Evaluating Large Language Models with Enterprise Benchmarks

Bing Zhang[*1], **Mikio Takeuchi**[*2], **Ryo Kawahara**[*2], **Shubhi Asthana**[1],
Md. Maruf Hossain[†2], **Guang-Jie Ren**[†3], **Kate Soule**[4], **Yifan Mai**[5] **Yada Zhu**[‡4],
[1]IBM Almaden Research Lab, CA, USA [2]IBM Research - Tokyo, Japan
[3]Adobe, CA, USA [4]MIT-IBM Watson AI Lab, MA, USA [5]Stanford University, CA, USA
bing.zhang@ibm.com, {mtake,ryokawa}@jp.ibm.com, sasthan@us.ibm.com,
w_maruf@outlook.com, gren@adobe.com, kate.soule@ibm.com,
yifan@cs.stanford.edu, yzhu@us.ibm.com

Abstract

The advancement of large language models (LLMs) has led to a greater challenge of having a rigorous and systematic evaluation of complex tasks performed, especially in enterprise applications. Therefore, LLMs need to be benchmarked with enterprise datasets for a variety of NLP tasks. This work explores benchmarking strategies focused on LLM evaluation, with a specific emphasis on both English and Japanese. The proposed evaluation framework encompasses 25 publicly available domain-specific English benchmarks from diverse enterprise domains like financial services, legal, climate, cybersecurity, and 2 public Japanese finance benchmarks. The diverse performance of 8 models across different enterprise tasks highlights the importance of selecting the right model based on the specific requirements of each task. Code and prompts are available on GitHub.

1 Introduction

Large Language Models (LLMs) have garnered significant attention and adoption across various domains due to their remarkable capabilities in natural language understanding and generation. To align with the new era of LLMs, new benchmarks have been proposed recently to probe a diverse set of LLM abilities. For example, BIG-bench (Beyond the Imitation Game benchmark) (Srivastava et al., 2022) and HELM (Holistic Evaluation of Language Models) (Liang et al., 2022) attempt to aggregate a wide range of natural language processing (NLP) tasks for holistic evaluation. Towards the application of LLMs in real world, it is expected that LLMs are capable of processing enterprise text data, which is generated and accumulated through business operations of enterprises. An important characteristics of such data is that it often contain expressions used in specific domains such as finance, legal, climate, and cybersecurity. However, the existing benchmarks often lack domain-specific datasets, particularly for those enterprise domains. This gap poses challenges for practitioners seeking to assess LLM performance tailored to their needs.

Enterprise datasets, though potentially useful as benchmarks, often face accessibility or regulatory issues. Evaluating LLMs with these datasets can be difficult due to sophisticated concepts or techniques needed to convert use case-based inputs to the standard input format of evaluation harness (e.g., BIG-bench or HELM), which indicates the need for standardized metrics and clear performance benchmarks. This highlights the necessity for robust evaluation frameworks that measure LLM performance in specialized domains.

Emerging enterprise-focused or domain-specific LLMs, such as Snowflake Arctic[1] and BloombergGPT (Wu et al., 2023), are evaluated with limited enterprise application scope and volume. For textual inputs, Snowflake Arctic is assessed on world knowledge, common sense reasoning, and math. However, such non-domain-specific benchmarks often fail to address the complexities of enterprise applications, such as financial Named Entity Recognition (NER), which requires precise domain language understanding. BloombergGPT is evaluated with several finance datasets, mostly proprietary, and does not include the summarization task.

Beyond the gaps in English LLM enterprise benchmarking, there are additional challenges in the availability and development of such benchmarks in other languages, especially Japanese. This gap includes a lack of comprehensive, high-quality datasets tailored specifically to Japanese financial

[*]Equal contribution.
[†]The contribution was made during employment at IBM Research.
[‡]Corresponding author.

[1]https://www.snowflake.com/blog/arctic-open-efficient-foundation-language-models-snowflake/

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Industry Track), pages 485–505

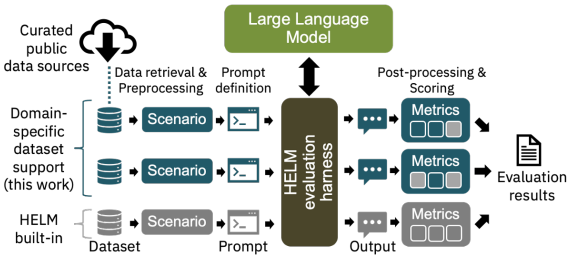

Figure 1: Overview of the enterprise benchmark framework for LLM evaluation.

terminology, regulations, and market dynamics. Additionally, there is limited benchmarking for tasks such as sentiment analysis, risk assessment, and financial forecasting in the Japanese context.

To narrow the gap between LLM development and evaluation in enterprises, we present a framework in Figure 1 by augmenting Stanford's HELM with emphasizes the use of enterprise benchmarks that cater specifically to domains such as finance, legal, climate, and cybersecurity. This framework aims to create and adopt standardized benchmarks reflecting real-world application requirements. This initiative not only addresses the current scarcity of domain-specific evaluation frameworks but also informs better decisions for deploying and optimizing LLM technologies across diverse enterprise environments.

Together, our work makes the following key contributions: **(i)** Developing a set of domain-specific benchmarks by curating datasets, enhancing metrics, and implementing prompts based on industry use cases and requirements. **(ii)** Conducting extensive experiments to demonstrate that LLMs show different performance trends in domain-specific settings. **(iii)** Enabling researchers and industry practitioners to assess and optimize LLMs tailored to specific domains by integrating the prompts and benchmark code into the widely adopted HELM evaluation harness. This paper does not aim to provide an exhaustive evaluation of LLM performance across all enterprise benchmarks; instead, it focuses on the evaluation process of LLMs in different domains.

In the next section, we delve into the current state-of-the-art LLM evaluation benchmarks. In Section 3, we introduce 27 enterprise datasets in four enterprise domains. Section 4 describes the key design considerations in the development of the benchmark. Experiments and primary results are presented in Section 5. The paper concludes in Section 6.

2 Related Work

Recently, researchers have developed several frameworks to assess the various capabilities of LLMs. Examples include HELM (Bommasani et al., 2023), MMLU (Hendrycks et al., 2020), Big-Bench (Lewkowycz et al., 2022), EleutherAI (Phang et al., 2022), and MMCU (Zeng, 2023), which are widely used to evaluate LLMs on multiple NLP tasks. Specifically, HELM categorizes potential scenarios and metrics of interest for LLMs. However, these frameworks lack benchmarks and metrics for assessing LLM performance in enterprise-focused problems. This work leverages the HELM platform, extending its benchmark scenarios and metrics to include domain-specific LLM evaluations.

Researchers are actively developing enterprise-specific LLM benchmarks in domains like finance, legal, and cybersecurity. For example, FinBen (Xie et al., 2024) introduces a finance-focused benchmark spanning 24 tasks, including information extraction, question answering, and risk management. However, its design is tailored to Chinese language tasks, limiting its applicability to English texts and American market data. Similarly, Xu et al. (Xu et al., 2024) provides an extensive analysis of finance-specific tasks, covering six domains and 25 specialized tasks in Chinese. Zhu et al. (Zhu et al., 2024) further propose CFLUE, the Chinese Financial Language Understanding Evaluation benchmark, but its relevance to non-Chinese languages remains constrained.

In another effort, Hirano (Hirano, 2024) makes an initial attempt to build a benchmark for Japanese financial tasks, including performance evaluations for several models. While promising, this benchmark lacks the depth and task diversity seen in Xu et al.'s comprehensive Chinese evaluation, highlighting the need for further exploration of Japanese-specific tasks for more robust assessments.

Enterprise benchmarks in legal are upcoming with works like Legalbench (Guha et al., 2024), Lawbench (Fei et al., 2023), and LAiW (Dai et al., 2023). Lawbench is evaluated on multilingual and Chinese-oriented LLMs while LAiW is the Chinese legal LLMs benchmark. Legalbench provides a benchmark on reasoning while the others evaluate legal foundation inference and complex legal

application tasks.

Lastly, in the cybersecurity domain, researchers have contributed to benchmarks like SEvenLLM (Ji et al., 2024), CyberBench (Liu et al.), Cyberseceval 2 (Bhatt et al., 2024) and CyberMetric (Tihanyi et al., 2024). These benchmarks analyze tasks like cyber advisory and reasoning, question-answering, and cybersecurity incident analysis. Compared to existing benchmarks, our enterprise benchmarks perform sentiment analysis and summarization tasks that have not been tackled in existing art. The benchmarks in our work are open-sourced and consolidated into a widely adopted evaluation framework to enable comprehensive evaluation across reasoning tasks.

3 Enterprise Benchmarks

Domain	Task	Dataset
Finance	Classification	Earnings Call Transcripts (Roozen and Lelli, 2021)
	Classification	News Headline (Sinha and Khandait, 2020)
	NER	Credit Risk Assessment (Salinas Alvarado et al., 2015)
	NER	KPI-Edgar (Deußer et al., 2022)
	NER	FiNER-139 (Loukas et al., 2022)
	QA	Opinion-based QA (FiQA) (Maia et al., 2018)
	QA	Sentiment Analysis (FiQA SA) (Maia et al., 2018)
	QA	Insurance QA (Feng et al., 2015)
	Summarization	Financial Text Summarization (EDT) (Zhou et al., 2021)
Finance (Jpn.)	Classification	MultiFin (Jørgensen et al., 2023)
	Summarization	Bank of Japan Outlook (Bank of Japan, 2024)
	E2J Translation	(same as above)
	J2E Translation	(same as above)
Legal	Classification	Legal Sentiment Analysis [2]
	Classification	UNFAIR-ToS (Lippi et al., 2019)
	Classification	Legal Judgement Prediction (Chalkidis et al., 2019)
	QA	CaseHOLD (Zheng et al., 2021)
	Summarization	BillSum (Eidelman, 2019)
	Summarization	Legal Summarization (Manor and Li, 2019)
Climate	Classification	Reddit Climate Change [3]
	Classification	Wildfires and Climate Change Tweets [4]
	Summarization	SUMO Climate Claims (Mishra et al., 2020)
Cyber-security	Classification	SPEC5G (Karim et al., 2023)
	Classification	CTI-to-MITRE with NLP (Orbinato et al., 2022)
	Classification	TRAM [5]
	Classification	SecureNLP (Phandi et al., 2018)
	Classification	IoTSpotter (Jin et al., 2022)
	Summarization	SPEC5G (Karim et al., 2023)

Table 1: List of benchmarks.

This work introduces benchmark datasets from four specific domains (Table 1), where natural language understanding is crucial for productivity and decision-making. All datasets are curated from open data sources to cover a broad range of natural language tasks and diverse industry use cases within these domains. Datasets without reference answers or with fewer than 100 test cases were excluded from the benchmarks.

Although the collected tasks are mostly conventional, the combination of such tasks and domain-specific datasets are still rare and understudied in the field of LLM applications. The focus of this paper is in catering a means for practitioners to evaluate the performance of processing domain-specific datasets. This is because it is known that a general domain LLM might suffer from the degradation of performance when it processes domain-specific data due to the unique terminology and knowledge that are only used in a specific industry.

As summarized in Appendix A.1/Table 6, the English finance benchmarks include 10 datasets collected from important use cases such as market prediction based on earnings call transcripts, entity recognition for retrieving information from U.S. Securities and Exchange Commission (SEC) filings, and understanding news and reports. The tasks range from classification and NER to QA and long document summarization. NER is crucial for many applications in digital finance, and numerical NER is a particularly challenging task for language models. ConvFinQA provides multi-turn conversational financial QA data involving information extraction from tables and numerical reasoning, offering a critical lens for evaluating LLMs' numerical reasoning capabilities.

As summarized in Appendix A.1/Table 7, the Japanese finance benchmarks encompass several datasets tailored to crucial use cases within the financial sector. These use cases include classification using the MultiFin dataset, which covers financial article headlines in multiple languages; summarization utilizing the Bank of Japan Outlook dataset, which provides insights from quarterly monetary policy meetings; and translation tasks for both English to Japanese and Japanese to English, exploiting the same dataset. LLM performance in multilingual settings is an important concern in enterprise use cases, and a translation is a typical task that represents the demands in such settings.

Similarly, the seven legal benchmarks in Ap-

[2] https://osf.io/zwhm8/

[3] https://huggingface.co/datasets/SocialGrep/the-reddit-climate-change-dataset

[4] https://github.com/reabdi/WildFiresTopicModeling/tree/master/DataSet

[5] https://github.com/center-for-threat-informeddefense/tram

pendix A.1/Table 8 contain rich NLP tasks and important use cases, such as legal opinion classification, legal judgment prediction, and legal contract summarization. Climate is an emerging domain for LLM applications, including summarizing claims and understanding human concerns like wildfires and climate change. Given the scarcity of open-source datasets with high-quality labels, three benchmarks have been curated, as detailed in Appendix A.1/Table 9. Cybersecurity-related tasks, including classification and summarization of textual documents such as network protocol specifications, malware reports, vulnerability, and threat reports are curated and shown in Appendix A.1/Table 10.

4 Benchmark Development

Recent LLMs, primarily based on the decoder-only transformer architecture, have unique capabilities and limitations, such as in-context learning (few-shot learning) and input token length constraints. Domain-specific benchmark datasets are often designed for different architectures (such as BERT), necessitating adaptations in datasets and task implementations.

In HELM, a *scenario* represents an evaluation task with a specific dataset and corresponding metrics. These adaptations are incorporated into the development of the scenarios. The prompt for each scenario is included in the Appendix A.3. The developed scenarios are adopted to a specific edition of HELM, called HELM Classic, which collects the largest number of NLP scenarios among the HELM editions. In this study, HELM v0.4.0 is used.

4.1 Classification Task

In a classification task, a model is asked to generate the name of a class of the input sample directly as an output. It is better to use natural language words as the class names (e.g., positive/neutral/negative) than to use symbolic names (see the discussion in Section 4.2). One usually needs to provide few-shot examples to ensure that a model does not generate tokens other than the class names.

For classification tasks with more than 20 classes, defining all classes in a prompt and covering them in in-context learning examples is challenging due to input token length limits. This work simplifies the task by selecting samples that belong to the top-k classes based on their distributions, where k is typically less than 10. Related topics on the estimation of the token consumption and other possible implementation options are discussed in Appendix A.5.

In addition to HELM's built-in micro- and macro-F1 scores, the Weighted F1 score as implemented in (scikit-learn developers, 2024) is added as a performance metric.

4.2 Named Entity Recognition Task

A conventional NER task is formalized as a sequence-to-sequence task, where the input is a sequence of tokens. A system classifies whether each token is a part of a named entity and identifies its category (e.g., person, location, organization, etc.). Then the system generates a sequence of corresponding tags (so-called BIO tags) in the same order as the input tokens (Cui et al., 2021). However, in our preliminary experiments, this approach did not work well with LLMs. This seems to be because BIO tags are unknown to pre-trained LLMs.

Due to the challenges, alternative implementation methods are discussed in Appendix A.6. In this work, a simplified approach (Wu et al., 2023) is employed. In this approach, a model extracts only named entities and their categories in a natural language (e.g., "New York (location), John Smith (person)".). In some scenarios, the number of categories is reduced, as explained in the previous Section 4.1.

To support the above extraction-based NER, a new metric called Entity F1 is added. For each test sample, predicted named entities and the categories of those are compared with those in the ground-truth, to compute true positives, false positives, and false negatives. Those are aggregated population-wide to compute the Entity F1 score.

4.3 Question and Answering Task

There are several types of QA tasks, some of which overlap with information retrieval tasks. In many business applications, one is requested to answer a question based on a given set of documents (e.g., product manuals, FAQs, medical papers, regulations, etc.). This involves a ranking of answer candidates with respect to their relevance to the user's question. However, LLMs struggle with these operations because handling multiple answer candidates in a single prompt consumes many tokens.

Alternatively, the "point-wise" approach provided in HELM is adopted (Liang et al., 2022). For a question q_i, there are k pre-defined answer candidates $\{a_{ij}|j = 1, \cdots k\}$ and one prompts the fol-

lowing question to a model: "Does a_{ij} answer the question q_i? Answer in yes or no." Then, one can obtain a pair of the output text $b_{ij} \in$ {"yes", "no"} and its log probability c_{ij} from the model. An answer candidate with "yes" and higher c_{ij} is ranked higher, while "no" and higher c_{ij} is ranked lower.

4.4 Summarization Task

In a summarization task, one needs to handle a long document as an input. Therefore, the input token length limit becomes a severe issue. In this study, this issue is handled by selecting relatively shorter samples and truncating the end of the samples to preserve the original context as much as possible. For the English benchmark evaluation, performance is measured using conventional ROUGE scores (see also Section 4.5 for Japanese tasks).

4.5 Supporting Non-English Datasets and Tasks

In this study, some of the Japanese datasets are supported as examples of extending the model evaluation capability to non-English languages. There are some considerations in implementing the non-English language support.

First, most of the Japanese LLMs are fine-tuned with Japanese instruction data to improve the instruction-following capability in that language. In addition, models often require the use of model-specific system prompts. Therefore, the instruction in a prompt is set to Japanese. The use of the model-specific system prompt is also examined and the best prompt of a scenario is selected for each model.

Second, the language of the labels is also assumed to be Japanese. This is because an LLM often exploits its knowledge about the label as a natural language phrase.

Third, language-specific metrics need to be introduced. In particular, the use of a language-specific tokenizer is crucial to accurately compute the metrics. Implementation details of the language-specific metrics are described in Appendix A.7.

5 Experiments and Results

This evaluation is conducted by augmenting HELM's framework to encompass 27 publicly available datasets from multiple domains, namely financial, legal, climate and cybersecurity. For each benchmark, the evaluation is conducted on a specific configuration. The intention of this section is to demonstrate the usefulness for practitioners of our benchmarks in evaluating candidate models with their own settings.

5.1 Evaluated Models

Here, the evaluation models are selected from the best-performing open-sourced models under 70 billion parameters based on model size, type of training data, accessibility, and model tuning method. Specifically, 1) **Llama 3.1** (Dubey et al., 2024) is a collection of LLMs optimized for multilingual dialogue use cases and outperforms many of the available open-source and proprietary models on common industry benchmarks. In this study, we use 8 and 70-billion-parameter instruction-tuned models. 2) **Flan UL2** (Tay et al., 2022) is another state-of-the-art model that has been pre-trained with a framework that combines diverse pre-training paradigms. This is the only encoder-decoder Transformer model among the models we tested. 3) **Phi 3.5** (Abdin et al., 2024) is a family of powerful and small language models (SLMs) with a modern architecture that supports a long context window of 128k tokens. 4) **Mistral 7B** (Jiang et al., 2023) is a series of 7-billion-parameter language models. This version (v0.3) supports function calling and relatively a long context length of 32k tokens. 5) **Granite 3** (Granite Team, 2024) is a set of the latest open-sourced enterprise-focused models. The datasets used in the training of these models include some finance and legal datasets, such as FDIC, Finance Text Books, EDGAR Filings, etc. 6) **Granite 8B Japanese** is an instruction-tuned model and is designed and developed with the same philosophy of the Granite model stated above and then tailored for Japanese. 7) **Llama 3 ELYZA JP 8B** model is based on the llama-3-8b-instruct model, which has been enhanced for Japanese usage through additional pre-training and instruction tuning. Other information about the models is summarized in Table 2.

Model	Context length	Release date
phi-3-5-mini-instruct (3.8b)	131072	2024-08-01
mistral-7b-instruct-v0-3	32768	2024-05-22
llama-3-1-8b-instruct	131072	2024-07-23
llama-3-1-70b-instruct	131072	2024-07-23
granite-3-8b-instruct	4096	2024-10-21
flan-ul2 (20b)	4096	2023-02-28
granite-8b-japanese	4096	2024-02-29
llama-3-elyza-jp-8b	4096	2024-06-26

Table 2: Model information

All 8 models are evaluated in our benchmarks, regardless of the purposes of the models (i.e., for

chat, etc.). As we will see in the following sections, the relation between the performance of a task and the intended purpose of a model is not straightforward.

5.2 Evaluation Setup

In this study, the data source-provided train and test splits are used whenever possible. Model performance is reported based on test or validation examples, depending on the availability of test labels. If the train and test splits do not exist, a task-specific ratio of the data is selected as the test split, with the remainder used as the train split.

In-context learning examples are sampled from the train split. The number of few-shot examples provided to the model varies by task and is detailed in Table 3, and Appendix A.2/Tables 4 - 13. Note that, in HELM, only one set of randomly sampled examples is used across all test cases of a given benchmark. For in-context learning, this work adopts HELM's sampling strategy, which includes samples from minority classes. This is different from the conventional uniformly random sampling, where samples in a minority class tend to be ignored in the case of a few-shot sampling.

For the current evaluation, all the models use the same parameters and the same context examples. The prompts used are shown in Appendix A.3. To ensure reproducibility, a fixed random seed and the greedy decoding method (i.e., temperature zero) without repetition penalty are used. Standard text normalization (i.e., moving articles, extra white spaces, and punctuations followed by lowering cases) is applied to the generated output before matching texts.

5.3 Evaluation Results

English Finance Benchmark

Table 3 provides the evaluation results of 6 models across a range of financial NLP tasks, including classification, NER, QA, and summarization. Each task was assessed using the best-fitted metrics to determine the performance of different models.

For classification tasks, the highest Weighted F1 scores were achieved by the llama-3-1-70b-instruct model in the Earnings Call Transcripts classification and the News Headline classification demonstrating its strong performance in extracting relevant information from earnings calls as well as indicating its effectiveness in handling short text classification tasks.

NER was evaluated using three different tasks: Credit Risk Assessment, KPI-Edgar, and FiNER-139. The llama-3-1-70b-instruct model outperformed others in all three tasks showcasing its capability in identifying financial entities accurately.

Among the diverse QA tasks, the llama-3-1-70b-instruct model excelled in FiQA-Opinion and ConvFinQA with the highest RR scores and the highest accuracy, respectively highlighting their proficiency in answering complex questions with limited context as well as indicating its robustness in handling multi-turn financial QA tasks involving numerical reasoning. The granite-3-8b-instruct model obtained the highest Weighted F1 score in FiQA SA, The flan-ul2 model excelled in Insurance QA with the highest RR scores.

For Text Summarization, the llama-3-1-8b-instruct model achieved the highest Rouge-L score, demonstrating its ability to generate concise and relevant summaries from financial texts.

Legal Benchmark The results in Appendix A.2/Table 4 highlight the performance of various models across legal tasks. For classification, the mistral-7b-instruct-v0-3 model achieved the highest score in Legal Sentiment Analysis (Weighted F1 of 0.727), the llama-3-1-70b-instruct model excelled in UNFAIR-ToS (Weighted F1 of 0.824), while mistral-7b-instruct-v0-3 led in Legal Judgement Prediction (Weighted F1 of 0.845). In QA, llama-3-1-70b-instruct achieved the highest F1 score (0.816) in the CaseHOLD task. The granite-3-8b-instruct model was best in summarization tasks, such as BillSum (Rouge-L of 0.312) and Legal Summarization (Rouge-L of 0.271).

Results of other domains are summarized in Appendix A.2. Across all domains, the results indicate that different models excel in various tasks depending on their training process and architecture.

How these results differ from the case of general (non-domain-specific) NLP performance is summarized in Table 5 in the case of the summarization tasks, as well as discussed in detail in Appendix A.4. We usually expect that larger models in terms of the parameter sizes perform better. However, for example, flan-ul2 (20B) shows large drops of relative performance in some of the legal benchmarks, while granite-3-8b-instruct keeps stable performance there, possibly due to to the difference of the training datasets. This kind of observation is particularly useful when there are requirements on the inference cost or the latency, which are correlated with the parameter sizes.

Task	Classification		Named Entity Recognition			Question Answering				Summarization
	Earnings Call Transcripts	News Headline	Credit Risk Assessment	KPI-Edgar	FiNER-139	FiQA-Opinion	Insurance QA	FiQA SA	Conv-FinQA	EDT
Metrics	Weighted F1	Weighted F1	Entity F1	Adj F1	Entity F1	RR@10	RR@5	Weighted F1	Accuracy	Rouge-L
N-shot Prompt	5-shot	5-shot	20-shot	20-shot	10-shot	5-shot	5-shot	5-shot	1-shot	5-shot
phi-3-5-mini-instruct (3.8b)	0.411	0.800	0.417	0.421	0.677	0.605	0.350	0.824	0.277	0.368
mistral-7b-instruct-v0-3	0.453	0.794	0.396	0.588	0.686	0.569	0.414	0.838	0.280	0.390
llama-3-1-8b-instruct	0.411	0.838	0.473	0.563	0.772	0.624	0.388	0.835	0.531	**0.435**
llama-3-1-70b-instruct	**0.602**	**0.874**	**0.539**	**0.697**	**0.802**	**0.808**	0.645	0.855	**0.629**	0.394
granite-3-8b-instruct	0.411	0.791	0.332	0.571	0.706	0.701	0.388	**0.859**	0.296	0.412
flan-ul2 (20b)	0.411	0.829	0.259	0.011	0.446	0.804	**0.723**	0.811	0.254	0.428

Table 3: Finance benchmark evaluation results per task.

Task	Classification			Question Answering	Summarization	
	Legal Sentiment Analysis	UNFAIR-ToS	Legal Judgement Prediction	CaseHOLD	BillSum	Legal Summarization
Metrics	Weighted F1	Weighted F1	Weighted F1	F1	Rouge-L	Rouge-L
N-shot Prompt	5-shot	5-shot	5-shot	2-shot	0-shot	0-shot
phi-3-5-mini-instruct (3.8b)	0.594	0.464	0.739	0.767	0.311	0.205
mistral-7b-instruct-v0-3	**0.727**	0.720	**0.845**	0.696	**0.312**	0.255
llama-3-1-8b-instruct	0.652	0.592	0.794	0.723	0.282	0.252
llama-3-1-70b-instruct	0.703	**0.824**	0.839	**0.816**	0.291	0.228
granite-3-8b-instruct	0.705	0.485	0.616	0.800	**0.312**	**0.271**
flan-ul2 (20b)	0.646	0.302	0.073	0.780	0.234	0.173

Table 4: Legal benchmark evaluation results per task.

Scenario	CNN-DM		EDT		Legal Summ.	
Domain	General		Finance		Legal	
N-shot Prompt	5-shot		5-shot		0-shot	
Metrics	[R-L]	rank	rank	(Δ)	rank	(Δ)
phi-3-5-mini-instruct (3.8b)	[0.237]	6	6	(0)	5	(-1)
mistral-7b-instruct-v0-3	[0.263]	5	5	(0)	2	(-3)
granite-3-8b-instruct	[0.270]	4	3	(-1)	1	(-3)
llama-3-1-8b-instruct	[0.273]	3	1	(-2)	3	(0)
flan-ul2 (20b)	[0.299]	1	2	(+1)	6	(+5)
llama-3-1-70b-instruct	[0.276]	2	4	(+2)	4	(+2)

Table 5: Comparison with non-domain-specific data: Summarization task. The number of test samples in CNN-DM is 54. The metrics of this task is Rouge-L [R-L]. The difference of a rank on each benchmark from the rank on CNN-DM is indicated as (Δ).

These evaluations underscore the importance of selecting the appropriate model based on the specific requirements and nature of the task at hand. The diversity in performance also highlights the potential for further model optimization and specialization in these domains.

6 Conclusion

In summary, this work advances the evaluation of LLMs in domain-specific contexts by consolidating benchmark datasets and incorporating unique performance metrics into Stanford's HELM framework. This enables researchers and industry practitioners to assess and optimize LLMs for specific domains. This work demonstrated that one can get non-trivial evaluation results that are not expected from general-purpose NLP benchmarks. This was done on widely used 18 LLMs through extensive experiments on 27 publicly available benchmarks in financial, legal, climate, and cybersecurity domains, providing practical prompts for practitioners. Our analysis offers valuable insights and highlights future needs for benchmarking LLMs in specialized applications.

For the deployment of this work, we open-sourced the code and prompts. In addition, a merge of the benchmark into the HELM repository is ongoing to facilitate community adoption of this work.

Acknowledgments

This work is funded by IBM Research and MIT-IBM Watson AI Lab. We would like to thank David Cox and Rameswar Panda for their guidance, Naoto Satoh, Futoshi Iwama, and Alisa Arno for their constructive comments. The views and conclusions are those of the authors and should not be interpreted as representing those of IBM or the government.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav

Chaudhary, Dong Chen, Dongdong Chen, Weizhu Chen, Yen-Chun Chen, Yi-Ling Chen, Hao Cheng, Parul Chopra, Xiyang Dai, Matthew Dixon, Ronen Eldan, Victor Fragoso, Jianfeng Gao, Mei Gao, Min Gao, Amit Garg, Allie Del Giorno, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Wenxiang Hu, Jamie Huynh, Dan Iter, Sam Ade Jacobs, Mojan Javaheripi, Xin Jin, Nikos Karampatziakis, Piero Kauffmann, Mahoud Khademi, Dongwoo Kim, Young Jin Kim, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Xihui Lin, Zeqi Lin, Ce Liu, Liyuan Liu, Mengchen Liu, Weishung Liu, Xiaodong Liu, Chong Luo, Piyush Madan, Ali Mahmoudzadeh, David Majercak, Matt Mazzola, Caio César Teodoro Mendes, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Liliang Ren, Gustavo de Rosa, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Yelong Shen, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Praneetha Vaddamanu, Chunyu Wang, Guanhua Wang, Lijuan Wang, Shuohang Wang, Xin Wang, Yu Wang, Rachel Ward, Wen Wen, Philipp Witte, Haiping Wu, Xiaoxia Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Jilong Xue, Sonali Yadav, Fan Yang, Jianwei Yang, Yifan Yang, Ziyi Yang, Donghan Yu, Lu Yuan, Chenruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

Bank of Japan. 2024. Outlook for economic activity and prices.

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

Manish Bhatt, Sahana Chennabasappa, Yue Li, Cyrus Nikolaidis, Daniel Song, Shengye Wan, Faizan Ahmad, Cornelius Aschermann, Yaohui Chen, Dhaval Kapil, et al. 2024. Cyberseceval 2: A wide-ranging cybersecurity evaluation suite for large language models. *arXiv preprint arXiv:2404.13161*.

Rishi Bommasani, Percy Liang, and Tony Lee. 2023. Holistic evaluation of language models. *Annals of the New York Academy of Sciences*, 1525(1):140–146.

Ilias Chalkidis, Ion Androutsopoulos, and Nikolaos Aletras. 2019. Neural legal judgment prediction in English. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4317–4323, Florence, Italy. Association for Computational Linguistics.

Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. 2022. Convfinqa: Exploring the chain of numerical reasoning in conversational finance question answering. *Preprint*, arXiv:2210.03849.

Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. 2021. Template-based named entity recognition using bart. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1835–1845.

Yongfu Dai, Duanyu Feng, Jimin Huang, Haochen Jia, Qianqian Xie, Yifang Zhang, Weiguang Han, Wei Tian, and Hao Wang. 2023. Laiw: A chinese legal large language models benchmark (a technical report). *arXiv preprint arXiv:2310.05620*.

Tobias Deußer, Syed Musharraf Ali, Lars Hillebrand, Desiana Nurchalifah, Basil Jacob, Christian Bauckhage, and Rafet Sifa. 2022. KPI-EDGAR: A novel dataset and accompanying metric for relation extraction from financial documents. In *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE.

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.

Vladimir Eidelman. 2019. Billsum: A corpus for automatic summarization of us legislation. In *Proceed-*

ings of the 2nd Workshop on New Frontiers in Summarization. Association for Computational Linguistics.

Zhiwei Fei, Xiaoyu Shen, Dawei Zhu, Fengzhe Zhou, Zhuo Han, Songyang Zhang, Kai Chen, Zongwen Shen, and Jidong Ge. 2023. Lawbench: Benchmarking legal knowledge of large language models. *arXiv preprint arXiv:2309.16289*.

Minwei Feng, Bing Xiang, Michael R. Glass, Lidan Wang, and Bowen Zhou. 2015. Applying deep learning to answer selection: A study and an open task. *Preprint*, arXiv:1508.01585.

IBM Granite Team. 2024. Granite 3.0 language models.

Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, et al. 2024. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in Neural Information Processing Systems*, 36.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Masanori Hirano. 2024. Construction of a japanese financial benchmark for large language models. *arXiv preprint arXiv:2403.15062*.

Hangyuan Ji, Jian Yang, Linzheng Chai, Chaoren Wei, Liqun Yang, Yunlong Duan, Yunli Wang, Tianzhen Sun, Hongcheng Guo, Tongliang Li, et al. 2024. Sevenllm: Benchmarking, eliciting, and enhancing abilities of large language models in cyber threat intelligence. *arXiv preprint arXiv:2405.03446*.

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.

Xin Jin, Sunil Manandhar, Kaushal Kafle, Zhiqiang Lin, and Adwait Nadkarni. 2022. Understanding iot security from a market-scale perspective. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 1615–1629.

Rasmus Jørgensen, Oliver Brandt, Mareike Hartmann, Xiang Dai, Christian Igel, and Desmond Elliott. 2023. MultiFin: A dataset for multilingual financial NLP. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 894–909, Dubrovnik, Croatia. Association for Computational Linguistics.

Imtiaz Karim, Kazi Samin Mubasshir, Mirza Masfiqur Rahman, and Elisa Bertino. 2023. Spec5g: A dataset for 5g cellular network protocol analysis. *arXiv preprint arXiv:2301.09201*.

Aitor Lewkowycz, Ambrose Slone, Anders Andreassen, Daniel Freeman, Ethan S Dyer, Gaurav Mishra, Guy Gur-Ari, Jaehoon Lee, Jascha Sohl-dickstein, Kristen Chiafullo, et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. Technical report, Technical report.

Zhuowan Li, Cheng Li, Mingyang Zhang, Qiaozhu Mei, and Michael Bendersky. 2024. Retrieval augmented generation or long-context llms? a comprehensive study and hybrid approach.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D. Manning, Christopher Ré, Diana Acosta-Navas, Drew A. Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. 2022. Holistic evaluation of language models. *Preprint*, arXiv:2211.09110.

Marco Lippi, Przemysław Pałka, Giuseppe Contissa, Francesca Lagioia, Hans-Wolfgang Micklitz, Giovanni Sartor, and Paolo Torroni. 2019. Claudette: an automated detector of potentially unfair clauses in online terms of service. *Artificial Intelligence and Law*, 27:117–139.

Zefang Liu, Jialei Shi, and John F Buford. Cyberbench: A multi-task benchmark for evaluating large language models in cybersecurity.

Lefteris Loukas, Manos Fergadiotis, Ilias Chalkidis, Eirini Spyropoulou, Prodromos Malakasiotis, Ion Androutsopoulos, and Paliouras George. 2022. Finer: Financial numeric entity recognition for xbrl tagging. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL 2022)*. Association for Computational Linguistics.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Macedo Maia, André Freitas, Alexandra Balahur, Siegfried Handschuh, Manel Zarrouk, Ross McDermott, and Brian Davis. 2018. Financial opinion mining and question answering.

Laura Manor and Junyi Jessy Li. 2019. Plain English summarization of contracts. In *Proceedings of the*

Natural Legal Language Processing Workshop 2019, pages 1–11, Minneapolis, Minnesota. Association for Computational Linguistics.

Rahul Mishra, Dhruv Gupta, and Markus Leippold. 2020. Generating fact checking summaries for web claims. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 81–90, Online. Association for Computational Linguistics.

Vittorio Orbinato, Mariarosaria Barbaraci, Roberto Natella, and Domenico Cotroneo. 2022. Automatic mapping of unstructured cyber threat intelligence: An experimental study:(practical experience report). In *2022 IEEE 33rd International Symposium on Software Reliability Engineering (ISSRE)*, pages 181–192. IEEE.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, R. I. S. H. I. T. A. ANUBHAI, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. In *International Conference on Learning Representations*.

Sachin Pawar, Nitin Ramrakhiyani, Anubhav Sinha, Manoj Apte, and Girish Palshikar. 2024. Why generate when you can discriminate? a novel technique for text classification using language models. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1099–1114. Association for Computational Linguistics.

Peter Phandi, Amila Silva, and Wei Lu. 2018. Semeval-2018 task 8: Semantic extraction from cybersecurity reports using natural language processing (securenlp). In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 697–706.

Jason Phang, Herbie Bradley, Leo Gao, Louis Castricato, and Stella Biderman. 2022. Eleutherai: Going beyond" open science" to" science in the open". *arXiv preprint arXiv:2210.06413*.

Matt Post. 2018. A call for clarity in reporting bleu scores. *arXiv preprint arXiv:1804.08771*.

Dexter Roozen and Francesco Lelli. 2021. Stock values and earnings call transcripts: a sentiment analysis. *Preprints 2021, 2021020424*.

Julio Cesar Salinas Alvarado, Karin Verspoor, and Timothy Baldwin. 2015. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 84–90, Parramatta, Australia.

scikit-learn developers. 2024. *sckikit-learn User Guide*.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Ankur Sinha and Tanmay Khandait. 2020. Impact of news on the commodity market: Dataset and results. *Preprint*, arXiv:2009.04202.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Siamak Shakeri, Dara Bahri, Tal Schuster, Huaixiu Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. 2022. Ul2: Unifying language learning paradigms. *Preprint*, arXiv:2205.05131.

Norbert Tihanyi, Mohamed Amine Ferrag, Ridhi Jain, and Merouane Debbah. 2024. Cybermetric: A benchmark dataset for evaluating large language models knowledge in cybersecurity. *arXiv preprint arXiv:2402.07688*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *Preprint*, arXiv:2303.17564.

Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, et al. 2024. The finben: An holistic financial benchmark for large language models. *arXiv preprint arXiv:2402.12659*.

Liang Xu, Lei Zhu, Yaotong Wu, and Hang Xue. 2024. Superclue-fin: Graded fine-grained analysis of chinese llms on diverse financial tasks and applications. *arXiv preprint arXiv:2404.19063*.

Hui Zeng. 2023. Measuring massive multitask chinese understanding. *arXiv preprint arXiv:2304.12986*.

Lucia Zheng, Neel Guha, Brandon R. Anderson, Peter Henderson, and Daniel E. Ho. 2021. When does pretraining help? assessing self-supervised learning for law and the casehold dataset. In *Proceedings of the 18th International Conference on Artificial Intelligence and Law*. Association for Computing Machinery.

Zhihan Zhou, Liqian Ma, and Han Liu. 2021. Trade the event: Corporate events detection for news-based event-driven trading. *Preprint*, arXiv:2105.12825.

Jie Zhu, Junhui Li, Yalong Wen, and Lifan Guo. 2024. Benchmarking large language models on cflue – a chinese financial language understanding evaluation dataset. *Preprint*, arXiv:2405.10542.

A Appendix

A.1 Benchmarks Overview

Table 6, 7 and 8 to 10 present the overview of English and Japanese benchmarks in the domain of finance, legal, climate, and cybersecurity, respectively. The data tables summarize key benchmarking information. Each table includes the *Task*, which specifies the problem, and the *Task Description*, explaining its nature. The *Dataset* column names the data used, with the *Dataset Description* detailing its characteristics. Lastly, the *Metric* column outlines the evaluation metrics used to measure model performance.

A.2 Evaluation Results

Table 4, 11, 12 and 13 show the LLMs evaluation results of legal, climate, and cybersecurity benchmarks, respectively. We have discussed the finance and legal results in section 5.3. Other results are summarized below.

Climate Benchmark Appendix/Table 11 shows the evaluation of models on climate and sustainability tasks. The flan-ul2 model performed best in Reddit Climate Change classification (0.560 Weighted F1) and SUMO Climate Claims summarization (0.258 Rouge-L), while the phi-3-5-mini-instruct model led in Wildfires and Climate Change Tweets classification (0.796 Weighted F1).

Cybersecurity Benchmark Table 12 presents the performance of models on cybersecurity tasks. In classification tasks, the llama-3-1-70b-instruct model excelled in SPEC5G (0.564 Weighted F1), CTI-to-MITRE with NLP (0.896 F1), TRAM (0.708 Macro F1), and IoTSpotter (0.928 Binary F1), while the flan-ul2 model achieved the highest score in SecureNLP (0.369 Binary F1). In summarization, flan-ul2 was the best in SPEC5G Summarization (0.331 Rouge-L).

Japanese Finance Benchmark The results in Table 13 show that the granite-8b-japanese model outperformed llama-3-elyza-jp-8b across all tasks (classification, summarization and translation) in the Japanese Finance Benchmark. Granite-8b-japanese achieved the highest scores with a Weighted-F1 of 0.454 in MultiFin, a Japanese Rouge-L of 0.456 in BoJ Outlook summarization, a Japanese BLEU of 0.123 in English-to-Japanese translation, and a BLEU of 0.075 in Japanese-to-English translation, consistently surpassing the scores of llama-3-elyza-jp-8b.

A.3 Prompts

Prompts that are used in the experiments are shown in this section. Figures 2 to 5 show the prompts for English finance, legal, climate, and cybersecurity scenarios, respectively. Figure 6 shows the prompts for Japanese finance scenarios.

A prompt consists of an "instruction" block, which is shown above a dotted line, and an "input-output" block, which is shown below the dotted line. The instruction block contains an instruction, which is placed at the beginning of a prompt. Some scenarios may not have the instruction block. The input-output block contains a pair of the input and output of each sample. This is located after the instruction block. Within a block, a text enclosed with curly brackets { ... } is replaced with an input text of each sample. A text enclosed with square brackets [...] is a placeholder of the generated text by an LLM as an output. In the case of a few-shot learning setting, the input-output block can be used to show a training example for in-context learning. In that case, the placeholder of the output is filled with the ground truth label of the sample.

Such instances of input-output blocks that correspond to the few-shot examples are iterated after the instruction block for n times, where n is the number of the shots of the in-context learning. After the in-context learning examples, another input-output block is placed without filling the output with a ground truth label.

Standard prompts (see the techniques of few-shot-prompting and zero-shot-prompting and examples of prompts[7]) without chain-of-thought prompting (Wei et al., 2023) or system prompts are used.

For News Headline and FiQA SA, the prompts are taken from BloombergGPT (Wu et al., 2023).

A.4 Comparison with existing non-domain-specific benchmarks

In this paper, importance of using domain-specific data is emphasized to evaluate the model performance for industry applications. Conversely, the use of non-domain-specific data such as pure language capability benchmarks or common sense benchmarks is discussed in this section.

[7] https://www.promptingguide.ai/techniques/fewshot

Task	Task Description	Dataset	Dataset Description	Metric
Classification	2 Classes	Earnings Call Transcripts (Roozen and Lelli, 2021)	Earnings call transcripts, the related stock prices and the sector index in terms of volume	Weighted F1
	9 Classes	News Headline (Sinha and Khandait, 2020)	The gold commodity news annotated into various dimensions	Weighted F1
Named Entity Recognition	4 numerical entities	Credit Risk Assessment (NER) (Salinas Alvarado et al., 2015)	Eight financial agreements (totalling 54,256 words) from SEC filings were manually annotated for entity types: location, organization person and miscellaneous	Entity F1
	4522 Numerical Entities	KPI-Edgar (Deußer et al., 2022)	A dataset for Joint NER and Relation Extraction building on financial reports uploaded to the Electronic Data Gathering, Analysis, and Retrieval (EDGAR) system, where the main objective is to extract Key Performance Indicators (KPIs) from financial documents and link them to their numerical values and other attributes	Adj F1
	139 Numerical Entities	FiNER-139 (Loukas et al., 2022)	1.1M sentences annotated with extensive Business Reporting Language (XBRL) tags extracted from annual and quarterly reports of publicly-traded companies in the US, focusing on numeric tokens, with the correct tag depending mostly on context, not the token itself.	Entity F1
Question Answering	Document relevance ranking	Opinion-based QA (FiQA) (Maia et al., 2018)	Text documents from different financial data sources (microblogs, reports, news) for ranking document relevance based on opinionated questions, targeting mined opinions and their respective entities, aspects, sentiment polarity and opinion holder.	RR@10
	3 Classes	Sentiment Analysis (FiQA SA) (Maia et al., 2018)	Text instances in the financial domain (microblog message, news statement or headline) for detecting the target aspects which are mentioned in the text (from a pre-defined list of aspect classes) and predict the sentiment score for each of the mentioned targets.	Weighted F1
	Ranking	Insurance QA (Feng et al., 2015)	Questions from real world users and answers with high quality composed by professionals with deep domain knowledge collected from the website Insurance Library [6]	RR@10
	Exact Value Match	Chain of Numeric Reasoning (ConvFinQA) (Chen et al., 2022)	Multi-turn conversational finance question answering data for exploring the chain of numerical reasoning.	Accuracy
Summarization	Long Documents	Financial Text Summarization (EDT) (Zhou et al., 2021)	303893 news articles ranging from March 2020 to May 2021 for abstractive text summarization.	Rouge-L

Table 6: Finance benchmarks overview

Task	Task Description	Dataset	Dataset Description	Metric
Classification	Japanese 6 classes	MultiFin (Jørgensen et al., 2023)	MultiFin is a financial dataset consisting of real-world article headlines covering 15 languages across different writing systems and language families.	Weighted F1
Summarization	Japanese	Bank of Japan Outlook (Bank of Japan, 2024)	The Bank of Japan's outlook for economic activity and prices at the quarterly monetary policy meetings.	Japanese Rouge-L
Translation	English to Japanese			Japanese BLEU
	Japanese to English			BLEU

Table 7: Japanese finance benchmarks overview

Among such benchmarks, three popular benchmark scenarios are selected:

- MMLU(Hendrycks et al., 2020) is a benchmark for multi-choice QA task. There are 57 sub-categories and in this experiment, "high school world history" is used as an example of a common-sense QA data.
- IMDb(Maas et al., 2011) is a benchmark for sentiment classification of movie reviews. There are two classes (Positive or Negative).
- CNN-DM(See et al., 2017) is a benchmark for news article summarization task, where the news articles were obtained from CNN and Daily Mail.

These benchmark scenarios are already available as a part of HELM. The labels of those are all manually created. These scenarios use text data that are written in plain English.

Table 14, 15, and 5 show the performance of the models on the above non-domain-specific data (shown as "General"). The models are sorted in the order of the parameter sizes. Also, the rankings of the models in terms of each metric are compared with the rankings of the models on scenarios of the corresponding task categories in the finance and legal domains (Tables 3 and 4).

Roughly speaking, there is a trend where the larger models show higher performance, with some exceptions, in the scenarios of non-domain-specific

Task	Task Description	Dataset	Dataset Description	Metric
Classification	3 Classes	Legal Sentiment Analysis [2]	Legal opinion categorised by sentiment	Weighted F1
	Multi-classes	UNFAIR-ToS (Lippi et al., 2019)	The UNFAIR-ToS dataset contains 50 Terms of Service (ToS) from online platforms. The dataset has been annotated on the sentence-level with 8 types of unfair contractual terms, meaning terms (sentences) that potentially violate user rights according to EU consumer law.	Weighted F1
	2 Classes	Legal Judgement Prediction (Chalkidis et al., 2019)	Legal judgment prediction is the task of automatically predicting the outcome of a court case, given a text describing the case's facts. This English legal judgment prediction dataset contains cases from the European Court of Human Rights.	Weighted F1
Question Answering	Multi-choice QA	CaseHOLD (Zheng et al., 2021)	The CaseHOLD dataset (Case Holdings On Legal Decisions) provides 53,000+ multiple choice questions with prompts from a judicial decision and multiple potential holdings, one of which is correct, that could be cited.	F1
Summarization	Summarization of US Legislations	BillSum (Eidelman, 2019)	The BillSum dataset consists of three parts: US training bills, US test bills and California test bills. The US bills were collected from the Govinfo service provided by the United States Government Publishing Office (GPO). For California, bills from the 2015-2016 session were scraped directly from the legislature's website; the summaries were written by their Legislative Counsel.	Rouge-L
	Contract Summarization	Legal Summarization (Manor and Li, 2019)	Legal text snippets paired with summaries written in plain English. The summaries involve heavy abstraction, compression, and simplification.	Rouge-L

Table 8: Legal benchmarks overview

Task	Task Description	Dataset	Dataset Description	Metric
Classification	2 Classes	Reddit Climate Change [3]	All the mentions of climate change on Reddit before Sep 1 2022.	Weighted F1
	2 Classes	Wildfires and Climate Change Tweets [4]	Tweets during the peach of the wildfire season in late summer and early fall of 2020 from public and government agencies.	Weighted F1
Summarization	Generating Fact Checking Summaries	SUMO Climate Claims (Mishra et al., 2020)	Climate claims from news or webs.	Rouge-L

Table 9: Climate benchmarks overview

data. However, different rankings can be seen in the cases of domain-specific datasets.

- Multi-choice QA: In the case of the legal dataset, we observe a drop of the rank of llama-3-1-8b-instruct. Note also that even llama-3-1-70b-instruct is still the best, its advantage shrinks. Both MMLU (general) and CaseHOLD (legal) have similar format of questions and similar text length. In contrast, the terminologies used in those scenarios are largely different. In CaseHOLD, an expert-level legal vocabularies and knowledge are needed to answer the questions.

- Sentiment classification: There is a large drop of the rank of flan-ul2 in both the finance and legal datasets. This is because there is unique terminology to express positive or negative situations (e.g., comparison of a financial result to that of the last year), and hence one cannot identify whether it is positive or not from the polarity of the used words (e.g., like, good, disappointing, etc.).

- Summarization: In the case of the legal dataset, we observe that the ranks of flan-ul2 and llama-3-1-70b-instruct drop, while other smaller models relatively work better. As we can see in Table 8, the labels of this dataset include heavy abstraction, compression, and simplifications, which requires deeper understanding of domain-specific terms. The result of BillSum (legal) has a similar trend. For the finance dataset, the rank drop of flan-ul2 is suppressed, while llama-3-1-8b-instruct rises to a position higher than llama-3-1-70b-instruct. This behavior is somewhat exceptional in our benchmarks. The reason of this is still unclear, but one should note that this task is actually a title generation task. Its expected output is much shorter than other summarization tasks.

In average, though some exceptions exist, there is a tendency that the rank of flan-ul2 drops in both finance and legal domains, and the ranks of llama-3-1 series slightly drop in the legal domain. Although it is difficult to explain these trends in

Task	Task Description	Dataset	Dataset Description	Metric
Classification	3 Classes	SPEC5G (Karim et al., 2023)	SPEC5G is a dataset for the analysis of natural language specification of 5G Cellular network protocol specification. SPEC5G contains 3,547,587 sentences with 134M words, from 13094 cellular network specifications and 13 online websites. It is designed for security-related text classification and summarisation.	Weighted F1
	Multi-classes	CTI-to-MITRE with NLP (Orbinato et al., 2022)	This dataset contains samples of CTI (Cyber Threat Intelligence) data in natural language, labeled with the corresponding adversarial techniques from the MITRE ATT&CK framework.	F1
	Multi-classes	TRAM [5]	The Threat Report ATT&CK Mapper dataset contain sentences from CTI reports labeled with the ATT&CK techniques	Macro F1
	2 Classes	SecureNLP (Phandi et al., 2018)	Semantic Extraction from CybersecUrity REports using Natural Language Processing (SecureNLP), a dataset on annotated malware report.	Binary F1
	2 Classes	IoTSpotter (Jin et al., 2022)	The IoTSpotter dataset is a collection of corpus and IoTSpotter identification results related to Internet of Things (IoT) devices and their security vulnerabilities.	Binary F1
Summarization	Text to Summary	SPEC5G (Karim et al., 2023)	*The same as above.This is the sub-dataset for summarization*	Rouge-L

Table 10: Cybersecurity benchmarks overview

Task	Classification		Summarization
	Reddit Climate Change	Wildfires and Climate Change Tweets	SUMO Climate Claims
Metrics	Weighted F1	Weighted F1	Rouge-L
N-shot Prompt	5-shot	5-shot	0-shot
phi-3-5-mini-instruct (3.8b)	0.470	**0.796**	0.190
mistral-7b-instruct-v0-3	0.457	0.761	0.210
llama-3-1-8b-instruct	0.448	0.746	0.225
llama-3-1-70b-instruct	0.418	0.736	0.235
granite-3-8b-instruct	0.461	0.784	0.216
flan-ul2 (20b)	**0.560**	0.747	**0.258**

Table 11: Climate benchmark evaluation results per task.

terms of the training data because the sources of the training data are usually not disclosed in most of the models, the reason of the above trends can be attributed to the training data in some cases. In the case of flan-ul2, the model uses the C4 corpus, which is a filtered English dataset of the Common Crawl, for pre-training(Tay et al., 2022)[8]. Since the model is published earlier than other models, it might be plausible that the training data for the model was not as diverse as other recent models to include finance and legal domain data. In the case of granite model series, it is known that some domain-specific datasets are intentionally included (see Section 5.1). From Tables 14, 15, and 5, one can observe that granite-3-1-8b-instruct keeps relatively a stable rank throughout these domains.

To conclude, the ranking of the models can be different in domain-specific scenarios from that in non-domain-specific scenarios even if the tasks are similar. It is not necessarily true that a larger model is better than a smaller model in terms of the parameter sizes. The reasons of those are that there are unique vocabularies and expressions that need to be understood to complete the task in those domains, while domain-specific training data is not common to all the models in general.

A.5 Classification Methods for many-class data

In our experiments, the model's input token length limit is usually around 1K to 8K. In the case of multi-class classification, the definition of a class tends to be highly domain-specific or task-specific. Therefore, the definition of classes must be described in a prompt. This roughly consumes CL tokens where C is the number of classes and L is the length of such a description of one class. In addition, the in-context learning examples need to cover all the classes at least once, to avoid the ignorance of minor classes. This will consume CQ tokens, where Q is the length of a question. For example, assuming that Q~50 tokens and L~50 tokens in the case of English classification task, 2K tokens are required when there are C~20 classes.

Recent models support a larger input token length limit such as 32K-128K tokens. There are interesting discussions on-going, such as its effectiveness in in-context learning(Li et al., 2024) and the trade-off between its benefit and the increase of the cost and latency(Bertsch et al., 2024). Evaluation of many-class classification tasks with such models is our future study. It is also possible that

[8]See also https://www.yitay.net/blog/flan-ul2-20b

Task	Classification					Summarization
	SPEC5G	CTI-to-MITRE with NLP	TRAM	SecureNLP	IoTSpotter	SPEC5G Summarization
Metrics	Weighted F1	F1	Macro F1	Binary F1	Binary F1	Rouge-L
N-shot Prompt	5-shot	10-shot	20-shot	5-shot	14-shot	0-shot
phi-3-5-mini-instruct (3.8b)	0.527	0.801	0.532	0.328	0.814	0.179
mistral-7b-instruct-v0-3	0.517	0.798	0.532	0.283	0.812	0.187
llama-3-1-8b-instruct	0.521	0.844	0.417	0.301	0.915	0.165
llama-3-1-70b-instruct	**0.564**	**0.896**	**0.708**	0.287	**0.928**	0.188
granite-3-8b-instruct	0.483	0.848	0.608	0.339	0.817	0.306
flan-ul2 (20b)	0.077	0.764	0.349	**0.369**	0.869	**0.331**

Table 12: Cybersecurity benchmark evaluation results per task.

Task	Classification	Summarization	Translation	
	MultiFin	BoJ Outlook (Summarization)	BoJ Outlook (E-to-J Translation)	BoJ Outlook (J-to-E Translation)
Metric	Weighted-F1	Japanese Rouge-L	Japanese BLEU	BLEU
N-shot Prompt	20-shot	0-shot	0-shot	0-shot
granite-8b-japanese	**0.454**	**0.456**	**0.123**	**0.075**
llama-3-elyza-jp-8b	0.436	0.398	0.110	0.053

Table 13: Japanese finance benchmark evaluation results per task.

users choose short input token length models due to this trade-off.

In this section, two different LLM-based implementation methods of the classification task are compared. One is the method proposed by (Pawar et al., 2024), and the other one is the naive method explained in Section 4.1. Pawar's method adopts a two-step approach, where in the first step, perplexity and log-likelihood based features are retrieved from an LLM by giving a prompt "X. This text is about K_c" where X is an input text and K_c is a key phrase associated with a specific class c, and a separate classification model outputs the final label from the features using a conventional machine learning model in the second step. Pawar's method has an advantage that it is not affected by the context length limit of a model even when the number of classes is large.

However, one side-effect of the method is the increase of the latency that is proportional to the number of classes. To evaluate this, the inference times of these two methods are measured for various number of classes, which can be seen Figure 7. From this result, we can see that the inference time increases almost linearly to the number of classes in the case of the method proposed by (Pawar et al., 2024), while that of the naive method increases weakly. The main factor of this difference is the length of the output. In the case of the naive method, the output length is almost constant (i.e., the length of a class label) regardless of the number of classes. In the case of Pawar et al., the output length is proportional to the number of classes because the computation of log-likelihood or perplexity of generating the key phrases for a class c must be iterated for all the classes. Since an LLM generates output tokens one-by-one, the inference time increases linearly to the number of output tokens, while the input tokens can be processed within one step as far as it is smaller than the input context length limit.

As a conclusion, in the case of an LLM with a short context length limit (e.g., 1k - 4k tokens), the only solution for the many-class classification task is the method by (Pawar et al., 2024). However, this method is also not practical because usually there is a latency requirement in a classification task. Therefore, many-class (e.g., 100 classes) classification is still challenging for LLMs with short context length. We expect that recent long context length models (e.g., 32k - 128k tokens) or fine-tuning of a model can mitigate this issue, but of course there is a trade-off with the computational cost.

The detail of the experiment are described as follows. To implement the method proposed by (Pawar et al., 2024), a question for the original classification task is converted into a set of C sub-questions in the pre-process, each of which can be used to generate a log probability or a perplexity of a specific class name. For each sub-question, the number of in-context learning examples is fixed to four, including both positive and negative cases. In the case of the naive implementation, the number of in-context learning examples is set to C.

The configuration of the experiment is as follows. In this experiment, CTI-to-mitre dataset (Table 8) is used. The dataset originally has 199 classes. From this dataset, subsets whose samples belong to top 10, 20, ..., 60 classes in terms of frequency are

Earnings Call Transcripts (classification)

Classify the sentences into one of the 2 sentiment categories. Possible labels: positive, negative.

{Sentence}
Label: [positive/negative]

News Headline (classification)

{Sentence}
Question: Is the passage above about {topic}?
Answer: [Yes/No]

Credit risk assessment (NER)

Extract named entities from the input sentence below. Also, classify each of the extracted named entities into one of the following categories: person, organization, location, and miscellaneous.

Input: {Sentence}
Task: Extract named entities.
Answer: [person name (person), organization name (organization), location name (location), ...]

KPI-Edgar (NER)

Context: {Sentence}
Task: Extract key performance indicators (KPIs) and values from the above text. Also, specify one of the following categories to each of the extracted KPIs and values in brackets.
kpi: Key Performance Indicators expressible in numerical and monetary value, cy: Current Year monetary value, py: Prior Year monetary value, py1: Two Year Past Value.
Answer:[...]

FINER-139 (NER)

Passage: {Sentence}
Answer: [Numeric entities]

Opinion-based QA (FiQA) (QA)

Passage: {Passage}
Query: {Question}
Does the passage answer the query?
Answer: [Yes/No]

Sentiment Analysis (FiQA SA) (QA)

{sentence}
Question: what is the sentiment on {target}?
Answer: [negative/neutral/positive]

Insurance QA (QA)

Read the passage and query below, and identify whether the passage answers the query. Use yes or no to respond.

Passage: {Passage}
Query: {Question}
Does the passage answer the query?
Answer: [Yes/No]

Chain of Numeric Reasoning (ConvFinQA) (QA).

Passage: Table:
{Table}
Text:
Questions: Question: {Question}? The answer is {Answer}
{Question}? The answer is {Answer}
{Question}? The answer is {Answer}
{Question}? The answer is
Answer:

Financial text summarization (EDT) (summarization)

Generate the title of the following article.

{text}
Title:

Figure 2: Prompts of English finance scenarios.

extracted, and the inference times for those subsets are measured. The number of test samples is fixed to 100 in all the cases. The model is llama-3-1-70b-instruct, which is executed in a shared cloud server. The inference time includes the computation time of the inference by the LLM and the network communication time to access the API of the model, but does not include the pre-processing time and post-processing time. The access to the model API is parallelized using four threads.

A.6 Other NER Methods for LLMs

As explained in Section 4.2, a conventional NER task is formalized as a sequence-to-sequence task from natural language text to a BIO tag sequence, which denotes the category of corresponding tokens (e.g., B_PERSON, I_LOCATION, O, etc., where the prefixes B, I, and O indicate the beginning, internal, and outside of an entity name, respectively). However, in our preliminary experiments, this approach did not work well with LLMs. This seems to be because BIO tags are unknown to pre-trained LLMs.

In addition, Wu et al.(Wu et al., 2023) reports that one needs 20 or more shots for in-context learning. This number of shots is larger than that of classification tasks. In the case of the naive seq-to-seq method, few-shot examples consume many tokens since the inputs and the tags in the labels are both provided in a seq-to-seq manner.

Recently, several alternative approaches have been proposed for LLM-based NER. These methods exploit the knowledge of a pre-trained LLM on natural language phrases that appear in the inputs as well as in the category labels. Such approach helps improving the performance especially in low-resource domains(Cui et al., 2021).

The template-based method(Cui et al., 2021) is originally proposed for the encoder-decoder architecture, but can be applied to the decoder-only architecture. In this method, the task is formalized as a translation from the input text to another text which is generated from a template such as "X is a Y entity", where X is a candidate of an named entity in the input text and Y is a category of an entity. In the inference phase, one measures the log

Legal Sentiment Analysis (classification)

Classify the sentences into one of the 3 sentiment categories. Possible labels: positive, neutral, negative.

{Sentence}
Label: [positive/neutral/negative]

UNFAIR-ToS (classification)

In this task, you are given a sentence from a Terms of Service (ToS) document from online platforms. Predict the types of unfair contractual terms out of the following: 0: Limitation of liability, 1: Unilateral termination, 2: Unilateral change, 3: Content removal, 4: Contract by using, 5: Choice of law, 6: Jurisdiction, 7: Arbitration. If there is no label reply n/a, if there are multiple labels specify all of them separated by a comma.

Passage: {passage}

Answer: [answer0,answer1,...]

Legal judgement prediction (classification)

Is the following case a violation of human rights?

Case: Human rights have not been violated.
Answer: No

Case: Human rights have been violated.
Answer: Yes.

Case: {Text}
Answer: [Yes/No]

Case-HOLD (QA)

Give a letter answer among A, B, C, D, or E.

Passage: {text}
Answer: [A/B/C/D/E]

BillSum (summarization)

BillSum is a dataset for summarization of US Congressional and California state bills. Write the summary of the above article in 300 words.

Article: {Text}
Summary:

Legal summarization (summarization)

Text: {Text}
Write the summary of the text article in 100 words.
Summary: [Summary]

Figure 3: Prompts of legal scenarios.

Redit climate change dataset (classification)

Passage: {text}
Question: The above sentence is from a collection of comments related to climate change. Classify the comments into one of the 3 sentiment categories.
Possible labels:
positive
neutral
negative
Answer: [negative/neutral/positive]

Wildfires and climate change Tweets (classification)

Passage: {text}.
Question: Classify the tweets into one of the 3 sentiment categories.
Possible labels:
positive
neutral
negative
Answer: [negative/neutral/positive]

SUMO climate claims (summarization)

Generate the title of the following article.

{text}
Title:

Figure 4: Prompts of climate scenarios

probability of generating a specific instance of the template text (e.g., "Bangkok is a location entity") from the model and determines whether the named entity and its category. Therefore, the length of the label is usually shorter than the input text, while it requires multiple inferences to exhaust all the named entity candidates.

Another approach is the use of an augmented natural language(Paolini et al., 2021). This method formalizes NER as a translation from input text to the same text with annotations inserted. The annotation specifies the range of an named entity as well as its category. In this case, the output text is longer than the input text.

A simplified approach is proposed, where named entities are extracted from the input text(Wu et al., 2023). In this method, a model is instructed to report only named entities and the categories of those (e.g., New York (location), etc.). Thus, the length of the output and label is usually shorter than the input.

These methods are compared with the naive method in Table 16. In the table, "Position" column indicates the capability of retrieving positional information of the detected entities. "Input token consumption" is identified from the label length of in-context learning. "Latency / cost" is related to the output length. "Accuracy" is related to the exploitation of knowledge of a pre-trained LLM. The evaluation is relative to the case of the naive seq-to-seq method.

In this paper, the extraction-based method is cho-

SPEC5G (classification)

Categorize the following sentence into 3 classes. Possible labels: 1. Non-Security 2. Security 3. Undefined.

Sentence: {Text}
Label: [Output]

SecureNLP (classification)

Passage: {passage}
Question: Is the passage above critically relevant to malware?
Answer:[yes/no]

CTI-to-MITRE with NLP (classification)

Answer the possible security attacks in each of the following situations from each of the options below.

Situation: {description}
A. {attack_category_name_1}
B. {attack_category_name_2}
...
Y. {attack_category_name_(k-1)}
Z. Others
Answer: [correct_answer]

IoTSpotter (classification)

Read the given passage below and identify whether the passage is a description of an IoT mobile app. Answer in yes or no. Note that an IoT mobile app is a mobile app that is used for managing and controlling IoT devices such as smart appliances.

Passage: {passage}
Question: Is the passage above a description of an IoT mobile app?
Answer:[yes/no]

TRAM (classification)

Passage: {passage}
Question: Is the passage above about {class}?
Answer:[yes/no]

SPEC5G (summarization)

Text: {Text}
Write the summary of the above text.
Summary:

Figure 5: Prompts for cybersecurity scenarios.

Scenario	MMLU		CaseHOLD
Domain	General		Legal
N-shot Prompt	5-shot		-
Metrics	accuracy	rank	rank (diff.)
phi-3-5-mini-instruct (3.8b)	0.775	4	4 (0)
mistral-7b-instruct-v0-3	0.742	5.5	6 (+0.5)
granite-3-8b-instruct	0.854	3	2 (-1)
llama-3-1-8b-instruct	0.865	2	5 (+3)
flan-ul2 (20b)	0.742	5.5	3 (-2.5)
llama-3-1-70b-instruct	0.944	1	1 (0)

Table 14: Comparison with non-domain-specific data: Multi-choice QA task. For MMLU, the sub-category is high school world history and the number of test samples is 89.

sen so that both short-context models and long-context models can be compared in a same benchmark. See Table 2 for the context length limit of each model. Additional simplifications are: (i) In some scenarios, the number of categories is reduced, due to a similar reason with the case of classification tasks (Appendix A.5). (ii) Questions without any labeled named entity are removed, which is similar to (Wu et al., 2023).

A.7 Details of additional metrics

In ConvFinQA, the answers are floating point numbers. A regular expression is used to match the floating-point numbers.

In Japanese scenarios, a language-specific tokenizer is introduced to compute the metrics (Section 4.5). Japanese BLEU (for English-to-Japanese translation) and BLEU (for Japanese-to-English translation) are implemented with the sacreBLEU library (Post, 2018) using ja-mecab[9] and the default (13a) tokenizers, respectively. Japanese Rouge-L is implemented with the same ja-mecab tokenizer and used for the summarization task.

[9] https://taku910.github.io/mecab/

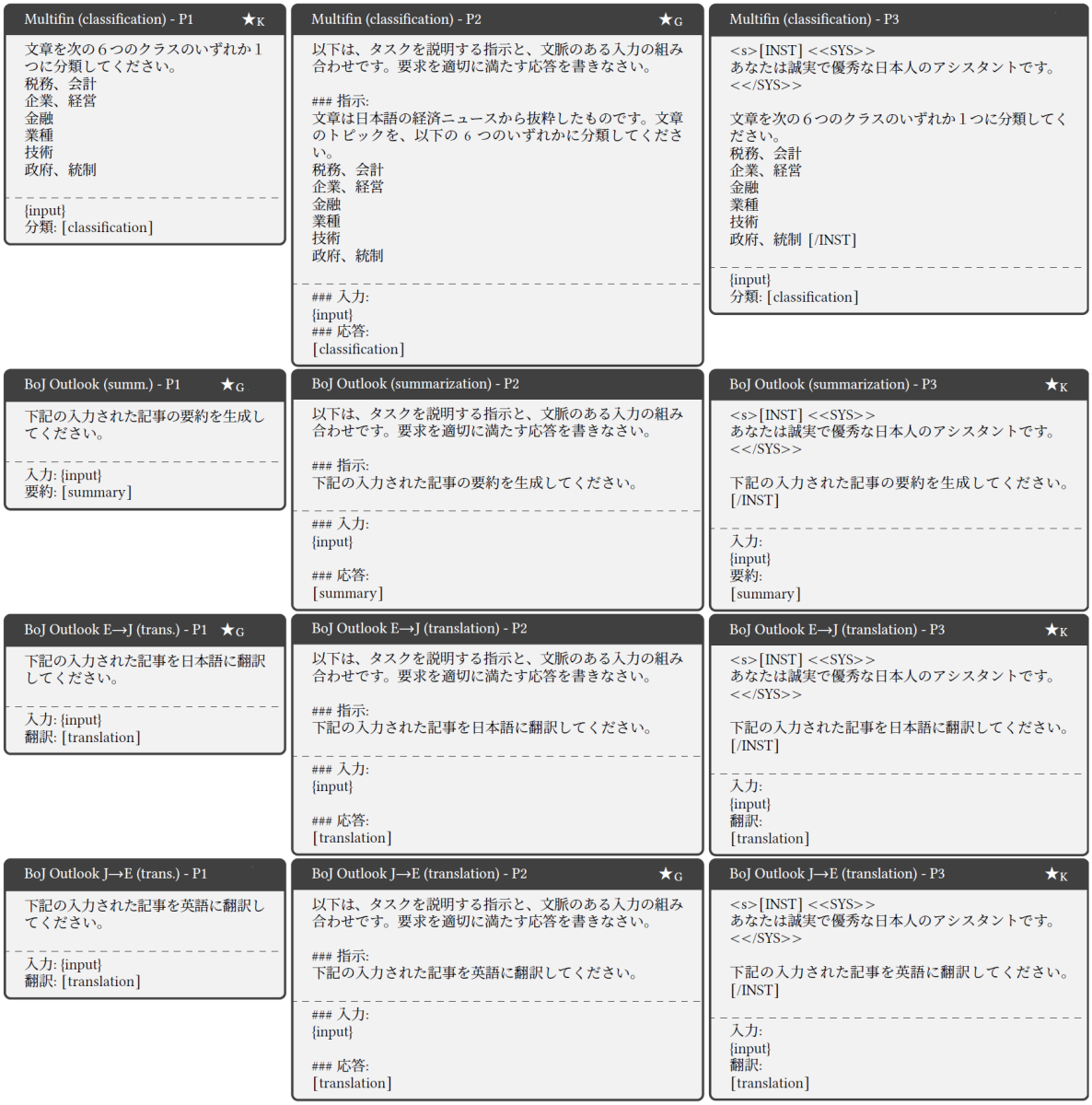

Figure 6: Prompts for Japanese finance scenarios. Each scenario has the following prompts. P1: This is a standard prompt without a system prompt. P2: The system prompt for granite-8b-japanese. P3: The system prompt for japanese-llama-2-7b-instruct and llama-3-elyza-jp-8b. $\bigstar_{\mathrm{G}}$ and $\bigstar_{\mathrm{K}}$ indicate the best prompts for granite-8b-japanese and llama-3-elyza-jp-8b, respectively.

Scenario	IMDB		FiQA-SA	Legal Sentiment Analysis
Domain	General		Finance	Legal
N-shot Prompt	N-shot		-	-
Metrics	accuracy	rank	rank (diff.)	rank (diff.)
phi-3-5-mini-instruct (3.8b)	0.935	4	5 (+1)	6 (+2)
mistral-7b-instruct-v0-3	0.950	3	3 (0)	1 (-2)
granite-3-8b-instruct	0.960	2	1 (-1)	2 (0)
llama-3-1-8b-instruct	0.920	5.5	4 (-1.5)	4 (-1.5)
flan-ul2 (20b)	0.975	1	6 (+5)	5 (+4)
llama-3-1-70b-instruct	0.920	5.5	2 (-3.5)	3 (-2.5)

Table 15: Comparison with non-domain-specific data: Sentiment classification task. The number of test samples in IMDB is 200.

Method	Position	Input token consumption	Latency / cost	Accuracy
BIO tag seq. (naive)	**Yes**	High	High	Low
Template-based(Cui et al., 2021)	No	**Low**	High	**High**
Augmented NL(Paolini et al., 2021)	**Yes**	High	High	**High**
Extraction-based (Wu et al., 2023)	No	**Low**	**Low**	**High**

Table 16: Comparison of various NER methods for LLMs.

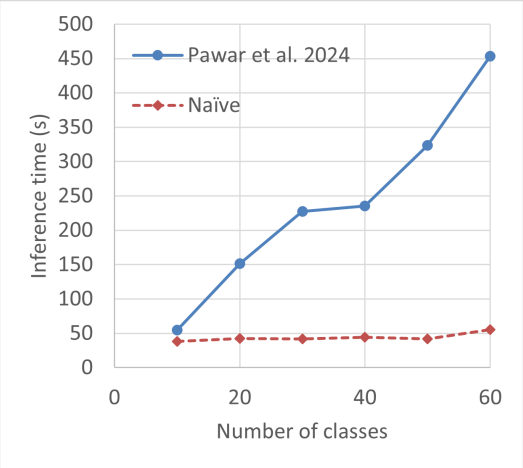

Figure 7: Dependence of the inference time of CTI-to-MITRE scenario (classification) to the number of classes.