain $\mathcal{T}$, we use ST to denote the *Cross-domain Performance*, which is the score (e.g., F1) achieved

SS	Source In-domain Performance
TT	Target In-domain Performance
ST	Cross-domain Performance
SD	Source Drop (Observed Drop): SS − ST
TD	Target Drop (Unobserved Drop): TT − ST
IDD	In-domain difference: SS − TT
$\overline{\text{SS}}$	Average In-domain: $\mathbb{E}[\text{SS}] = \mathbb{E}[\text{TT}]$
$\overline{\text{ST}}$	Average Cross-domain: $\mathbb{E}[\text{ST}]$
$\overline{\Delta}$	Average Drop: $\overline{\text{SS}} - \overline{\text{ST}} = \mathbb{E}[\text{SD}] = \mathbb{E}[\text{TD}]$
W_{SD}	Worst SD: $\max_{(S,T)}$ SD
W_{TD}	Worst TD: $\max_{(S,T)}$ TD

Table 1: The notations of Domain Robustness concepts and metrics we use in this study. Toy example in Table 7.

when training a model on data $S \in \mathcal{S}$ and testing it on $T \in \mathcal{T}$. When training and testing the model with data from the source domain, we use SS to denote the *Source In-domain Performance*. Likewise, TT is the *Target In-domain Performance*.

Finally, we define the *in-domain difference* to be IDD = SS − TT. A positive IDD may indicate a shift towards an inherently more challenging target domain, for example, the shifts $A \rightarrow C$ and $A \rightarrow B$ from Figure 1. The cornerstone of this paper is that *a truthful DR characterization requires considering* SS*,* TT*, and* ST. Specifically, full characterization requires understanding the joint distribution of SS, TT, and ST (see Appendix §A).

Nevertheless, identifying these random variables and their relationships is not tractable without further assumptions, and therefore, we introduce practical and interpretable metrics that quantify the degradation in performance when shifting domains. We denote the *Average In-domain Performance* by $\overline{\text{SS}} = \mathbb{E}[\text{SS}]$, and the *Average Cross-domain Performance* by $\overline{\text{ST}} = \mathbb{E}[\text{ST}]$. The difference between these metrics is the *Average Drop*, denoted by $\overline{\Delta} = \overline{\text{SS}} - \overline{\text{ST}}$. Intuitively, *the larger the* $\overline{\Delta}$ *is, the more severe the DR challenge of the model is.*

3.2 The Source and Target Drops

Although characterizing the DR challenge ideally requires task-level analysis across various domain shifts, this approach can be impractical or less relevant when focusing on a specific shift. Hence, we introduce shift-level degradation metrics. The *Source Drop (*SD*)* and the *Target Drop (*TD*)* are the drops in performance caused by a domain shift, alternately using the source and target's in-domain performance as a point of reference:

$$\text{SD} = \text{SS} - \text{ST}$$
$$\text{TD} = \text{TT} - \text{ST}$$

Notice that the training data from the target domain may not be available in a real-life scenario, and in this case, the TT can not be computed. The performance degradation we observe in practice is the SD. The TD is a more theoretical measure: "*What would the drop be compared to if the model were trained on data from the target domain?*"

From the above definitions, it follows that: SD = TD + IDD. This is a solid justification for using both SD and TD when quantifying the DR challenge. *Using only one could potentially paint an image obscured by the* IDD*, which is not a by-product of the domain shift itself.* For instance, in studies involving challenge sets that report a large SD, the drop may be primarily influenced by a large IDD rather than both SD and TD being large (e.g., the shift $A \rightarrow C$ in Figure 1). In §6.4, we found that this is the case in many domain shifts. We refer the readers to **Appendix §A for an extended discussion and theorem** on the relationships between the DR metrics.

Finally, other task-level metrics we use are the *Worst* SD *(*W_{SD}*) and Worst* TD *(*W_{TD}*)*, which measure the highest SD and TD observed across all domain shifts and identify challenging shifts.

3.3 Domain Shift Scenarios

We next introduce a novel framework for classifying domain shifts into four possible scenarios. These scenarios are defined by the sign (positive or negative) of the source and target drops, which can help us understand the nature of the DR challenge. In Appendix §A.2, we further discuss these scenarios and motivate when each might occur.

The Classic Scenario ($A \rightarrow B$ in Figure 1) In this scenario both SD and TD are positive. Accordingly, we deduce that the model is not effectively generalizing from the source domain to the target.

The Observed Scenario ($A \rightarrow C$ in Figure 1) This scenario occurs when the shift is to a harder domain and TT < ST < SS. In this scenario, only the observed drop, SD is positive. Although we observe a performance drop, it might be explained by moving to a harder domain and not due to a genuine DR challenge since the model achieves generalization to the target domain and even exhibits higher performance than TT.

The Unobserved Scenario ($C \rightarrow A$ in Figure 1) This scenario occurs when the shift is from a harder domain to an easier one: SS < ST < TT. In this

Task		#D	Train	Dev	Test
SA	Sentiment Analysis	6	10K	2.5K	2.5K
NLI	Natural Language Inference	5	50K	2.5K	2K
AB	Aspect Based SA (ABSA)	5	2K	500	1.4K
QA	Question Answering	6	9K	1K	2.5K
QG	Question Generation	6	7.5K	900	1K
AS	Abstractive Summarization	5	10K	1K	500
TG	Title Generation	6	17.5K	1K	1K

Table 2: Details about the tasks in The Domain Robustness Benchmark. "#D" is the number of domains. "Train", "Dev", "Test" columns present the size of the splits of each domain. Note that we present the average size for the test split since it differs between domains. More details can be found in the project repository.

scenario, only SD is negative, and we do not observe a performance drop. However, since TD is positive, we know the model can potentially generalize better and it might suffer from a DR challenge.

The No Challenge Scenario ($C \rightarrow B$ in Figure 1) Occurs when ST is larger than both SS and TT, therefore, SD and TD are negative.

4 The Domain Robustness Benchmark

In Sections 1 and 2, we identified shortcomings in existing DR benchmarks. These include an overemphasis on challenge sets and synthetic datasets, coupled with neglecting key NLP tasks such as token-level classification, QA, and particularly generation tasks. To our knowledge, this is the first DR study focusing on various generation tasks, which have gained prominence with the widespread use of LLMs and GenAI. Moreover, most benchmarks consider only a single or very few domains and often use target domains with only test splits, preventing measuring target drops. These limitations restrict a complete understanding of the state of the DR challenge in "natural settings".

To bridge these gaps, we curated a novel DR benchmark that focuses on natural shifts and covers seven downstream tasks. Each task consists of several domains with the same amount of labeled data, enabling using any domain as a source or a target and computing the metrics from §3. Table 2 details the number of examples in each task domain. In Appendix §D, we describe the preprocessing we performed and discuss technical assumptions.

Sentiment Analysis (SA) Following Ziser and Reichart (2018) and Calderon et al. (2022), we combine five domains of the Amazon product review dataset (Blitzer et al., 2007) with the airline review dataset (Nguyen, 2015) into a single dataset with six domains: *Appliances, Beauty, Books, Games, Software, and Airline*.

Natural Language Inference (NLI) We use five domains from MNLI dataset (Williams et al., 2018): *Fiction, Government, Slate, Telephone, and Travel*.

Aspect Based Sentiment Analysis (AB) Following Lekhtman et al. (2021), we combine the SemEval 2014, 2015, and 2016 (Pontiki et al., 2014, 2015, 2016) ABSA datasets, together with the MAMs dataset (Jiang et al., 2019) into a single dataset with four domains: *Device, Laptops, Restaurants, Service, and MAMs*.

Question Answering (QA) We rely on the SQuAD v2 dataset (Rajpurkar et al., 2016, 2018), one of the most common QA datasets. We asked human annotators to categorize the documents according to the Wikipedia's taxonomy,[3] and created six domains: *Geography, History, Philosophy, Science, Society, and Technology*.

Question Generation (QG) We rely on our domain partition of the SQuAD dataset (Rajpurkar et al., 2016) and only use examples with an answer. Given a Wikipedia document and an answer to the question, the task of the NLP model is to generate the question (Calderon et al., 2023).

Abstractive Summarization (AS) We rely on the Webis-TLDR-17 dataset (Völske et al., 2017), which consists of Reddit posts and their "TL;DR" summary. We asked human annotators to categorize subreddits into five domains: *Drugs, Fitness, LoL (video game), Politics, and Relationships*.

Title Generation (TG) We focus on generating titles for Amazon product reviews (Yang et al., 2023a). Our dataset contains six domains: *Beauty, Books, DVD, Kitchen, Sports, and Wireless*.

5 Experimental Setup

Table 3 presents details about the participating models. Additional implementation details, including hyperparameters and prompts are in Appendix §E.

Fine-tuning Models For classification tasks (SA, NLI, AB, QA) we employ encoder-only models. Specifically, we use RoBERTa (Liu et al., 2019) and DeBERTa-v3 (He et al., 2021a), as well as the smaller DistilBERT (Sanh et al., 2019). For conditional generation tasks (QG, AS, TG), we

[3] We merged the vital articles categories: `https://en.wikipedia.org/wiki/Wikipedia:Vital_articles`, into eight categories and used six of them as domains.

Arch.	Name	#P	#L	Name	#P	#L
fine-tuned EO	DistilBert	66m	6	DeBERTa-XS	70m	12
				DeBERTa-S	142m	6
	RoBERTa-B	125m	12	DeBERTa-B	184m	12
	RoBERTa-L	355m	24	DeBERTa-L	435m	24
fine-tuned ED	T5-S	60m	12	BART-B	139m	12
	T5-B	220m	24			
	T5-L	737m	48	BART-L	406m	24
few-shot DO	Orca-7b	7b	32	Orca-13b	13b	40
	Mistral 7b	7b	32	NeuralChat	7b	32
	Llama2-7	7b	32	Llama2-13b	13b	40
	Llama2-70b	70b	40			
	GPT3.5	?	?	GPT4	?	?

Table 3: Details about the participating models in this study. 'Arch.' states the architecture type: EO for Encoder-only, ED for Encoder-decoder, and DO for Decoder-only. '#P' is the number of parameters in millions (m) or billions (b), and '#L' is the number of layers.

Task	Model	$\overline{SS}$	$\overline{ST}$	$\overline{\Delta}$	W_{SD}	W_{TD}
SA	RoBERTa-L	95.76	92.79	2.97	13.92	19.82
	DeBERTa-L	**96.21**	**94.10**	**2.11**	**9.60**	**10.25**
NLI	RoBERTa-L	89.29	87.81	**1.48**	**4.83**	**2.89**
	DeBERTa-L	**90.43**	**88.92**	1.51	5.47	3.10
AB	RoBERTa-L	**73.31**	49.42	23.90	**35.28**	**32.41**
	DeBERTa-L	71.98	**50.19**	**21.80**	35.54	34.49
QA	RoBERTa-L	**82.01**	**81.72**	**0.29**	**6.01**	**2.53**
	DeBERTa-L	74.54	74.10	0.44	6.29	2.72
QG	T5-L	**77.36**	**77.24**	0.13	**4.26**	1.16
	BART-L	76.30	76.30	**0.00**	4.43	**0.80**
AS	T5-L	**62.40**	61.42	0.98	**4.62**	2.55
	BART-L	62.33	**61.62**	**0.71**	4.96	**1.93**
TG	T5-L	**66.48**	**65.22**	1.26	6.78	5.06
	BART-L	65.87	64.72	**1.15**	**6.61**	**4.58**

Table 4: Comparison between different large fine-tuned models. The columns are: $\overline{SS}$ - Average In-domain, $\overline{ST}$ - Average Cross-domain, $\overline{\Delta}$ - Average Drop, W_{SD} - Worst Source Drop and W_{TD} - Worst Target Drop.

utilize two common encoder-decoder models: T5 (Raffel et al., 2020) and BART (Lewis et al., 2020). We chose these open-source models because they offer a variety of sizes (see Table 3).

We conduct hyperparameter tuning for each model and source domain, selecting optimal parameters based on the source domain's validation set, and then evaluate the model across all target domains. See Appendix §E for more details.

Zero-shot and Few-shot LLMs We examine LLMs with an API, including GPT3.5 (`turbo`) and GPT4 (OpenAI, 2023), as well as the open-sourced LLMs LLama v2 (Touvron et al., 2023), Orca v2 (Mitra et al., 2023) (which is based on LLama v2 and fine-tuned using signals from GPT4), Mistral-7b (Jiang et al., 2023) and NeuralChat (Lv et al., 2023) (which is based on Mistral and fine-tuned using the Orca dataset (Mukherjee et al., 2023)).

For each test example from a target domain, the LLM receives an input comprising a task instruction and the example. In few-shot setups, the input is augmented with additional demonstrations from the source domain. Task instructions and prompt examples are provided in Appendix §E.1.

Due to the high costs of API calls and the quadratic increase in the number of experiments with the number of domains, we limit our presentation of few-shot results to three domains and 600 examples for each task (see Appendix §D.3).

Metrics For classification tasks (SA, NLI, AB, QA) we report the F1 score. For generation tasks (QG, AS, TG) we report the BertScore (Zhang et al., 2020) with a pre-trained SBERT model (Reimers and Gurevych, 2019). Please see our note in §8.L1.

6 Results

6.1 Fine-tuned Models

In Table 4, we present the results of large fine-tuned models. As can be seen, for every task the average in-domain performance consistently exceeds the average cross-domain performance. An exception to this is the QA and QG tasks, which share the same partition of domains, explaining why they behave similarly. Moreover, the vast majority of tasks exhibit non-negligible drops in performance upon domain shift. *This leads to the conclusion that the DR problem still exists in fine-tuned models*, though in varying severity, depending on the task. Some tasks (e.g., AB) exhibit significant drops in most domain shifts, while other tasks (e.g., QA) exhibit minor drops, but we can still expect to have challenging shifts for every domain.

In Appendix §C we provide additional results for fine-tuned models. Specifically, in §C.1 we explore the effect of the model size. We observe that larger models improve absolute in-domain and cross-domain performance and exhibit an apparent reduction in performance drops, especially in classification tasks. In §C.4, we examine the impact of the source dataset size. We find enhancements in both in-domain and cross-domain performance, however, the performance drop is only reduced in classification tasks and worsens in generation tasks.

6.2 Few-shot Models

Unlike fine-tuning, a domain shift occurs for few-shot models when the domain of the prompt demonstrations differs from the test example's domain. Table 5 presents the results of 4-shots LLMs.

Model	SA $\overline{SS}$	SA $\overline{ST}$	SA W_{SD}	SA W_{TD}	NLI $\overline{SS}$	NLI $\overline{ST}$	NLI W_{SD}	NLI W_{TD}	AB $\overline{SS}$	AB $\overline{ST}$	AB W_{SD}	AB W_{TD}
Orca-7b	80.9	79.2	8.7	6.3	70.7	70.4	16.8	**2.5**	44.0	41.9	24.5	8.2
Orca-13b	92.6	92.3	10.0	2.7	75.7	74.4	15.5	6.3	52.6	49.2	38.3	11.9
Mistral	83.8	80.9	11.0	5.7	49.0	45.8	17.1	10.3	49.9	43.5	34.5	19.4
Neural	92.4	92.4	12.0	1.3	79.8	77.0	16.0	8.8	42.6	39.3	**20.3**	10.1
Llama-70b	94.1	93.9	**8.3**	1.3	56.6	56.3	**5.3**	4.5	51.4	48.6	35.9	8.9
GPT3.5	92.1	92.9	10.0	**0.0**	72.9	71.7	16.4	7.2	52.7	**51.9**	37.0	**2.4**
GPT4	95.2	**94.7**	11.0	2.0	87.0	86.0	6.4	3.9	51.0	47.9	28.9	6.9
Best FT	**95.5**	91.7	9.6	10.2	**91.0**	**89.0**	5.5	3.1	**74.4**	47.2	35.3	32.4

Model	QA $\overline{SS}$	QA $\overline{ST}$	QA W_{SD}	QA W_{TD}	QG $\overline{SS}$	QG $\overline{ST}$	QG W_{SD}	QG W_{TD}	AS $\overline{SS}$	AS $\overline{ST}$	AS W_{SD}	AS W_{TD}	TG $\overline{SS}$	TG $\overline{ST}$	TG W_{SD}	TG W_{TD}
Orca-7b	25.8	23.6	5.0	3.8	65.9	65.5	3.0	1.5	53.4	53.2	2.3	1.2	52.6	52.7	**0.6**	1.0
Orca-13b	62.8	62.4	4.5	2.5	73.3	73.3	**2.2**	0.8	61.0	60.4	2.0	3.2	58.9	58.8	1.8	0.8
Mistral	48.7	46.8	6.7	7.2	69.8	69.5	5.2	1.2	59.0	58.5	2.4	4.7	57.4	57.4	1.2	1.0
Neural	50.8	49.5	5.5	4.6	72.0	72.1	3.9	0.8	61.6	61.4	**1.6**	1.6	58.4	58.3	1.6	**0.4**
Llama-70b	36.6	36.0	4.4	3.5	73.3	73.1	4.6	0.6	60.5	59.2	2.7	3.5	57.7	57.7	2.3	0.7
GPT3.5	60.1	59.7	6.4	2.3	74.6	74.5	4.4	**0.3**	**64.7**	**64.4**	3.0	0.9	58.5	58.4	1.3	0.8
GPT4	71.0	71.1	6.0	**0.8**	76.0	75.8	3.5	0.5	64.1	64.0	2.5	**0.6**	58.0	57.9	1.5	**0.4**
Best FT	**83.7**	**83.5**	**3.1**	2.0	**77.7**	**77.5**	4.3	0.4	63.2	62.0	3.5	1.7	**65.1**	**63.2**	6.8	5.1

Table 5: Comparison between fine-tuned and (4) few-shot models. The 'Best FT' selects the best performing fine-tuned model according to the source development set: DebERTa-L for SA and NLI, RoBERTa-L for AB and QA, and T5-L for QG, AS, TG. All the results are for the same examples and three domains (see Appendix §D.3).

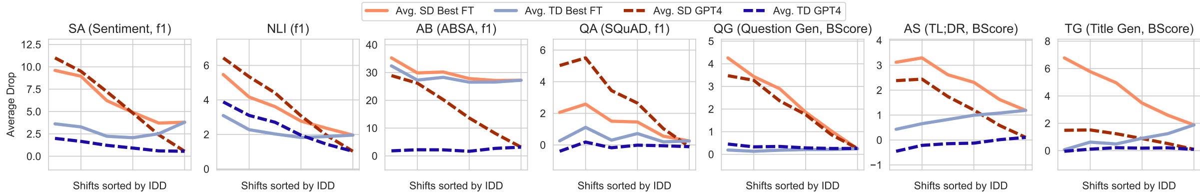

Figure 2: Average SD (orange lines) and Average TD (blue lines) as a function of challenging domain shifts. Specifically, we sort the domain shifts by their In-domain Difference (IDD) and as we move to the right on the x-axis, we incrementally include an additional domain shift in the average drop calculation. Consequently, the leftmost point represents the shift with the largest IDD, while the rightmost point encompasses all shifts. The best fine-tuned model (see caption of Table 5, solid lines) against GPT4 (dashed lines). This figure illustrates three key findings: (1) The SD is larger than the TD, and when including all shifts their averages are equal; (2) Generally, fine-tuned models exhibit larger drops; (3) Examining only challenging shifts and focusing solely on the SD, obscure the true DR state. Incorporating the TD can compensate for this and provide a clearer understanding.

Similar to fine-tuned models, in most tasks and few-shot models, in-domain performance surpasses cross-domain performance, *indicating that the domain of the demonstration has an effect.* However, the average drops in few-shot models, particularly in GPT3.5 and GPT4, are lower than in fine-tuned models (see also Figure 2). This probably stems from weaker anchoring to the source domain since in few-shot setups, the parameters are not updated based on source domain optimization. Yet, few-shot models experience large worst drops, although, except for NLI and QA, they are much lower than the worst drops of fine-tuned models.

Nevertheless, the robustness of few-shot models comes at a cost of absolute performance. As shown in Table 5, fine-tuned models outperform all non-GPT models in both in-domain and cross-domain settings. For GPT models, aside from the AS task, the fine-tuned models achieve higher in-domain performance. However, in certain tasks (SA, AB, AS), GPT models exceed the cross-domain performance of fine-tuned models. *This discrepancy highlights the importance of Domain Adaptation research of fine-tuned NLP models.*

In Appendix §C.2 we study the effect of the number of demonstrations, finding that a larger number of demonstrations usually improves in-domain and cross-domain performance, though in some cases

Figure 3: The proportion of each domain shift scenario (see §3.3) for fine-tuned (top chart) and few-shot models (bottom). For each task, the proportion is measured over all the models and domain shifts. More details in §C.8.

mildly increasing the drop between them (by causing a stronger "source domain anchoring").

In Appendix §C.3, we also analyze the impact of few-shot model size. Same as for fine-tuned models, increasing model size generally improves absolute performance and tends to reduce drops.

6.3 Characterizing the DR Challenge

To understand the nature of the domain shifts, we present the proportion of the four scenarios (from §3.3) in Figure 3. In Appendix C.8, we provide details on this analysis and confirm its statistical

	Task	σ_{SD}	σ_{TD}	W_{SD}	W_{TD}	ρ_{SS}	ρ_{TT}	R^2_{SD}	R^2_{TD}
Fine-tuning	SA	3.62	3.33	13.23	17.05	0.28	0.42	0.34	0.08
	NLI	3.06	1.29	7.14	5.12	-0.28	0.82	0.83	0.06
	AB	7.09	6.53	36.05	36.55	-0.15	0.10	0.27	0.12
	QA	3.71	2.07	6.76	4.52	-0.06	0.68	0.75	0.14
	QG	2.29	0.46	4.55	1.21	-0.28	0.95	0.96	0.02
	AS	1.91	0.65	4.81	2.49	0.06	0.77	0.95	0.58
	TG	2.70	1.47	6.94	4.88	0.31	0.70	0.92	0.60
Few-shot	SA	7.49	1.77	10.58	3.73	-0.26	0.82	0.94	0.36
	NLI	8.63	3.93	15.54	8.27	-0.47	0.85	0.80	0.39
	AB	22.68	4.64	32.70	10.57	-0.13	0.86	0.99	0.55
	QA	4.20	1.92	5.78	3.09	-0.25	0.53	0.71	0.28
	QG	3.17	0.50	4.11	0.75	-0.36	0.88	0.95	0.25
	AS	1.74	1.13	2.55	2.01	0.01	0.31	0.75	0.55
	TG	1.21	0.55	1.62	0.91	-0.24	0.79	0.82	0.34

Table 6: Statistics of the SD and the TD. We first calculate the statistic for each model and then present the mean statistic for the task. This includes: (1) The standard deviation of the SD (σ_{SD}) and the TD (σ_{TD}); (2) The Worst SD (W_{SD}) and TD (W_{TD}); (3) Spearman's correlation between the ST and SS (ρ_{SS}) or TT (ρ_{TT}); (4) The R-squared of IDD and SD (R^2_{SD}) or TD (R^2_{TD}).

significance. Notably, for fine-tuned models, the Classic scenario, marked by positive SD and TD, emerges as the most dominant and occurs in most tasks with a frequency exceeding 50%, which indicates *the prevalent DR challenge*. On the other hand, all four scenarios are common across few-shot tasks, suggesting that *the effect of domain shift on few-shot models is weaker and more nuanced*. This is also true in fine-tuned QA and QG tasks, which share the same domain partitions.

Although there is a positive TD in most cases, many are Unobserved scenarios. This finding is essential since many past works overlooked the TD. Our study implies that a DR challenge can exist even when the shift is to an easier domain (SS $<$ TT) and even if practitioners do not observe a performance degradation. In comparison, the Observed scenario (positive SD but negative TD), is less frequent but still appears in half of the fine-tuning and few-shot tasks. *This also underscores the necessity for both metrics* and calls for a deeper analysis: which metric more accurately estimates the average drop and cross-domain performance?

6.4 Comparing SD and TD

In Table 6, we see that for every task and for both fine-tuning and few-shot, the variance of the SD is larger than the variance of the TD. In addition, for almost all tasks (except for fine-tuning SA and AB) the Worst SD is higher than the worst TD. These findings indicate that *the TD is a more robust estimator of the average drop*.

Moreover, we find that *the ST behaves more like the TT rather than the SS*, as can be seen by Table 6, where the correlation between ST and TT is much stronger than the correlation with SS (typically above 0.7). This suggests that attempting to estimate the cross-domain performance without incorporating knowledge of the TT is challenging.

Additionally, Table 6 shows the R^2 between the in-domain difference (IDD $=$ SS $-$ TT) and the drops. These values indicate the extent to which drop variations can be predicted by the IDD. The high R^2 of the SD, compared to the TD, suggests that *observing a large SD is likely attributed to shifting to a harder domain and not by genuine DR issues*. This raises a red flag for the NLP community since many works measure DR by source performance degradation on challenge sets.

The issue becomes clear in Figure 2, which shows the average SD and TD calculated over challenging shifts. The figure reveals that when focusing on challenging shifts (as shown on the left x-axis), the SD appears extremely large. Consequently, focusing on challenge sets and relying on the SD tend to portray a severe picture of the DR state. Examining the TD and additional domain shifts provides a more accurate depiction.

In Appendix §A, we provide a detailed discussion of the analysis from this subsection and present a theorem that unifies our findings, demonstrating their equivalence. In Appendix §A.1 we explore the connection between the domain divergence and drop metrics. Our study underscores using both metrics, however, when only one is available, the TD is the preferable choice.

7 Discussion

In this work, we study the DR challenge in modern NLP models. To this end, we constructed a new DR benchmark comprising various NLP tasks and domain shifts. We proposed shift-level and task-level metrics for precise evaluation and benchmarked numerous fine-tuned models and few-shot LLMs while examining the effect of multiple factors.

Our extended discussion in Appendix §B (to be included with an extra page) delves into the key implications of our findings. Specifically, our comprehensive study highlights the need for a nuanced approach to assessing robustness, and that current research can paint a skewed picture. Finally, our work underscores the ongoing relevance of Domain Adaptation research in NLP and the importance of developing robust, adaptable models capable of handling the diverse nature of real-world data.

8 Limitations

L1. Prompt Engineering Noteworthy, we experimented with various prompts and task instruction revisions but saw no significant change. Following Gao et al. (2021), we also tried selecting demonstrations from the source domain most similar to the target test example using a pre-trained SBERT model (Reimers and Gurevych, 2019). However, this approach did not enhance performance and introduced biases, such as demonstrations from only one class, leading the LLM to classify the test example with this class.

L2. Larger Models Although we examined a broad range of models of various sizes, we did not fine-tune models with more than one billion parameters. This decision stems from two reasons. The first is our belief that fine-tuned models should be relatively fast and compact. Otherwise, few-shot LLMs like those we examined in the study can be used. Second, the volume of experiments (including hyperparameter tuning) imposed practical limitations, and examining larger fine-tuned models was not feasible due to their computational resource requirements. Nonetheless, we believe that the trends observed in the smaller fine-tuned models will likely persist, and we leave the examination of larger models to future research.

L3. Domain Adaptation Solutions Although a wide array of DA solutions exists to address the DR challenge and improve the OOD generalization of NLP models, our study specifically focuses on the diagnostic aspect. We aim to explore whether this challenge is prevalent in modern NLP models, and our findings confirm it is prevalent. We anticipate that future research could leverage our new DR benchmark for diagnostic purposes as well as for benchmarking DA solutions. Furthermore, we hope our study will facilitate further research in this vital area and inspire novel DA methods.

L4. Text Generation Evaluation Text generation evaluation is an open research problem, and many techniques exist. Although we report BERTScore for simplicity, we did conduct a comprehensive analysis using various metrics (BLEU, ROUGE, METEOR, BLEURT, etc...) and observed similar trends to our findings. We chose BERTScore because it captures semantic similarity and context. In addition, upon manual inspection of LLM outputs, we found them comparable or even superior to the reference texts used for benchmarking. Yet, automatic evaluation with references is useful for assessing the extent to which models learn and capture the dataset distribution $P(Y|X)$. This perspective shifts the focus from human preference to a more technical objective. Supporting this viewpoint is the fact that increasing the number of demonstrations also enhances the performance.

References

Yonatan Belinkov and Yonatan Bisk. 2018. Synthetic and natural noise both break neural machine translation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Eyal Ben-David, Nadav Oved, and Roi Reichart. 2022a. PADA: example-based prompt learning for on-the-fly adaptation to unseen domains. *Trans. Assoc. Comput. Linguistics*, 10:414–433.

Eyal Ben-David, Carmel Rabinovitz, and Roi Reichart. 2020. PERL: pivot-based domain adaptation for pre-trained deep contextualized embedding models. *Trans. Assoc. Comput. Linguistics*, 8:504–521.

Eyal Ben-David, Yftah Ziser, and Roi Reichart. 2022b. Domain adaptation from scratch. *CoRR*, abs/2209.00830.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Mach. Learn.*, 79(1-2):151–175.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic*. The Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, and et al. Jared Kaplan. 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*.

Pawel Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 2018. Multiwoz - A large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 5016–5026. Association for Computational Linguistics.

Nitay Calderon, Eyal Ben-David, Amir Feder, and Roi Reichart. 2022. Docogen: Domain counterfactual generation for low resource domain adaptation. 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 7727–7746. Association for Computational Linguistics.

Nitay Calderon, Subhabrata Mukherjee, Roi Reichart, and Amir Kantor. 2023. A systematic study of knowledge distillation for natural language generation with pseudo-target training. *CoRR*, abs/2305.02031.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: the muppets straight out of law school. *CoRR*, abs/2010.02559.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, and et. al. . 2022. Palm: Scaling language modeling with pathways. *CoRR*, abs/2204.02311:30.

Alexandra Chronopoulou, Matthew E. Peters, and Jesse Dodge. 2022. Efficient hierarchical domain adaptation for pretrained language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 1336–1351. Association for Computational Linguistics.

Kostadin Cvejoski, Ramsés J. Sánchez, and César Ojeda. 2022. The future is different: Large pre-trained language models fail in prediction tasks. *CoRR*, abs/2211.00384.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Hady ElSahar and Matthias Gallé. 2019. To annotate or not? predicting performance drop under domain shift. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2163–2173. Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 3816–3830. Association for Computational Linguistics.

Chunjiang Ge, Rui Huang, Mixue Xie, Zihang Lai, Shiji Song, Shuang Li, and Gao Huang. 2023. Domain adaptation via prompt learning. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–11.

Zorik Gekhman, Jonathan Herzig, Roee Aharoni, Chen Elkind, and Idan Szpektor. 2023a. Trueteacher: Learning factual consistency evaluation with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 2053–2070. Association for Computational Linguistics.

Zorik Gekhman, Nadav Oved, Orgad Keller, Idan Szpektor, and Roi Reichart. 2023b. On the robustness of dialogue history representation in conversational question answering: A comprehensive study and a new prompt-based method. *Transactions of the Association for Computational Linguistics*, 11:351–366.

Karan Goel, Nazneen Fatema Rajani, Jesse Vig, Zachary Taschdjian, Mohit Bansal, and Christopher Ré. 2021. Robustness gym: Unifying the NLP evaluation landscape. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 42–55. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasovic, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8342–8360. Association for Computational Linguistics.

Bernal Jimenez Gutierrez, Nikolas McNeal, Clayton Washington, You Chen, Lang Li, Huan Sun, and Yu Su. 2022. Thinking about GPT-3 in-context learning for biomedical ie? think again. In *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 4497–4512. Association for Computational Linguistics.

Xiaochuang Han and Jacob Eisenstein. 2019. Unsupervised domain adaptation of contextualized embeddings for sequence labeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4237–4247. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021a. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing. *CoRR*, abs/2111.09543.

Ruidan He, Linlin Liu, Hai Ye, Qingyu Tan, Bosheng Ding, Liying Cheng, Jia-Wei Low, Lidong Bing, and Luo Si. 2021b. On the effectiveness of adapter-based tuning for pretrained language model adaptation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th*

International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2208–2222. Association for Computational Linguistics.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2744–2751. Association for Computational Linguistics.

Dieuwke Hupkes, Mario Giulianelli, Verna Dankers, Mikel Artetxe, Yanai Elazar, Tiago Pimentel, Christos E. Christodoulopoulos, Karim Lasri, Naomi Saphra, Arabella Sinclair, Dennis Ulmer, Florian Schottmann, Khuyagbaatar Batsuren, Kaiser Sun, Koustuv Sinha, Leila Khalatbari, Maria Ryskina, Rita Frieske, Ryan Cotterell, and Zhijing Jin. 2023. A taxonomy and review of generalization research in NLP. *Nat. Mac. Intell.*, 5(10):1161–1174.

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

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 6279–6284. 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.

Abhinav Ramesh Kashyap, Devamanyu Hazarika, Min-Yen Kan, and Roger Zimmermann. 2021. Domain divergences: A survey and empirical analysis. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 1830–1849. Association for Computational Linguistics.

Divyansh Kaushik, Eduard H. Hovy, and Zachary Chase Lipton. 2020. Learning the difference that makes A difference with counterfactually-augmented data. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton Earnshaw, Imran S. Haque, Sara M. Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. 2021. WILDS: A benchmark of in-the-wild distribution shifts. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 5637–5664. PMLR.

Hao Lang, Yinhe Zheng, Yixuan Li, Jian Sun, Fei Huang, and Yongbin Li. 2023. A survey on out-of-distribution detection in NLP. *CoRR*, abs/2305.03236.

Entony Lekhtman, Yftah Ziser, and Roi Reichart. 2021. DILBERT: customized pre-training for domain adaptation with category shift, with an application to aspect extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 219–230. Association for Computational Linguistics.

Yoav Levine, Itay Dalmedigos, Ori Ram, Yoel Zeldes, Daniel Jannai, Dor Muhlgay, Yoni Osin, Opher Lieber, Barak Lenz, Shai Shalev-Shwartz, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2022. Standing on the shoulders of giant frozen language models. *CoRR*, abs/2204.10019.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Jian Liang, Ran He, and Tieniu Tan. 2023. A comprehensive survey on test-time adaptation under distribution shifts. *CoRR*, abs/2303.15361.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Kaokao Lv, Wenxin Zhang, Haihao Shen, and Intel Corporation. 2023. Supervised fine-tuning and direct preference optimization on intel gaudi2.

Inbal Magar and Roy Schwartz. 2022. Data contamination: From memorization to exploitation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 157–165. Association for Computational Linguistics.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3428–3448. Association for Computational Linguistics.

John Miller, Karl Krauth, Benjamin Recht, and Ludwig Schmidt. 2020. The effect of natural distribution shift on question answering models. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 6905–6916. PMLR.

Timothy Miller, Egoitz Laparra, and Steven Bethard. 2021. Domain adaptation in practice: Lessons from a real-world information extraction pipeline. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 105–110, Kyiv, Ukraine. Association for Computational Linguistics.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 11048–11064. Association for Computational Linguistics.

Arindam Mitra, Luciano Del Corro, Shweti Mahajan, Andrés Codas, Clarisse Simões, Sahaj Agrawal, Xuxi Chen, Anastasia Razdaibiedina, Erik Jones, Kriti Aggarwal, Hamid Palangi, Guoqing Zheng, Corby Rosset, Hamed Khanpour, and Ahmed Awadallah. 2023. Orca 2: Teaching small language models how to reason. *CoRR*, abs/2311.11045.

Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. 2023. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 12284–12314. Association for Computational Linguistics.

Subhabrata Mukherjee, Arindam Mitra, Ganesh Jawahar, Sahaj Agarwal, Hamid Palangi, and Ahmed Awadallah. 2023. Orca: Progressive learning from complex explanation traces of GPT-4. *CoRR*, abs/2306.02707.

Quang Nguyen. 2015. The airline review dataset.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Aitor Ormazabal, Mikel Artetxe, and Eneko Agirre. 2023. Comblm: Adapting black-box language models through small fine-tuned models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 2961–2974. Association for Computational Linguistics.

Barbara Plank and Gertjan van Noord. 2011. Effective measures of domain similarity for parsing. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 1566–1576. The Association for Computer Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, and et al. Suresh Manandhar. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, pages 19–30. The Association for Computer Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2015, Denver, Colorado, USA, June 4-5, 2015*, pages 486–495. The Association for Computer Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014*, pages 27–35. The Association for Computer Linguistics.

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

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. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, pages 784–789. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2383–2392. The Association for Computational Linguistics.

Alan Ramponi and Barbara Plank. 2020. Neural unsupervised domain adaptation in NLP - A survey. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 6838–6855. International Committee on Computational Linguistics.

Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younès Bennani. 2020. A survey on domain adaptation theory. *CoRR*, abs/2004.11829.

Machel Reid, Victor Zhong, Suchin Gururangan, and Luke Zettlemoyer. 2022. M2D2: A massively multi-domain language modeling dataset. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 964–975. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Robert Remus. 2012. Domain adaptation using domain similarity- and domain complexity-based instance selection for cross-domain sentiment analysis. In *12th IEEE International Conference on Data Mining Workshops, ICDM Workshops, Brussels, Belgium, December 10, 2012*, pages 717–723. IEEE Computer Society.

Guy Rotman and Roi Reichart. 2019. Deep contextualized self-training for low resource dependency parsing. *Trans. Assoc. Comput. Linguistics*, 7:695–713.

Sebastian Ruder, Parsa Ghaffari, and John G. Breslin. 2017. Data selection strategies for multi-domain sentiment analysis. *CoRR*, abs/1702.02426.

Barbara Rychalska, Dominika Basaj, Alicja Gosiewska, and Przemyslaw Biecek. 2019. Models in the wild: On corruption robustness of neural NLP systems. In *Neural Information Processing - 26th International Conference, ICONIP 2019, Sydney, NSW, Australia, December 12-15, 2019, Proceedings, Part III*, volume 11955 of *Lecture Notes in Computer Science*, pages 235–247. Springer.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2023. Detecting pretraining data from large language models. *CoRR*, abs/2310.16789.

Hugo Touvron, Louis Martin, Kevin Stone, and et. al. Peter Albert. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Lifu Tu, Garima Lalwani, Spandana Gella, and He He. 2020. An empirical study on robustness to spurious correlations using pre-trained language models. *Trans. Assoc. Comput. Linguistics*, 8:621–633.

Vincent Van Asch and Walter Daelemans. 2010. Using domain similarity for performance estimation. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pages 31–36, Uppsala, Sweden. Association for Computational Linguistics.

Neeraj Varshney, Swaroop Mishra, and Chitta Baral. 2022. Towards improving selective prediction ability of NLP systems. In *Proceedings of the 7th Workshop on Representation Learning for NLP, RepL4NLP@ACL 2022, Dublin, Ireland, May 26, 2022*, pages 221–226. Association for Computational Linguistics.

Dave Van Veen, Cara Van Uden, Maayane Attias, Anuj Pareek, Christian Bluethgen, Malgorzata Polacin, Wah Chiu, Jean-Benoit Delbrouck, Juan Manuel Zambrano Chaves, Curtis P. Langlotz, Akshay Chaudhari, and John M. Pauly. 2023. Radadapt: Radiology report summarization via lightweight domain adaptation of large language models. In *The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks, BioNLP@ACL 2023, Toronto, Canada, 13 July 2023*, pages 449–460. Association for Computational Linguistics.

Tomer Volk, Eyal Ben-David, Ohad Amosy, Gal Chechik, and Roi Reichart. 2022. Example-based hypernetworks for out-of-distribution generalization. *CoRR*, abs/2203.14276.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. 2017. Tl;dr: Mining reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization, NFiS@EMNLP 2017, Copenhagen, Denmark, September 7, 2017*, pages 59–63. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. 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 3261–3275.

Thomas Wang, Adam Roberts, Daniel Hesslow, Teven Le Scao, Hyung Won Chung, Iz Beltagy, Julien Launay, and Colin Raffel. 2022a. What language model architecture and pretraining objective works best for zero-shot generalization? In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 22964–22984. PMLR.

Xuezhi Wang, Haohan Wang, and Diyi Yang. 2022b. Measure and improve robustness in NLP models: A survey. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 4569–4586. Association for Computational Linguistics.

Lucas Weber, Elia Bruni, and Dieuwke Hupkes. 2023. Mind the instructions: a holistic evaluation of consistency and interactions in prompt-based learning. In *Proceedings of the 27th Conference on Computational Natural Language Learning, CoNLL 2023, Singapore, December 6-7, 2023*, pages 294–313. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel R. 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, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Bang Yang, Fenglin Liu, Zheng Li, Qingyu Yin, Chenyu You, Bing Yin, and Yuexian Zou. 2023a. Multimodal prompt learning for product title generation with extremely limited labels. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 2652–2665. Association for Computational Linguistics.

Linyi Yang, Yaoxian Song, Xuan Ren, Chenyang Lyu, Yidong Wang, Jingming Zhuo, Lingqiao Liu, Jindong Wang, Jennifer Foster, and Yue Zhang. 2023b. Out-of-distribution generalization in natural language processing: Past, present, and future. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 4533–4559. Association for Computational Linguistics.

Ping Yu, Tianlu Wang, Olga Golovneva, Badr AlKhamissi, Siddharth Verma, Zhijing Jin, Gargi Ghosh, Mona Diab, and Asli Celikyilmaz. 2023. ALERT: Adapt language models to reasoning tasks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1055–1081, Toronto, Canada. Association for Computational Linguistics.

Tiezheng Yu, Zihan Liu, and Pascale Fung. 2021. Adaptsum: Towards low-resource domain adaptation for abstractive summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 5892–5904. Association for Computational Linguistics.

Yu Yu, Abdul Rafae Khan, and Jia Xu. 2022. Measuring robustness for NLP. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 3908–3916. International Committee on Computational Linguistics.

Lifan Yuan, Yangyi Chen, Ganqu Cui, Hongcheng Gao, Fangyuan Zou, Xingyi Cheng, Heng Ji, Zhiyuan Liu, and Maosong Sun. 2023. Revisiting out-of-distribution robustness in NLP: benchmark, analysis, and llms evaluations. *CoRR*, abs/2306.04618.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, and Dragomir R. Radev. 2021. Qmsum: A new benchmark for query-based multi-domain meeting summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 5905–5921. Association for Computational Linguistics.

Yftah Ziser and Roi Reichart. 2017. Neural structural correspondence learning for domain adaptation. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), Vancouver, Canada, August 3-4, 2017*, pages 400–410. Association for Computational Linguistics.

Yftah Ziser and Roi Reichart. 2018. Pivot based language modeling for improved neural domain adaptation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1241–1251. Association for Computational Linguistics.

Yftah Ziser and Roi Reichart. 2019. Task refinement learning for improved accuracy and stability of unsupervised domain adaptation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5895–5906. Association for Computational Linguistics.

Appendix

A On The Relationship Between SS, TT, ST, SD and TD

In this subsection, we expand the discussion from §3.1 and §3.2 about the Domain Robustness (DR) metrics introduced in our study. Our aim is to address and clarify any questions that might arise from the nuanced definitions presented earlier. Additionally, we offer a theoretical perspective on our findings discussed in the results subsection §6.4.

In §3 we define the DR challenge as *the inherent inability of an NLP model to generalize from the source domain to the target domains.* This inability is closely linked to the in-domain and cross-domain performance of the model, and *full characterization of it requires understanding the joint distribution of* SS*,* TT *and* ST. These three *performance measures* are random variables, with their variability stemming from the selection of source and target domains, the sampling of training and testing data from these domains, and the variabilities in the training and inference processes.

Nonetheless, identifying these random variables and their relationships is not tractable without further assumptions. We hence introduce simple, practical, and interpretable metrics that quantify the properties of the joint distribution: $\overline{\mathrm{SS}} = \mathbb{E}[\mathrm{SS}]$, $\overline{\mathrm{ST}} = \mathbb{E}[\mathrm{ST}]$ and $\overline{\Delta} = \overline{\mathrm{SS}} - \overline{\mathrm{ST}}$. Although our definitions rely on expectations, in practice, these metrics are task-level statistics (averages) that estimate them. Intuitively, the average drop ($\overline{\Delta}$) estimates the expected task-level performance degradation when shifting domains and *the larger it is, the more severe the DR challenge of the model is.*

Other three metrics that are derived from the joint distribution and quantify performance degradation at the shift level are SD, TD, and IDD. A positive in-domain difference may indicate a shift to a *harder domain*, and in contrast to the SD and the TD, *the* IDD *is not a genuine by-product of the DR challenge since it does not consider the* ST.

Based on our assertion that the joint distribution of SS, TT and ST is needed for characterizing the DR challenge, then it follows that we need at least two of the *degradation metrics* (SD, TD, and IDD) to do so. Moreover, the following trivial equation:

$$\mathrm{SD} = \mathrm{IDD} + \mathrm{TD}$$
$$\mathrm{TD} = \mathrm{IDD} - \mathrm{SD}$$

presents how the three metrics are connected. Accordingly, looking solely on one drop metric (SD or TD) can lead to incorrect conclusions, as large

Source	Target	ST	SS	TT	IDD	SD	TD	Scenario
A	A		90	90				
B	B		80	80				
C	C		70	70				
A	B	75	90	80	10	15	5	Classic
A	C	75	90	70	20	15	-5	Observed
B	A	95	80	90	-10	-15	-5	No Challenge
B	C	65	80	70	10	15	5	Classic
C	A	80	70	90	-20	-10	10	Unobserved
C	B	75	70	80	-10	-5	5	Unobserved

Table 7: Toy example of domain shifts. The task-level statistics are: $\overline{\text{SS}} = 80$; $\overline{\text{ST}} = 77.5$; $\overline{\Delta} = 2.5$; $W_{\text{SD}} = 15$; $W_{\text{TD}} = 10$. Notice that the mean of SD is 2.5, equal to that of TD and $\overline{\Delta}$. However, as many previous studies have done, examining only the challenging shifts (with IDD > 0, indicated by gray rows) and focusing on SD alone can obscure the real DR state. In these shifts, the mean SD is 15, which might be misconstrued due to large IDD. Incorporating the TD into the analysis can rectify this and avoid misinterpretations. Nonetheless, the most comprehensive approach to understanding task-level behavior is to consider all domains both as sources and targets, as we do. In this case, the means of all drops are equal: $\overline{\Delta} = \mathbb{E}[\text{SD}] = \mathbb{E}[\text{TD}]$.

drops might be attributed to the IDD. Notably, when a range of experiments is conducted **using all domains for both training and testing**, it follows that $\mathbb{E}[\text{SS}] = \mathbb{E}[\text{TT}]$, and from the linearity of the expectation, $\overline{\Delta} = \mathbb{E}[\text{SD}] = \mathbb{E}[\text{TD}]$. Importantly, while SD and TD have equal expected values, they are distinct random variables with differing variances. See the toy example in Table 7.

Although an accurate and truthful understanding of the DR challenge requires considering both metrics, many works measure only the SD. However, this is the least indicative option, as we empirically show that the TD is a more robust estimator of the average drop, $\overline{\Delta}$. This is because the TD tends to have a lower extreme magnitude and variance than the SD, and the IDD explains a larger portion of the SD than the TD.

Below, we introduce a theorem that binds these properties together and demonstrates their equivalence. But even more, it reveals them to be equivalent to the case when the ST is more akin to the TT (e.g. when $\text{Cov}[\text{ST}, \text{TT}] > \text{Cov}[\text{ST}, \text{SS}]$). In other words, if we believe that in our task the potential of the model to perform well cross-domain is determined by the difficulty of the target domain, as in the case of challenge sets, then the reference point for measuring a degradation should be the TT and not the SS, and the TD would be indeed the better drop metric.

Theorem 1. *Let (S, T) be different source and target domains sampled independently from the domain space, and let $(\text{SS}, \text{TT}, \text{ST})$ be RVs of their performances. The following are equivalent:*

(1) $\text{Cov}[\text{TT}, \text{ST}] > \text{Cov}[\text{SS}, \text{ST}]$

(2) $\text{Cov}[\text{IDD}, \text{SD}]^2 > \text{Cov}[\text{IDD}, \text{TD}]^2$

(3) $\text{Var}[\text{SD}] > \text{Var}[\text{TD}]$

(4) $\mathbb{E}[|\text{SD}|] > \mathbb{E}[|\text{TD}|]$

Remark 1. Although in Theorem 1 we employ fundamental probability concepts such as expectation, variance, and covariance, our results utilize well-established and easily interpretable statistics: (1) We use the Pearson's correlation between the ST and the SS or the TT; (2) We use the R-squared (R^2) between the IDD and the SD or the TD. Notably, the R-squared indicates the proportion of the variability in a dependent variable (SD) that is explained by the independent variable (IDD), serving as a gauge of the goodness of fit. We use Peasron's correlation to understand the relationship of SS, TT, and ST because it considers the directionality of the relationship, indicated by the sign. In contrast, here we use the R^2 since it focuses on the degree, ignoring the sign; (3) We use the sample standard deviation of the drops; (4) We use the maximum drops (Worst SD or TD); While the concepts in Theorem 1 are not direct equivalents of these statistics, they are closely related and help elucidate our findings.

Remark 2. Notice that, $\text{ST} = \text{TT} - \text{TD}$ and $\text{SD} = \text{IDD} + \text{TD}$. Although we found a strong relationship between the ST and the TT (e.g., $\rho = 0.95$ in the fine-tuning QG task) or between the SD and the IDD (e.g., $R^2 = 0.96$ in fine-tuning QA task), this does not imply that the TD is zero and no DR challenge exist. These strong correlations or high R^2 values merely reflect the TD has a low variability. Its magnitude cannot be inferred from the correlation or R^2 alone.

Proof. We start be denoting $x = \text{Var}[\text{SS}] > 0$ and $y = \text{Cov}[\text{TT}, \text{ST}] - \text{Cov}[\text{SS}, \text{ST}]$. Notice that $\mathbb{E}[\text{SS}] = \mathbb{E}[\text{TT}]$ and $\text{Var}[\text{SS}] = \text{Var}[\text{TT}]$. From the linearity of expectation, we get:

$$\begin{aligned} \mathbb{E}[\text{SD}] &= \mathbb{E}[\text{SS}] - \mathbb{E}[\text{ST}] \\ &= \mathbb{E}[\text{TT}] - \mathbb{E}[\text{ST}] = \mathbb{E}[\text{TD}] \end{aligned}$$

(1) $\Leftrightarrow$ (2): Since S and T are independent then $\mathrm{Cov}[\mathrm{SS}, \mathrm{TT}] = 0$. From the bilinearity of the covariance, we get:

$$\mathrm{Cov}[\mathrm{IDD}, \mathrm{SD}] = \mathrm{Var}[\mathrm{SS}] + \mathrm{Cov}[\mathrm{SS}, \mathrm{TT}]$$
$$-\mathrm{Cov}[\mathrm{SS}, \mathrm{ST}] + \mathrm{Cov}[\mathrm{TT}, \mathrm{ST}] = x + y$$

Similarly, $\mathrm{Cov}[\mathrm{IDD}, \mathrm{SD}] = -x + y$.

If (1) holds, then $y > 0$. Since x and y are both positive, then $(x + y)^2 > (-x + y)^2$ and (2) holds. The same is true for the other direction: if (2) holds, then y must be positive, and (1) holds.

(1) $\Leftrightarrow$ (3): From the variance of a sum, we get:

$$\mathrm{Var}[\mathrm{SD}] = \mathrm{Var}[\mathrm{SS}] - 2\mathrm{Cov}[\mathrm{ST}, \mathrm{SS}] + \mathrm{Var}[\mathrm{ST}]$$
$$\mathrm{Var}[\mathrm{TD}] = \mathrm{Var}[\mathrm{TT}] - 2\mathrm{Cov}[\mathrm{ST}, \mathrm{TT}] + \mathrm{Var}[\mathrm{ST}]$$

If (1) holds, then:

$$\mathrm{Var}[\mathrm{SD}] - \mathrm{Var}[\mathrm{TD}] =$$
$$2(\mathrm{Cov}[\mathrm{ST}, \mathrm{TT}] - \mathrm{Cov}[\mathrm{ST}, \mathrm{SS}]) > 0$$

and (3) holds. Invert the order to prove (3) $\Rightarrow$ (1).

(1) $\Leftrightarrow$ (4): Notice that:

$$\mathbb{E}[\mathrm{SD}^2] = \mathbb{E}[\mathrm{SS}]^2 - 2\mathbb{E}[\mathrm{SS} \cdot \mathrm{ST}] + \mathbb{E}[\mathrm{ST}]^2$$
$$\mathbb{E}[\mathrm{TD}^2] = \mathbb{E}[\mathrm{SS}]^2 - 2\mathbb{E}[\mathrm{TT} \cdot \mathrm{ST}] + \mathbb{E}[\mathrm{ST}]^2$$

Since $\mathbb{E}[\mathrm{SS}] = \mathbb{E}[\mathrm{TT}]$, we get:

$$\mathbb{E}[\mathrm{SD}^2] - \mathbb{E}[\mathrm{TD}^2] = 2(\mathbb{E}[\mathrm{TT} \cdot \mathrm{ST}] - \mathbb{E}[\mathrm{SS} \cdot \mathrm{ST}])$$

From the definition of covariance:

$$\mathrm{Cov}[\mathrm{ST}, \mathrm{SS}] = \mathbb{E}[\mathrm{SS} \cdot \mathrm{ST}] - \mathbb{E}[\mathrm{SS}]\mathbb{E}[\mathrm{ST}]$$
$$\mathrm{Cov}[\mathrm{ST}, \mathrm{TT}] = \mathbb{E}[\mathrm{TT} \cdot \mathrm{ST}] - \mathbb{E}[\mathrm{TT}]\mathbb{E}[\mathrm{ST}]$$

and therefore:

$$\mathrm{Cov}[\mathrm{ST}, \mathrm{TT}] - \mathrm{Cov}[\mathrm{ST}, \mathrm{SS}]$$
$$= \mathbb{E}[\mathrm{TT} \cdot \mathrm{ST}] - \mathbb{E}[\mathrm{SS} \cdot \mathrm{ST}]$$

Now, if (1) holds, then $\mathbb{E}[\mathrm{TT} \cdot \mathrm{ST}] > \mathbb{E}[\mathrm{SS} \cdot \mathrm{ST}]$ and $\mathbb{E}[\mathrm{SD}^2] > \mathbb{E}[\mathrm{SD}^2]$, and (4) holds. To prove the converse, reverse the implications. $\square$

A.1 Domain Divergence and Performance Drops

Many past works have explored the connection between domain divergence, a notion of distance between two domains, and the performance drops (Remus, 2012; Ruder et al., 2017). This includes theoretical works that upper-bound the cross-domain performance based on domain divergence (Ben-David et al., 2010; Redko et al., 2020), and empirical studies that have identified a degree of correlation between divergence metrics and SD (El-Sahar and Gallé, 2019; Kashyap et al., 2021).

While divergence is indeed connected to cross-domain performance and thus to the performance drop, in practice, numerous other factors may influence robustness and performance drops, for example, the IDD $=$ SS $-$ TT, which serves to quantify the transition to a more challenging domain and is not a byproduct of a domain shift or a divergence (because it is defined only by SS and TT, and not by ST). In this subsection, we aim to explore the correlation between domain divergence and the performance drop metrics introduced in this paper.

Following Remus (2012) and Ruder et al. (2017), we decided to use the Jensen Shannon Divergence (JS-Div). This decision is based on findings from Kashyap et al. (2021), which demonstrated that, among various divergence metrics, the JS-Div typically shows the highest average correlation. We utilize word frequency distribution to compute the JS-Div, excluding stop-words and considering only the top 10k frequent words (Kashyap et al., 2021). We then compute for each model and task the correlation between the divergence and the SD or TD across all pairs of domains. Table 8 presents the average Spearman's correlations.

Our results indicate that stronger correlations between domain divergence and performance drops occur when the DR challenge is more severe. For

		SA	NLI	AB	QA	QG	AS	TG
	$JS-Div$	0.23	0.32	0.27	0.30	0.27	0.18	0.18
Fine-tuning	$\overline{\Delta}$	3.45	2.72	22.99	0.60	0.17	0.93	1.24
	$\rho(Div, \mathrm{SD})$	0.43	0.02	0.53	-0.02	0.02	0.07	0.09
	$\rho(Div, \mathrm{TD})$	0.73	0.16	0.54	0.02	0.19	0.15	0.38
	$\rho(\mathrm{IDD}, \mathrm{SD})$	0.53	0.91	0.51	0.86	0.98	0.98	0.96
	$\rho(\mathrm{IDD}, \mathrm{TD})$	-0.27	0.01	-0.30	-0.29	-0.08	-0.61	-0.78
Few-shot	$\overline{\Delta}$	1.29	2.20	3.37	0.49	0.13	0.45	0.18
	$\rho(Div, \mathrm{SD})$	0.00	0.02	0.01	-0.04	0.01	0.20	0.12
	$\rho(Div, \mathrm{TD})$	0.12	0.16	0.19	-0.08	0.07	0.26	0.00
	$\rho(\mathrm{IDD}, \mathrm{SD})$	0.97	0.88	0.99	0.83	0.97	0.86	0.79
	$\rho(\mathrm{IDD}, \mathrm{TD})$	-0.29	0.33	-0.64	0.01	-0.26	-0.71	0.07

Table 8: Correlations between domain divergence (Jensen-Shannon) and performance drop metrics. We first calculate the statistic for each model and then present the mean statistic for the task. The first row presents the average JS-divergence in the task. $\rho(\cdot, \cdot)$ presents the Spearman's correlation. We also present the correlation between the IDD and the performance drop for comparison.

instance, these correlations are higher for fine-tuned models compared to few-shot models, corresponding with larger average drops ($\overline{\Delta}$). Additionally, we see stronger correlations in tasks such as SA, AB, and TG, which also have larger drops.

In addition, we also present in Table 8 correlations between the IDD and drops. We see that the IDD is a strong predictor (larger magnitude) of the SD, while the opposite holds for domain divergence, which is a better predictor of the TD. This is interesting because the domain divergence is theoretically linked to the cross-domain performance, while the IDD is not, further suggesting that the TD is a more reliable estimator of the DR.

Finally, DR studies typically measure robustness by analyzing shifts only to synthetic, adversarial, or challenge sets, which are known to exhibit high IDD. These studies also tend to rely solely on the SD, with high drops suggesting a lack of model robustness. However, our findings raise concerns about the validity of these assessments, which tend to overestimate the severity of the DR challenge, which is generally milder. A more balanced approach would analyze the TD as well, which could help mitigate this bias.

A.2 Intuition for Domain Shift Scenarios

In §3.3 we introduce a framework for classifying types of domain shifts into four scenariosL *Classic* (SD > 0 and TD > 0), *Observed* (SD > 0 but TD < 0), *Unobserved* (SD < 0 but TD > 0), and *No Challenge* (SD < 0 and TD < 0).

While performance degradation with respect to TT (positive TD) seems intuitive (as we do not expect the model to perform better than it would have had it been trained on data from the target domain), one may wonder about the cases where TD is negative. Specifically the *Observed* and *No Challenge* scenarios which can be counter-intuitive.

In what follows, we will elaborate on these scenarios. First, notice that every scenario can occur if the effect of the domain shift is noisy. Second, consider the following motivation:

The No Challenge scenario (SS $>$ ST and TT $>$ ST): Imagine a model trained on advanced math problems (graduate level) being applied to basic math problems (elementary level). In this case, we anticipate a *No Challenge* scenario due to the simplicity of elementary problems compared to graduate-level problems (SS $>$ ST) and the model's capability to understand complex graduate-level content, which implies it can certainly handle elementary-level problems (TT $>$ ST).

The Observed scenario (SS $>$ ST $>$ TT): Now consider the opposite direction. The model is trained on elementary math problems and applied to graduate-level problems. Obviously, we anticipate SS to be larger than ST. In addition, within the set of graduate-level problems, there are some introductory or "warmup" problems (that the model trained on the elementary-level problem can solve). Despite the presence of simpler problems within the graduate-level set, the overall complexity of this domain can prevent the model from learning even the elementary concepts when trained on graduate-level problems, and thus, ST $>$ TT.

Notice that, indeed, the *Observed* and *No challenge* scenarios are the least common scenarios (see Figure 3). They occur mostly in the few-shot setups and can be attributed to the weaker effect of the domain shift on few-shot models. In addition, they also occur for FT models in the QA and QG tasks where the shift effect is also weak (see Table 5).

B Extended Discussion

In the section, we extend the discussion from §7 and summarise and discuss the key implications of our work.

On Domain Robustness Research As discussed in the paper, most past DR works focused solely on the observed performance degradation (SD) as a measure of the DR challenge. However, as asserted in this paper, a full characterization of the DR challenge requires deriving the joint distribution of SS, TT, and ST, which is not tractable. Therefore, we propose practical metrics to quantify the performance degradation: SD and TD.

We need both metrics for a single domain shift because large drops might be attributed to the in-domain difference (IDD) and obscure the DR challenge of the shift. Indeed, our findings indicate that a large SD commonly coexists with a large IDD. At the task level, the expected values of both drop metrics (SD and TD) are equal and correspond to the average drop ($\overline{\Delta}$). However, we empirically find that the TD is a better estimator of the $\overline{\Delta}$. This implies that when examining a limited number of domain shifts, it is crucial to include the TD.

In addition, we suggest that current research may paint an inaccurate picture of the state of domain robustness. This stems from two of our findings. First, performance degradation is larger when measured with the SD than with the TD. Second, every task has shifts with severe performance drops, even when most shifts are not remotely as bad. This means that past works focused only on the SD and challenging shifts such as challenge sets and other highly-curated datasets present a distorted image of the actual state of DR, which is actually much milder. Nevertheless, we acknowledge the importance of challenge sets as diagnostic tools.

On the Relevance of Fine-tuning Zero-shot and few-shot LLMs can perform various tasks without the additional cost of annotating data or training a model. However, their usage can be very costly, as they require massive computational resources, and their latency can be extremely high. Additionally, when the data cannot be sent to external servers because of privacy constraints or when the domain is unique or specific (e.g., in national security settings or human conversations), LLMs that cannot be fine-tuned may be less effective. Moreover, with enough task-specific labeled data that few-shot LLMs can cheaply annotate, it is possible to develop a small, high-performing, fine-tuned model (Calderon et al., 2023; Gekhman et al., 2023a; Ormazabal et al., 2023). For these reasons, fine-tuning a smaller model that does not have few-shot capabilities is still the de-facto standard (Levine et al., 2022).

Moreover, there is strong evidence that few-shot language models underperform fine-tuned models in specific domains that require expertise, such as biomedical (Gutierrez et al., 2022) or when the training size is large enough (Yuan et al., 2023). This study also shows that task-specific fine-tuned models outperform few-shot models in-domain, although this gap may be closed soon. Nevertheless, we also found that few-shot LLMs are more robust to domain shifts and can outperform fine-tuned models cross-domain. This calls for further Domain Adaptation research of fine-tuned models.

On the Relevance of Domain Adaptation Domain Adaptation (DA) is a field that addresses solutions to the DR problem. DA research considers various setups, each having different assumptions on the availability of data from the target domain at the model training time (Blitzer et al., 2007; Plank and van Noord, 2011; Ziser and Reichart, 2017, 2019; Rotman and Reichart, 2019; Ben-David et al., 2020; Ramponi and Plank, 2020; He et al., 2021b; Ben-David et al., 2022a; Calderon et al., 2022; Volk et al., 2022; Ge et al., 2023; Lang et al., 2023; Liang et al., 2023; Veen et al., 2023).

Modern NLP models are believed to be robust due to the pretraining process, where the models have seen a vast amount of diverse data from various domains. Another reason could be data contamination (Magar and Schwartz, 2022; Shi et al., 2023), i.e., pretraining on data from a downstream task improves the performance on it (Radford et al., 2019; Han and Eisenstein, 2019; Gururangan et al., 2020). This belief questions the relevance or the necessity of Domain Adaptation research.

However, in this study, we demonstrated that the DR challenge still exists. We show that there is a performance drop due to domain shift in every task or model, and moreover, some shifts are remarkably challenging. We believe that DA research remains essential and relevant, particularly for NLP. To facilitate further research, we provide an NLP benchmark with natural topic shifts, which has some challenging setups for various NLP tasks. We hope this benchmark will be used to evaluate and improve DA methods.

On Predicting Cross-domain Performance Estimating performance has an important impact on the deployment and maintenance of NLP models and related financial decisions (e.g., the need for annotation) (Van Asch and Daelemans, 2010; ElSahar and Gallé, 2019; Varshney et al., 2022; Ben-David et al., 2022b). We found that the TT is a better predictor of the cross-domain performance (ST) than the SS. Accordingly, knowledge about the target domain is essential, and without it, estimators may struggle to predict cross-domain performance. In addition, previous studies have attempted to predict performance drops (specifically, only the SD) using domain divergence (Kashyap et al., 2021). Our study (see A.1) suggests that domain divergence is a better predictor of the TD than the SD.

C Additional Results

C.1 Fine-tuned Model Size

Larger fine-tuned models often lead to better performance, but the question remains: How does the model size affect its domain robustness? To address this question, we have conducted comprehensive experiments using models of different sizes within the same architectural families, as detailed in Table 3. In Figure 4, we compare the absolute performance of various model sizes within the same model families. Conversely, Figure 5 presents the performance drops for these models.

Same as our finding in 6, we observe that also across all model sizes and all tasks (except QA

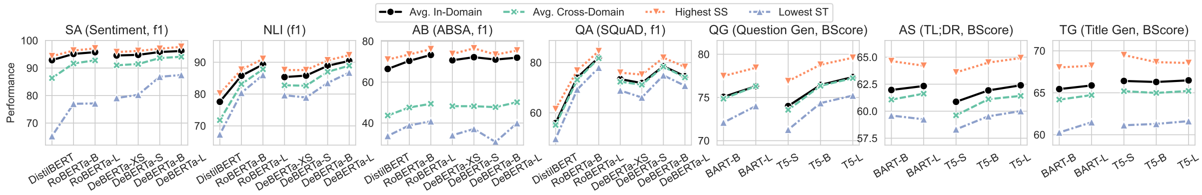

Figure 4: Fine-tuning performance for the seven tasks of different models with varying sizes. The plots present the F1 and BertScore scores of the average in-domain (black line) and cross-domain (green line) performance. In addition, the highest in-domain score (orange line) and the lowest cross-domain score (blue line) are displayed.

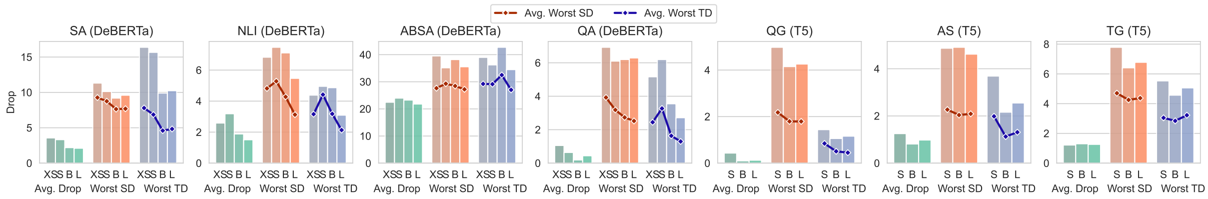

Figure 5: Fine-tuning drops of DeBERTa and T5 families. The plots present: The Average Drop (green bars); The Worst SD (orange bars); and the Worst TD (blue bars). The lines on the bars present the Average Worst SD and TD, i.e., for each source domain we first find the worst drop and then take the average over all source domains.

and QG), the average in-domain performance consistently exceeds the average cross-domain performance and the Worst SD surpasses the Worst TD.

When examining the influence of increasing model size, we find that, as expected, larger models within the same architectural family improve the absolute in-domain cross-domain performance. Regarding the performance drop, the general trend is that larger models reduce performance drops, a trend that is more pronounced in classification tasks. This indicates that utilizing larger models could enhance not just the absolute performance, but also the DR of these models.

C.2 Number of Few-shot Demonstrations

In contrast to fine-tuning, in few-shot setups there is potentially a weaker anchoring of the model in the source domain since it is not trained on domain-specific data. Instead, the few-shot model is simply provided with a few demonstrations from the source domain. We investigate whether increasing the number of demonstrations strengthens this anchoring, thereby potentially affecting the model's domain robustness. Figures 6 and 7 illustrate the impact of the number of demonstrations on both the absolute performance and performance drops of few-shot models, respectively.

Unsurprisingly, when comparing zero-shot to few-shot, we see that incorporating demonstrations generally enhances performance for most tasks and models. Nevertheless, in many instances, particularly with GPT3.5, using just a single demonstration surprisingly leads to poorer performance. This could imply that a single demonstration might introduce a bias detrimental to performance (e.g., the LLM predicts the same label as the demonstration).

For tasks other than SA, we observe that a greater number of demonstrations tends to improve both in-domain and cross-domain performance. The influence on performance drops is less straightforward - it appears that increasing the number of demonstrations may either exacerbate the drop in performance or have no significant effect.

In conclusion, it is better to use a greater number of demonstrations, with a preference for those originating from the target domain.

C.3 Few-shot Model Size

In this subsection, we explore the effect of the few-shot model size on DR. For this analysis, we experimented with LLMs from the Orca and Llama2 families. These families support 2 (Orca) and 3 (Llama2) of different sizes, all of which have undergone similar training and alignment procedures. Due to hardware constraints, we were unable to load the Llama2-70b model. Therefore, all Llama2 models were loaded with NF4 quantization, and computations were performed in 16-bit FP.

Although the results are inconclusive, since in some tasks (QA and TG) the performance of the 70b model sharply drops, we can still observe in Figure 8 that increasing the model size generally

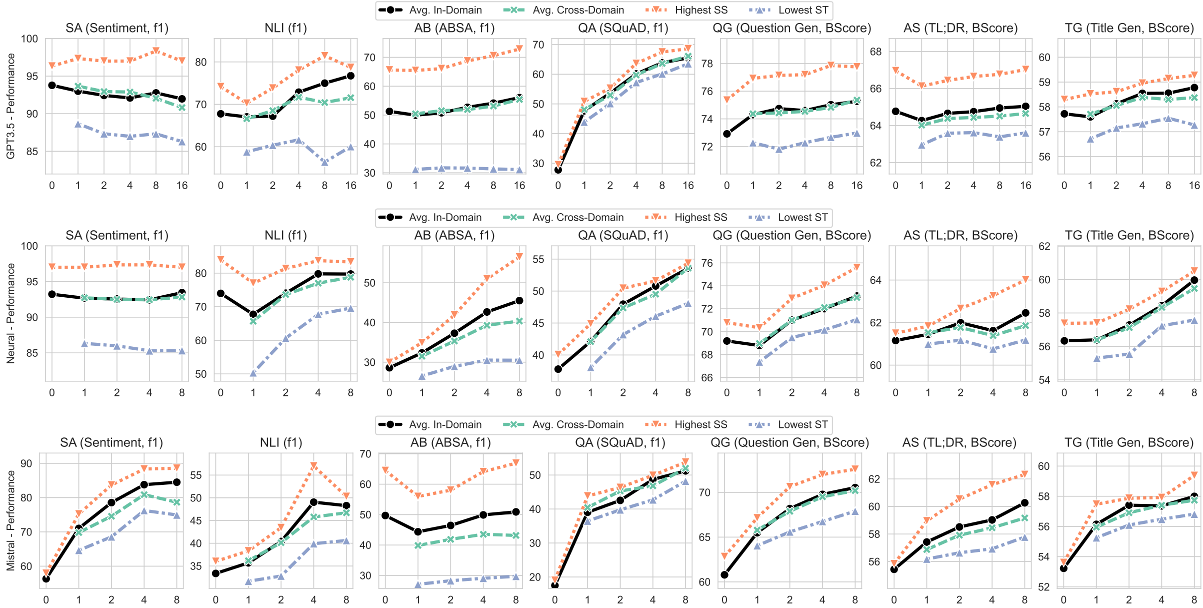

Figure 6: Performance of GPT3.5 (top), NeuralChat (middle) and Mistral (bottom) as a function of the number of few-shot demonstrations. The plots present the F1 and BertScore scores of the average in-domain (black line) and cross-domain (green line) performance. In addition, the highest in-domain score (orange line) and the lowest cross-domain score (blue line) are displayed.

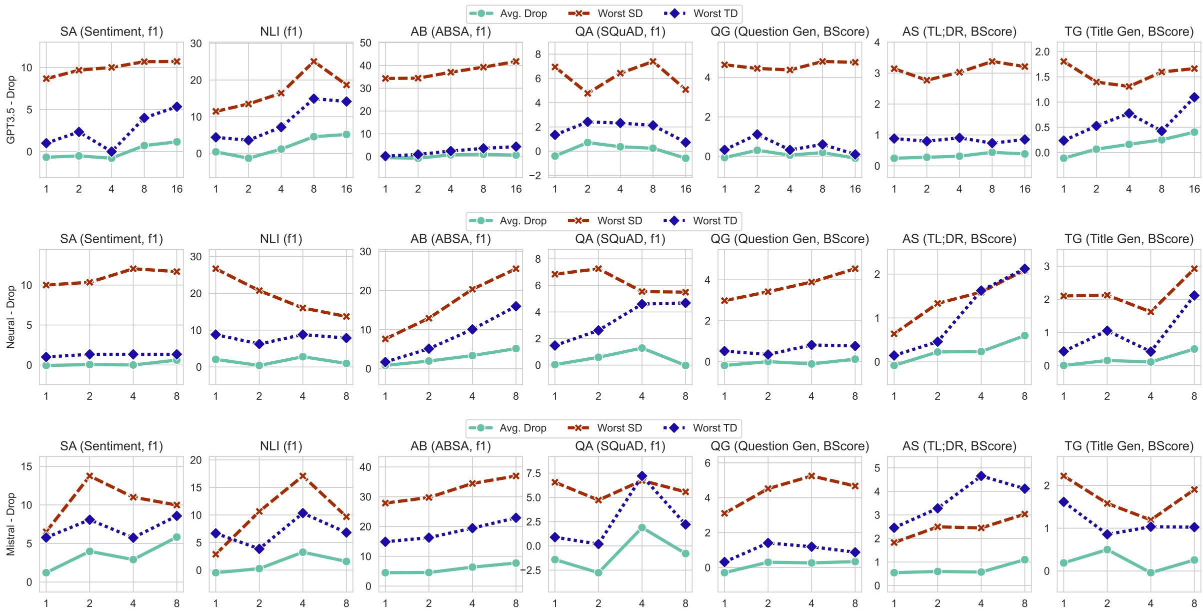

Figure 7: Performance drops of GPT3.5 (top), NeuralChat (middle) and Mistral (bottom) as a function of the number of few-shot demonstrations. The plots present: The Average Drop (green line); The Worst SD (orange line); and the Worst TD (blue line).

improves the absolute in-domain and cross-domain performance. This behavior is not surprising and is similar to what is observed in fine-tuning setups. Regarding the drops presented in Figure 9, the trends can be mixed. Yet, it appears that both the average drops and the worst drops are decreasing as the size increases.

C.4 Dataset Size

Our next analysis aims to explore how the number of training samples from the source domain influences the domain robustness. Figures 10 and 11 depict the impact of the size of the source training dataset on the performance of models in classification and generation tasks, respectively.

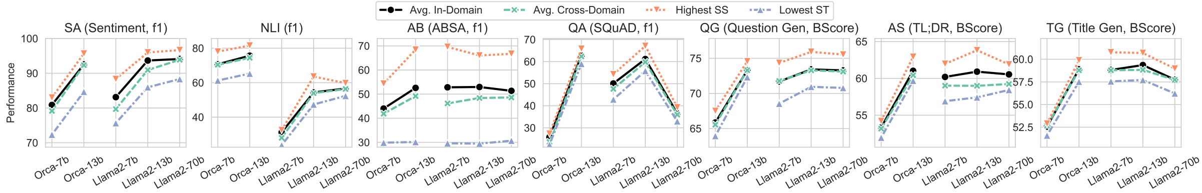

Figure 8: Few-shot (4 demonstrations) performance for the seven tasks of Llama2-family models with varying sizes. The plots present the F1 and BertScore scores of the average in-domain (black line) and cross-domain (green line) performance. In addition, the highest in-domain score (orange line) and the lowest cross-domain score (blue line) are displayed. Due to hardware constraints, all models were loaded with NF4 quantization, and computations were performed in 16-bit FP.

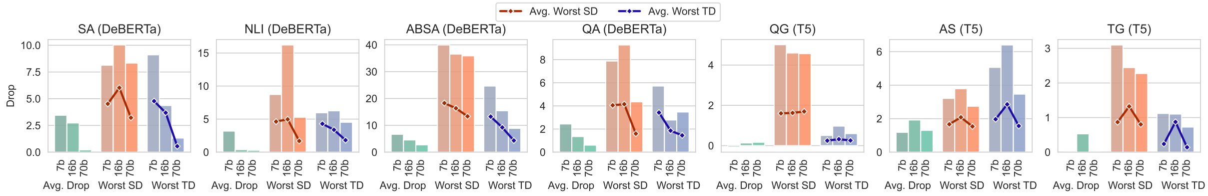

Figure 9: Few-shot (4 demonstrations) drops for the seven tasks of Llama2-family models with varying sizes. The plots present: The Average Drop (green bars); The Worst SD (orange bars); and the Worst TD (blue bars). The lines on the bars present the Average Worst SD and TD, i.e., for each source domain we first find the worst drop and then take the average over all source domains. Due to hardware constraints, all models were loaded with NF4 quantization, and computations were performed in 16-bit FP.

As expected, an increase in the dataset size enhances performance in both in-domain and cross-domain. For classification tasks, while an increase in sample size tends to decrease the worst SD and TD, it does not affect the average drop. On the other hand, in generation tasks, the effect varies across different tasks. Interestingly, in the TG and AS tasks, we observe larger drops when increasing the number of samples.

C.5 Epochs and Model Selection

In the standard fine-tuning process, a model is trained until it no longer shows improvement on the validation set and the model selected for deployment is the one that attains the highest validation score. However, this approach does not guarantee optimal performance in the target domain, nor does it necessarily lead to the best model selection. We therefore wish to measure how the in-domain and cross-domain performance evolve over the course of the fine-tuning procedure, across different epochs.

As seen in Figure 12, in most cases, models appear to reach convergence in terms of average in-domain and cross-domain performance within a few epochs. Yet, it is noteworthy that the lowest cross-domain performance exhibits significant variability, undergoing substantial fluctuations during the training process. A similar pattern is observed in the performance drops.

These findings raise an interesting research question: Considering the significant variability of the cross-domain performance during the fine-tuning process, what is the optimal strategy for selecting a domain robust model? This question opens an interesting avenue for further research.

C.6 Token Embeddings

Every Transformer-based model employs an embedding matrix to transform tokens into continuous vectors. One strategy, known as 'freezing' this matrix, involves not updating its weights during fine-tuning (Ben-David et al., 2020). This tactic is motivated by the idea that, given the vocabulary differences across domains, maintaining the original embeddings might prevent the introduction of biases specific to the source domain. Consequently, this approach could potentially enhance the ability to generalize across different domains.

The results, presented in Table 9, indicate that freezing embeddings during fine-tuning does not harm the in-domain performance while increasing the cross-domain performance by approximately 0.5 points in SA and 0.2 points in NLI. Regarding the worst drops, in the SA task this approach remarkably improves the drops while in the NLI task,

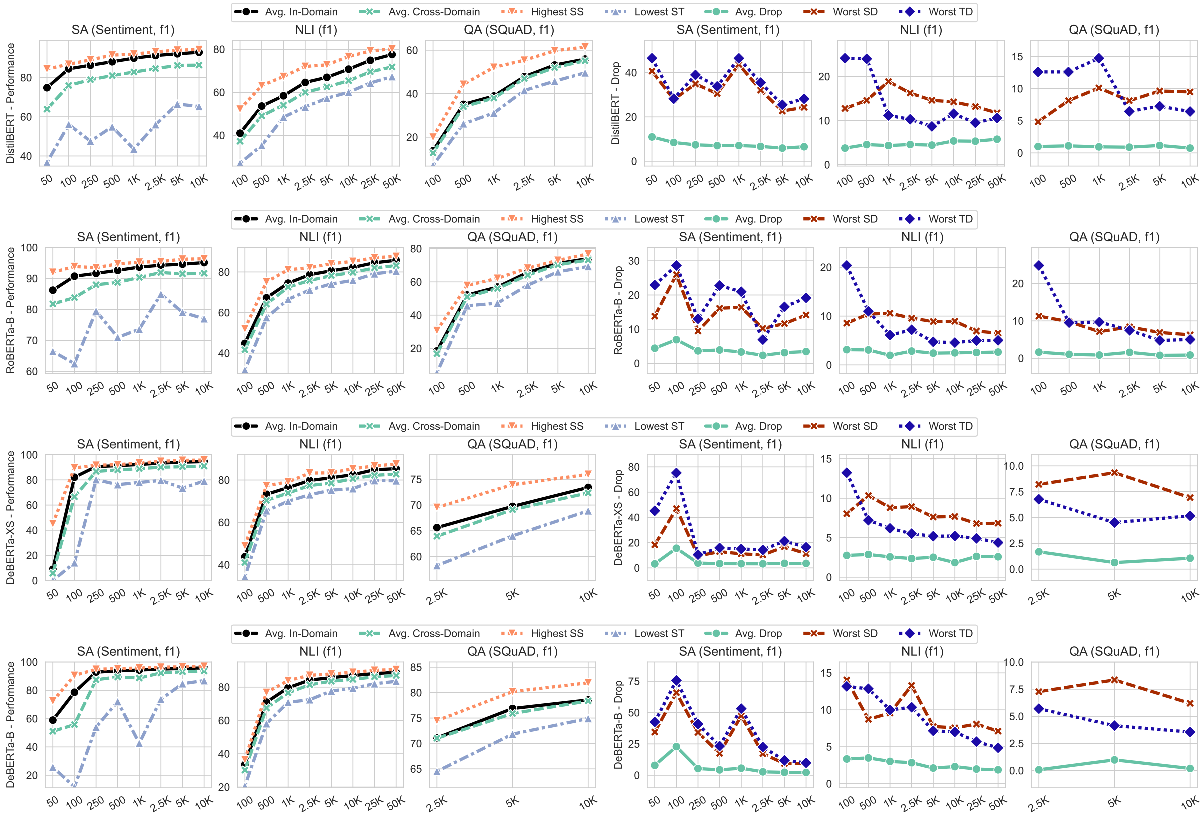

Figure 10: Classification performance and drops of DistilBERT (first row), RoBERTa-B (second row), DeBERTa-XS (third row) and DeBERTa-B (fourth row) as a function of the training dataset size of the source domain. In the leftmost three columns: The F1 scores of the average performance in-domain (black line); cross-domain (green line); The highest in-domain score (orange line); The lowest cross-domain score (blue line). In the rightmost three columns: The Average Drop (green line); The Worst SD (orange line); and the Worst TD (blue line).

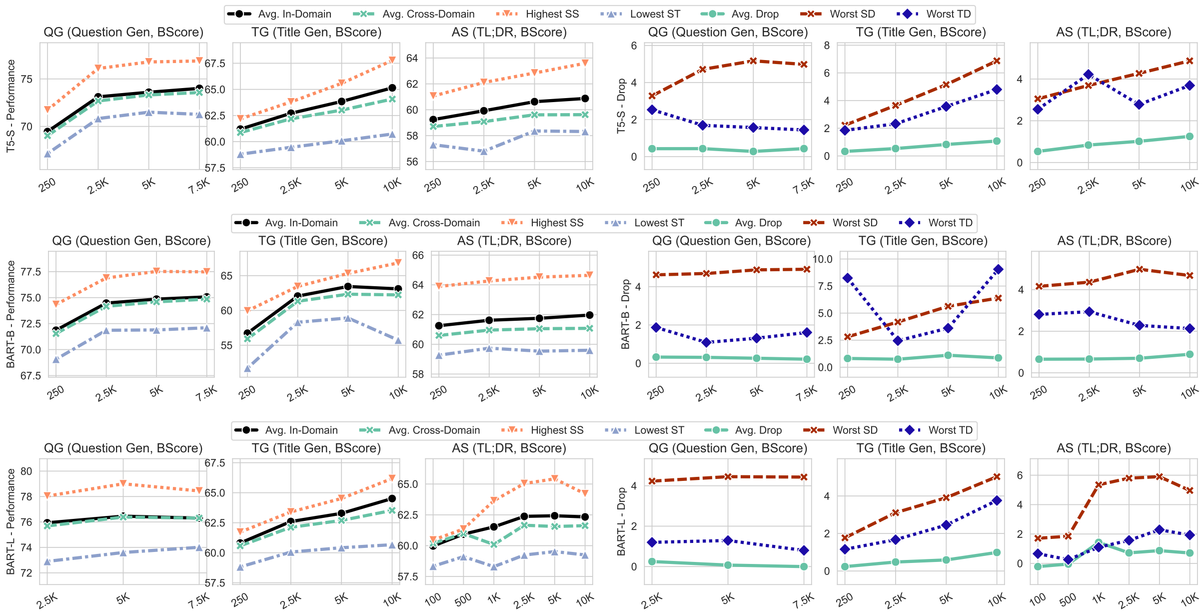

Figure 11: Generation performance and drops of T5-S (first row), BART-B (second row) and BART-L (third row) as a function of the training dataset size of the source domain. In the leftmost three columns: The BERTScores of the average performance in-domain (black line); cross-domain (green line); The highest in-domain score (orange line); The lowest cross-domain score (blue line). In the rightmost three columns: The Average Drop (green line); The Worst SD (orange line); and the Worst TD (blue line).

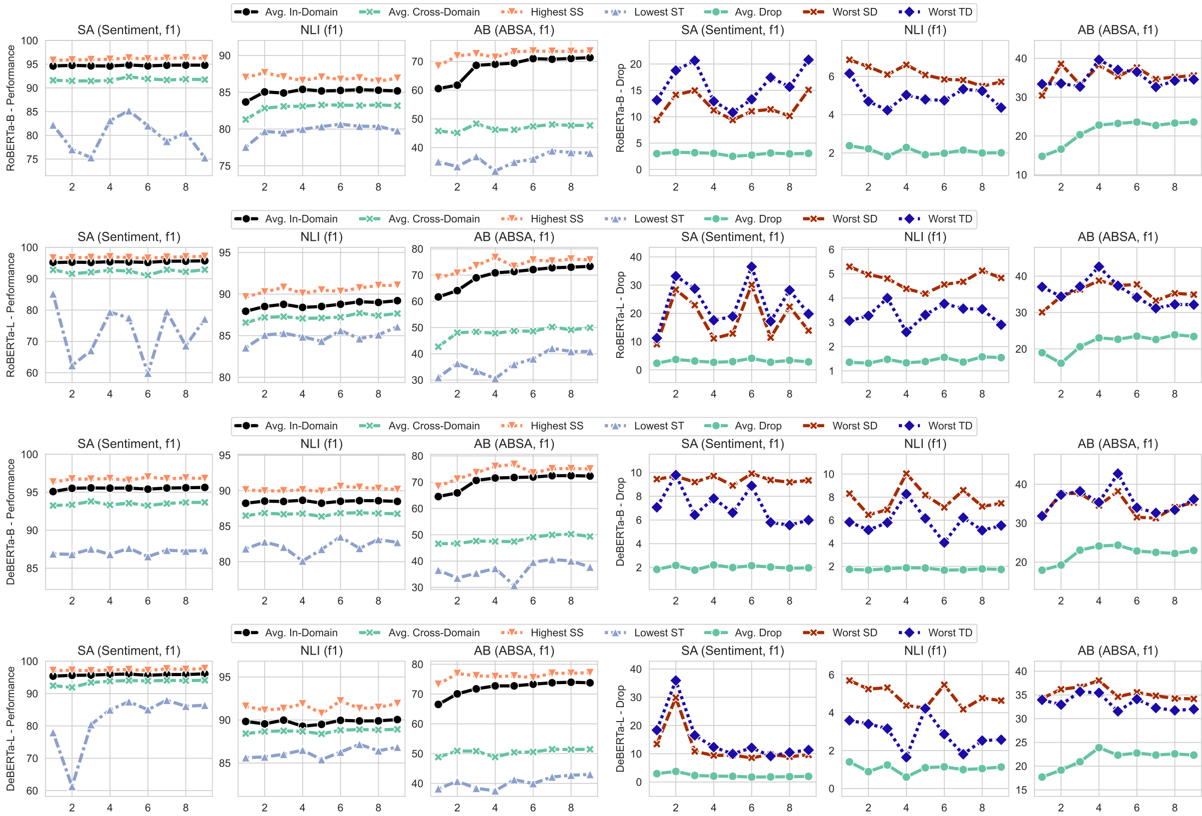

Figure 12: Performance and drops of RoBERTa-B (first row), RoBERT-L (second row), DeBERTa-B (third row) and DeBERTa-L (fourth row) as a function of the epoch. In the leftmost three columns: The F1 scores of the average performance in-domain (black line); cross-domain (green line); The highest in-domain score (orange line); The lowest cross-domain score (blue line). In the rightmost three columns: The Average Drop (green line); The Worst SD (orange line); and the Worst TD (blue line).

it slightly degrades them. These findings suggest that freezing embeddings could serve as a simple baseline for future research in domain adaptation.

Task	$\overline{\text{SS}}$	$\overline{\text{ST}}$	$\overline{\Delta}$	W_{SD}	W_{TD}
SA	95.13	91.67	3.46	14.16	19.18
+ FZ	95.13	92.17	2.96	10.04	12.99
NLI	85.76	83.13	2.63	6.51	5.03
+ FZ	85.60	83.33	2.27	7.04	5.18

Table 9: Results of RoBERTa-B in the SA and NLI tasks under two scenarios: First, when the token embeddings matrix is trainable (SA and NLI), and second, when it is frozen (+ FZ). The columns are: $\overline{\text{SS}}$ - Average In-domain Performance, $\overline{\text{ST}}$ - Average Cross-domain Performance, $\overline{\Delta}$ - Average Drop, W_{SD} - Worst SD and W_{TD} - Worst TD.

C.7 Prior Shift

When developing our benchmark, we decided to restrict it to several technical assumptions (described in §D.2). These assumptions enable a precise and clear analysis in a "controlled experiment" manner. One of the assumptions is that the prior distribution

Model	Task	$\overline{\text{SS}}$	$\overline{\text{ST}}$	$\overline{\Delta}$	W_{SD}	W_{TD}
DistilBERT	QA	55.89	55.15	0.75	9.49	6.44
	+ IB	53.76	54.13	-0.37	7.51	17.62
RoBERTa-B	QA	74.01	73.15	0.86	6.29	5.03
	+ IB	70.66	69.96	0.71	15.67	24.42
RoBERTa-L	QA	82.01	81.72	0.29	6.01	2.53
	+ IB	81.09	80.62	0.47	13.74	10.55
DeBERTa-XS	QA	73.41	72.36	1.06	6.93	5.16
	+ IB	70.50	70.01	0.48	17.86	21.51
DeBERTa-S	QA	71.83	71.19	0.64	6.10	6.19
	+ IB	66.10	67.00	-0.90	13.67	18.10
DeBERTa-B	QA	78.56	78.37	0.19	6.20	3.55
	+ IB	78.57	78.21	0.36	14.95	10.00
DeBERTa-L	QA	74.54	74.10	0.44	6.29	2.72
	+ IB	79.82	79.06	0.77	17.67	15.84

Table 10: Results of fine-tuned models in the QA task under two scenarios: First, when all domains have an identical ratio of questions without answers (QA), and second, when the distribution of 'no answer' questions varies between domains (+ IB - imbalanced). The columns are: $\overline{\text{SS}}$ - Average In-domain Performance, $\overline{\text{ST}}$ - Average Cross-domain Performance, $\overline{\Delta}$ - Average Drop, W_{SD} - Worst SD and W_{TD} - Worst TD.

$P(Y)$ remains relatively consistent across various domains. For classification tasks, every domain has the same class distribution. In the QA task, it translates to each domain having the same ratio of 'no answer' examples (0.2). This subsection explores what happens when this assumption does not hold and a prior shift occurs. To this end, we reconstruct the QA dataset by resampling examples from each domain, reflecting their original 'no answer' distribution. Accordingly, the ratio of 'no answer' examples can vary between 0.05 and 0.4.

In Table 10, we present the results of several encoder-only models trained on the balanced and imbalanced QA datasets. Our observations indicate that while the impact on the average is relatively low, the worst drops are much more prominent when the prior shift occurs. We analyzed the results and found a simple explanation for this.

The increased diversity across different domains leads to greater variability in absolute performance. For example, domains with a higher proportion of 'no answer' questions, which are typically more challenging, tend to have a lower absolute in-domain performance (or lower cross-domain performance when shifting to those domains). This increased variability leads to more pronounced discrepancies between in-domain and cross-domain performance, resulting in larger drops. Although the average drop remains consistently low – because sometimes the shift is to an easier domain, compensating drops when the shift is to a harder domain – the worst drops are significantly more pronounced. This experiment effectively illustrates that as the shift becomes more prominent (affecting both X and Y variables), there is a notable increase in performance variability across domains, leading to more substantial drops in some cases.

C.8 Scenarios Statistical Validation

In §3.3 we introduce four possible scenarios of domain shift: Classic, Unobserved, Observed, and No Challenge. Each scenario is determined by the sign of the SD and the TD of a single domain shift. We present the proportion of each scenario in Figure 3, taking into account the results of all domain shifts and all participating models. For all few-shot models, we use 4-shots. Since we conducted experiments with more shots in §C.2, we also include results 8-shots for GPT3.5, Neural, and Mistral, and 16-shots for GPT3.5.

We next validate whether the domain shift has a statistically valid effect on the model performance. Consider that if there is no effect, we would expect the order of $(\mathrm{SS}, \mathrm{TT}, \mathrm{ST})$ to be distributed uniformly. There are six possible sequences, where two belong to the Classic scenario ($\mathrm{ST} < \mathrm{SS} < \mathrm{TT}$ or $\mathrm{ST} < \mathrm{TT} < \mathrm{SS}$), two belong to the No Challenge scenario ($\mathrm{SS} < \mathrm{TT} < \mathrm{ST}$ or $\mathrm{TT} < \mathrm{SS} < \mathrm{ST}$), one belongs to the Observed scenario ($\mathrm{TT} < \mathrm{ST} < \mathrm{SS}$), and one to the Unobserved scenario ($\mathrm{SS} < \mathrm{ST} < \mathrm{TT}$). Under the assumption of uniform distribution, each sequence would have a probability of $1/6$.

We conduct a Chi-square test with a significance threshold of 0.05, applying a Bonferroni correction for multiple comparisons (14 tests in total, adjusting the significance level to 0.0036). The test results show that all P-values are below 0.001, except for the QA task in few-shot models, which is at 0.004. These findings confirm that the effect of domain shift on model performance is statistically significant. Notably, the results highlight that the demonstration domain used in few-shot models influences the cross-domain performance.

D The Domain Robustness Benchmark: Technical Details

D.1 Preprocessing

Sentiment Analysis (SA) We removed links from texts since they were tokenized to dozens of tokens and significantly increased the input length.

Question Answering (QA) We split the documents of each category (and their corresponding questions) into train, development, and test sets.

Question Generation (QG) The input is a concatenation of the document and the answer, separated by the "answer:" token.

Abstractive Summarization (AS) Since the summaries of the Webis-TLDR-17 dataset were automatically extracted and not verified, they may be of low quality. After manually examining dozens of them, we decided to use only summaries that have 15-60 words, and at least 75% of them appear in the post.

Title Generation (TG) After manually examining examples, we found many reviewers misused the title option: They started writing a long review in the title and continued it in the body box. We therefore decided to use only titles that have 5-20 words, and at least 75% of them appear in the grounding review.

Motivation					
Practical □	*Cognitive*	*Intrinsic*	*Fairness*		
Generalisation type					
Compositional	*Structural*	*Cross Task*	*Cross Language*	*Cross Domain* □	*Robustness* □
Shift type					
Covariate □	*Label*	*Full*	*Assumed*		
Shift source					
Naturally occuring □	*Partitioned natural*	*Generated shift*	*Fully generated*		
Shift locus					
Train–test □	*Finetune train–test* □	*Pretrain–train*	*Pretrain–test* □		

Table 11: Categorization of our study according to the GenBench taxonomy (Hupkes et al., 2023).

D.2 Technical Domain Shift Assumptions

As discussed in §3, a domain can be characterized by various attributes such as topic, style, syntax, and medium. When one of these attributes changes, the joint distribution $P(X, Y)$ changes, and a domain shift occurs. In developing our benchmark, we grounded it in technical assumptions aimed at facilitating a controlled experimental analysis, as detailed §D.2. One of these assumptions is to focus on natural topic shifts (although other factors are likely to change as well, such as the style and syntax). This contrasts with other studies that explore synthetic shifts, such as adversarial attacks, challenge sets, or transitions to datasets from different data-generating processes (e.g., having other annotation guidelines).

Our rationale was to isolate and control a single variable and facilitate a "controlled experiment" approach, allowing for a precise and clear analysis and characterization of the DR challenge. In line with this objective, we have established the following technical assumptions:

1. Our benchmark focuses on natural topic shift, e.g., training an NLP model on book reviews and applying it to kitchen product reviews. In contrast to many other works (Hendrycks et al., 2020; Miller et al., 2020; Koh et al., 2021; Yuan et al., 2023), our natural topic shift allows us to avoid complexities that arise when the shift is a byproduct of constructing a challenge set or transitioning to another dataset that was constructed by a different data generating process (e.g., different annotation guidelines).

2. Each task consists of several domains, facilitating a more comprehensive and accurate estimation of average performance and performance degradation.

3. For each task, all the domains have the same number of training examples, enabling its use as a source and as a target domain. Moreover, it helps mitigate (non-DR) biases that may arise when transitioning from a domain with sufficient training data to a domain with scarce labeled data.

4. We try to reduce the effect of the prior shift, i.e., changes in $P(Y)$: For classification tasks, we create balanced datasets (for QA, same ratio of 'no answer'), while for generation tasks, we sample examples with similar output length distributions. In Appendix §C.7, we discuss experiments exploring changes in $P(Y)$ upon a domain shift. We found that this variation leads to increased performance variability across domains, resulting in larger worst drops but minimally impacting the average drop (because shifts to easier domains compensate for shifts to harder domains).

While our assumptions simplify the domain shift, we argue that if the DR challenge exists under these assumptions (and it does), then it will definitely exist more severely when our assumptions are violated and a complex shift occurs. Researchers who wish to focus on a specific type of prior shift (e.g., unbalanced domains) can easily use our publicly available benchmark to construct more challenging setups.

D.3 Domains for Few-shot Experiments

As mentioned in Section 5, due to the high costs associated with API calls, we limit our presentation of few-shot results to only three domains for each task, rather than encompassing all five or six domains. In addition, we randomly sample 200 test examples for each target domain. This cost constraint arises from the quadratic increase in the number of experiments relative to the number of domains (for instance, six domains lead to 36 domain-shift setups, whereas three domains result in just 9). Additionally, the extended input length, a consequence of augmenting it with multiple demonstrations, also contributes to this decision. For a fair comparison, we present results for the same three domains for both few-shot and fine-tuned models in Table 5. The specific domains we focus on are:

- SA - Airline, Beauty, Books.
- NLI - Fiction, Telephone, Traval.
- AB - Device, Laptops, MAMs.
- QA - History, Science, Society.
- QG - Geography, History, Science.
- AS - Fitness, LoL, Relationships.
- TG - Beauty, Books, DVDs.

E Implementation Details

Our experiments are conducted in the PyTorch and HuggingFace frameworks and optimize the fine-tuning models with the AdamW optimizer. An exception is the OpenAI's models, which were run via their paid API service and their results are correct as of January 2023. The data, results and code are provided in the project repository.

Hyperparameter Tuning For each model and source domain, we initially conduct hyperparameter tuning, selecting the optimal set based on the source domain's validation set. Subsequently, we evaluate the model across all target domains. In the hyperparameter tuning phase for classification models, we experiment with the following learning rates: [1e-5, 5e-5, 1e-4] and batch sizes: [4, 8, 16, 64] and 10 epochs. For generation models, we explore learning rates of [1e-3, 5e-4, 1e-4, 5e-5, 1e-5], use a batch size of 64 and 15 epochs.

Instructions and Demonstrations For each test example from a target domain, the LLM input includes a system prompt detailing the task instruction, and a user prompt presenting the example. In few-shot setups, we augment this with additional demonstrations (input and target) from the source domain. This involves adding extra user-assistant turns: the user turn shows the demonstration input, and the assistant turns present the demonstration target (label). We randomly select demonstrations from the source domain's training set for each test. In classification tasks, for $N > 1$-shots the prompt includes demonstrations of all labels. Task instructions and prompt examples are in Appendix E.1. In addition, to not exceed the maximum input length of several models, we truncate the maximum length of each demonstration to 256 tokens (but no truncation was applied to the test example). Please see L2 in §8 for other prompting attempts.

The classification results of few-shot LLMs are based on "long-form generation". Notice that we mentioned the labels in the prompt and asked the LLM to respond only with them (see examples §E.1). The LLMs we used in our study underwent SFT with instructions and, therefore, almost always followed our instructions and responded with a label (we also used temperature=0.0). When they did not–such as when they began generating an explanation before or after stating the label–we extracted the first mentioned label (lowercase). We found labels 100% of the time (except for CodeLlama-70b).

E.1 Prompts

Prompt for SA (Sentiment Analysis)

SYSTEM
You will be provided with a review and asked to classify its sentiment.
You can only response "negative" or "positive".

USER
Review:
[text]

Prompt for NLI (Multi-NLI)

SYSTEM
You will be provided with a premise and a hypothesis and asked to classify their

relationship.
You can only response "entailment", "neutral" or "contradiction".

USER
Premise:
[premise]

Hypothesis:
[hypothesis]

Prompt for AB (ABSA)

SYSTEM
You will be provided with a sequence of words and asked to extract the aspect and the polarity of each word.
You can only response with a sequence of tags corresponding to each word. The tags are: "O", "T-POS", "T-NEG", "T-NEU", where "O" indicates a non aspect word. For example, the answer of: "The good boy", is: "O O T-POS".

USER
Text:
[text]

Prompt for QA (SQuAD v2)

SYSTEM
You will be provided with a context and a question and asked to extract the answer from the context.
You can only response with a copied span of text from the context. If there is no answer, response: "No answer".

USER
Context:
[context]

Question:
[question]

Prompt for QG (Question Generation)

SYSTEM
You will be provided with a context and an answer, and asked to generate a question that would lead to the answer.
You can only response with the question.

USER
Context:
[context]

Answer:
[answer]

Prompt for AS (TL;DR Abstractive Summarization)

SYSTEM
You will be provided with a reddit post and asked to generate a short TL;DR summary of the post that the Redditor might have written at the end of the post.
You can only response with the summary.

USER
Post:
[text]

Prompt for TG (Title Generation)

SYSTEM
You will be provided with a product review and asked to generate a title that the reviewer might have given to the review.
You can only response with the title.

USER
Review:
[text]

Example of 2-shot SA prompt

SYSTEM
You will be provided with a review and asked to classify its sentiment.
You can only response "negative" or "positive".

USER
Review:
[text1]

ASSISTANT
negative

USER
Review:
[text2]

ASSISTANT
positive

USER
Review:
[text]