MICLBench: Benchmarking LLMs with Hundreds of In-Context Examples

Anonymous ACL submission

Abstract

In-context learning (ICL) has emerged as a key capability that enables large language models (LLMs) to adapt effectively to specific tasks, offering both flexibility and improved performance. Recent advancements in extending the context window allow LLMs to process longer sequences, enabling them to benefit from many-shot in-context learning, which has shown to further enhance their performance. To systematically evaluate this emerging many-shot capability, we introduce *MICLBench*, a **M**any-shot **ICL Bench**mark spanning four core categories—classification, translation, summarization, and question answering—across 14 diverse tasks, unified under a standardized prompt and API framework. This benchmark is used to evaluate over 19 high-performance models, including both proprietary and open-source ones. Our experiments yield critical insights: (i) most models can benefit from many-shot ICL across various tasks, (ii) many-shot prompts are more effective for tasks that are coarse-grained and require less reasoning, and (iii) performance in the few-shot scenario is not necessarily positively correlated with the ability to effectively utilize many-shot prompts. The benchmark highlights pronounced performance disparities across tasks and models in many-shot settings, while exposing limitations in current LLMs' ability to harness escalating context examples. These findings provide actionable pathways for advancing models to better leverage expanding context windows. By offering a rigorous, automated evaluation framework, this work underscores the challenges and opportunities in scaling in-context learning to many-shot paradigms. The code is available at https://anonymous.4open.science/r/MICLBench.

1 Introduction

With the rapid scaling of model and data sizes, large language models (LLMs) have demonstrated enhanced linguistic capabilities and broader knowledge bases compared to traditional models. A key feature is In-Context Learning (ICL), which enables LLMs to learn from a few examples provided within a given context (Brown et al., 2020; Chowdhery et al., 2023). This ability allows LLMs to perform a variety of complex tasks without requiring fine-tuning (Wei et al., 2022). Initially, early research focused primarily on the *few-shot* scenario, constrained by the limited context length (Mavromatis et al., 2023; Liu et al., 2021).

Recent advancements (Chen et al., 2023; Peng et al., 2023; Ding et al., 2024), which expands the context window size, have paved the way for research in long-context settings. This has given rise to *many-shot* ICL, where a significantly larger number of examples are provided within a single context window. Many-shot ICL has attracted considerable attention (Bertsch et al., 2024b; Agarwal et al., 2024), as it has demonstrated the potential to enhance LLM performance across tasks, highlighting their ability to comprehend and learn from a rich set of examples within a given context. This capability is particularly valuable for real-world applications, such as document analysis, story generation, and specialized knowledge comprehension, and offers an alternative to traditional fine-tuning.

Despite these advances, evaluating LLMs' ability to leverage long-context many-shot learning remains fragmented. Existing benchmarks often focus on retrieving information from extended contexts (e.g., needle-in-a-haystack tasks (Song et al., 2025)) or solving long-dependency questions (Li et al., 2024a). Critical gaps persist in assessing how effectively models *learn* from abundant in-context examples to generalize to new cases. Prior efforts (Lee et al., 2024; Li et al., 2024b) have explored many-shot capabilities but lack task diversity, standardized evaluation protocols, and scalability to extreme context lengths, limiting their utility for holistic analysis.

To address these challenges, we introduce *MICLBench*, a comprehensive benchmark designed to

assess the ability of LLMs to leverage many-shot examples. The benchmark spans four core categories—classification, translation, summarization, and question answering—across 14 diverse tasks, all unified under a standardized prompt and API framework. Hence it enables systematic evaluation of LLMs' ability to harness escalating context lengths (up to millions of tokens). We evaluate 19 state-of-the-art models, including both proprietary and open-source architectures, through an automated pipeline with established metrics (e.g., Rouge-L (Lin, 2004), chrF2++ (Popović, 2017), accuracy). Our experiments reveal three critical insights:(i) most models can benefit from many-shot ICL across various tasks, (ii) many-shot prompts are more effective for tasks that are coarse-grained and require less reasoning, and (iii) performance in the few-shot scenario is not necessarily positively correlated with the ability to effectively utilize many-shot prompts. These findings underscore pronounced performance disparities across tasks and models, exposing limitations in current LLMs' capacity to exploit many-shot examples despite extended context windows.

For evaluation, we have implemented an automated pipeline for each scenario, employing established and reliable metrics such as Rouge-L (Lin, 2004), chrF2++ (Popović, 2017), and accuracy. This approach offers a straightforward and efficient method for benchmarking other models.

In short, our work makes the following contributions:

- **The *MICLBench***: A rigorous evaluation framework for many-shot ICL, spanning 14 tasks across four categories critical for real-world generalization. The benchmark includes standardized prompts and scalable methodologies to extend examples to extreme context lengths.

- **Automated Evaluation Pipeline**: A unified API and metrics pipeline for efficient, reproducible benchmarking of diverse LLMs.

- **Empirical Analysis:** Comprehensive evaluation of 19 LLMs, revealing performance trends, architectural limitations, and actionable pathways to improve context utilization.

By bridging gaps in task diversity, scalability, and standardization, *MICLBench* provides a foundation for advancing models to better leverage expanding context windows, while highlighting challenges in many-shot learning. At the same time, *MICLBench* also provides a new perspective to evaluate LLM's long context capability.

2 Related Work

Our work is closely related with the long context LLMs and the many-shot in context learning.

Long-context language models. Recent advances in extending LLMs' context windows have enabled many-shot in-context learning (ICL). Key approaches include architectural innovations like Rotary Position Embedding (RoPE) extrapolation (Chen et al., 2023; Zhu et al., 2024) and dynamic token compression (Tworkowski et al., 2024), as well as efficiency-focused methods such as retrieval-augmented caching (Bertsch et al., 2024a) and infinite attention mechanisms (Martins et al., 2022). These techniques allow models like CodeLlama (Rozière et al., 2024) and Mistral-8x22B to process inputs exceeding 1M tokens, making many-shot ICL feasible.

Concurrently, benchmarks for long-context understanding have emerged, focusing on tasks like retrieval (e.g., needle-in-a-haystack (Song et al., 2025)), summarization (Bai et al., 2024), and complex reasoning over extended texts (Li et al., 2024a). While these evaluate information extraction or dependency resolution, they do not assess how models learn from abundant in-context examples to solve unseen problems—a critical gap given many-shot ICL's potential to replace fine-tuning (Agarwal et al., 2024). Recent studies (Lee et al., 2024; Li et al., 2024b) explore many-shot performance but are limited to narrow tasks (e.g., classification) or fail to standardize evaluation across scales.

Our work bridges these gaps by introducing a unified benchmark for many-shot ICL across classification, translation, summarization, and question answering—domains where systematic learning from examples is essential but underexplored.

Many-shot in context learning. Many-shot ICL, which leverages hundreds to thousands of in-context examples, has emerged as a promising paradigm to reduce reliance on fine-tuning while maintaining task flexibility. Enabled by advances in long-context LLMs (e.g., Gemini 1.5 (Agarwal et al., 2024)), recent work explores its potential and limitations. For instance, Bertsch et al. (2024b) demonstrate that many-shot ICL rivals fine-tuned Llama2 on select tasks, while Baek et al. (2025)

find minimal gains from advanced example selection strategies for most many-shot tasks.

Existing benchmarks, however, remain narrow in scope. LOFT (Lee et al., 2024) focuses on classification and QA with contexts $\leq$32K tokens, while LongICLBench (Li et al., 2024b) tests extreme-label classification up to 50K tokens. These lack task diversity (e.g., summarization, question answering) and fail to stress-test modern LLMs supporting million-token contexts.

Our benchmark addresses these gaps by spanning classification, translation, summarization, and question answering tasks, scaling to 128K+ tokens. It offers a comprehensive framework for evaluating the learning abilities of LLMs across diverse domains within the *many-shot* setting.

3 *MICLBench* Construction

Scenario	Task	Data Source	Avg. Length
Translation	Bemba	FLORES-200 (Team et al., 2022)	91
	Kurdish	FLORES-200 (Team et al., 2022)	90
	French	FLORES-200 (Team et al., 2022)	73
	German	FLORES-200 (Team et al., 2022)	73
Summarization	News	XLSum (Narayan et al., 2018)	587
	Dialogue	DialogSum (Chen et al., 2021)	232
	Bill	BillSum (Eidelman, 2019)	2310
Classification	Sentiment	Yelp Review Full (Zhang et al., 2015)	180
	Topic	Yahoo Answers (Zhang et al., 2015)	135
	Intent	Banking77 (Casanueva et al., 2020)	25
Question Answering	Science	GPQA (Rein et al., 2023)	412
	Medical	MedMCQA (Pal et al., 2022)	180
	Retrieval	PubMedQA (Jin et al., 2019)	405
	Commonsense	CommonsenseQA (Talmor et al., 2019)	47

Table 1: Task descriptions in the *MICLBench*. The *MICLBench* includes 4 scenarios and 14 tasks, with each task showing its data source and average example length.

To simulate real-world many-shot utilization, we developed four scenarios and 14 tasks, including translation, summarization, classification, and cross-domain question answering. For most tasks, we provide enough examples to reach at least 128k tokens, aligning with typical LLM context windows and supporting scalability. However, we limit the number of examples in experiment to a few hundred, balancing few-shot learning and full fine-tuning while prioritizing performance and efficiency, particularly in key-value caching for in-context examples. A list of tasks is provided in Table 1.

3.1 Problem Definition

The many-shot regime involves providing hundreds or thousands of example demonstrations within a single context window (Agarwal et al., 2024). The prompts consist of three main components: a preamble outlining the task and answer format, the many-shot context with separated examples, and the final question for the model to answer.

In this setup, only the final question varies across evaluations, while the preamble and many-shot context remain constant. Typically, the preamble and final question are brief, with the many-shot context being the longest and comprising the majority of the input. The detailed prompt format for each task is shown in Appendix A.

3.2 Datasets Construction

Summarization Summarization tasks are crucial for information extraction and content condensation in various applications. To assess the ability of LLMs to summarize texts across domains and varying lengths in a many-shot scenario, we classify the articles into three categories: News, Bills, and Dialogues. Notably, Bill summaries are typically much longer, presenting challenges in managing extended contexts while ensuring summary clarity.

For each dataset, we sample 100 test queries. Summarization performance is evaluated using the Rouge-L score (Lin, 2004), which measures the overlap between generated summaries and reference texts, ensuring content relevance and similarity.

Translation Translation performance offers insights into a model's ability to handle linguistic diversity, essential for advancing multilingual applications. To examine the impact of pretraining data scale on LLMs performance in many-shot scenarios, we design tasks for both low-resource languages (Bemba, Northern Kurdish) and high-resource languages (French, German).

For each language, we sample 100 test queries. Translation quality is assessed using the chrF2++ score (Popović, 2017), a reliable metric that evaluates character and word-level similarities across languages with varying resources.

Classification The classification scenario includes three tasks: Intent, Sentiment, and Topic Classification, offering a comprehensive benchmark across diverse linguistic and contextual domains. With label sets ranging from 5 to 77 categories, the dataset allows for a thorough evaluation of LLMs' ability to identify nuanced patterns in text classification.

For each task, we curate test queries to ensure full representation of all label categories. To avoid attributing performance gains to the model's unfamiliarity with the label space, we include all possible labels in the prompt preamble. Performance

is evaluated using accuracy, a reliable metric for classification.

Question Answering The question answering scenario includes four tasks: Natural Science, Medical, Commonsense, and Retrieval, ensuring a comprehensive evaluation across diverse reasoning domains. All tasks involve answering questions directly, except for Retrieval, which requires responses based on provided research papers. Rationale is given for answers in Science, Medical, and Retrieval tasks, but not in Commonsense. This design allows for a thorough evaluation of reasoning capabilities and analysis of how different strategies impact performance in many-shot settings. Reasoning is fundamental for advanced AI applications, making this scenario a key benchmark for assessing LLMs' cognitive processes across real-world domains.

To ensure reliability, we select test queries that cover all relevant subthemes, minimizing assessment errors. All tasks are formatted as multiple-choice questions for consistency, allowing accuracy to be directly measured by the proportion of correct responses.

4 Evaluation

4.1 Experiment Setup

Evaluated Models To examine the influence of LLMs' internal capabilities on their performance in a many-shot learning regime, we selected 19 widely used LLMs with long-context capabilities. This selection includes both open-source models, such as LLaMA (Dubey et al., 2024), Qwen (Qwen et al., 2025), Mistral (Jiang et al., 2023) and Mixtral (Jiang et al., 2024), as well as proprietary models, specifically GPT-4o-mini-0718 (Achiam et al., 2023), Gemini-1.5-Pro (Team et al., 2024), and Claude-3.5-Sonnet (The). These models encompass a broad range of parameter sizes and context window lengths (ranging from 32k to 2M tokens). A comprehensive list of the evaluated models is presented in Table 2 and Table 3

Example Selection Due to the limitations of context window size in early LLMs, much prior research has focused on strategies for selecting examples in the few-shot regime (An et al., 2023; Mavromatis et al., 2023). However, with the rapid expansion of context lengths, several recent studies have shown that various sample selection strategies do not result in statistically significant gains in the many-shot scenario (Bertsch et al., 2024b; Baek et al., 2025). In light of this, we focus on utilizing randomly sampled demonstrations from the dataset, as this approach enhances efficiency through key-value caching of in-context examples. To ensure that the context captures a broader range of information as the number of demonstrations increases, we incrementally introduce additional examples into the context, thereby increasing the number of shots.

Evaluation Methods and Metrics Our evaluation framework includes 4 scenarios and 14 tasks, supported by an automated grading pipeline for efficiency and precision. In summarization, we use the ROUGE-L metric to measure the longest common subsequence between reference and generated summaries. For translation, we employ the chrF2++ score, assessing character- and word-level n-gram overlaps. In classification and question answering, we use exact match scores. To address varying answer formats, we conduct multiple rounds of answer extraction for improved accuracy.

To mitigate the impact of random sample selection on the trend from the few-shot to the many-shot regime, we use five randomly selected subsets of the prompt datasets and average the results for open-source models. For efficiency, the experiment is conducted once when assessing proprietary models.

4.2 Results on *MICLBench*

The primary results are presented in Table 2 and Table 3. In the following section, we offer a comprehensive analysis of these findings.

Model Size We categorize open-source models into two groups based on size: small (0.5B-14B) and large (32B-72B). While closed-source and larger models typically outperform smaller ones across most tasks, our analysis of many-shot prompt utilization reveals more nuanced findings. Specifically, smaller models often benefit from many-shot in-context learning and can even outperform larger models in a few-shot setting. For instance, in the Bemba task, Qwen2.5-3B-Instruct achieves a top score of 20.46 at 200 shots, whereas Qwen2.5-7B-Instruct scores 18.49 at 5 shots. This suggests that, in some cases, smaller models can achieve target performance by simply providing sufficient examples. This observation also applies to base models and those with supervised instruction tuning. The latter consistently outperform their base counterparts in a few-shot setting, while base models gradually improve their ability to follow instructions

Model	Context Window	Translation								Summarization					
		Bemba		Kurdish		French		German		News		Dialogue		Bill	
		5-shots	Best	5-shots	Best	5-shots	Best	5-shots	Best	5-shots	Best	5-shots	Best	1-shots	Best
Gemini-1.5-pro	2M	35.58	**43.49(800)**	39.16	**41.67(800)**	72.65	72.65(5)	65.54	**65.99(800)**	0.3109	**0.3986(75)**	0.3389	**0.3893(50)**	0.2841	0.3291(10)
Claude-3.5-Sonnet	200k	38.73	**43.95(200)**	38.89	**40.74(200)**	70.92	**71.77(100)**	66.00	**67.61(1000)**	0.2616	0.3552(25)	0.3093	**0.3595(400)**	0.2243	**0.2930(40)**
GPT-4o-mini-0718	128k	24.79	**31.18(1000)**	30.00	**32.92(1000)**	70.80	**72.03(100)**	65.12	65.18(1)	0.2148	0.2278(10)	0.2973	**0.3224(400)**	0.2165	0.2486(5)
Qwen2.5-0.5B-Inst.	32k	10.02	**14.73(50)**	7.317	**11.41(200)**	47.23	**47.66(50)**	36.61	**37.22(50)**	0.1857	0.2338(25)	0.1595	**0.2927(100)**	0.1272	0.2237(10)
Qwen2.5-1.5B-Inst.	32k	11.95	**16.40(200)**	9.498	**13.79(200)**	57.17	57.17(5)	46.91	**47.08(200)**	0.2555	0.2724(25)	0.2497	**0.3474(100)**	0.1807	0.1807(1)
Qwen2.5-3B-Inst.	32k	13.36	**20.46(200)**	10.63	**15.25(200)**	61.46	**62.30(200)**	50.90	**52.44(100)**	0.2491	0.2895(25)	0.2201	**0.3370(50)**	0.1672	0.2550(10)
Qwen2.5-7B	128k	12.88	**22.81(500)**	10.91	**17.99(500)**	63.57	**64.21(200)**	55.48	**56.30(1000)**	0.3188	**0.3309(50)**	0.3255	0.3376(25)	0.2859	0.2903(10)
Qwen2.5-7B-Inst.	32k	18.49	**24.27(200)**	16.07	**18.91(200)**	63.97	**64.62(50)**	54.68	**55.65(50)**	0.2988	0.3203(25)	0.3165	**0.3562(100)**	0.2534	0.2722(10)
Mistral-7B-Inst.-v0.2	32k	17.20	**23.36(200)**	15.14	**18.73(200)**	63.57	**64.74(50)**	54.79	**55.34(50)**	0.3134	0.3248(25)	0.3210	**0.3521(100)**	0.2637	0.2758(5)
Ministral-8B-Inst.-2410	32k	12.51	**18.40(200)**	23.18	**25.34(200)**	68.92	**69.41(50)**	60.88	60.92(25)	0.3032	0.3143(25)	0.3347	0.3584(25)	0.2666	0.2953(10)
Llama-3.1-8B-Inst.	128k	14.88	**29.48(1000)**	24.65	**28.29(800)**	66.83	66.85(25)	59.25	59.49(1)	0.3047	**0.3167(50)**	0.3337	0.3419(10)	0.1871	**0.2992(30)**
Qwen2.5-14B-Inst.	32k	15.83	**24.17(200)**	19.01	**21.89(200)**	65.79	66.81(1)	59.70	**60.42(200)**	0.3296	0.3353(10)	0.3067	**0.3652(100)**	0.2080	0.2625(10)
Qwen2.5-32B	128k	13.16	**29.83(500)**	17.38	**25.64(500)**	63.59	68.83(25)	60.46	**61.13(50)**	0.3557	**0.3746(75)**	0.3306	**0.3499(400)**	0.2932	0.3005(5)
Qwen2.5-32B-Inst.	32k	17.39	**26.11(200)**	20.12	**25.00(200)**	63.01	**68.45(100)**	57.65	**61.26(100)**	0.2809	0.3539(25)	0.2612	**0.3582(100)**	0.2409	0.2849(10)
Mixtral-8x7B-Inst.-v0.1	32k	18.83	**24.85(200)**	18.82	**21.07(200)**	68.88	**69.53(50)**	62.33	**63.26(200)**	0.3522	0.3603(25)	0.3445	**0.3676(100)**	0.2823	0.2841(5)
Llama-3.1-70B-Inst.	128k	21.99	**36.34(800)**	35.05	**36.98(500)**	70.04	70.40(1)	62.42	**63.50(500)**	0.3256	**0.3574(100)**	0.3473	**0.3503(200)**	0.2664	**0.3323(35)**
Llama-3.3-70B-Inst.	128k	24.18	**36.36(800)**	34.16	**36.77(800)**	70.20	70.20(5)	63.72	64.18(25)	0.3211	**0.3615(100)**	0.3460	**0.3536(300)**	0.2594	**0.3361(30)**
Qwen2.5-72B	128k	15.02	**28.06(800)**	20.27	**26.54(500)**	69.95	**70.61(800)**	62.40	62.89(25)	0.3700	**0.3908(50)**	0.3460	0.3497(10)	0.3147	**0.3298(35)**
Qwen2.5-72B-Inst.	32k	19.32	**27.09(200)**	21.82	**25.77(200)**	67.95	69.22(25)	60.65	62.26(25)	0.3173	0.3572(25)	0.2810	**0.3623(50)**	0.2679	0.3438(10)

Table 2: Performance of Translation and Summarization tasks. The scores for the few-shot regime (mostly 5-shot, except for the Bill task, which uses 1-shot due to the length of the examples in the Bill dataset—each example is much longer, so 50 examples already account for nearly 128k tokens) and the best performance for each task are displayed. The numbers in parentheses indicate the corresponding shot count. The best scores are highlighted in **bold** when the shot count exceeds 50 (except for the Bill task, where the boundary is set to 25), marking the transition from few-shot to many-shot. Notably, for the Qwen2.5 Instruct series, we maintained a context window of 32k tokens, but used the rope scaling parameter to extend it to 128k tokens in the Exploratory Insight.

and learn the answer format as the number of examples increases, as seen in tasks like Kurdish (e.g., Qwen2.5-32B vs. Qwen2.5-32B-Instruct). This suggests that a model's performance in a few-shot setting does not necessarily predict its ability to learn from many-shot prompts. Models with varying few-shot performance can be comparable or even have their rankings reversed when evaluated with many-shot prompts.

Training Data Size The scale of pre-training data, which influences the internal knowledge of specific tasks, significantly affects the ability of LLMs to utilize many-shot prompts. As demonstrated in the results for translation, the improvement in low-resource translation tasks is considerably greater when transitioning from few-shot to many-shot, compared to high-resource translation tasks, where many models show only marginal gains with few-shot prompts. This can be attributed to the fact that LLMs are likely trained on large datasets, exposing them to numerous examples of similar tasks. As a result, their ability to improve further is limited, as they have already learned most of the relevant patterns from the data.

Task Difficulty Intuitively, it is difficult for a model to extract useful information from the given context if the task's difficulty exceeds the model's upper capacity limit. As observed in the summarization scenario, models with large context windows (which can handle more than 50 examples) generally show improved performance with many-shot prompts in the News and Dialogue tasks. However, they struggle with the Bill task. As the example length increases, the model's ability to effectively learn from the context diminishes. Despite this, the largest models still demonstrate potential benefits from many-shot prompt, suggesting that as model capabilities grow, there may be further opportunities for performance improvements.

When evaluating large models on simpler tasks, we find that larger models benefit from many-shot prompting in the Science task, while smaller models show greater improvement in the Medical task. This difference is likely due to the increased complexity of the Science dataset, which presents challenges for smaller models. In contrast, larger models may reach performance saturation on the relatively simpler Medical task, where many-shot prompt can disrupt their responses and degrade final scores.

Task Categories Many-shot prompting significantly improves performance in tasks such as Low-resource Translation, Dialogue Summarization, Classification and Retrieval, with results strongly tied to task categories. It appears particularly effective for humanities and social sciences tasks, which

are generally coarse-grained and require less reasoning. In contrast, fine-grained tasks like science question answering, which demand step-by-step reasoning, may offer more potential for improvement. Notably, some models show strong performance in retrieval tasks, demonstrating the value of leveraging long-context capabilities in many-shot settings for short-dependency question answering. However, the reasons behind the consistent performance increase in tasks like Bemba Translation and Intent Classification remain unclear and warrant further exploration.

Role of Rationale For the Commonsense task, unlike previous datasets, we did not provide rationales, nor did we use a "think step by step" prompt, allowing the LLMs to generate answers directly. Surprisingly, several models still showed performance improvements. This aligns with the findings of Agarwal et al. (2024), who demonstrate the effectiveness of "Unsupervised ICL", where the prompt is provided without the answer, suggesting the need for further investigation into the necessity of rationales and answers in many-shot settings.

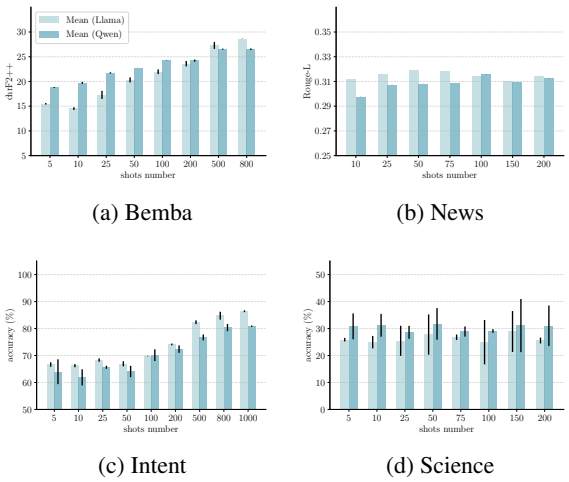
4.3 Exploratory Insight

In this section, we conduct a comprehensive analysis of the factors that significantly influence the performance of LLMs in the many-shot regime, exploring these factors from multiple perspectives. We identify distinct characteristics that may provide valuable insights for advancing many-shot performance and facilitating the expansion of LLMs' context window size. To ensure the reliability of our results, we evaluate two widely used models—LLaMA-3.1-8B-Instruct and Qwen2.5-7B-Instruct—and select some representative tasks from 4 scenarios for testing.

4.3.1 Robustness to Example Order

The large number of examples in many-shot scenarios raises important questions about the impact of example order , factor that have been shown to play a crucial role in few-shot scenarios (Lu et al., 2022; Xiang et al., 2024).

Several studies (Agarwal et al., 2024; Bertsch et al., 2024b; Baek et al., 2025) have investigated the effect of example order. However, Agarwal et al. (2024) and Baek et al. (2025) focus solely on the many-shot regime without comparing it to the few-shot setting, while Bertsch et al. (2024b) limit their analysis to classification tasks. To comprehensively examine the influence of example order as we transition from a few-shot to a many-shot scenario, we progressively add examples to the prompt and calculate the average across 5 different example orders for each given number of demonstrations. The results is showed in the Figure 1.



(a) Bemba (b) News

(c) Intent (d) Science

Figure 1: Variance across different shot numbers. The two bars represent the mean scores across five example orders: the left bar for Llama-3.1-8B-Instruct and the right for Qwen2.5-7B-Instruct. The error bar depict the variance for each model.

As the number of examples increases, we anticipated a decrease in the importance of any single example, leading to improved robustness against example order. However, this was not universally observed, and performance remained sensitive to the order of examples, particularly in more complex tasks, such as science question answering, even in the many-shot scenario. Notably, the lowest variance occurred when the number of shots was moderate (e.g., Qwen in the Bemba translation and science question answering tasks, Llama in intent classification and science question answering tasks). Building on our findings in 4.3.3, this suggests that while additional examples reduce the weight of individual instances, LLMs may not fully utilize the context window. As a result, early examples may be less effectively learned, yet they could have a greater impact on overall performance. Optimizing the order of examples could improve performance. Identifying the optimal order remains a key area for future research.

4.3.2 Analysis on Noisy Ratios

As more information is incorporated into the context window, the likelihood of including noisy messages—whether incorrect or of low quality—inevitably increases, which can negatively

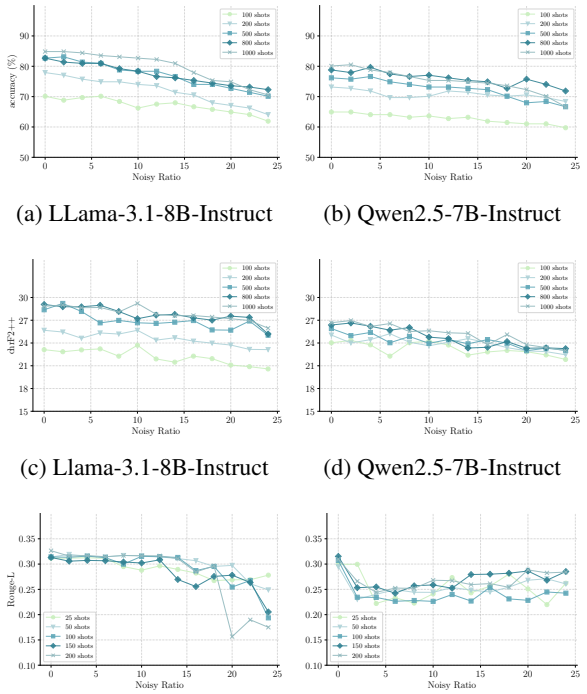
Model	Context Window	Classification						Question Answering							
		Sentiment		Topic		Intent		Science		Medical		Retrieval		Commonsense	
		5-shots	Best	5-shots	Best	5-shots	Best	5-shots	Best	5-shots	Best	5-shots	Best	5-shots	Best
Gemini-1.5-pro	2M	0.6700	**0.7300(400)**	0.6850	**0.7100(100)**	0.7489	**0.9307(2000)**	0.6010	0.6313(10)	0.8762	0.9048(25)	0.7444	**0.8000(100)**	0.8000	0.8500(25)
Claude-3.5-Sonnet	200k	0.7550	0.7700(10)	0.7150	0.7350(10)	0.7576	**0.9221(2000)**	0.5960	0.6465(25)	0.9143	**0.9333(400)**	0.8333	0.8333(5)	0.8400	0.8600(25)
GPT-4o-mini-0718	128k	0.6800	0.7250(1)	0.6900	**0.7050(50)**	0.7013	**0.8745(2000)**	0.3788	0.3990(10)	0.8571	0.8571(5)	0.7000	0.7333(25)	0.8000	**0.8100(50)**
Qwen2.5-0.5B-Inst.	32k	0.4340	**0.5370(100)**	0.2330	**0.4350(100)**	0.2597	**0.6329(1000)**	0.2152	0.2323(25)	0.2724	**0.3429(100)**	0.3822	**0.4222(50)**	0.4660	**0.5360(50)**
Qwen2.5-1.5B-Inst.	32k	0.5400	0.5830(25)	0.5530	**0.5670(200)**	0.4909	**0.7420(1000)**	0.2556	**0.2636(50)**	0.3676	**0.4267(100)**	0.5400	0.5644(25)	0.6860	**0.7180(200)**
Qwen2.5-3B-Inst.	32k	0.6280	**0.7130(50)**	0.5200	**0.6230(100)**	0.5792	**0.7922(1000)**	0.2808	0.3141(25)	0.5962	**0.6133(50)**	0.6689	**0.6867(50)**	0.7300	0.7520(10)
Qwen2.5-7B	128k	0.6290	**0.6690(400)**	0.6030	**0.6420(200)**	0.6416	**0.8320(2000)**	0.2970	0.3030(10)	0.5810	0.6305(1)	0.6400	0.6756(25)	0.8540	**0.8640(100)**
Qwen2.5-7B-Inst.	32k	0.6620	0.6970(10)	0.6470	**0.6530(200)**	0.6649	**0.8182(1000)**	0.3030	0.3323(25)	0.6838	0.7010(10)	0.7156	0.7400(1)	0.8600	**0.884(200)**
Mistral-7B-Inst.-v0.2	32k	0.6370	**0.6980(50)**	0.5660	**0.6190(100)**	0.5810	**0.8597(1000)**	0.2586	0.2747(10)	0.5695	0.6133(25)	0.6756	0.6978(10)	0.6100	**0.6880(200)**
Ministral-8B-Inst.-2410	32k	0.6850	**0.7150(100)**	0.6770	**0.7050(100)**	0.6459	**0.8355(1000)**	0.3000	0.3131(10)	0.6152	0.6152(5)	0.6800	**0.6889(50)**	0.7140	0.7140(5)
Llama-3.1-8B-Inst.	128k	0.6830	**0.7400(400)**	0.6410	**0.6740(500)**	0.6511	**0.8693(2000)**	0.2475	**0.2970(75)**	0.7067	0.7238(1)	0.6733	**0.6867(100)**	0.6840	0.7100(1)
Qwen2.5-14B-Inst.	32k	0.6730	0.7060(10)	0.6610	**0.6800(200)**	0.6857	**0.8459(1000)**	0.3586	0.3879(1)	0.7333	0.7448(10)	0.7444	**0.7489(50)**	0.8620	**0.8700(200)**
Qwen2.5-32B	128k	0.6760	**0.7440(500)**	0.6540	**0.6800(800)**	0.6823	**0.8494(2000)**	0.3747	**0.3950(75)**	0.6990	0.7067(1)	0.6933	**0.7200(200)**	0.8820	0.9080(1)
Qwen2.5-32B-Inst.	32k	0.6990	0.7260(10)	0.6740	0.6740(5)	0.6892	**0.8424(1000)**	0.3828	**0.4364(50)**	0.7943	0.8095(25)	0.7622	0.7644(25)	0.8700	0.8760(10)
Mixtral-8x7B-Inst.-v0.1	32k	0.7140	**0.7560(50)**	0.6400	**0.6620(100)**	0.6701	**0.9022(1000)**	0.3040	0.3162(10)	0.6667	0.6990(25)	0.6978	0.7156(1)	0.7420	0.7420(5)
Llama-3.1-70B-Inst.	128k	0.6970	**0.7620(200)**	0.6850	**0.7020(500)**	0.6866	**0.8675(2000)**	0.4192	0.4303(25)	0.8610	0.8610(5)	0.6800	**0.7422(200)**	0.8360	**0.8520(200)**
Llama-3.3-70B-Inst.	128k	0.6870	**0.7600(300)**	0.6900	0.7020(1)	0.6970	**0.8719(2000)**	0.4374	0.4545(1)	0.9124	0.9124(5)	0.6711	**0.7111(200)**	0.8220	**0.8460(800)**
Qwen2.5-72B	128k	0.7020	**0.7550(400)**	0.6700	0.6870(1)	0.6866	**0.8581(2000)**	0.3737	**0.4040(50)**	0.7371	0.7524(1)	0.6889	**0.7022(100)**	0.8880	0.9000(1)
Qwen2.5-72B-Inst.	32k	0.6940	**0.7250(100)**	0.6909	0.7030(1)	0.7333	**0.8545(1000)**	0.4485	**0.4929(50)**	0.7943	**0.8171(50)**	0.7111	**0.7467(50)**	0.8520	0.8540(25)

Table 3: Performance of Classification and Question Answering tasks. The scores for the few-shot regime and the best performance for each task are displayed. The numbers in parentheses indicate the corresponding shot count. The best scores are highlighted in **bold** when the shot count exceeds 50, marking the transition from few-shot to many-shot. Notably, for the Qwen2.5 Instruct series, we maintained a context window of 32k tokens, but used the rope scaling parameter to extend it to 128k tokens in the Exploratory Insight.

impact performance. Previous research (Agarwal et al., 2024) has shown that many-shot prompts can help overcome pre-training biases. However, whether they are also capable of shielding LLMs from the effects of noisy information still requires further investigation.

To create the noisy sample, we replace the answer of examples in the prompt by examples not in the prompt. Especially, when test the performance in classification scenario, we replace the original label by other labels in the whole label space. In addition, as the noisy ration gradually increasing give the number of examples, we ensure the selected noisy examples in high ratio include all the noisy examples in low ratio.

As illustrated in Figure 2, which presents results for intent classification and news summarization, the scores for intent classification and Bemba translation show minimal decline when the noise ratio is below 10%, with performance remaining above 90% even when the noise ratio reaches 25%, relative to the baseline without noise. In contrast, performance on news summarization declines more rapidly, particularly for Qwen2.5-7B-Instruct, even in the many-shot regime. This is likely due to the longer length of each example, which amplifies the impact of noise on performance. Additionally, we find that while many-shot prompting does not significantly improve performance in high-resource language translation, it does enhance robustness to



(a) LLama-3.1-8B-Instruct (b) Qwen2.5-7B-Instruct

(c) Llama-3.1-8B-Instruct (d) Qwen2.5-7B-Instruct

(e) LLama-3.1-8B-Instruct (f) Qwen2.5-7B-Instruct

Figure 2: Performance Change with Increasing Noise Ratio on two representative datasets. The upper graphs show Intent Classification results, the middle graphs show Bemba Translation results and the lower ones show News Summarization results. Other dataset results are in Appendix B.1.

noise in certain contexts, as shown in the French translation results in Appendix B.1.

4.3.3 Examination of Input Utilization

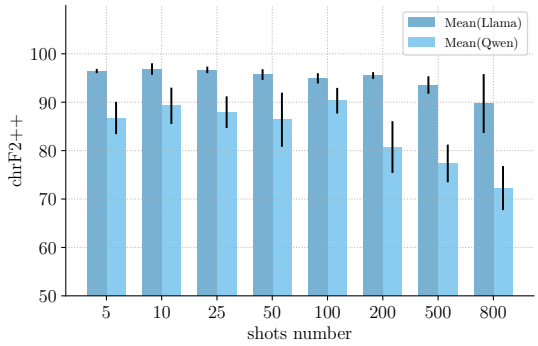
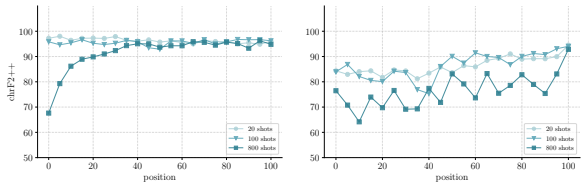


Figure 3: Performance of test queries incorporated into prompts in the Bemba translation task. We average the scores across five different sets of examples for each given number of shots. Other dataset results are in Appendix B.2.

Given the rapid expansion of the context window size, it is natural to question whether LLMs can fully utilize the available context. To investigate this, we adopt a simple approach used in Bertsch et al. (2024b), where each question is embedded into the prompt, extending the context across various tasks to gain different insights. If LLMs can fully utilize the context, they should be able to identify the question and provide the correct answer. Although all LLMs achieve high accuracy in this setting, none reach 100% correctness.

As shown in Figure 3, which presents the performance on the Bemba translation task, LLMs demonstrate limited ability to accurately detect the query within the prompt, especially in the many-shot scenario, where a noticeable decline in performance is observed. This likely occurs because, as the number of examples increases, the difficulty of locating the query also increases, leading to a drop in performance. Furthermore, the variance in this experiment is significantly large, highlighting a strong correlation between copy behavior and the position of the query in the prompt. This observation motivates further investigation in subsequent experiments.

Previous research (Lu et al., 2022) has demonstrated that LLMs often prioritize the final example in a prompt. To further explore how LLMs utilize different segments of the prompt, we designed an experiment in which the position of the test query was gradually shifted from the first to the last example. If LLMs exhibit a preference for specific sections of the prompt, performance should improve when the query is positioned accordingly. As shown in Figure 4, in the Bemba translation task, performance remains largely unchanged with a small number of shots. However, as the number of shots increases, performance significantly decreases when the query is positioned earlier in the prompt. This suggests that the models are unable to fully leverage the context window and continue to exhibit a tendency to focus on examples placed towards the end.



(a) LLama-3.1-8B-Instruct (b) Qwen2.5-7B-Instruct

Figure 4: Performance variations with query position. The graphs display the Bemba Translation results, with the x-axis representing the position of the query as a percentage of all examples. A value of 0% indicates the query is at the beginning, while 100% indicates it is at the end. Other dataset results are in Appendix B.2.

5 Conclusion

In this study, we introduce the *MICLBench*, a synthetic benchmark designed to assess the learning capacity of LLMs under many-shot prompting conditions. Using this benchmark, we evaluate the performance of 19 LLMs with long-context capabilities across four distinct scenarios, encompassing 14 tasks. We assess these tasks under both few-shot and many-shot prompts, analyzing how LLM performance varies as we transition from few-shot to many-shot settings. This evaluation aims to investigate the scaling laws associated with ICL performance. Our findings demonstrate the significant potential of many-shot ICL to generalize to out-of-distribution problems. Furthermore, we provide insights into the selection of tasks and models that benefit most from many-shot ICL and examine the factors that may limit performance, offering strategies to improve its effectiveness.

Limitation

Our work has several limitations. First, the benchmark lacks open-ended questions, which, while challenging to evaluate, are crucial for real-world applications. Addressing this gap will be important in future research. Additionally, the length of the context significantly impacts inference speed, increasing the time required for experiments and limiting the ability to further expand the datasets. Furthermore, as context window sizes rapidly grow, the current dataset scale may become insufficient. Future work could focus on developing a pipeline that automatically generates high-quality examples across diverse domains, enabling the system to keep pace with the expanding context window. Finally, due to limited data resources, we have not accounted for potential data contamination, where examples in the training set may have already been seen by the models during pre-training or supervised instruction tuning, which requires further attention.

References

The claude 3 model family: Opus, sonnet, haiku.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Rishabh Agarwal, Avi Singh, Lei M. Zhang, Bernd Bohnet, Luis Rosias, Stephanie Chan, Biao Zhang, Ankesh Anand, Zaheer Abbas, Azade Nova, John D. Co-Reyes, Eric Chu, Feryal Behbahani, Aleksandra Faust, and Hugo Larochelle. 2024. Many-shot in-context learning. *Preprint*, arXiv:2404.11018.

Shengnan An, Zeqi Lin, Qiang Fu, Bei Chen, Nanning Zheng, Jian-Guang Lou, and Dongmei Zhang. 2023. How do in-context examples affect compositional generalization? *Preprint*, arXiv:2305.04835.

Jinheon Baek, Sun Jae Lee, Prakhar Gupta, Geunseob Oh, Siddharth Dalmia, and Prateek Kolhar. 2025. Revisiting in-context learning with long context language models. *Preprint*, arXiv:2412.16926.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024. Longbench: A bilingual, multitask benchmark for long context understanding. *Preprint*, arXiv:2308.14508.

Amanda Bertsch, Uri Alon, Graham Neubig, and Matthew Gormley. 2024a. Unlimiformer: Long-range transformers with unlimited length input. *Advances in Neural Information Processing Systems*, 36.

Amanda Bertsch, Maor Ivgi, Uri Alon, Jonathan Berant, Matthew R. Gormley, and Graham Neubig. 2024b. In-context learning with long-context models: An in-depth exploration. *Preprint*, arXiv:2405.00200.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. 2020. Efficient intent detection with dual sentence encoders. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 38–45, Online. Association for Computational Linguistics.

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. 2023. Extending context window of large language models via positional interpolation. *Preprint*, arXiv:2306.15595.

Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. 2021. Dialogsum: A real-life scenario dialogue summarization dataset. *Preprint*, arXiv:2105.06762.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Yiran Ding, Li Lyna Zhang, Chengruidong Zhang, Yuanyuan Xu, Ning Shang, Jiahang Xu, Fan Yang, and Mao Yang. 2024. Longrope: Extending llm context window beyond 2 million tokens. *Preprint*, arXiv:2402.13753.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Vladimir Eidelman. 2019. Billsum: A corpus for automatic summarization of us legislation. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, page 48–56. Association for Computational Linguistics.

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.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William W. Cohen, and Xinghua Lu. 2019. Pubmedqa: A dataset for biomedical research question answering. *Preprint*, arXiv:1909.06146.

Jinhyuk Lee, Anthony Chen, Zhuyun Dai, Dheeru Dua, Devendra Singh Sachan, Michael Boratko, Yi Luan, Sébastien M. R. Arnold, Vincent Perot, Siddharth Dalmia, Hexiang Hu, Xudong Lin, Panupong Pasupat, Aida Amini, Jeremy R. Cole, Sebastian Riedel, Iftekhar Naim, Ming-Wei Chang, and Kelvin Guu. 2024. Can long-context language models subsume retrieval, rag, sql, and more? *Preprint*, arXiv:2406.13121.

Jiaqi Li, Mengmeng Wang, Zilong Zheng, and Muhan Zhang. 2024a. Loogle: Can long-context language models understand long contexts? *Preprint*, arXiv:2311.04939.

Tianle Li, Ge Zhang, Quy Duc Do, Xiang Yue, and Wenhu Chen. 2024b. Long-context llms struggle with long in-context learning. *Preprint*, arXiv:2404.02060.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What makes good in-context examples for gpt-3? *Preprint*, arXiv:2101.06804.

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. *Preprint*, arXiv:2104.08786.

Pedro Henrique Martins, Zita Marinho, and André F. T. Martins. 2022. ∞-former: Infinite memory transformer. *Preprint*, arXiv:2109.00301.

Costas Mavromatis, Balasubramaniam Srinivasan, Zhengyuan Shen, Jiani Zhang, Huzefa Rangwala, Christos Faloutsos, and George Karypis. 2023. Which examples to annotate for in-context learning? towards effective and efficient selection. *Preprint*, arXiv:2310.20046.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on health, inference, and learning*, pages 248–260. PMLR.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. 2023. Yarn: Efficient context window extension of large language models. *arXiv preprint arXiv:2309.00071*.

Maja Popović. 2017. chrf++: words helping character n-grams. In *Proceedings of the second conference on machine translation*, pages 612–618.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2023. Gpqa: A graduate-level google-proof qa benchmark. *Preprint*, arXiv:2311.12022.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. 2024. Code llama: Open foundation models for code. *Preprint*, arXiv:2308.12950.

Mingyang Song, Mao Zheng, and Xuan Luo. 2025. Counting-stars: A multi-evidence, position-aware, and scalable benchmark for evaluating long-context large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3753–3763.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *Preprint*, arXiv:1811.00937.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume

Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *Preprint*, arXiv:2207.04672.

Szymon Tworkowski, Konrad Staniszewski, Mikołaj Pacek, Yuhuai Wu, Henryk Michalewski, and Piotr Miłoś. 2024. Focused transformer: Contrastive training for context scaling. *Advances in Neural Information Processing Systems*, 36.

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. *Preprint*, arXiv:2206.07682.

Yanzheng Xiang, Hanqi Yan, Lin Gui, and Yulan He. 2024. Addressing order sensitivity of in-context demonstration examples in causal language models. *Preprint*, arXiv:2402.15637.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Dawei Zhu, Nan Yang, Liang Wang, Yifan Song, Wenhao Wu, Furu Wei, and Sujian Li. 2024. Pose: Efficient context window extension of llms via positional skip-wise training. *Preprint*, arXiv:2309.10400.

A Prompt

In this section, we outline the prompt format used in our experiments for each task, which is adapted from the prompts provided in Agarwal et al. (2024),Baek et al. (2025), and Bertsch et al. (2024b).

News, Dialogue and Bill Summarization:

> You are an expert in article summarization. I am going to give you one or more example pairs of article and its summary in fluent English. The pairs will be written as the following format:
> Article: <article>
> Summary: <summary>
> After the example pairs, I am going to provide another article and I want you to summarize it. Give only the summary, and no extra commentary, formatting, or chattiness.
>
> {Examples}
>
> Article: <article>
> Summary:

Bemba, Kurdish, French, German Translation:

> You are an expert translator. I am going to give you one or more example pairs of text snippets where the first is in English and the second is a translation of the first snippet into {Target Language}. The sentences will be written:
> English: <first sentence>
> {Target Language}: <translated first sentence>
> After the example pairs, I am going to provide another sentence in English and I want you to translate it into {Target Language}. Give only the translation, and no extra commentary, formatting, or chattiness. Translate the text from English to {Target Language}.
>
> {Examples}
>
> English: <first sentence>
> {Target Language}:

Intent Classification:

> I am going to give you one or more example pairs of customer service query and its intent. The pairs will be written as the following format:
> service query: <query>
> intent category: <category>
> After the example pairs, I am going to provide another customer service query and I want you to classify the label of it that must be one among the intent categories provided in the examples. Give only the category, and no extra commentary, formatting, or chattiness.
> Here are all possible intent categories for classification:
>
> {Label Space}
>
> {Examples}
>
> service query: <query>
> intent category:

Topic Classification:

I am going to give you one or more example sets of question-answer pairs and the topic associated with them. The sets will be written as the following format:
Question: <question>
Answer: <answer>
Topic: <topic>
After the example sets, I am going to provide another question-answer pair and I want you to classify the label of it that must be one among the topic provided in the examples. Give only the topic, and no extra commentary, formatting, or chattiness.
Here are all possible topics for classification:

{Label Space}

{Examples}

Question: <question>
Answer: <answer>
Topic:

Sentiment Classification:

I am going to give you one or more example pairs of review and the score associated with the review. The pairs will be written as the following format:
Review: <review>
Score: <score>
After the example pairs, I am going to provide another review and I want you to classify the label of it that must be one among the score provided in the examples. Give only the score, and no extra commentary, formatting, or chattiness.
Here are all possible scores for classification:

{Label Space}

{Examples}

Review: <review>
Score:

Science, Medical Question Answering:

You are an expert in multiple-choice question answering tasks. I am going to give you one or more example pairs consisting of a question along with its solution procedure and answer in a multiple-choice question answering format. The pairs will be written as the following format:
Question: <question>
Solution: <solution>
Answer: <answer>
After the example pairs, I am going to provide another question and I want you to predict its answer. Think step by step before giving a final answer to this question and give the final answer that follows a consistent format as in the provided examples, and no extra commentary, formatting, or chattiness.

{Examples}

Question: <question>

Retrieval Question Answering:

You are an expert in multiple-choice question answering tasks. I am going to give you one or more examples, each containing a text, a question about the text, the solution procedure to derive the answer from the text, and the final answer in a multiple-choice question answering format. The examples will be written as the following format:
Text: <text>
Question: <question>
Solution: <solution>
Answer: <answer>
After the examples, I am going to provide another text and a question about the text and I want you to predict its answer. Think step by step before giving a final answer to this question and give the final answer that follows a consistent format as in the provided examples, and no extra commentary, formatting, or chattiness.

{Examples}

Text: <text>
Question: <question>

Commonsense Question Answering:

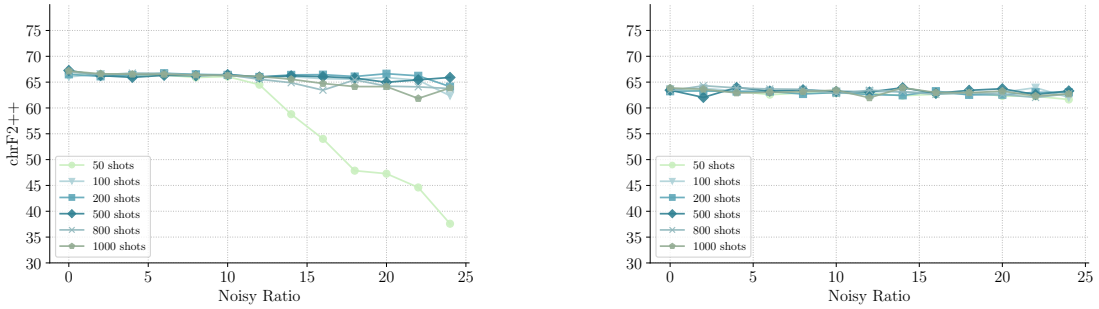
You are an expert in multiple-choice question answering tasks. I am going to give you one or more example pairs of question and its answer in a multiple-choice question answering format. The pairs will be written as the following format:
Question: <question>
Answer: <answer>
After the example pairs, I am going to provide another question and I want you to predict its answer. Give only the answer that follows a consistent format as in the provided examples, and no extra commentary, formatting, or chattiness.
{Examples}

Question: <question>
Answer:

B Full Results

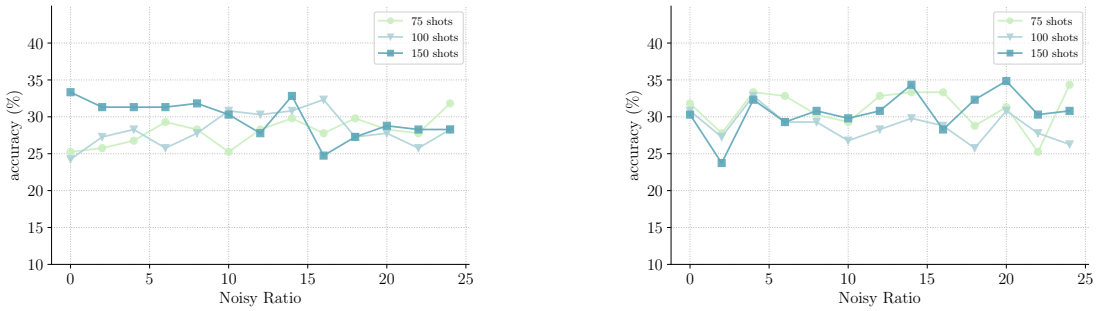
To save space, we present only 1-2 representative results for analysis in the paper. In this section, we provide additional results from the experiments.

B.1 Supplementary Results from Experiments on Noise Ratio



(a) Llama-3.1-8B-Instruct (b) Qwen2.5-7B-Instruct

Figure 5: Performance Change with Increasing Noise Ratio on French Transaltion task.



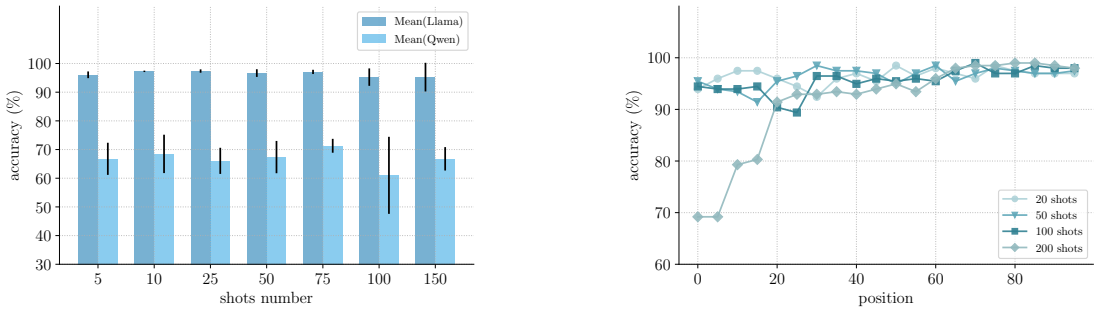
(a) Llama-3.1-8B-Instruct (b) Qwen2.5-7B-Instruct

Figure 6: Performance Change with Increasing Noise Ratio on Science Question Answering task.

As shown in Figure 5 and Figure 6, model performance demonstrates robustness to noise in both French Translation and Science Question Answering tasks within the many-shot scenario, particularly when the noise ratio is below 10%, which is common in real-world applications. Furthermore, although performance in the French Translation task does not show significant improvement, there is a notable enhancement in robustness to noisy input.

B.2 Supplementary Results from Experiments on Input Utilization

As illustrated in Figure 7, the Science Question Answering task shows patterns similar to the Bemba Translation task discussed in the paper, where performance does not reach 100% and declines in the many-shot scenario. As the position of the query is gradually shifted from the beginning to the end of the prompt, performance significantly decreases when the query is positioned earlier. Moreover, the copying ability of LLaMA-3.1-8B-Instruct appears to be much stronger than that of Qwen2.5-7B-Instruct, as evidenced by both the Bemba Translation and Science Question Answering results.



(a) Science

(b) Llama-3.1-8B-Instruct

Figure 7: The left graph shows the performance of test queries incorporated into prompts in the Science Question Answering task, while the right graph presents the performance variations based on query position for Llama-3.1-8B-Instruct.