In-Context Learning (and Unlearning) of Length Biases

Stephanie Schoch Yangfeng Ji
Department of Computer Science
University of Virginia
Charlottesville, VA 22903
{sns2gr,yangfeng}@virginia.edu

Abstract

Large language models have demonstrated strong capabilities to learn in-context, where exemplar input-output pairings are appended to the prompt for demonstration. However, existing work has demonstrated the ability of models to learn lexical and label biases in-context, which negatively impacts both performance and robustness of models. The impact of other statistical data biases remains under-explored, which this work aims to address. We specifically investigate the impact of length biases on in-context learning. We demonstrate that models do learn length biases in the context window for their predictions, and further empirically analyze the factors that modulate the level of bias exhibited by the model. In addition, we show that learning length information in-context can be used to counter the length bias that has been encoded in models (e.g., via fine-tuning). This reveals the power of in-context learning in debiasing model prediction behaviors without the need for costly parameter updates.

1 Introduction

In-context learning (ICL) has emerged as a new ability in large language models (LLMs), representative of a novel learning paradigm (Wei et al., 2022). With in-context learning, an LLM learns to perform an unseen task by seeing a number of demonstrations in the context window (Brown et al., 2020). Whereas previous methods such as fine-tuning update the model parameters to teach the model a desired task, ICL provides the model with input-output pairs as task exemplars directly at inference, with no parameter updates. While the goal of increased task accuracy is the same, the underlying mechanisms contributing to in-context learning are still being understood.

This has motivated a body of work aiming to understand how in-context learning works. Some works have demonstrated simlarities between fine-tuning and in-context learning. For example, instability due to the choice of examples occurs both in few-shot finetuning (Schick and Schütze, 2021; Gao et al., 2021) and in-context learning (Rubin et al., 2022; Liu et al., 2022; Wu et al., 2023). However, other work has shown counterintuitive results when comparing the apparent learning mechanisms of in-context learning and finetuning (Min et al., 2022).

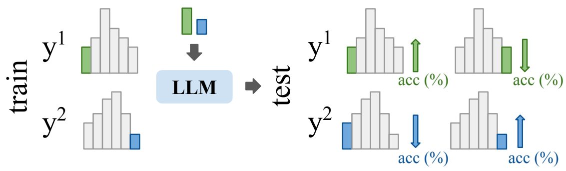


Figure 1: An illustration of our experiment setup and hypothesis. When sampling from the tails of the distribution (left of image), we introduce a data length bias. If the model can learn this shortcut feature in-context, we expect class performance on the data of similar length to be higher than data of the opposite length than what was seen in the context window (right of image).

A key area that is underexplored is whether in-context learning exhibits similar biases to finetuning with regard to statistical data biases. Statistical data biases can be defined as correlations between features and class labels. Under traditional learning paradigms such as fine-tuning, language models can learn exploitable statistical biases in the data. Such biases, or shallow features, can be exploited by a model as discriminatory features when they exhibit biased distributions across classes or are correlated with a specific class. This can lead to overestimates of a model's performance on the underlying task (Poliak et al., 2018; Gururangan et al., 2018).

Prior work has identified length as an exploitable statistical bias in natural inference datasets. For example, in the MultiNLI and SNLI datasets, length has been shown to be a discriminatory feature (Gururangan et al., 2018), and on the ROC story cloze

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 7633–7671

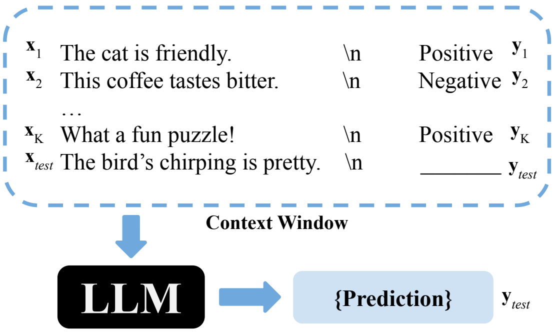


Figure 2: An overview of in-context learning using K input-output demonstrations concatenated to the test input $\{x_{test}, y_{test}\}$.

task choosing the longer ending performs above random baseline (Cai et al., 2017). However, length biases have been largely ignored in prior work on ICL, and some existing studies on which factors impact ICL have treated length as a static variable, selecting examples with similar lengths to test inputs (Min et al., 2022). It is therefore unclear whether models can exploit length biases in the data under an in-context learning setting (similar to finetuning) or whether length is another factor with counterintuitive tendencies.

In this work, we perform a series of empirical studies to investigate the ability of LLMs to learn statistical data biases in the context window during ICL (Figure 1). This has been studied in the finetuning literature, yet is underexplored in the ICL literature. We demonstrate empirically the ability of LLMs to learn length biases in-context. In the sections to follow, we analyze which factors influence these results, and we show the efficacy of ICL in debiasing finetuned models. Our results show that ICL can introduce biases to LLMs that negatively influence task performance. Specifically, our contributions are as follows:

1. We empirically demonstrate the ability of a range of LLM families to learn length biases in-context.

2. We investigate the influence of number of examples, number of model parameters, and class-length difference on how models learn biases.

3. We show that ICL can debias a model that contains existing length biases.

2 Background

In-context learning In-context learning is an emergent ability of LLMs that enables pre-trained models to learn an unseen task using a set of exemplars concatenated in the context window (see Figure 2). Formally, given a test example x, in-context learning concatenates K demonstration examples to the task instruction I, where $S = \{x_i, y_i\}_{i=1}^{K}$ denotes the example set. The performance of in-context learning, however, is highly dependent on both the selected examples (Rubin et al., 2022; Liu et al., 2022; Wu et al., 2023; Ye et al., 2023) and their orderings (Lu et al., 2022; Chen et al., 2023), resulting in performance variation from nearly random to comparable with finetuned models.

In-Context Learning & Bias While in-context learning has shown significant potential as a way to extract relevant information from an LLM and align the model with user expectations, it has also exhibited brittleness to an assortment of factors. These include selected examples (Rubin et al., 2022; Liu et al., 2022; Wu et al., 2023; Ye et al., 2023) and their orderings (Lu et al., 2022; Chen et al., 2023), which have recently been categorized under the umbrella of demonstration biases (Li et al., 2024).

Beyond demonstration bias, instability of ICL has been attributed to biases in the model toward predicting certain answers due to majority label bias, recency bias, and common token bias (Zhao et al., 2021). Correspondingly, several works have looked at identifying and mitigating label bias (Zhao et al., 2021; Fei et al., 2023) (Fei et al., 2023) with respect to lexical information, and Ali et al. (2024) have looked at mitigating "copy bias", where LLMs copy lexical information from demonstrations rather than learning underlying task information.

However, statistical data biases such as length information have been largely ignored in the in-context learning literature, yet received wide attention in the natural language inference literature with respect to traditional finetuned models (McCoy et al., 2019b; Poliak et al., 2018; Cai et al., 2017; Gururangan et al., 2018). Our work bridges this gap by looking at in-context learning with relation to a specific statistical bias: length bias.

3 Experiment Setup

[1](Dagan et al., 2006; Bar Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009)

Category	Dataset	Task	#Train	#Val
Inference	QNLI (Wang et al., 2018)	Natural Language Inference	105k	5.46k
	RTE [1]	Natural Language Inference	2.49k	277
	WNLI (Levesque et al., 2011)	Natural Language Inference	635	71
	HANS (McCoy et al., 2019a)	Natural Language Inference	30k	30k
Single Sentence	SST-2 (Socher et al., 2013)	Sentiment Analysis	67.3k	872
Paraphrase Detection	MRPC (Dolan et al., 2004)	Paraphrase Detection	3.67k	408
	PAWS-X_{EN} (Yang et al., 2019)	Paraphrase Detection	49.4k	2k

Table 1: Datasets used in our experiments. We use the distributions available from Huggingface (Lhoest et al., 2021), and use the respective validation sets to measure performance. Dataset descriptions can be found in Table 3.

Model	Parameters
LLaMa 3 (Dubey et al., 2024)	8B
LLaMa 2 (Touvron et al., 2023)	7B
Mistral (Jiang et al., 2023)	7B
OPT (Zhang et al., 2022)	6.7B
GPT-Neo (Black et al., 2021)	2.7B

Table 2: Models used in section 4.

In this section, we describe the experiment setup used in our analyses.

Datasets We use 7 binary classification datasets, representing natural language inference, sentiment analysis, and paraphrase detection tasks. As we sample from the tails of the length distributions, binary classification is ideal for our setting. For each dataset, we utilize the splits available from Huggingface. Dataset statistics are provided in Table 1, with detailed descriptions in subsection A.1. To count the length of each input, we use the NLTK word-tokenize package (Bird and Loper, 2004) rather than the LLM-specific tokenizers, to maintain consistency across experiments. Prompts are adapted from (Gao et al., 2023) and provided in subsection A.2.

Models Experiments in section 4 are run using five models from different LLM families, listed in Table 2. The selected models vary in size from 2.7B parameters to 8B parameters. Notably, the upper bound of the parameter range is due to our resource constraint, as each experiment is run using a single NVIDIA A100 GPU. For experiments in section 5 and section 6, we use a subset of these models, Llama3 and GPT-Neo. For experiments in subsection 5.1, we use the OPT model family.

Other Details Following Min et al. (2022), unless otherwise noted, all experiments use $k = 16$ demonstrations. For finetuning experiments, we use $k = 200$ finetuning examples. To minimize the impact of ordering effects, each result represents the mean of 4 trials, with standard deviation shown using error bars. Results are all run on the full validation split of each dataset.

In section 4, we investigate whether LLMs can learn length biases in-context. To further analyze these results, in section 5 we look at the impact of model parameter size, number of examples, and length distribution. Finally, section 6 demonstrates the utility of ICL to debias finetuned models that exhibit length biases.

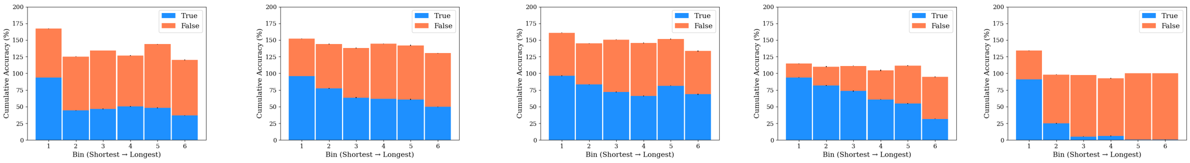
4 Length Biases in Finetuning and ICL

In this section, we investigate the question ***do models learn length biases in-context?*** We demonstrate empirically the ability of LLMs to learn length biases in-context.

4.1 Method

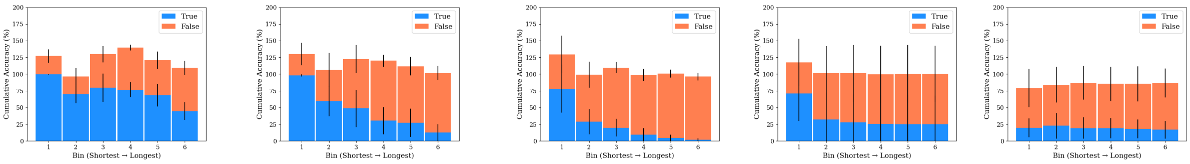
Consider a dataset $D = \{(x_i, y_i)\}_{i=1}^n$ that contains n training instances. In this work, we consider binary classification datasets, where $Y = \{y^1, y^2\}$. We aim to introduce a distributional bias in the input lengths with respect to class. To introduce a length bias in k demonstrations drawn from D, we sample from the tails of each class's input length distribution. Specifically, we sample the top-$\frac{k}{2}$ examples belonging to y^1 and the bottom-$\frac{k}{2}$ from y^2 (and vice versa). This effectively produces a "worst-case scenario" in maximizing the distance between the classes under the given length distributions.

To provide a baseline for comparison, we compare against finetuning. Specifically, we finetune each model (using LoRA adapters (Hu et al., 2022))



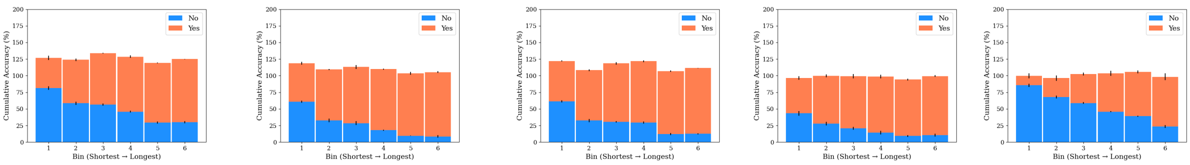
(a) Llama3 8B (b) Llama2 7B (c) Mistral 7B (d) OPT 6.7B (e) GPT Neo 2.7B

Figure 3: In-context learning validation performance across different models on the Hans dataset. For each graph, y_1 (Blue) was sampled from the short instances, and y_2 (Orange) was sampled from the long instances.



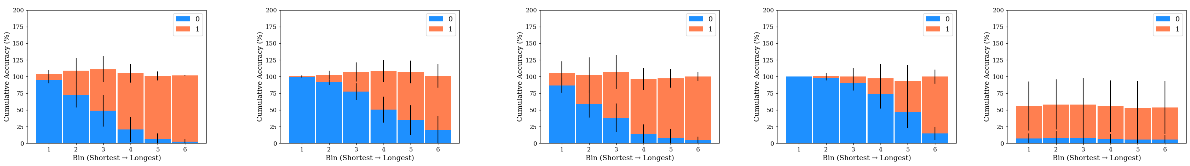
(a) Llama3 8B (b) Llama2 7B (c) Mistral 7B (d) OPT 6.7B (e) GPT Neo 2.7B

Figure 4: Finetuning validation performance across different models on the Hans dataset. For each graph, y_1 (Blue) was sampled from the short instances, and y_2 (Orange) was sampled from the long instances.



(a) Llama3 8B (b) Llama2 7B (c) Mistral 7B (d) OPT 6.7B (e) GPT Neo 2.7B

Figure 5: In-context learning validation performance across different models on the PAWS-X$_{\text{EN}}$ dataset. For each graph, y_1 (Blue) was sampled from the short instances, and y_2 (Orange) was sampled from the long instances.



(a) Llama3 8B (b) Llama2 7B (c) Mistral 7B (d) OPT 6.7B (e) GPT Neo 2.7B

Figure 6: Finetuning validation performance across different models on the PAWS-X$_{\text{EN}}$ dataset. For each graph, y_1 (Blue) was sampled from the short instances, and y_2 (Orange) was sampled from the long instances.

on $k = 200$ training instances selected using the same procedure as above. As an additional baseline, for all experiments, we compare against randomly sampling the demonstrations and finetuning examples.

For results, we utilize a binning procedure. Specifically, we bin the validation set based on length, with $b = 6$ bins. In this setting, bin 1 represents the shortest 16.67% of validation instances and bin 6 represent the longest 16.67% validation instances across both classes. If a model has learned a length bias, for the validation class with the training set drawn from the shortest instances, we expect performance on bin 1 to be higher than bin 6, and vice versa for the validation class where the training set is drawn from the longest instances.

As a performance measurement, we report the sum of the individual accuracy from each class. As there may be a slight imbalance across classes in each bin, reporting individual class accuracy rather than the percentage of the entire bin ensures we account for class imbalances across bins.

4.2 Results

We report results on HANS and PAWS-X$_{\text{EN}}$ under finetuning (Figure 4 and Figure 6) and ICL (Figure 3 and Figure 5), where y_1 demonstrations were sampled from short instances and y_2 demonstrations were sampled from long instances. y_1 and y_2 correspond to the Blue and Orange bars, respectively. Our results show decreased performance on validation examples that do not have a similar length as the demonstrations belonging to each respective class. This indicates that models can pick up length biases in-context. Additional results can be found in the Appendix.

5 Analysis of Influencing Factors

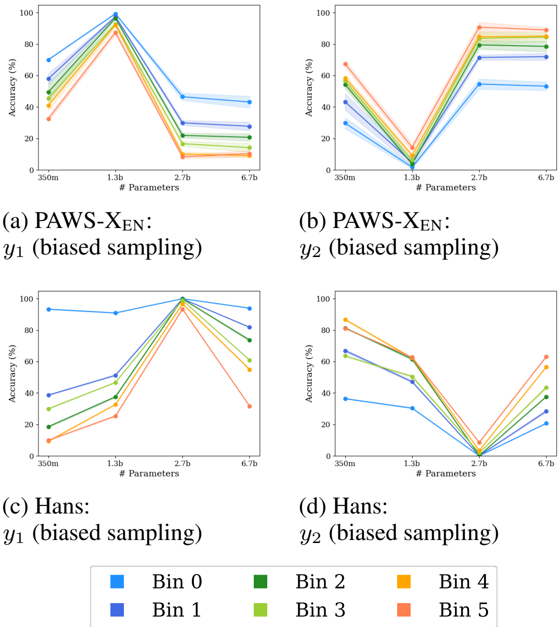
In this section, we investigate a further question of ***what factors influence how LLMs learn length biases in-context?*** We find that increased numbers of examples can exacerbate learned biases, and models across a range of sizes can learn length biases. Further, we find that length bias can be learned from as little as a few tokens of difference in average length between classes.

5.1 Number of Model Parameters

Existing work has suggested that the number of model parameters influences the ability of models to learn in-context, with larger models performing better (Milios et al., 2023; Lu et al., 2022). In this section, we investigate whether the number of parameters also influences the ability of models to learn length biases in-context. For example, if larger models are better at learning in-context, are they more susceptible or more resilient to learning statistical biases in the data?

We use the OPT model family (Zhang et al., 2022) across $p = \{350\text{M}, 1.3\text{B}, 2.7\text{B}, 6.7\text{B}\}$ parameters with $k = 16$ in-context examples. Note that the parameter count is upper-bounded based on computational resources. We use the procedure described in section 4 to introduce a length bias in the in-context demonstrations.

Results We report results on HANS and PAWS-X_{EN} in Figure 7. Notably, both datasets are designed to be challenging (see subsection A.1 for descriptions). Remaining datasets and conditions are reported in the Appendix. While we do observe length bias across varying model parameter sizes, there is not a consistent pattern of increased or decreased bias with increased model parameter sizes. Accordingly, we observe a dataset-model dependence with regard to the degree of length bias a model may learn.



(a) PAWS-X_{EN}: y_1 (biased sampling)

(b) PAWS-X_{EN}: y_2 (biased sampling)

(c) Hans: y_1 (biased sampling)

(d) Hans: y_2 (biased sampling)

Figure 7: Validation performance across different numbers of model parameters using the OPT model family, on the PAWS-X_{EN} and Hans datasets. In this figure, in-context examples from y_2 were sampled from long instances, and in-context examples from y_1 were sampled from short instances. Each subfigure shows results on the validation instances in the respective class, with Bin 0 containing the shortest demonstrations and Bin 5 containing the longest demonstrations. Additional results can be found in the Appendix.

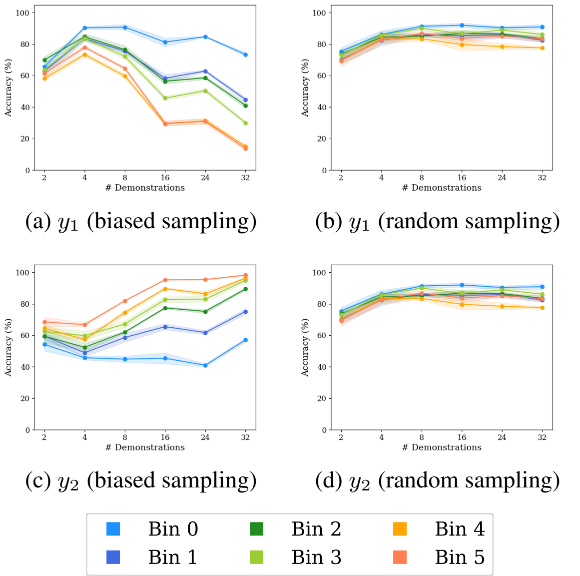
5.2 Number of Examples

The performance of ICL when using various numbers of examples has been studied in prior work (Wu et al., 2023; Min et al., 2022; Lu et al., 2022). As such, we investigate the sensitivity of LLM's ability to learn length bias across different numbers of in-context examples.

We use $k = \{2, 4, 8, 16, 24, 32\}$ in-context examples on the datasets in Table 1 using Llama3 8B and GPT Neo 2.7B. Following the procedure from section 4, we select the longest $\frac{k}{2}$ examples from y^1 and shortest $\frac{k}{2}$ examples from the y^2 (and vice versa), thereby introducing a bias in the length distribution of inputs across classes.

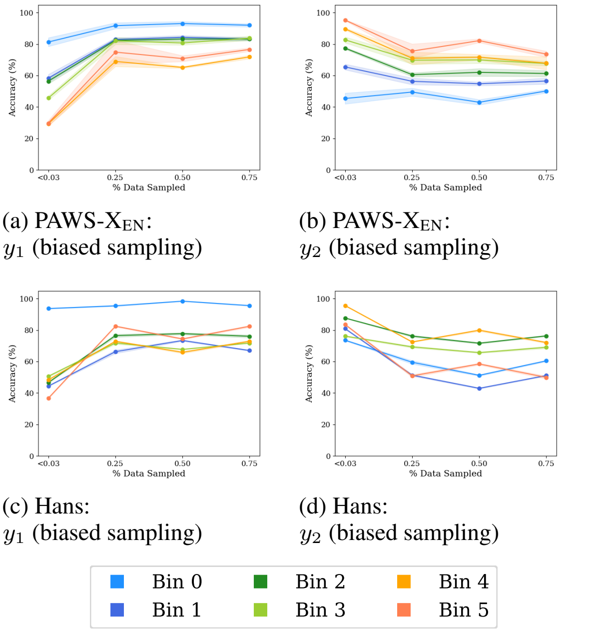
Results We report on the PAWS-X_{EN} dataset using Llama3 (8B) in Figure 8 and provide the average length for each class in subsection A.3. Our results show that models can generally begin learning biases around 8 in-context examples, with the effect typically strengthening with increased numbers of examples.

Longer context models are gaining traction,



(a) y_1 (biased sampling) (b) y_1 (random sampling) (c) y_2 (biased sampling) (d) y_2 (random sampling)

Figure 8: Validation performance of Llama3 (8B) across different numbers of demonstrations on the PAWS-X$_{\text{EN}}$ dataset. In this figure, in-context examples from y_2 were sampled from long instances, and in-context examples from y_1 were sampled from short instances. Each subfigure shows results on the validation instances in the respective class. The conditions that introduce length bias in the context window (a and c subfigures) demonstrate a larger spread between performance on short and long validation instances, indicating greater potential to learn bias with longer contexts. Additional results can be found in the Appendix.



(a) PAWS-X$_{\text{EN}}$: y_1 (biased sampling) (b) PAWS-X$_{\text{EN}}$: y_2 (biased sampling) (c) Hans: y_1 (biased sampling) (d) Hans: y_2 (biased sampling)

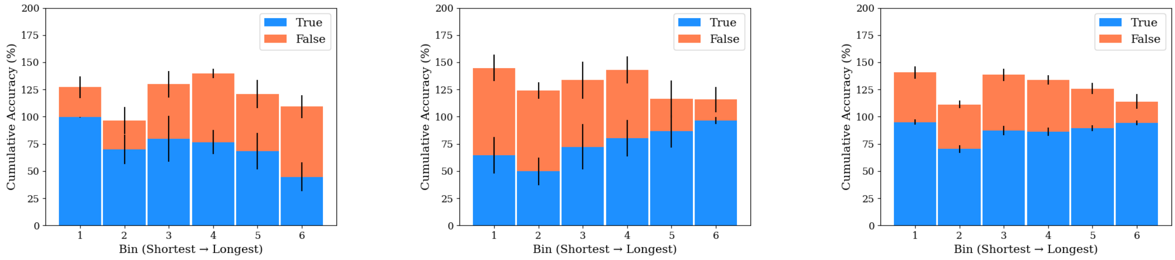
Figure 9: Validation performance across different data sampling percentages using Llama 3 (8B), on the PAWS-X$_{\text{EN}}$ and Hans datasets. In this figure, in-context examples from y_2 were sampled from long instances, and in-context examples from y_1 were sampled from short instances. Each subfigure shows results on the validation instances in the respective class, with Bin 0 containing the shortest demonstrations and Bin 5 containing the longest demonstrations. Additional results can be found in the Appendix.

with a recent line of work focusing on scaling in-context learning to larger numbers of demonstrations. Longer contexts can increase performance and decrease sensitivity to ordering effects (Cai et al., 2023; Hao et al., 2022), and contexts (beginning around $k = 8$) can decrease model calibration errors, where calibration is a measure of the faithfulness of a model's predictive uncertainty (Zhang et al., 2024). Our results demonstrate that longer contexts exhibit a greater potential for statistical data biases being learned in-context, and underscore the need for balanced selection methods with regard to potential data biases.

5.3 Difference in Average Demonstration Length Between Classes

Given the results from the previous section, we investigate whether the difference of the average demonstration length between classes influences the ability of LLMs to identify a length bias. We keep the number of examples consistent at $k = 16$ and sample from $p = \{0.25\%, 0.5\%, 0.75\%\}$ of the longest and shortest inputs for each class, respectively. For example, if y_1 is the long class, we sample $\frac{k}{2}$ instances from the longest $p = \{0.25\%, 0.5\%, 0.75\%\}$ of the instances belonging to y_1, and sample $\frac{k}{2}$ instances from the shortest $p = \{0.25\%, 0.5\%, 0.75\%\}$ of the instances belonging to y_2.

Results We report results using Llama 3 (8B) on the PAWS-X$_{\text{EN}}$ dataset in Figure 9, where 0.03 corresponds to an approximate sampling percentage from the previous experiment setup. We observe a length bias across different sampling percentages, despite the decrease in difference between average class lengths (see Table 7). Intuitively, as the difference increases, so does the spread between performance across bins of different lengths. This indicates that while models can learn length biases from a few tokens difference (approximately 3 tokens on HANS under 0.75 sampling), the biases are amplified in the model as they are amplified in the demonstrations.

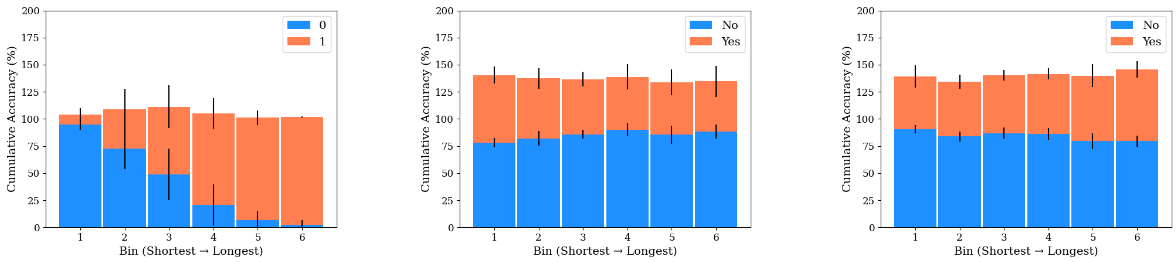


(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

Figure 10: HANS validation set performance on a finetuned Llama 3 (8B) model exhibiting a length bias (see Figure 10a for finetuning performance prior to intervention). Figure 10b and Figure 10c (respectively) show results on two debiasing conditions: ICL demonstrations ($k = 16$) sampled from the opposite lengths from what the model saw during finetuning (i.e. y_1 long demonstrations, y_2 short demonstrations), and random sampling.

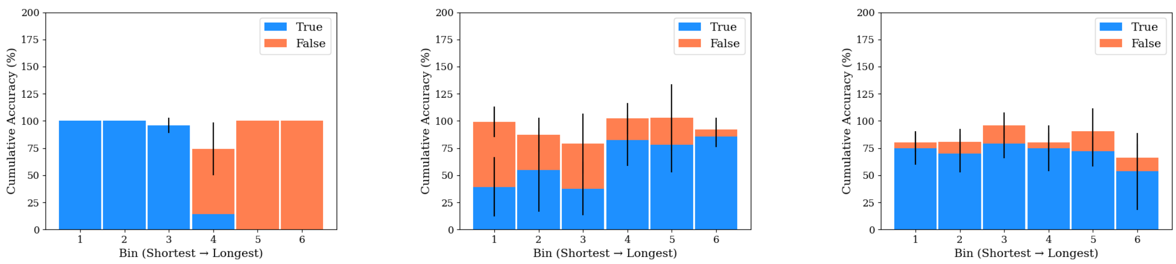


(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

Figure 11: PAWS-X_{EN} validation set performance on a finetuned Llama 3 (8B) model exhibiting a length bias (see Figure 11a for finetuning performance prior to intervention). Figure 11b and Figure 11c (respectively) show results on two debiasing conditions: ICL demonstrations ($k = 16$) sampled from the opposite lengths from what the model saw during finetuning (i.e. y_1 long demonstrations, y_2 short demonstrations), and random sampling.



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

Figure 12: WNLI validation set performance on a finetuned Llama 3 (8B) model exhibiting a length bias (see Figure 12a for finetuning performance prior to intervention). Figure 12b and Figure 12c (respectively) show results on two debiasing conditions: ICL demonstrations ($k = 16$) sampled from the opposite lengths from what the model saw during finetuning (i.e. y_1 long demonstrations, y_2 short demonstrations), and random sampling.

6 ICL for Debiasing Finetuned Models

In-context learning eliminates the need for expensive model parameter updates incurred when finetuning. However, it is often the case that a model may have encoded biases picked up from the pretraining and/or finetuning. As our previous experiments show that in-context learning can learn length information, a natural extension is to question whether ICL can be used to "unlearn" or mitigate previously learned length biases. In this section, we answer the question ***can ICL be used as an intervention to mitigate biases in finetuned models?***

We use previously finetuned models from section 4 and modify the length distribution to try to counteract the biases. Specifically we experiment with two conditions: 1) using in-context demonstrations drawn from the opposite tail of the length distribution from what was seen during finetuning, and 2) using randomly sampled in-context demonstrations. We again use Llama3 8B and GPT-Neo 2.7B for these experiments using the datasets in Table 1.

Results Results using Llama3 (8B) on HANS, PAWS-X$_{\text{EN}}$, and WNLI are reported in Figure 10, Figure 11 and Figure 12, respectively. We find that random sampling was able to counteract the bias, essentially "unlearning" the finetuned bias. This suggests that balanced data sampling is critical to minimize the likelihood of learning biases in-context. Further, if a dataset is balanced, random sampling may be sufficient. However, if a dataset contains shortcut features, more sophisticated sampling methods to mitigate the bias may be necessary.

Moreover, our results suggest balanced sampling over showing the models demonstrations of opposite lengths with respect to the finetuned bias. Specifically, the models learned the bias in the length information in the context window, regardless of how it contradicted what was seen during finetuning. One possible explanation is that the task may be implicitly encoded during pretraining and ICL extracts the knowledge (Xie et al., 2022; Min et al., 2022), however, further study is warranted on whether knowledge-extraction from ICL overrides knowledge-gain during finetuning.

7 Discussion

In this work, we investigate the impact of demonstration length bias on model performance when learning in-context. Under this setting, length is a statistical data bias, where the shallow feature (length) is correlated with class labels. It is important to make the distinction, however, between length as a linguistic feature containing information relevant to the underlying task, and length as an artifact of the data collection protocol. For example, length is an informative syntactic feature in classifying truthfulness vs. deceptiveness (Yancheva and Rudzicz, 2013) and authorship detection (Yule, 1939), however, length biases can also arise from artifacts reflective of heuristics used by human data annotators (Gururangan et al., 2018). Our work pertains to the latter settings where length is an artifact rather than a task-informative natural language feature.

Which variables have the greatest impact on models learning length biases? In subsection A.3, we observe that when varying the number of in-context examples, the distance between classes is greater with fewer in-context examples. However, the amount of bias increases with increased numbers of examples. Further, while we observe that bias increases with increased length difference, we still observe learned bias when class length difference is reduced to as few as 3 tokens on the HANS dataset. This suggests that a key factor in learning bias is the number of examples the model sees. Additionally, our results suggest that any model can learn bias, and model parameter size is not necessarily correlated with increased ability to learn biases in-context.

8 Conclusion

In this work, we empirically investigated the ability of LLMs to learn length biases under an in-context learning paradigm. Our results show that LLMs can learn statistical biases in the data. We further show the impact of model parameter sizes, number of examples, and class length difference on length biases. Finally, we demonstrate the potential for ICL to be used as a tool to debias fine-tuned models with previously learned length biases.

9 Limitations

While we test models up to 8B parameters, we acknowledge a limitation of this work is the parameter threshold due to available computational resources. We believe our results scale to larger models.

Acknowledgments

We thank the reviewers for their insightful feedback, which has helped us improve this paper significantly. We also thank Jason Stock for his helpful feedback and suggestions. This research was supported in part by NSF SaTC #2124538 and NSF III #2007492.

References

Ameen Ali, Lior Wolf, and Ivan Titov. 2024. Mitigating copy bias in in-context learning through neuron pruning. *arXiv preprint arXiv:2410.01288*.

Roy Bar Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second PASCAL recognising textual entailment challenge.

Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. 2009. The fifth PASCAL recognizing textual entailment challenge.

Steven Bird and Edward Loper. 2004. NLTK: The natural language toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 214–217, Barcelona, Spain. Association for Computational Linguistics.

Sid Black, Leo Gao, Phil Wang, Connor Leahy, and Stella Biderman. 2021. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow.

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

Tianle Cai, Kaixuan Huang, Jason D Lee, and Mengdi Wang. 2023. Scaling in-context demonstrations with structured attention. *arXiv preprint arXiv:2307.02690*.

Zheng Cai, Lifu Tu, and Kevin Gimpel. 2017. Pay attention to the ending:strong neural baselines for the ROC story cloze task. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 616–622, Vancouver, Canada. Association for Computational Linguistics.

Yanda Chen, Chen Zhao, Zhou Yu, Kathleen McKeown, and He He. 2023. On the relation between sensitivity and accuracy in in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 155–167, Singapore. Association for Computational Linguistics.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pages 177–190. Springer.

William Dolan, Chris Quirk, Chris Brockett, and Bill Dolan. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas

Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Maheswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vítor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Yu Fei, Yifan Hou, Zeming Chen, and Antoine Bosselut. 2023. Mitigating label biases for in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14014–14031, Toronto, Canada. Association for Computational Linguistics.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2023. A framework for few-shot language model evaluation.

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

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9. Association for Computational Linguistics.

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

Yaru Hao, Yutao Sun, Li Dong, Zhixiong Han, Yuxian Gu, and Furu Wei. 2022. Structured prompting: Scaling in-context learning to 1,000 examples. *arXiv preprint arXiv:2212.06713*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

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. *Preprint*, arXiv:2310.06825.

Hector J Levesque, Ernest Davis, and Leora Morgenstern. 2011. The Winograd schema challenge. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*, volume 46, page 47.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, Joe Davison, Mario Šaško, Gunjan Chhablani, Bhavitvya Malik, Simon Brandeis, Teven Le Scao, Victor Sanh, Canwen Xu, Nicolas Patry, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Clément Delangue, Théo Matussière, Lysandre Debut, Stas Bekman, Pierric Cistac, Thibault Goehringer, Victor Mustar, François Lagunas, Alexander Rush, and Thomas Wolf. 2021. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lvxue Li, Jiaqi Chen, Xinyu Lu, Yaojie Lu, Hongyu Lin, Shuheng Zhou, Huijia Zhu, Weiqiang Wang, Zhongyi Liu, Xianpei Han, and Le Sun. 2024. Debiasing in-context learning by instructing LLMs how to follow demonstrations. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 7203–7215, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114, Dublin, Ireland and Online. Association for Computational Linguistics.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019a. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *CoRR*, abs/1902.01007.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019b. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Aristides Milios, Siva Reddy, and Dzmitry Bahdanau. 2023. In-context learning for text classification with many labels. In *Proceedings of the 1st GenBench Workshop on (Benchmarking) Generalisation in NLP*, pages 173–184, Singapore. 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*, pages 11048–11064, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671, Seattle, United States. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352, Online. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

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

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Zhiyong Wu, Yaoxiang Wang, Jiacheng Ye, and Lingpeng Kong. 2023. Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1423–1436, Toronto, Canada. Association for Computational Linguistics.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*.

Maria Yancheva and Frank Rudzicz. 2013. Automatic detection of deception in child-produced speech using syntactic complexity features. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 944–953, Sofia, Bulgaria. Association for Computational Linguistics.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3687–3692, Hong Kong, China. Association for Computational Linguistics.

Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. 2023. Compositional exemplars for in-context learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 39818–39833. PMLR.

G Udny Yule. 1939. On sentence-length as a statistical characteristic of style in prose: With application to two cases of disputed authorship. *Biometrika*, 30(3/4):363–390.

Hanlin Zhang, YiFan Zhang, Yaodong Yu, Dhruv Madeka, Dean Foster, Eric Xing, Himabindu Lakkaraju, and Sham Kakade. 2024. A study on the calibration of in-context learning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6118–6136, Mexico City, Mexico. Association for Computational Linguistics.

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

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

A Appendix

A.1 Datasets

A.2 Prompts

A.3 Other Details

Binary NLI Datasets	
Dataset	**Description**
QNLI (Wang et al., 2018)	The Stanford Question Answering Dataset. A corpus of question-sentence pairs, with context sentences extracted from Wikipedia and questions written by a human annotator. The model is tasked with determining whether the context contains the answer to (entails) the question.
RTE [2]	The Recognizing Textual Entailment (RTE) datasets. A corpus constructed from annual textual entailment challenges based on news and Wikipedia text.
WNLI (Levesque et al., 2011)	The Winograd Schema Challenge. A corpus of reading comprehension sentence pairs, where ambiguous pronouns are replaced with each possible referent. The task is to predict if the substituted sentence is entailed by the original sentence.
HANS (McCoy et al., 2019a)	Heuristic Analysis for NLI Systems. A corpus of challenging premise and hypothesis pairs designed to target evaluation of lexical overlap, sub-sequence, and constituent heuristics.
Single-Sentence Datasets	
Dataset	**Description**
SST-2 (Socher et al., 2013)	The Stanford Sentiment Treebank. A corpus of sentences extracted from movie reviews, with human judgments of positive or negative sentiment.
Similarity & Paraphrase Detection Datasets	
Dataset	**Description**
MRPC (Dolan et al., 2004)	The Microsoft Research Paraphrase Corpus. A corpus of sentence pairs extracted from online news sources. Human raters judged semantic equivalence.
PAWS-X_{EN} (Yang et al., 2019)	Paraphrase Adversaries from Word Scrambling - Cross-lingual. A corpus of challenging paraphrase and non-paraphrase pairs created using data from Wikipedia. Sentence pairs were generated using controlled word swapping and back translation to ensure high lexical overlap, and human raters judged semantic equivalence. Our experiments utilize the English data split.

Table 3: Dataset descriptions.

Binary NLI Datasets	
Dataset	**Prompt**
QNLI (Wang et al., 2018)	{SENTENCE} QUESTION: {QUESTION} TRUE OR FALSE? ANSWER:
RTE [3]	{SENTENCE1} QUESTION: {SENTENCE2} TRUE OR FALSE? ANSWER:
WNLI (Levesque et al., 2011)	{SENTENCE1} QUESTION: {SENTENCE2} TRUE OR FALSE? ANSWER:
HANS (McCoy et al., 2019a)	{PREMISE} QUESTION: {HYPOTHESIS} TRUE OR FALSE? ANSWER:
Single-Sentence Datasets	
Dataset	**Description**
SST-2 (Socher et al., 2013)	{SENTENCE} QUESTION: IS THIS SENTENCE POSITIVE OR NEGATIVE? ANSWER:
Similarity & Paraphrase Detection Datasets	
Dataset	**Description**
MRPC (Dolan et al., 2004)	SENTENCE 1: {SENTENCE1} SENTENCE 2: {SENTENCE2} QUESTION: DO BOTH SENTENCES MEAN THE SAME THING? ANSWER:
PAWS-X$_{\text{EN}}$ (Yang et al., 2019)	SENTENCE 1: {SENTENCE1} SENTENCE 2: {SENTENCE2} QUESTION: DO BOTH SENTENCES MEAN THE SAME THING? ANSWER:

Table 4: Prompts used in our experiments.

Dataset	Condition	Class	# Examples					
			2	4	8	16	24	32
QNLI	Random	True	51.33	42.5.00	47.67	44.81	48.48	45.49
		False	48.60	36.5	49.94	47.70	46.86	48.44
	False-L	True	14.00	14.00	14.75	15.38	15.75	16.06
		False	227.00	207.50	191.50	175.00	165.75	159.50
	True-L	True	446.00	445.40	445.00	358.63	305.92	276.00
		False	15.00	15.50	15.75	15.88	16.00	16.25
RTE	Random	True	60.00	59.33	61.25	66.57	63.02	65.77
		False	44.00	57.71	53.00	63.06	56.10	63.36
	False-L	True	17.00	17.50	18.5	19.63	20.17	20.75
		False	277.00	253.00	233.75	216.75	206.83	200.75
	True-L	True	195.00	194.00	192.00	187.25	183.67	181.19
		False	16.00	17.50	18.75	19.75	20.75	21.50
WNLI	Random	True	37.75	45.50	41.07	41.33	38.20	39.62
		False	41.75	38.17	39.50	35.18	36.65	36.72
	False-L	True	19.00	19.50	19.75	20.38	20.83	21.25
		False	92.00	91.50	87.75	82.38	79.67	77.13
	True-L	True	93.00	91.50	89.50	84.88	82.42	80.63
		False	19.00	19.00	19.50	20.13	20.58	21.00
HANS	Random	True	18.80	19.25	21.50	20.21	20.17	20.91
		False	20.67	21.38	21.22	21.31	21.41	21.05
	False-L	True	13.00	13.00	13.00	13.00	13.00	13.00
		False	27.00	27.00	27.00	27.00	27.00	27.00
	True-L	True	27.00	27.00	27.00	27.00	27.00	27.00
		False	15.00	15.00	15.00	15.00	15.00	15.00
SST-2	Random	True	21.00	21.00	20.85	19.21	19.62	19.30
		False	22.33	18.00	19.16	18.00	17.78	19.16
	False-L	True	10.00	10.00	10.00	10.00	10.00	10.00
		False	61.00	61.00	60.50	60.00	59.58	59.19
	True-L	True	62.00	61.50	61.00	60.00	59.08	58.44
		False	10.00	10.00	10.00	10.00	10.00	10.00
MRPC	Random	True	59.00	57.00	59.33	58.67	59.13	60.22
		False	57.29	59.91	58.61	63.06	62.06	60.46
	False-L	True	34.00	34.00	34.50	35.13	35.67	36.19
		False	97.00	91.50	88.25	86.00	85.00	84.25
	True-L	True	82.00	81.50	81.25	80.25	79.50	79.00
		False	32.00	32.00	33.00	33.50	34.00	34.69
PAWS-X_{EN}	Random	True	63.33	57.50	58.50	57.84	58.11	58.00
		False	61.60	58.60	58.06	54.31	58.79	59.42
	False-L	True	26.00	27.00	27.50	28.25	28.50	28.63
		False	85.00	85.00	84.50	83.50	83.00	82.50
	True-L	True	86.00	86.00	86.00	85.63	85.25	84.94
		False	26.00	26.00	26.25	27.63	28.33	28.75

Table 5: Average input length (including prompt length) across different numbers of examples.

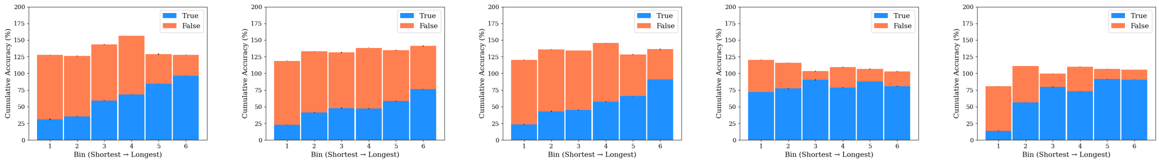
Dataset	Class	Average Length
QNLI	True	50.03
	False	47.50
RTE	True	63.25
	False	64.99
WNLI	True	40.23
	False	37.00
HANS	True	20.53
	False	20.99
SST2	True	28.17
	False	28.93
MRPC	True	57.72
	False	61.16
PAWS-X_{EN}	True	58.63
	False	58.63

Table 6: Average input length for each validation set. Reported input lengths include the prompt length (consistent across all inputs); prompts can be found in subsection A.2.

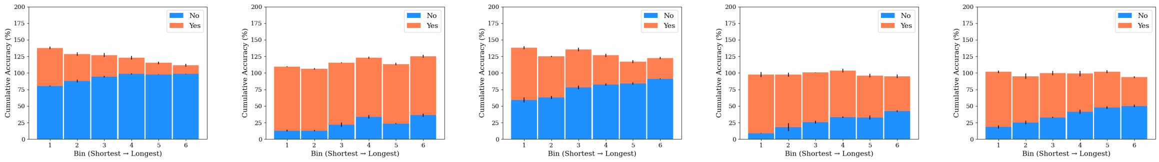
Dataset	Condition	Class	% Sampled		
			0.25	0.50	0.75
QNLI	False-L	True	31.35	36.27	40.91
		False	65.87	57.55	52.05
	True-L	True	69.98	60.16	53.93
		False	31.56	36.33	40.61
RTE	False-L	True	33.05	40.03	46.23
		False	122.98	90.66	76.35
	True-L	True	124.51	91.02	77.78
		False	33.44	39.98	47.21
WNLI	False-L	True	25.23	28.01	31.70
		False	56.01	47.21	40.60
	True-L	True	60.08	48.73	43.12
		False	24.72	28.12	31.05
HANS	False-L	True	16.92	18.72	19.65
		False	23.59	22.65	21.97
	True-L	True	23.29	22.39	21.77
		False	18.17	19.40	20.16
SST2	False-L	True	11.07	12.66	14.92
		False	29.57	23.90	20.36
	True-L	True	30.55	25.14	21.37
		False	11.06	12.26	14.27
MRPC	False-L	True	44.89	49.24	53.12
		False	74.31	69.98	65.67
	True-L	True	70.53	65.61	61.50
		False	47.12	52.35	56.78
PAWS-X_{EN}	False-L	True	44.61	49.51	53.98
		False	72.44	67.80	63.31
	True-L	True	72.58	67.93	63.46
		False	44.56	49.53	54.04

Table 7: Average input length across different sampling bins (by percentage of data sampled from). Reported input lengths include the prompt length (consistent across all inputs); prompts can be found in subsection A.2.

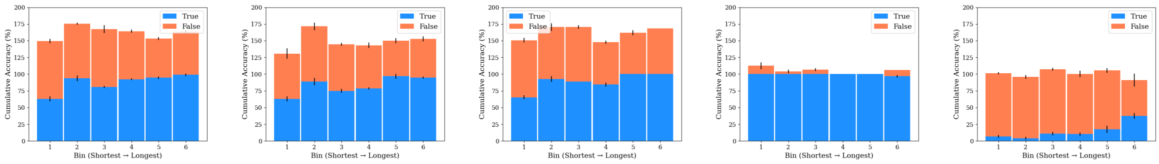
A.4 Additional Length Bias Results



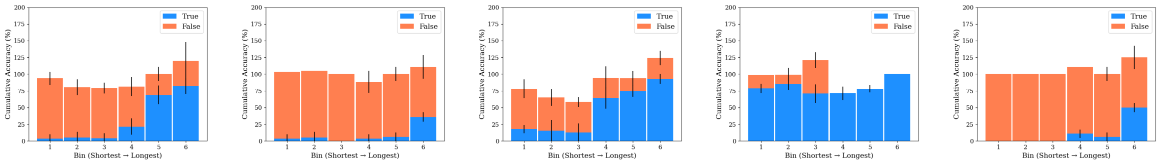
(a) Hans



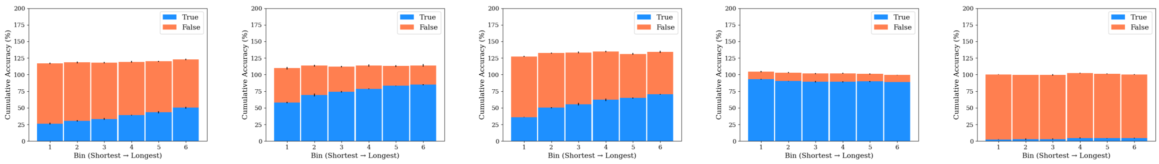
(b) PAWS-X_{EN}



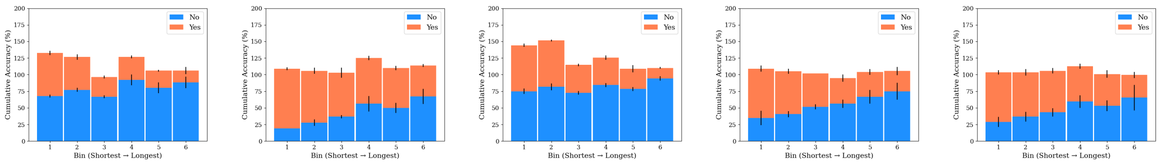
(c) RTE



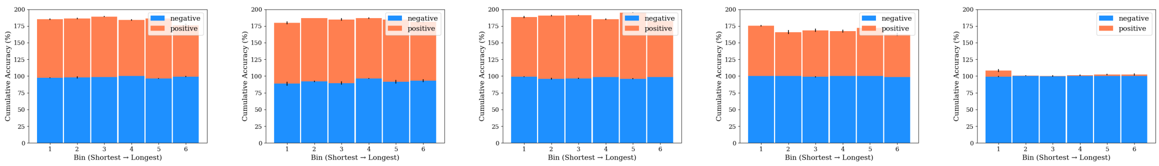
(d) WNLI



(e) QNLI

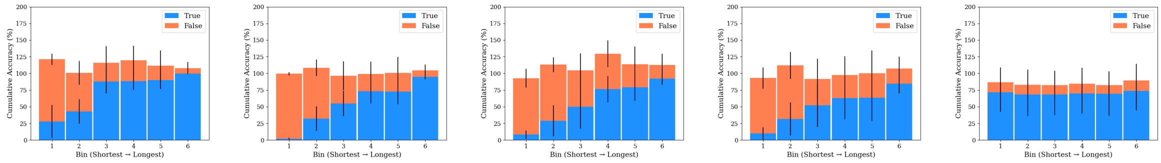


(f) MRPC

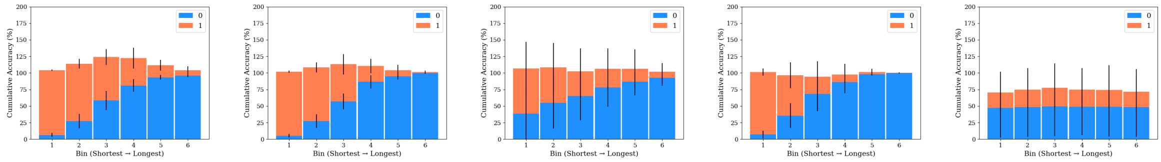


(g) SST-2

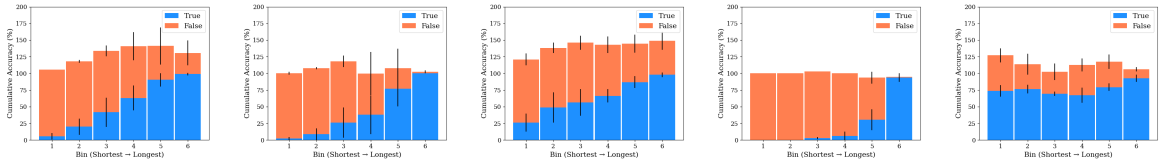
Figure 13: ICL performance of Llama3 8B, Llama2 7B, Mistral 7B, OPT 6.7B, and GPT Neo 2.7B (from left to right) where y_1 (Blue) samples long demonstrations and y_2 (Orange) samples short demonstrations.



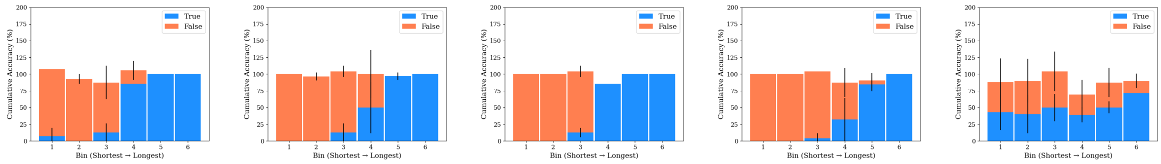
(a) Hans



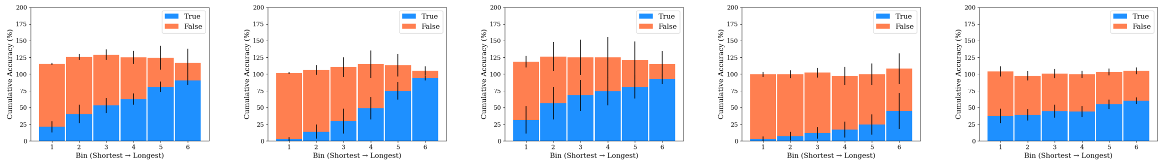
(b) PAWS-X_{EN}



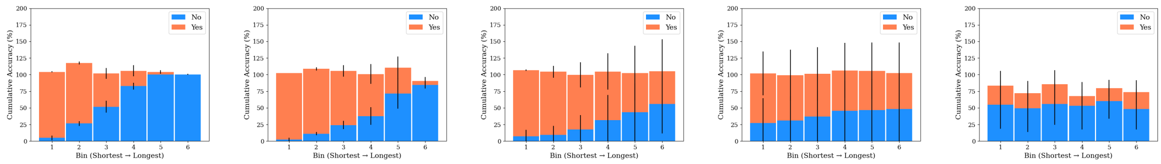
(c) RTE



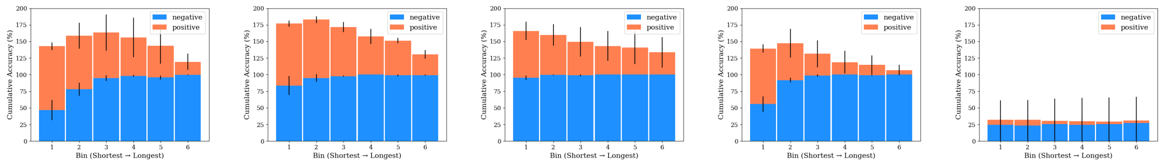
(d) WNLI



(e) QNLI

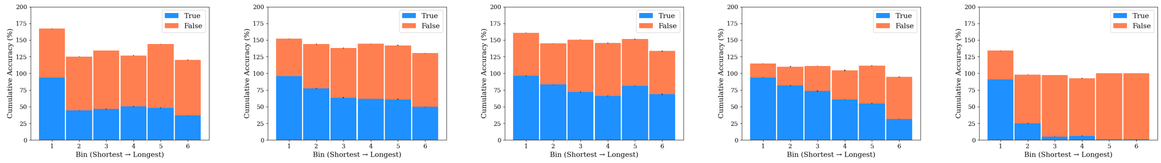


(f) MRPC

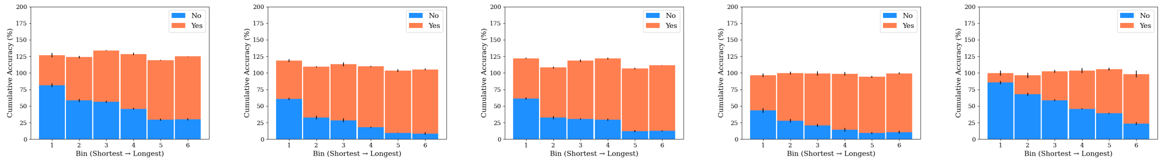


(g) SST-2

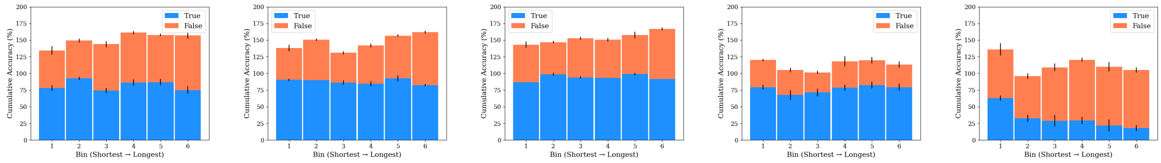
Figure 14: Finetuning performance of Llama3 8B, Llama2 7B, Mistral 7B, OPT 6.7B, and GPT Neo 2.7B (from left to right) where y_1 (Blue) samples long demonstrations and y_2 (Orange) samples short demonstrations.



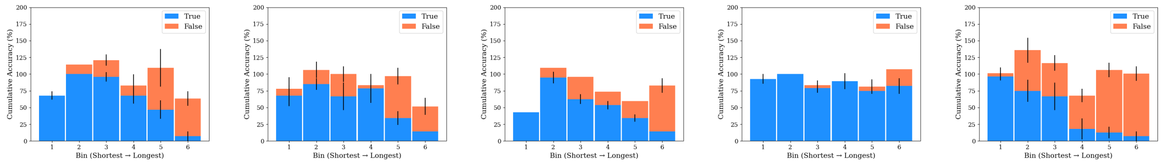
(a) Hans



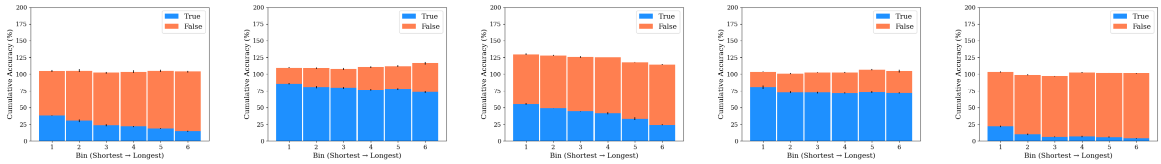
(b) PAWS-X_{EN}



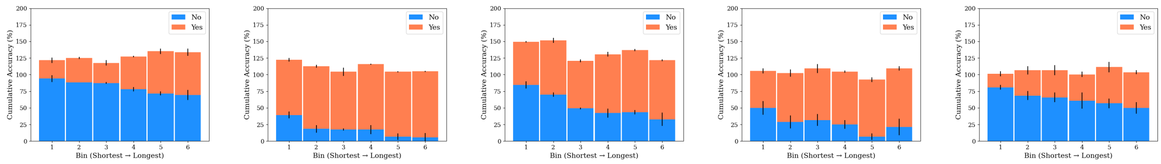
(c) RTE



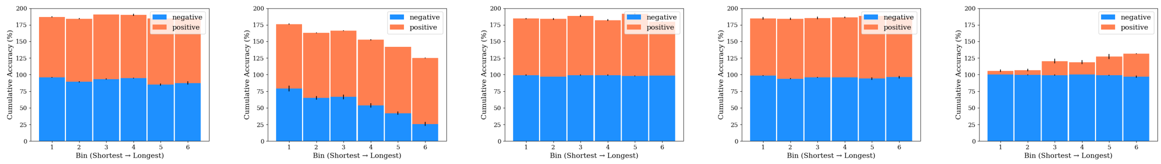
(d) WNLI



(e) QNLI

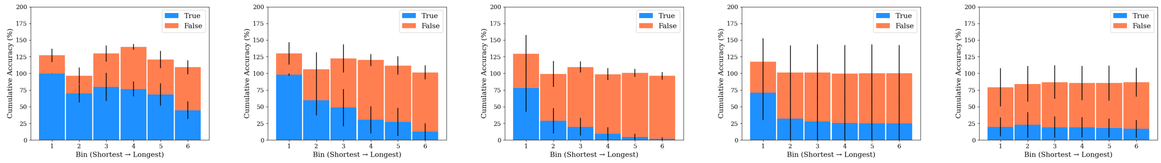


(f) MRPC

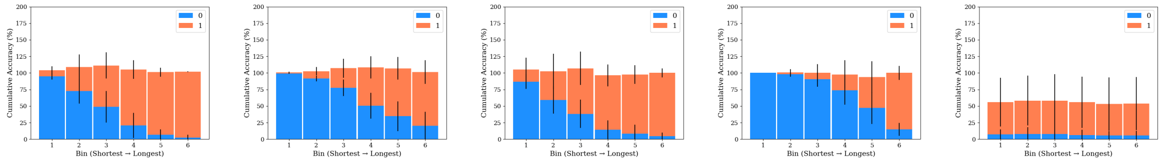


(g) SST-2

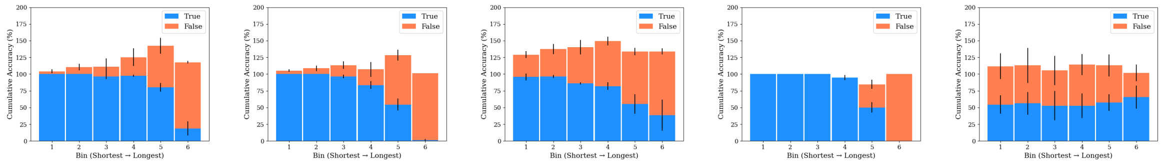
Figure 15: ICL performance of Llama3 8B, Llama2 7B, Mistral 7B, OPT 6.7B, and GPT Neo 2.7B (from left to right) where y_1 (Blue) samples short demonstrations and y_2 (Orange) samples long demonstrations.



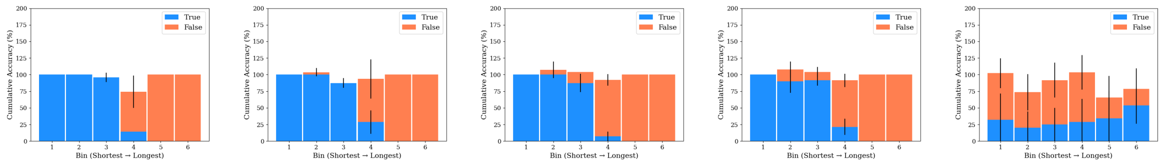
(a) Hans



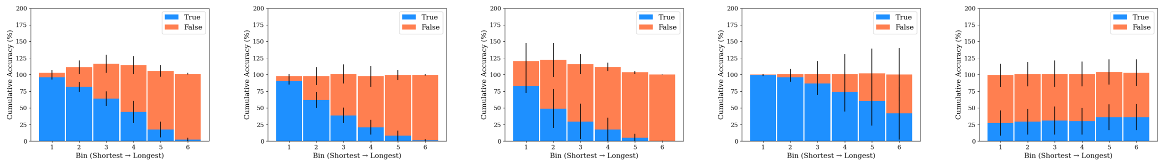
(b) PAWS-X_{EN}



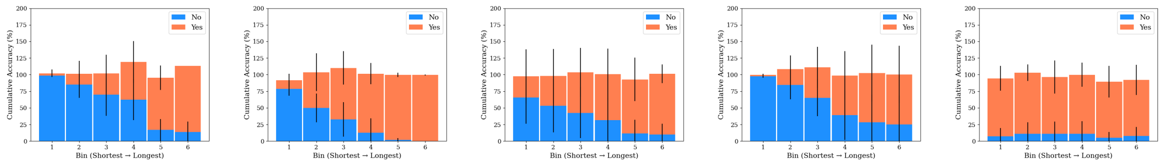
(c) RTE



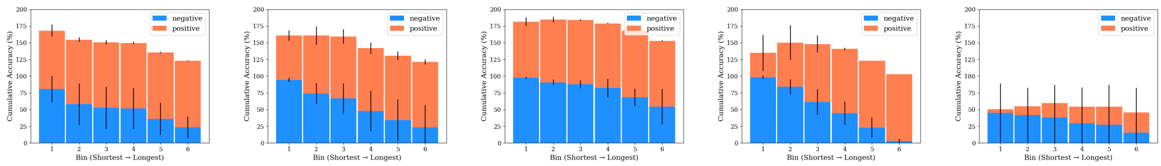
(d) WNLI



(e) QNLI

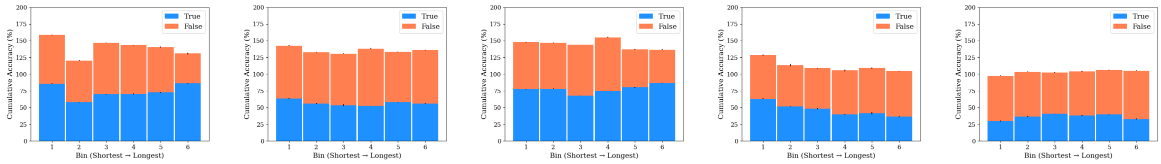


(f) MRPC

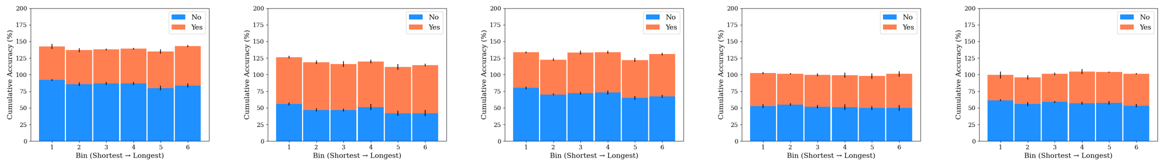


(g) SST-2

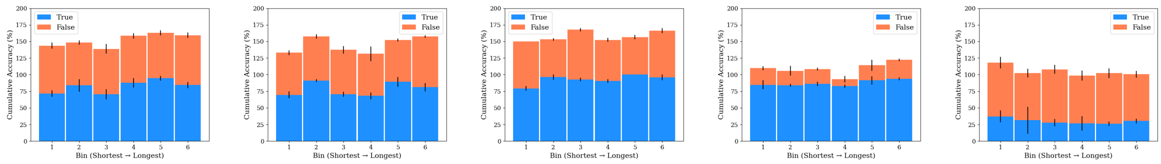
Figure 16: Finetuning performance of Llama3 8B, Llama2 7B, Mistral 7B, OPT 6.7B, and GPT Neo 2.7B (from left to right) where y_1 (Blue) samples short demonstrations and y_2 (Orange) samples long demonstrations.



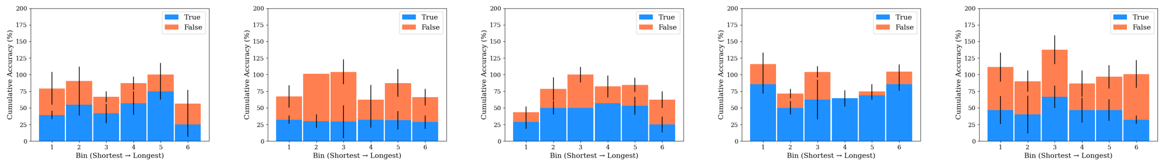
(a) Hans



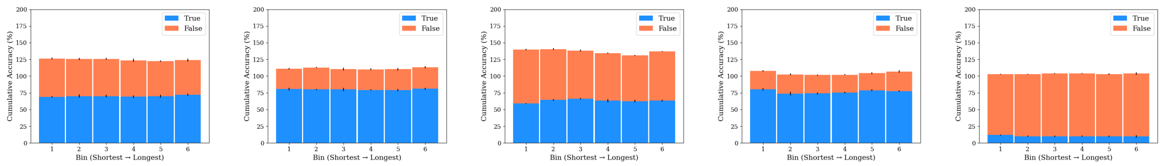
(b) PAWS-X_{EN}



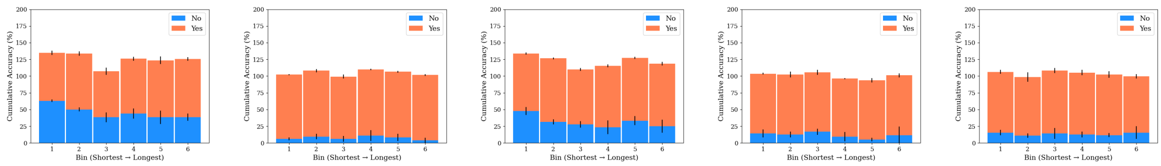
(c) RTE



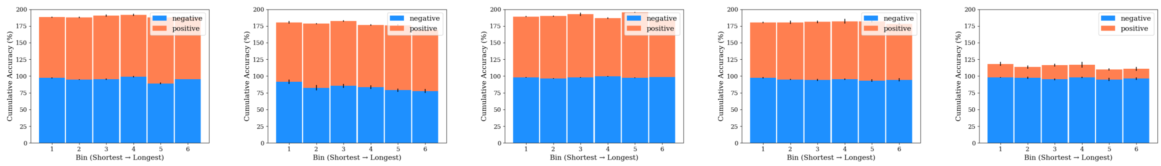
(d) WNLI



(e) QNLI

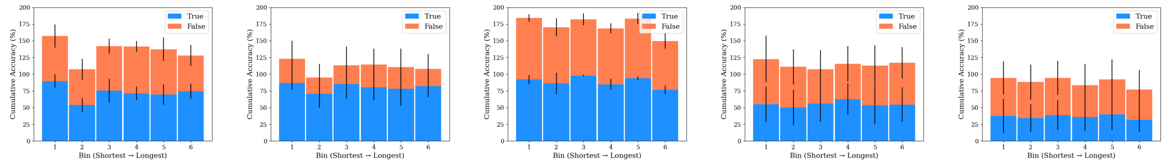


(f) MRPC

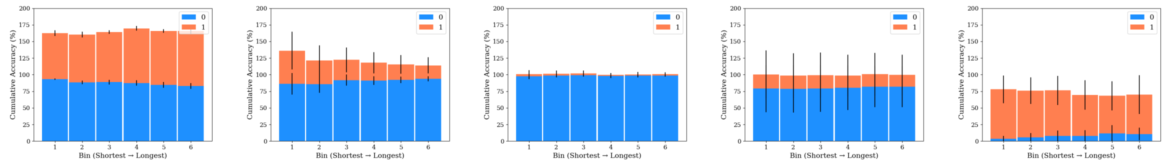


(g) SST-2

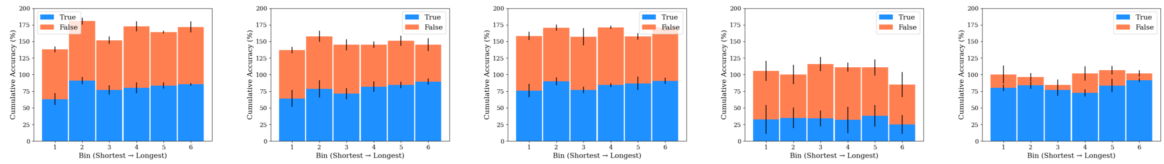
Figure 17: ICL performance of Llama3 8B, Llama2 7B, Mistral 7B, OPT 6.7B, and GPT Neo 2.7B (from left to right) where y_1 (Blue) and y_2 (Orange) are both randomly sampled.



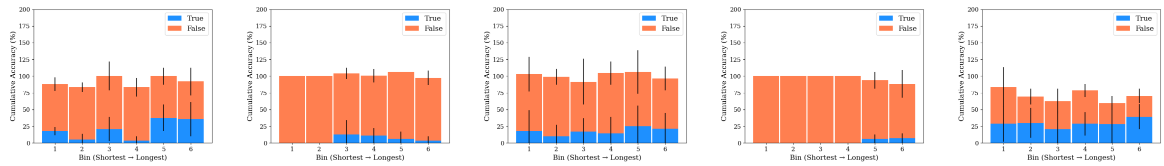
(a) Hans



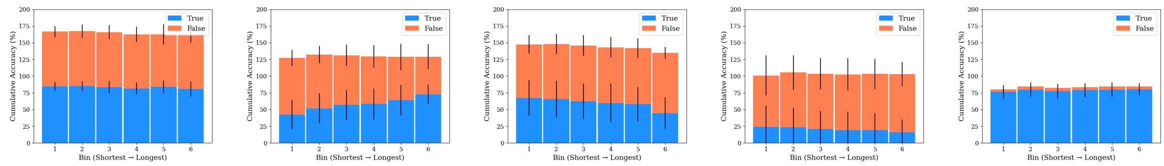
(b) PAWS-X_{EN}



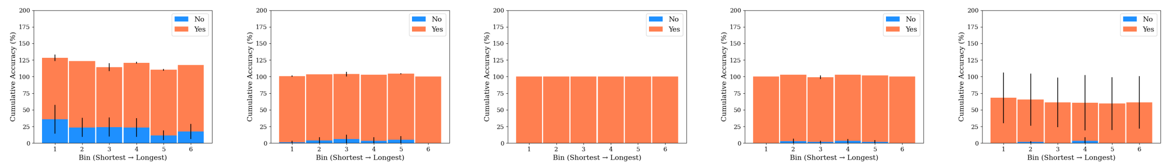
(c) RTE



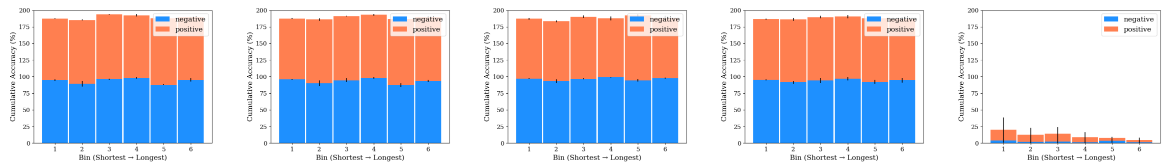
(d) WNLI



(e) QNLI



(f) MRPC



(g) SST-2

Figure 18: Finetuning performance of Llama3 8B, Llama2 7B, Mistral 7B, OPT 6.7B, and GPT Neo 2.7B (from left to right) where y_1 (Blue) and y_2 (Orange) are both randomly sampled.

A.5 Additional Model Parameter (OPT) Results

Each of the following figures shows validation performance when varying the number of model parameters using the OPT model family. Bin 0 contains the shortest demonstrations and Bin 5 contains the longest demonstrations. Each subfigure shows the validation accuracy on a single class when in-context instances belonging to the respective class were sampled from long instances, short instances, and randomly sampled (left to right).

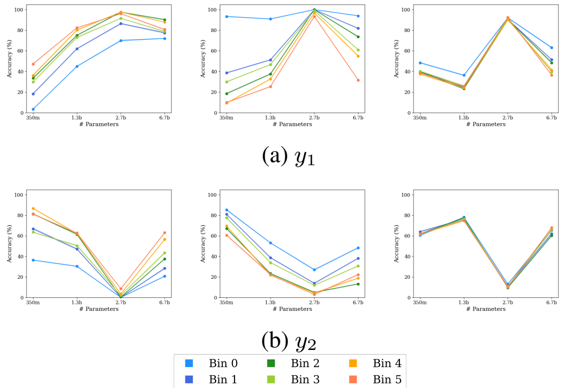


Figure 19: Hans dataset (OPT)

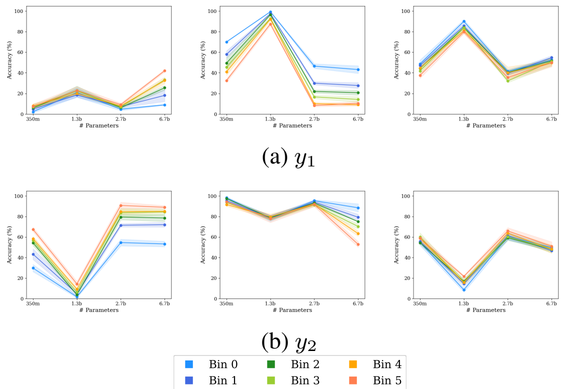


Figure 20: PAWS-X_{EN} dataset (OPT)

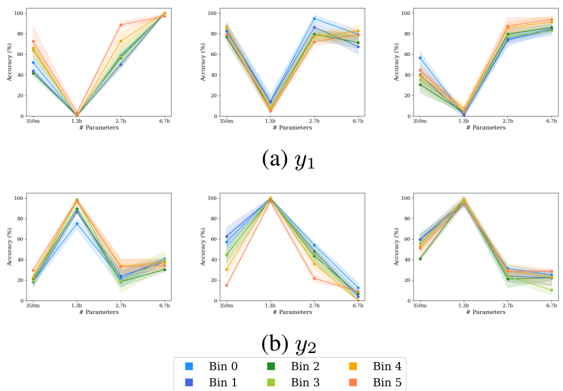


Figure 21: RTE dataset (OPT)

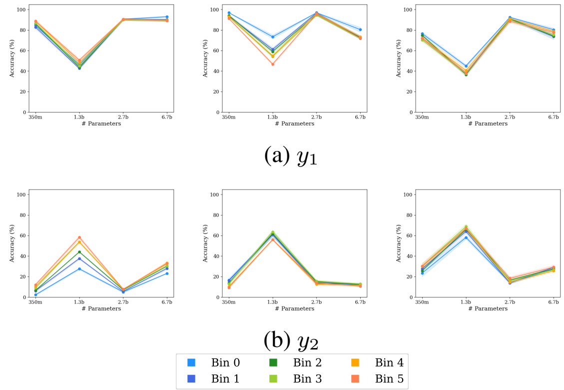


Figure 22: QNLI dataset (OPT)

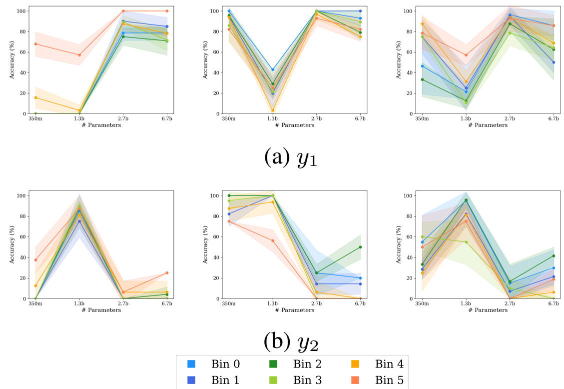


Figure 23: WNLI dataset (OPT)

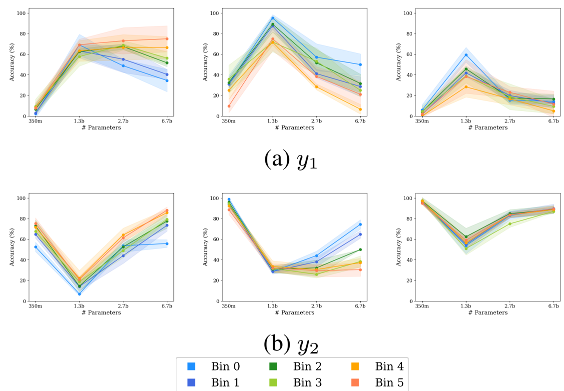


Figure 24: MPRC dataset (OPT)

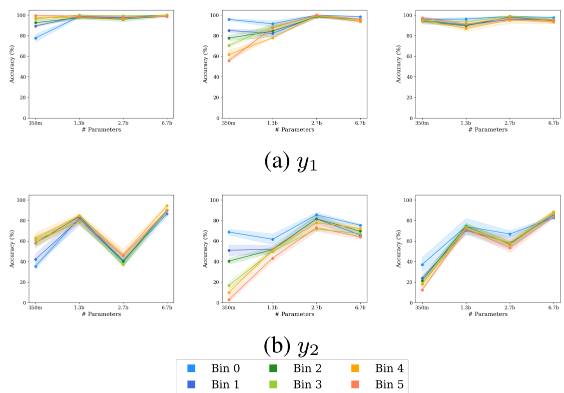
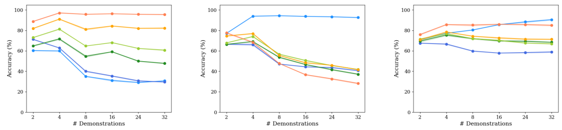


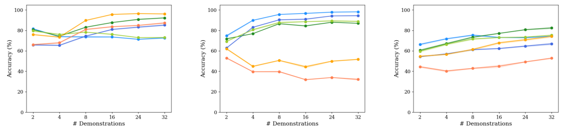
Figure 25: SST-2 dataset (OPT)

A.6 Additional Number of Examples Results

Each of the following figures shows validation performance when varying the number of examples using Llama3 8B and GPT Neo 2.7B. Bin 0 contains the shortest demonstrations and Bin 5 contains the longest demonstrations. Each subfigure shows the validation accuracy on a single class when in-context instances belonging to the respective class were sampled from long instances, short instances, and randomly sampled (left to right).



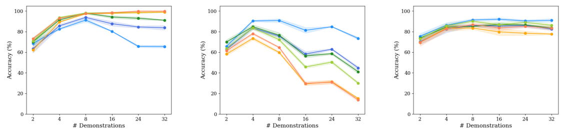
(a) y_1



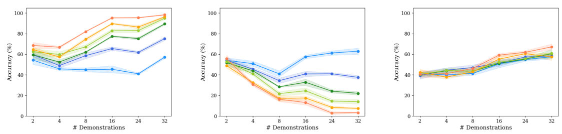
(b) y_2

Bin 0 Bin 2 Bin 4
Bin 1 Bin 3 Bin 5

Figure 26: Hans dataset (Llama 3 8B)



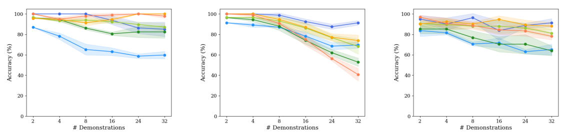
(a) y_1



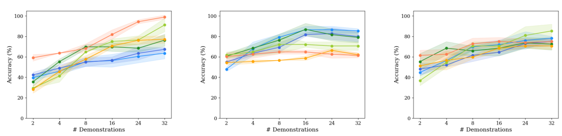
(b) y_2

Bin 0 Bin 2 Bin 4
Bin 1 Bin 3 Bin 5

Figure 27: PAWS-X_{EN} dataset (Llama 3 8B)



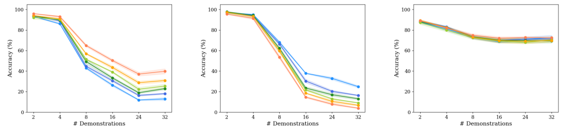
(a) y_1



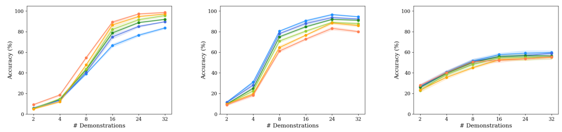
(b) y_2

Bin 0 Bin 2 Bin 4
Bin 1 Bin 3 Bin 5

Figure 28: RTE dataset (Llama 3 8B)



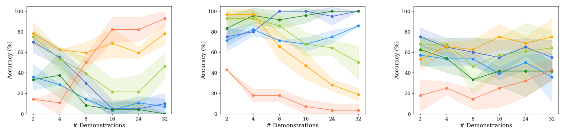
(a) y_1



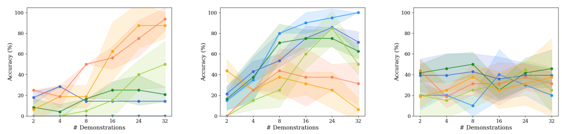
(b) y_2

Bin 0 Bin 2 Bin 4
Bin 1 Bin 3 Bin 5

Figure 29: QNLI dataset (Llama 3 8B)



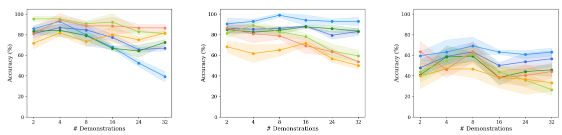
(a) y_1



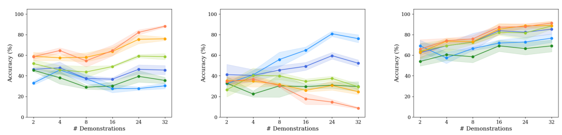
(b) y_2

Bin 0 Bin 2 Bin 4
Bin 1 Bin 3 Bin 5

Figure 30: WNLI dataset (Llama 3 8B)



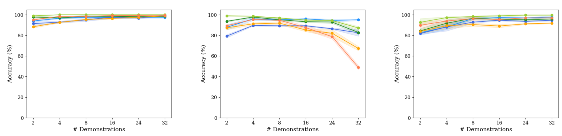
(a) y_1



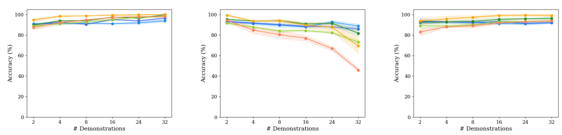
(b) y_2

Bin 0 Bin 2 Bin 4
Bin 1 Bin 3 Bin 5

Figure 31: MPRC dataset (Llama 3 8B)



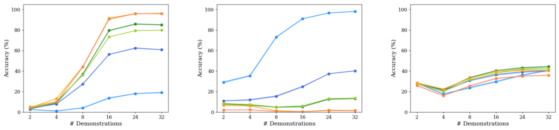
(a) y_1



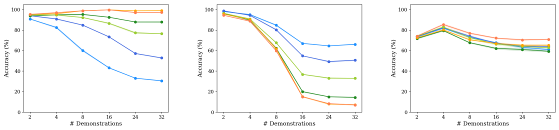
(b) y_2

Bin 0 Bin 2 Bin 4
Bin 1 Bin 3 Bin 5

Figure 32: SST-2 dataset (Llama 3 8B)



(a) y_1



(b) y_2

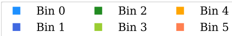
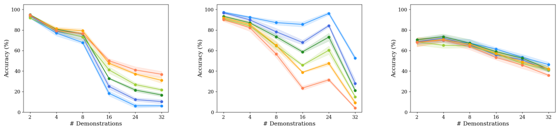
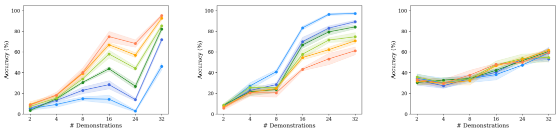


Figure 33: Hans dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

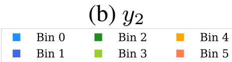
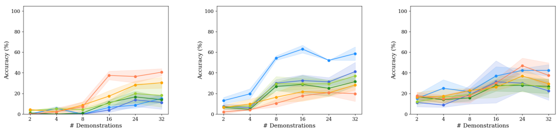
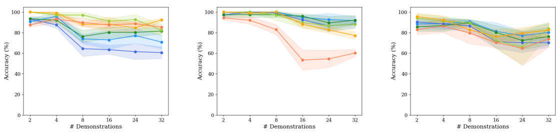


Figure 34: PAWS-X_{EN} dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

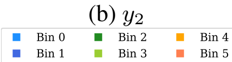
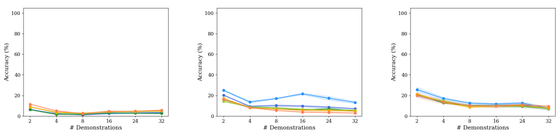
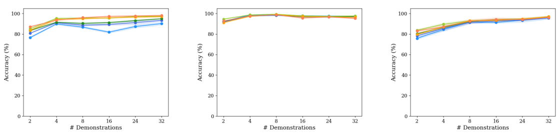


Figure 35: RTE dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

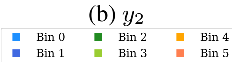
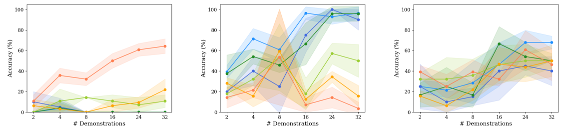
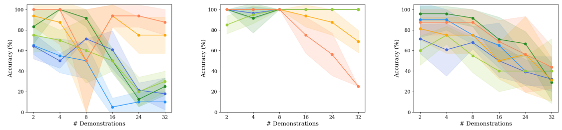


Figure 36: QNLI dataset (GPT Neo 2.7B)

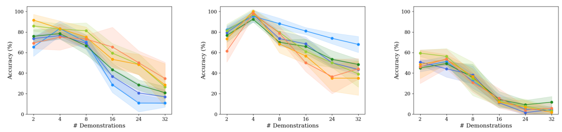


(a) y_1

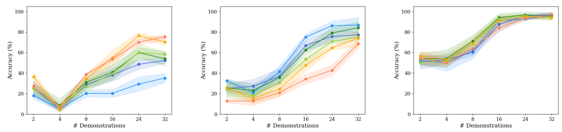


(b) y_2

Figure 37: WNLI dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

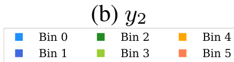
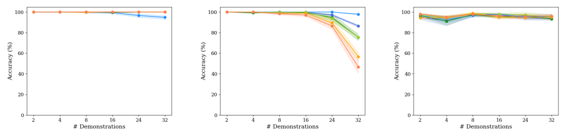
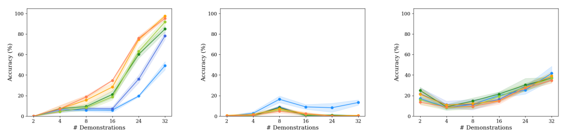


Figure 38: MPRC dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

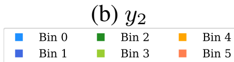
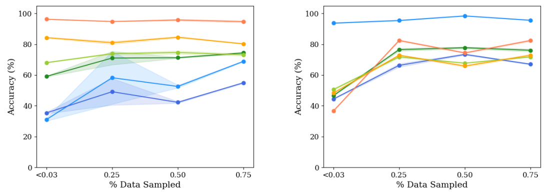


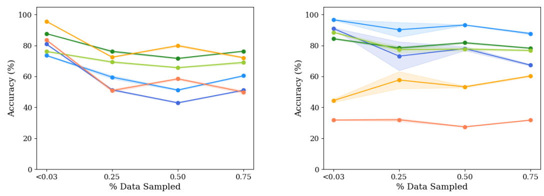
Figure 39: SST-2 dataset (GPT Neo 2.7B)

A.7 Additional Length Difference Results

Each of the following figures shows validation performance when varying the sampling percentage from each class using Llama3 8B and GPT Neo 2.7B. Bin 0 contains the shortest demonstrations and Bin 5 contains the longest demonstrations. Each subfigure shows the validation accuracy on a single class when in-context instances belonging to the respective class were sampled from long instances (left) and short instances (right).

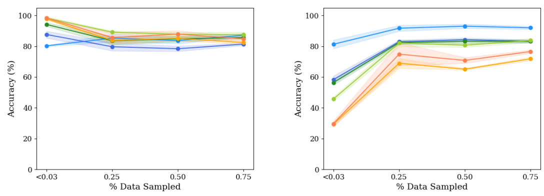


(a) y_1

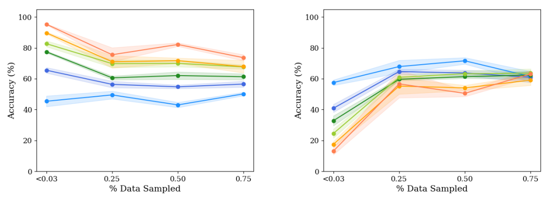


(b) y_2

Figure 40: Hans dataset (Llama 3 8B)

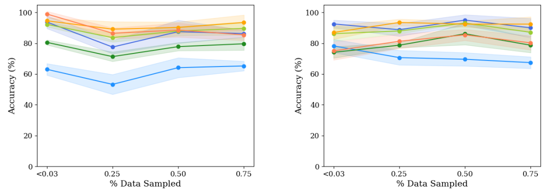


(a) y_1

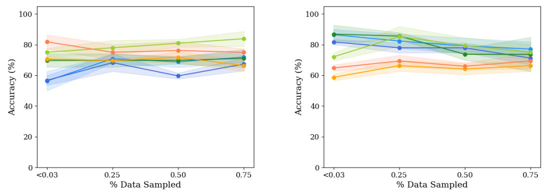


(b) y_2

Figure 41: PAWS-X_{EN} dataset (Llama 3 8B)

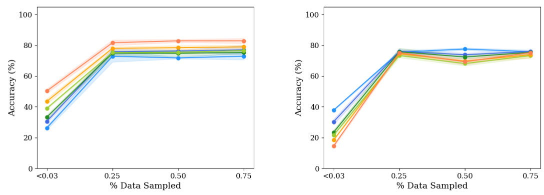


(a) y_1

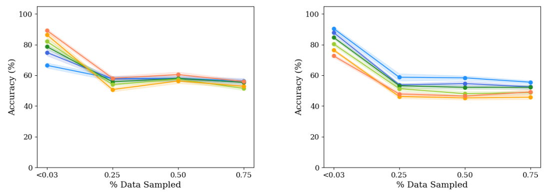


(b) y_2

Figure 42: RTE dataset (Llama 3 8B)

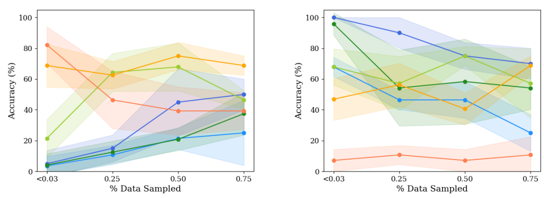


(a) y_1

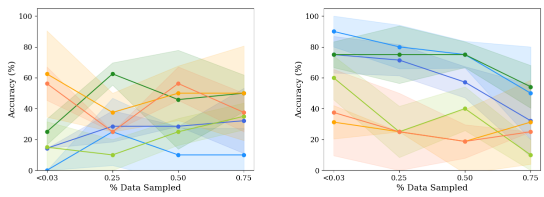


(b) y_2

Figure 43: QNLI dataset (Llama 3 8B)

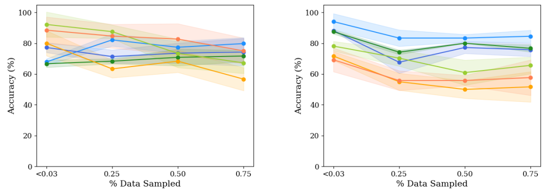


(a) y_1

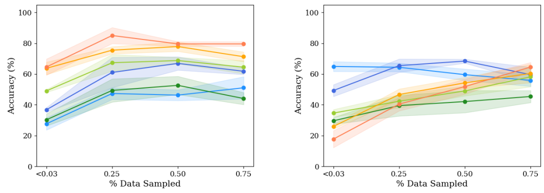


(b) y_2

Figure 44: WNLI dataset (Llama 3 8B)



(a) y_1



(b) y_2

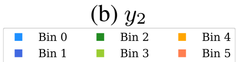
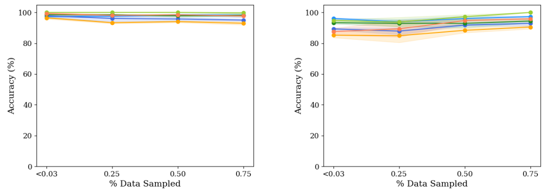
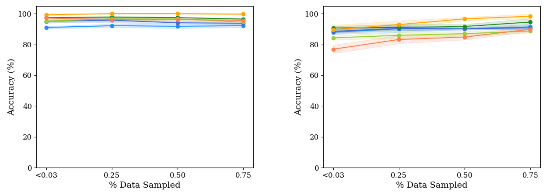


Figure 45: MPRC dataset (Llama 3 8B)



(a) y_1



(b) y_2

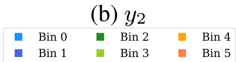
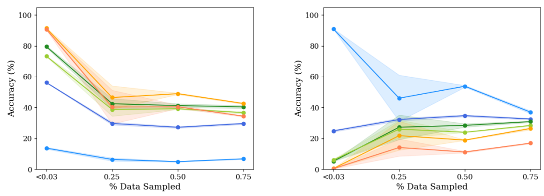
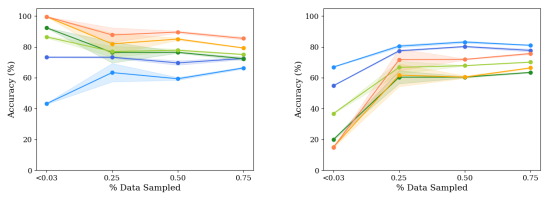


Figure 46: SST-2 dataset (Llama 3 8B)



(a) y_1



(b) y_2

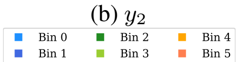
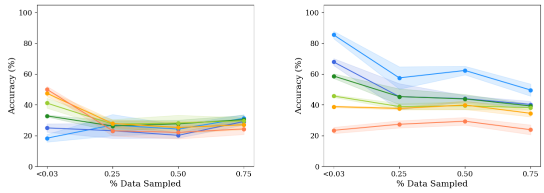
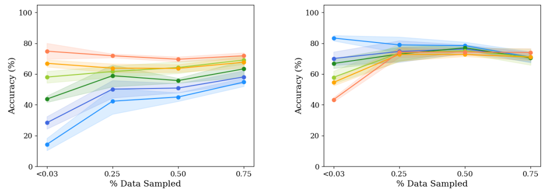


Figure 47: Hans dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

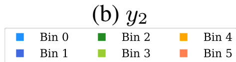
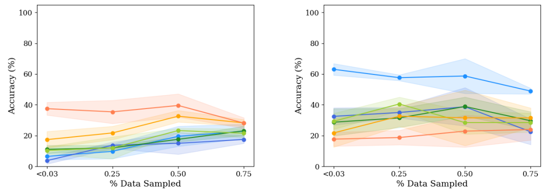
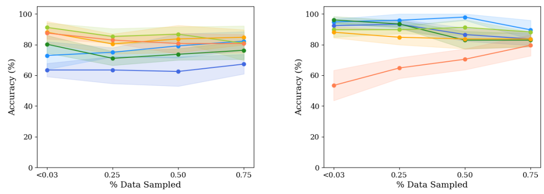


Figure 48: PAWS-X$_{\text{EN}}$ dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

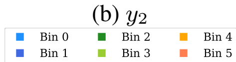
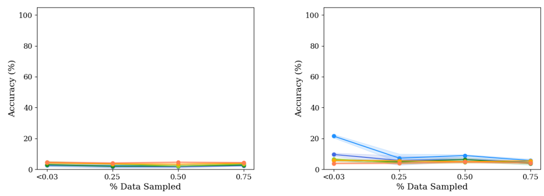
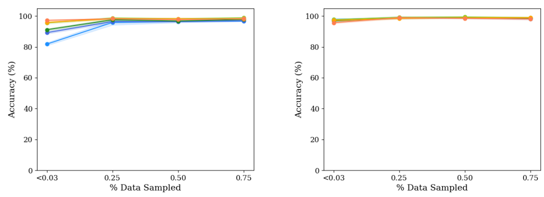


Figure 49: RTE dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

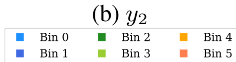
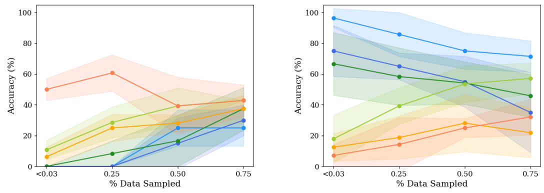
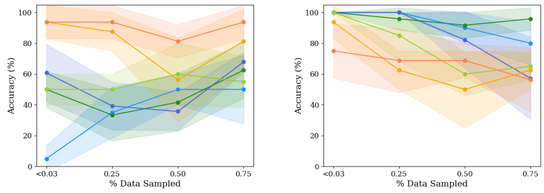


Figure 50: QNLI dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

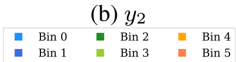
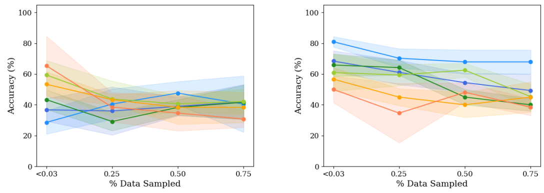
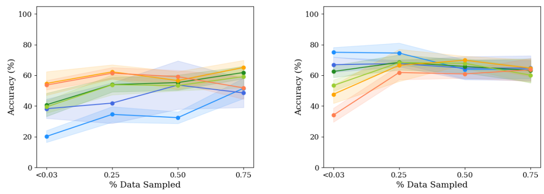


Figure 51: WNLI dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

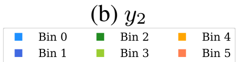
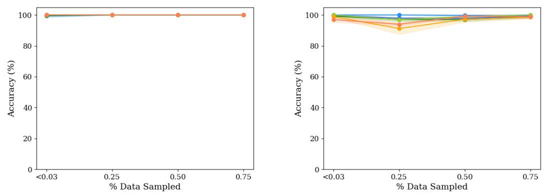
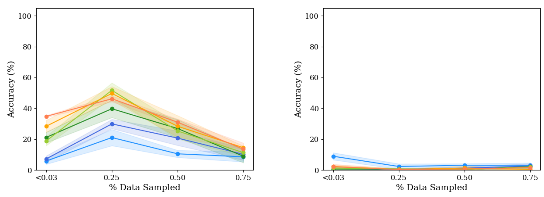


Figure 52: MPRC dataset (GPT Neo 2.7B)



(a) y_1



(b) y_2

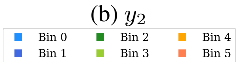
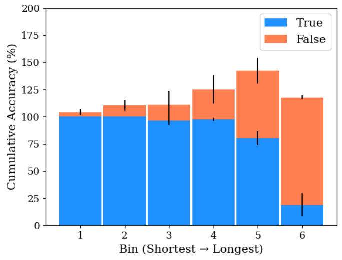


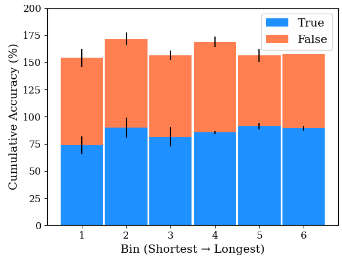
Figure 53: SST-2 dataset (GPT Neo 2.7B)

A.8 Additional Intervention Results

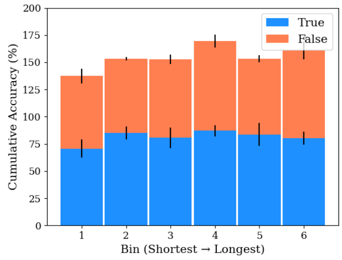
Each of the following figures shows validation set performance on a finetuned Llama 3 8B or GPT Neo 2.7B model exhibiting a length bias. For each figure, (a) shows finetuning performance prior to intervention). (b) and (c) show results on two debiasing conditions: ICL demonstrations ($k = 16$) sampled from the opposite lengths from what the model saw during finetuning (e.g y_1 long demonstrations, y_2 short demonstrations), and random sampling, respectively.



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

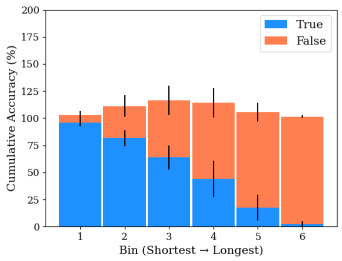


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

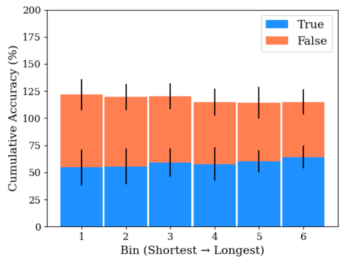


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

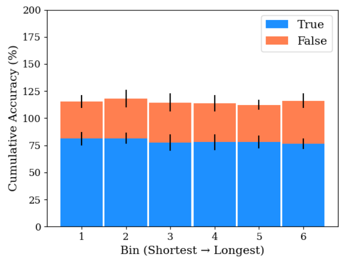
Figure 54: RTE (Llama 3 8B)



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

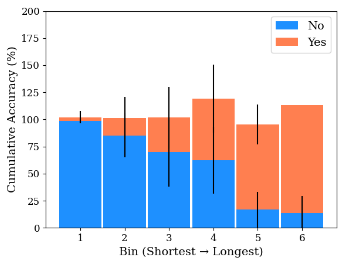


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

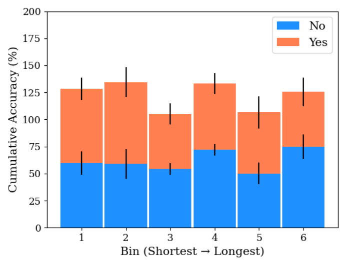


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

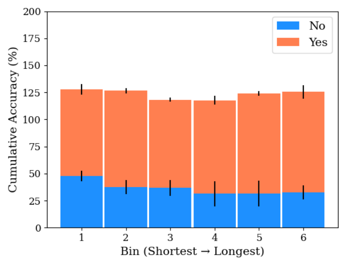
Figure 55: QNLI (Llama 3 8B)



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

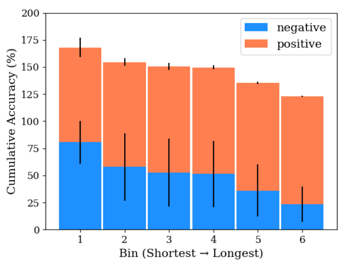


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

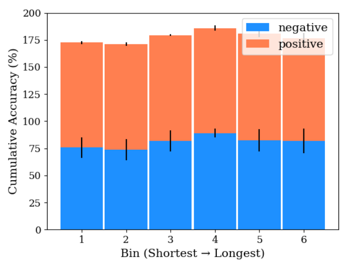


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

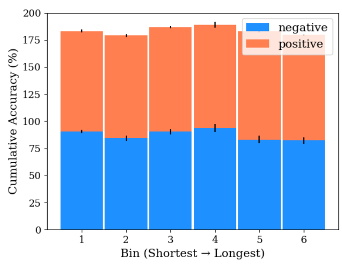
Figure 56: MRPC (Llama 3 8B)



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

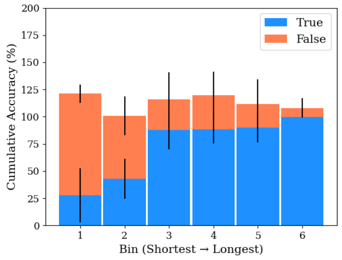


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

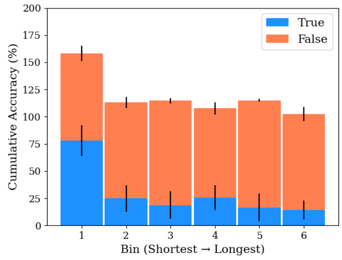


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

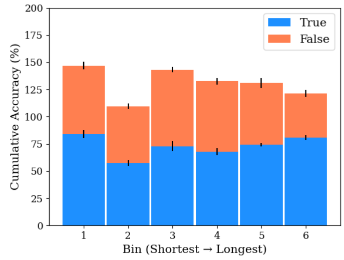
Figure 57: SST-2 (Llama 3 8B)



(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

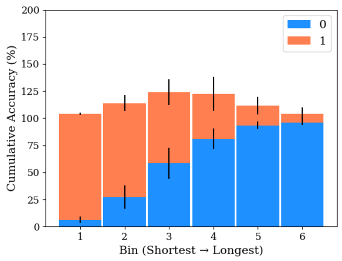


(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

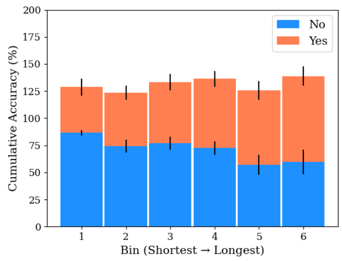


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

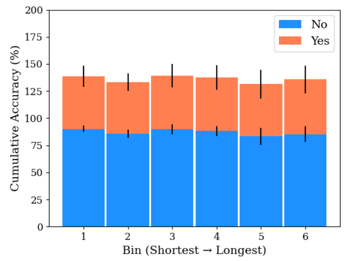
Figure 58: Hans (Llama 3 8B)



(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

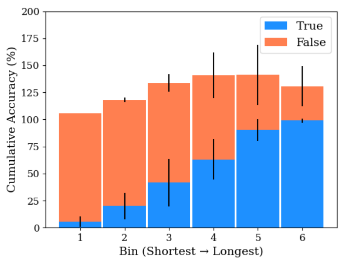


(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

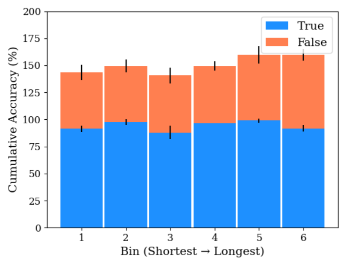


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

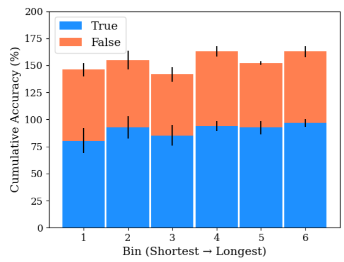
Figure 59: PAWS-X$_{\text{EN}}$ (Llama 3 8B)



(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

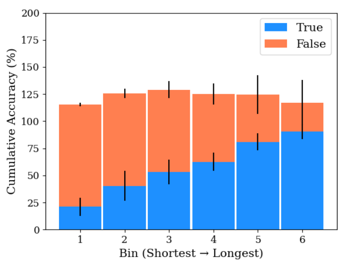


(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

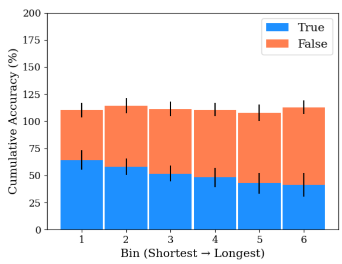


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

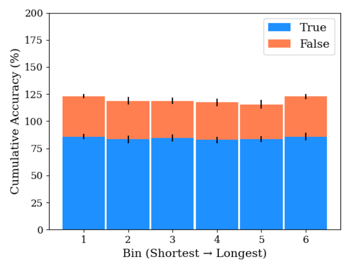
Figure 60: RTE (Llama 3 8B)



(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

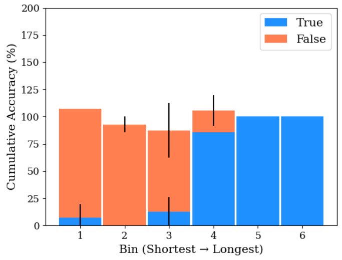


(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

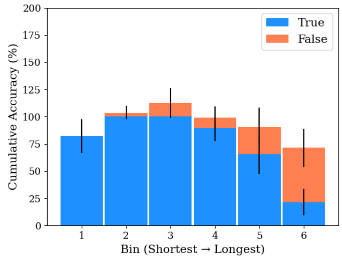


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

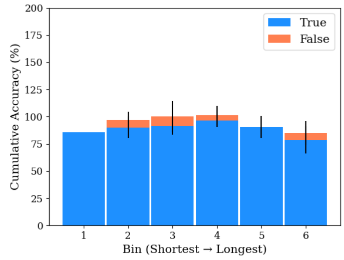
Figure 61: QNLI (Llama 3 8B)



(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

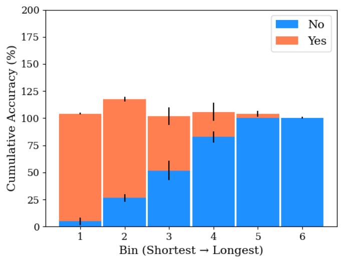


(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

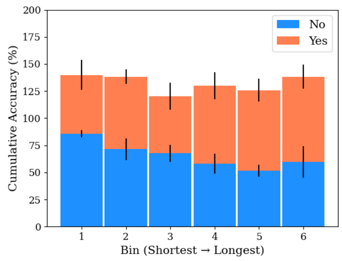


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

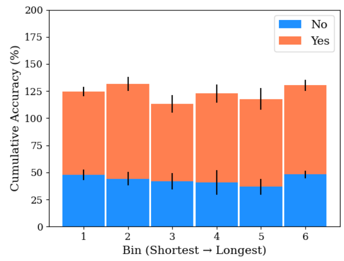
Figure 62: WNLI (Llama 3 8B)



(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

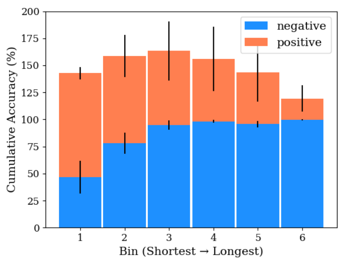


(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

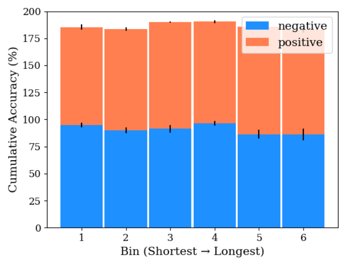


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

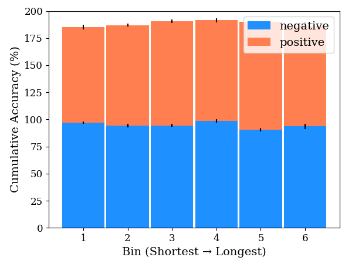
Figure 63: MRPC (Llama 3 8B)



(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

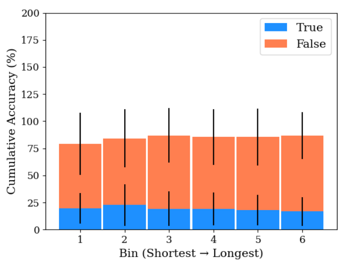


(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

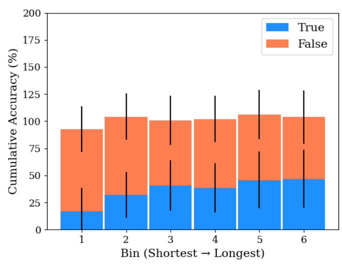


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

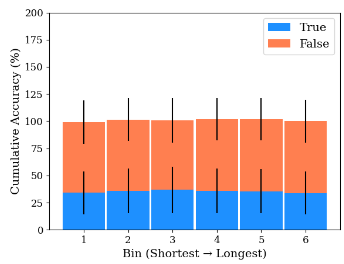
Figure 64: SST-2 (Llama 3 8B)



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

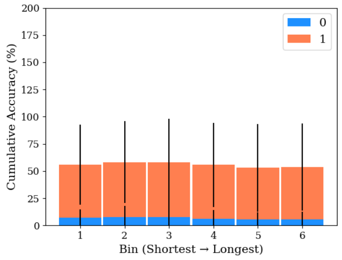


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

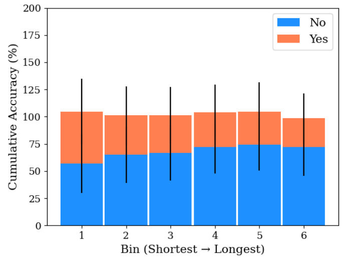


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

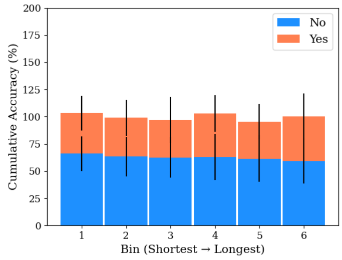
Figure 65: Hans (GPT Neo 2.7B)



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

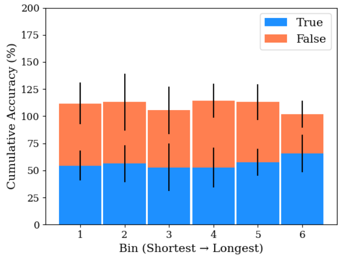


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

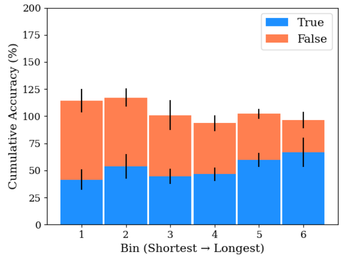


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

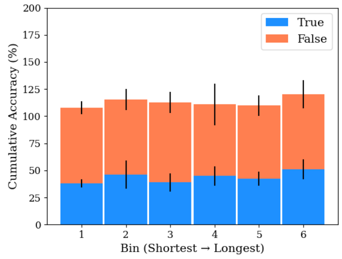
Figure 66: PAWS-X$_{\text{EN}}$ (GPT Neo 2.7B)



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

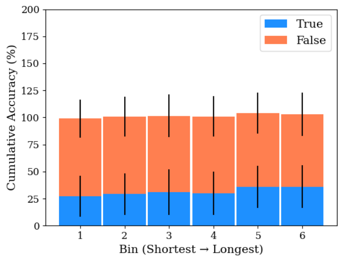


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

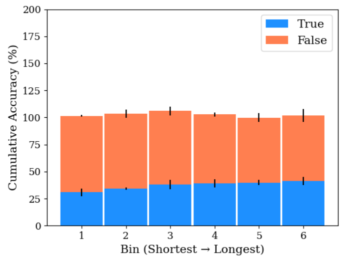


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

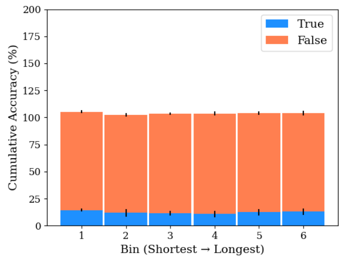
Figure 67: RTE (GPT Neo 2.7B)



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

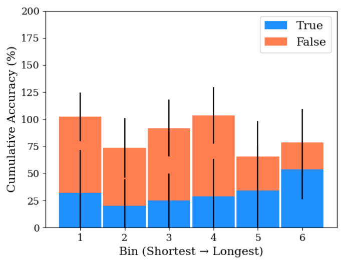


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

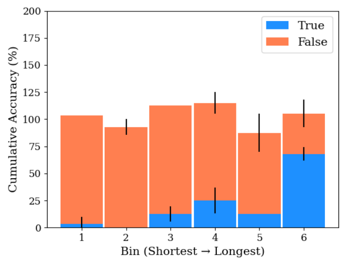


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

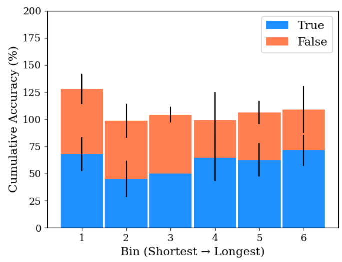
Figure 68: QNLI (GPT Neo 2.7B)



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

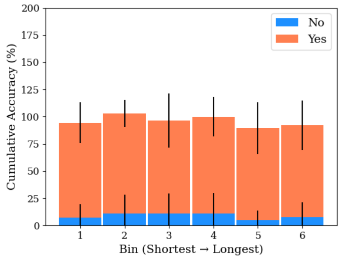


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

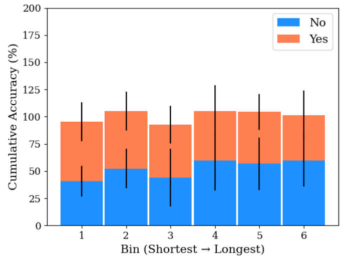


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

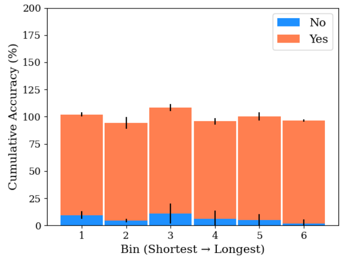
Figure 69: WNLI (GPT Neo 2.7B)



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

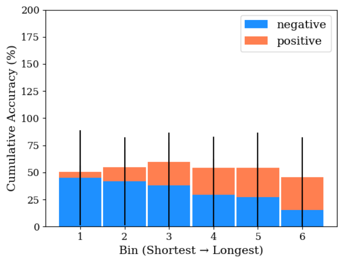


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

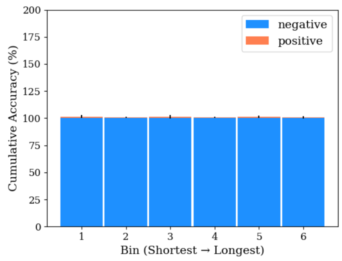


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

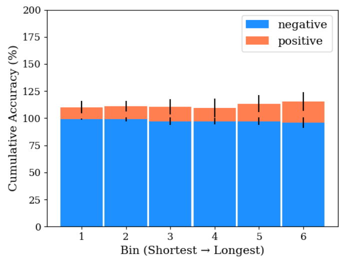
Figure 70: MRPC (GPT Neo 2.7B)



(a) Finetuning: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

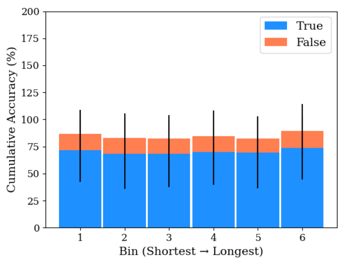


(b) Intervention: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

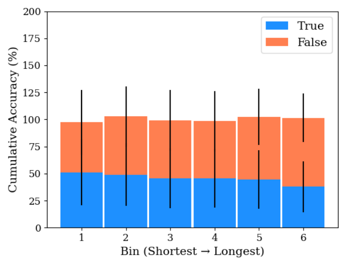


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

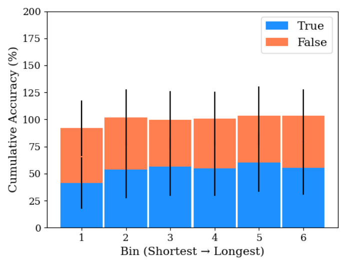
Figure 71: SST-2 (GPT Neo 2.7B)



(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

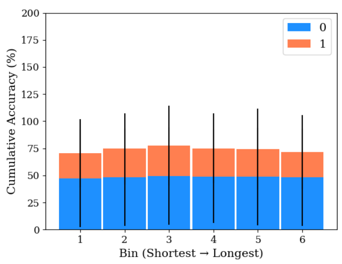


(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

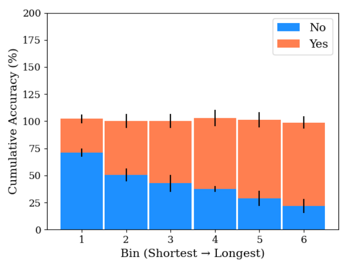


(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

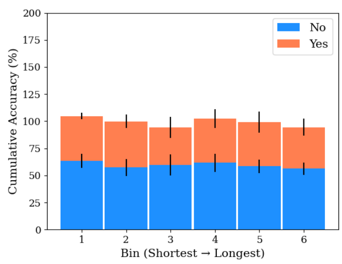
Figure 72: Hans (GPT Neo 2.7B)



(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

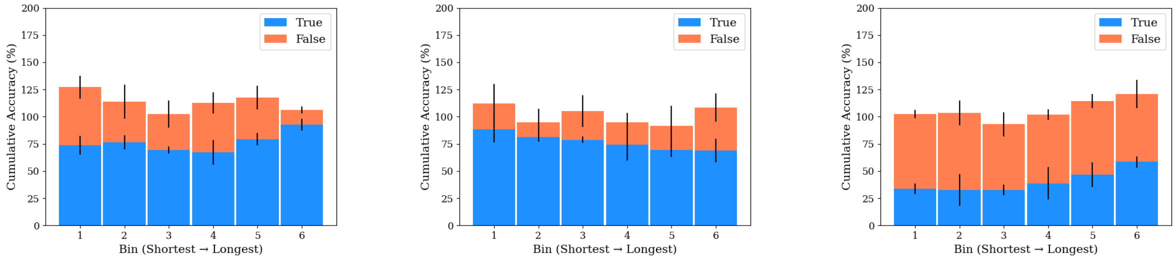


(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.



(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

Figure 73: PAWS-X$_{\text{EN}}$ (GPT Neo 2.7B)

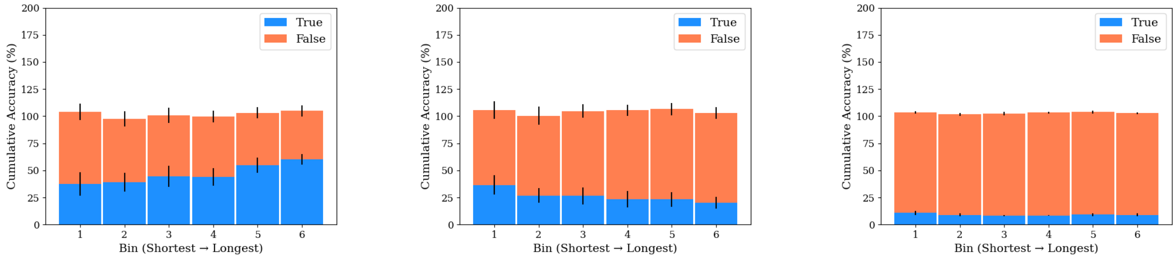


(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

Figure 74: RTE (GPT Neo 2.7B)

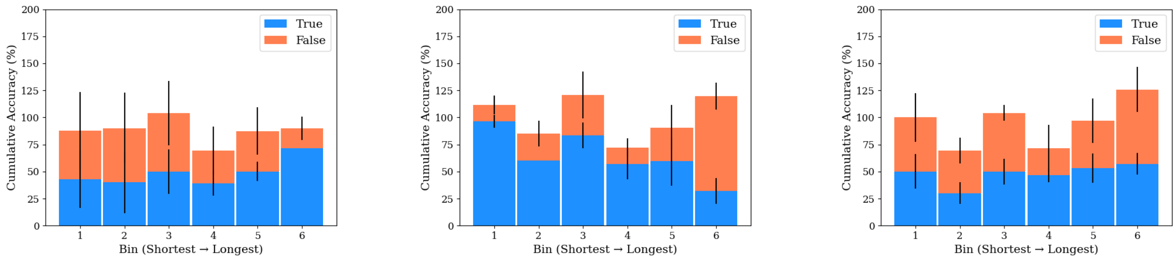


(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

Figure 75: QNLI (GPT Neo 2.7B)

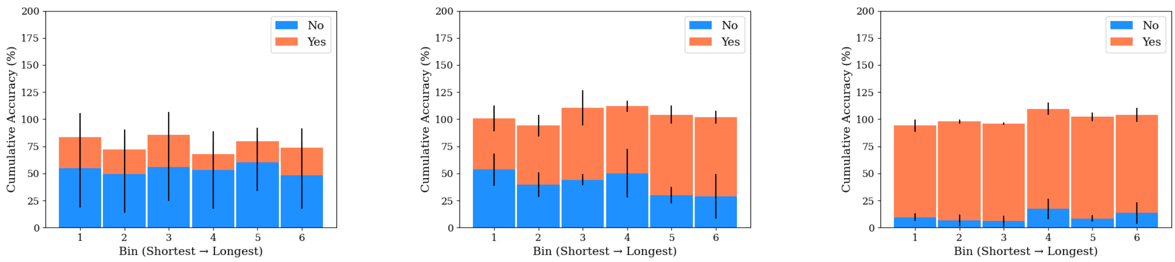


(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

Figure 76: WNLI (GPT Neo 2.7B)

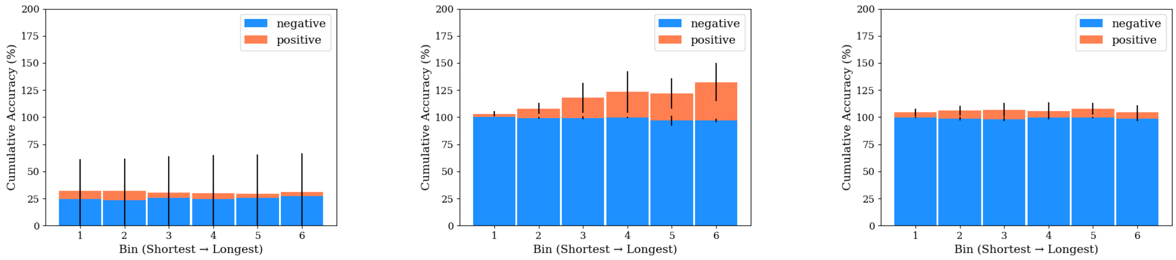


(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

Figure 77: MRPC (GPT Neo 2.7B)



(a) Finetuning: y_1 (Blue) long demonstrations, y_2 (Orange) short demonstrations.

(b) Intervention: y_1 (Blue) short demonstrations, y_2 (Orange) long demonstrations.

(c) Intervention: y_1 (Blue) and y_2 (Orange) demonstrations randomly sampled.

Figure 78: SST-2 (GPT Neo 2.7B)