Are Small Language Models Ready to Compete with Large Language Models for Practical Applications?

Neelabh Sinha[1], **Vinija Jain**[2*], and **Aman Chadha**[3†]

[1]Georgia Institute of Technology
[2]Meta AI
[3]Amazon GenAI
nsinha68@gatech.edu, hi@vinija.ai, hi@aman.ai

Abstract

The rapid rise of Language Models (LMs) has expanded their use in several applications. Yet, due to constraints of model size, associated cost, or proprietary restrictions, utilizing state-of-the-art (SOTA) LLMs is not always feasible. With open, smaller LMs emerging, more applications can leverage their capabilities, but selecting the right LM can be challenging as smaller LMs don't perform well universally. This work tries to bridge this gap by proposing a framework to experimentally evaluate small, open LMs in practical settings through measuring semantic correctness of outputs across three practical aspects: *task types*, *application domains* and *reasoning types*, using diverse prompt styles. It also conducts an in-depth comparison of 10 small, open LMs to identify best LM and prompt style depending on specific application requirement using the proposed framework. We also show that if selected appropriately, they can outperform SOTA LLMs like DeepSeek-v2, GPT-4o-mini, Gemini-1.5-Pro, and even compete with GPT-4o. [1]

1 Introduction

The field of NLP has advanced significantly with the rapid development of Language Models (LMs) (Brown et al., 2020; Touvron et al., 2023; Almazrouei et al., 2023; Team et al., 2024b; DeepSeek-AI, 2024), which has expanded their use across numerous types like Title Generation (Keles and Bayraklı, 2024), Data Exploration (Ma et al., 2023), Dialogue act recognition (Qiang et al., 2024); domains like Economics & Finance (Rajpoot et al., 2024; Yu et al., 2023), Politics (Feng et al., 2023), Nutrition & Food (Yang et al., 2024), News (Kuila and Sarkar, 2024); and reasoning types (Huang and Chang, 2023) like ANALOGICAL (Wijesiriwardene et al., 2023) and Multi-hop (Pan et al., 2021) reasoning.

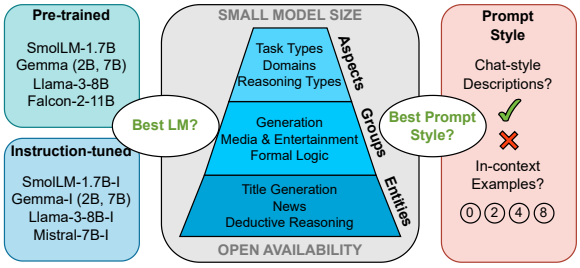


Figure 1: Outline of this work: Performance analysis of open, small-scale LMs and best prompt style for task types, application domains, and reasoning types.

Despite the growing variety of LMs, their usage in downstream applications is heavily skewed towards limited ones. Analyzing around 50 papers from 2024, we found that while 82.3% of methods utilized GPT-family LMs, only 41.1% used Llama variants, and less than 11.8% experimented with other alternatives like Mistral (Jiang et al., 2023) and Falcon (Almazrouei et al., 2023). Some studies also report issues like garbage output and hallucinations (Alhamed et al., 2024), but domain experts often lack the tools to address them effectively through informed LM choice, or correct ways to prompt them. Even the methods that experiment with multiple LMs often select models without a strong motivation (Kuila and Sarkar, 2024).

Apart from performance, many of the new LMs are smaller in size, and openly available. Despite the undeniable success of large, proprietary LMs like GPT-4 (OpenAI et al., 2024) and Llama-2 70B (Touvron et al., 2023), their inaccessibility due to limited API access, high costs (Jimenez Gutierrez et al., 2022), concerns around data privacy (for GPT), and massive computational demands (Ding et al., 2024) (for Llama) pose significant barriers of usage. Small, open LMs can navigate around

*Work does not relate to position at Meta.
†Work does not relate to position at Amazon.
[1]GitHub repository containing the code implementation of this work: https://github.com/neelabhsinha/lm-application-eval-kit

Proceedings of the 5th Workshop on Trustworthy NLP (TrustNLP 2025), pages 365–398

those, and also provide additional benefits like on-device usage, faster inference time, data privacy, easier compliance and security management, and low-cost maintainance. For many practitioners -— especially those in research, startups, or sectors with limited resources or high security risk – leveraging these presents an appealing alternative for functional, financial, or business reasons.

But these new, small, LMs vary a lot in terms of training data, pre-training strategies, and architectural decisions. Additionally, they may not perform globally well like SOTA LLMs due to limitations of scale (Kaplan et al., 2020). Utilization strategies of LMs in inference pipelines can also differ, like zero-shot usage, customizing pre-trained models (e.g., fine-tuning (Mosbach et al., 2023)), using in-context learning (Wei et al., 2022a; Dong et al., 2023), prompt engineering (Brown et al., 2020). Writing effective prompts also requires time and domain expertise. So, users need to conduct thorough analysis before choosing the right LM and usage strategy within constraints of time, money, computational resources, which is a complicated task. Although technical reports of some LMs (Team et al., 2024b,c) provide some insights, not all of them capture real-world, practical scenarios. Therefore, there is a need for a comprehensive practical evaluation framework which can enable determining capabilities of LMs in multiple practical applications, and effective ways to prompt them.

To bridge this gap, we propose a comprehensive framework for evaluating LMs in practical settings along three aspects: task types, application domains, and reasoning types. For each aspect, we select 12, 12, and 10 entities in English, grouping similar ones (e.g., 'Social Media' and 'News' under 'Media and Entertainment'). This three-tier structure (aspect, group, entity) helps identifying patterns in LM capabilities across multiple levels. Using Super-Natural Instructions (Wang et al., 2022), a meta-dataset encompassing various NLP benchmarks, we evaluate LMs on task instances within this framework. LM usage strategies vary significantly – ranging from fine-tuning (Mosbach et al., 2023), PEFT (Han et al., 2024) or direct usage with/without prompt engineering. Thus, we assess semantic correctness of outputs as an indicator of LMs' inherent abilities, evaluating five pre-trained and five instruction-tuned (IT) (Ouyang et al., 2022) models across eight prompt styles. Our results show that with careful selection, impact of scale can be reduced. Correctly chosen small, open LM can rival and even outperform models like GPT-4o-mini, GPT-4o (OpenAI, 2023), DeepSeek-v2 (DeepSeek-AI, 2024), and Gemini-1.5-Pro (Team et al., 2024a), while providing additional benefits. We also evaluate LMs with paraphrases of task definitions to show that results are robust against dataset-induced biases.

In this work, we aim to address these research questions: **(i)** Can small, open LMs compete with large, proprietary LMs in practical usage? **(ii)** What can be an exhaustive evaluation framework to conduct this analysis? **(iii)** For different application needs, how do current best small, open LMs perform in comparison, and which LM is the best choice? **(iv)** What type of prompt style should be used to extract best results from these LMs?

Consistent with Figure 1, we make the following **key contributions**:

(i) Propose a three-tier evaluation framework to analyze performance of LMs for different *task types*, *application domains* and *reasoning types*.

(ii) Conduct an in-depth experimental analysis of semantic correctness of outputs of 10 open, small LMs in 1.7B–11B size based on the framework.

(iii) Show that appropriate selection of open, small LMs can lead to outperforming SOTA LLMs like GPT-4o-mini, Gemini-1.5-Pro, and competing with GPT-4o.

(iv) Compare the performance of LMs with eight prompt styles and recommend the best alternative.

2 Evaluation Framework

We begin with describing our evaluation framework discussing dataset, prompt styles, selection process of aspects, evaluation metrics and experiments.

2.1 Experimental Dataset

We derive our experimental dataset from Super-Natural Instructions (Wang et al., 2022), which is not a single dataset but a meta-dataset constructed by combining many standard NLP datasets. In addition to the source datasets, it also has definition describing a task in chat-style instruction form and many in-context examples (refer Figure 2 for an example) curated by experts. Using datasets from here benefits us by allowing evaluation with various prompt styles and using chat-style instructions – the way users practically interact with LMs. It also provides labels of task type describing nature of a task (eg. question answering, data to text), domain describing the field of the task (eg. history,

news), and reasoning type, describing the type of reasoning (if any) needed in the task (eg. multihop, analogical, etc.), which we also use.

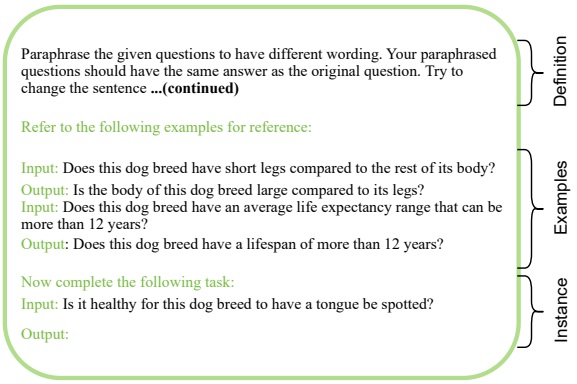


Figure 2: Example of a prompt with definition and 2 examples (text in Green is static text, and others are taken from the dataset).

We pick the test split of the dataset for which input and output is English, since most LMs are optimized for that, giving 119 tasks. To avoid redundancy but still take sufficient samples, we take 100 instances per tasks at maximum. Finally, we get 11810 task instances belonging to 12 task types, 36 domains and 18 reasoning types.

2.2 Prompt Styles

We conduct our experiments using multiple prompt styles - including/excluding chat-style task definitions, and with 0, 2, 4, 8 in-context examples for each instance. Examples help LMs (even pretrained) with in-context learning (Wei et al., 2022a; Dong et al., 2023) without altering their parameters. This is followed by an actual task instance. We select examples from positive examples section of the task. This gives 8 prompt styles per task instance. An example of prompt with definition and 2 examples is given in Figure 2. 'Input' and 'Output' is used since they are universal for all tasks.

2.3 Selection of Aspects

From the dataset, we divide each task instance into three aspects – task types, application domains and reasoning types. Since there were many instances for each entity, we filter and rearrange these to create a filtered set for brevity. Our objective was to cover a wide range of application area in each aspect. Therefore, first, we took all the 12 task types in the test set. Among them, for 36 domains and 18 reasoning types, we discarded subsets, very closely similar entities, or ones which didn't have many examples. For example, there were two domains 'Computer Science' and 'Coding', so we included only Computer Science as Coding can be considered a subset; among the two types of reasoning called 'Numerical' and 'Quantitative', we included only Quantitative since they were very similar, and so on. As the number of entities were not too many, we did this manually. We always included the more wider scoped entity when resolving these clashes. After taking a broad enough spectrum in all 3 aspects, we constructed groups in each entity and placed them to create a second-level hierarchy, with similar entities in same groups. Our final structure is shown in Figure 3. Here, Domains is an aspect, Social Sciences and Humanities is a group which contains 4 entities, Economics being one of them. Our intention with this is to provide a structure to this study and cover a broad spectrum of entities. Some of the definitions, specifically for reasoning types, are detailed more in a survey (Guo et al., 2023) and the dataset repository[2].

This allows analysis at three levels of hierarchy - aspect, group and entity level, which is how we address them in rest of this paper. Some tasks can overlap between entities of same aspect (Kuila and Sarkar, 2024) or different aspects (Keles and Bayraklı, 2024), and some may not belong to any aspect. There are more entities not included here for brevity but listed and evaluated in Appendix B with dataset statistics.

2.4 Evaluation Metrics

As per the analysis of recent works (Sai et al., 2021; Xiao et al., 2023), evaluating LM outputs using n-gram metrics like ROUGE (Lin, 2004), METEOR (Banerjee and Lavie, 2005), etc., have limitations in terms of coherence, consistency, relevance, and fluency. These works also show that BERTScore-recall (Zhang et al., 2019) limits this to a great extent. To be consistent, we evaluate LM's knowledge via semantic correctness of outputs using BERTScore (Zhang et al., 2019) recall with `roberta-large` (Liu et al., 2019).

Some tasks, like classification, aren't generation tasks, but we still consider them as one since they give a uniform evaluation paradigm. By aligning outputs using fine-tuning/ICL (Zhao et al., 2023), verbalizers (Hu et al., 2022), post-processing, labels can be obtained from language outputs.

[2] https://instructions.apps.allenai.org

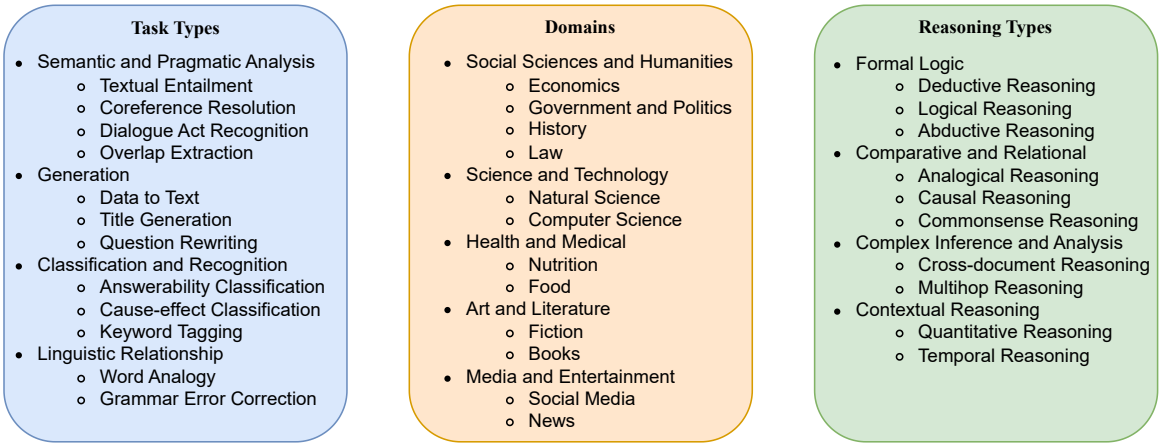


Figure 3: Sections and entities for which the performance of Language Models (LMs) is analyzed (each aspect is divided into groups (like formal logic), and the groups are divided into individual entities (like deductive reasoning). This three-level categorization allows analysis of performance across multiple hierarchies.

2.5 Language Models Used

The focus for this work is on open LMs from 1.7–11B parameters for adaptability and computational efficiency. Analysis of pre-trained models, trained for next-word prediction, will give an insight into LMs' ability and knowledge to perform the tasks. They can either be used directly or adapted/aligned further. IT models will suit out-of-the-box usage on chat-style human-like instructions due to a simple use-case or unavailability of sufficient data/resources to customize the models.

To cover a broad range of SOTA small, open LMs across sizes, families, our experiments utilize Gemma-2B, Gemma-7B (Team et al., 2024b), Llama-3-8B (Touvron et al., 2023; AI@Meta, 2024), Mistral-7B-v0.3 (Jiang et al., 2023), and Falcon-2-11B (Almazrouei et al., 2023; TIIUAE, 2024). We also take their instruction-tuned (IT) versions (except Falcon-2-11B - not available). But, we omit Mistral-7B pre-trained from discussion as its results weren't competitive, and Gemma-2 series (Team et al., 2024c) since their performance was below Gemma. Model and implementation details are discussed more in Appendix C, G. In this paper, suffix "-I" indicates instruction-tuned.

3 Experiments and Results

We use all the prompt styles with each of the task instance, do a forward pass on the LM, and decode the output using greedy decoding, which is evaluated with available references. We used greedy as it's reproducible, also other sampling techniques (Holtzman et al., 2020) didn't give any improvement (refer Appendix E). The following subsections discusses findings.

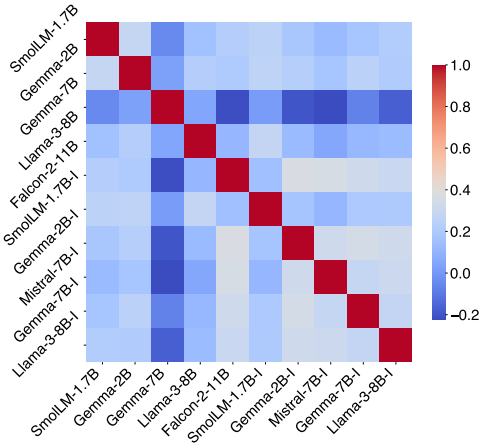


Figure 4: Correlation matrix of mean BERTScore recalls across different task instances for outputs of LMs.

3.1 Performance Correlation of LMs

One of the hypothesis was that different LMs would perform differently. To demonstrate that, we show the correlation between BERTScore recalls of LM outputs, shown in Figure 4, is low. This shows that their performance with different task types are inherently different, and therefore, selecting the right LM for a usage requirement becomes crucial. To analyze this, we detail their performance in our proposed evaluation framework. For these analyses, we use the best prompt style for that entity of that

aspect (refer Appendix D.2 to determine that).

3.2 Comparison Across Task Types

Figure 5a and Figure 5d show variation of performance on task-types for pre-trained and IT models.

Most of the pre-trained models perform reasonably well on most tasks. We see that Gemma-2B always and SmolLM-1.7B sometimes perform better than all 7B and 8B models, which is opposite to the general understanding that scale improves performance. So, other design factors are also relevant which contribute to their strengths. Gemma-2B is the best across 50% of the task types, with Falcon-2-11B leading in the remaining, except Word Analogy where SmolLM-1.7B is marginally the best. Considering the scale of the two models, Gemma-2B is a strong choice with resource constraints across all task types, unless Falcon-2-11B is needed purely on performance. Gemma-7B and Llama-3-8B hover below the top two with varying differences. We don't identify any patterns at group levels here but the difference between the top two models is similar across most tasks.

In IT models, Mistral-7B-I performs best on all task types, with Gemma-2B-I and SmolLM-1.7B-I competing for the second-best. At group level, we find the difference to be smaller for linguistic relationship and generation tasks, but large for semantic & pragmatic analysis tasks. Like their pre-trained variants, Gemma-7B-I and Llama-3-8B-I seldom compete with Gemma-2B-I in some tasks, but never outperform it. So, Gemma-2B, SmolLM-1.7B-I and Mistral-7B-I can be selected based on performance and resources trade-offs.

3.3 Comparison Across Application Domains

The behavior of LMs across application domains can be visualized in Figure 5b and 5e for pre-trained and IT models, respectively.

Particularly for pre-trained models, the performance is very sensitive across domains. For social sciences & humanities, and science & technology domain groups, Falcon-2-11B performs the best with Gemma-2B and Llama-3-8B following. Gemma-2B and Falcon-2-11B are not always the best ones. In health and medical tasks, Gemma-7B outperforms all models. Falcon-2-11B and Gemma-2B suffer a significant performance degradation in this group. Therefore, for domains, the choice of pre-trained LMs depends on the use case and other constraints. SmolLM-1.7B felt like a strong choice in task types, but here we see that it struggles with these domains. Its strength in Section 3.2 might be from other domains not considered here, showing its sensitivity with domains.

Among the IT models, we see similar trends as in task types - Mistral-7B-I being the best in all domains, and Gemma-2B-I and SmolLM-1.7B-I competing for second. The difference with Gemma-2B-I is closer in some domains like Computer Science, News, and Books, and largest in Economics. We also see that SmolLM-1.7B-I has strong limitations in Science and Technology group. Hence, Mistral-7B-I is still the best choice with best prompt style if the available resource allows, and if not, then Gemma-2B-I or SmolLM-1.7B is the way to proceed based on requirements.

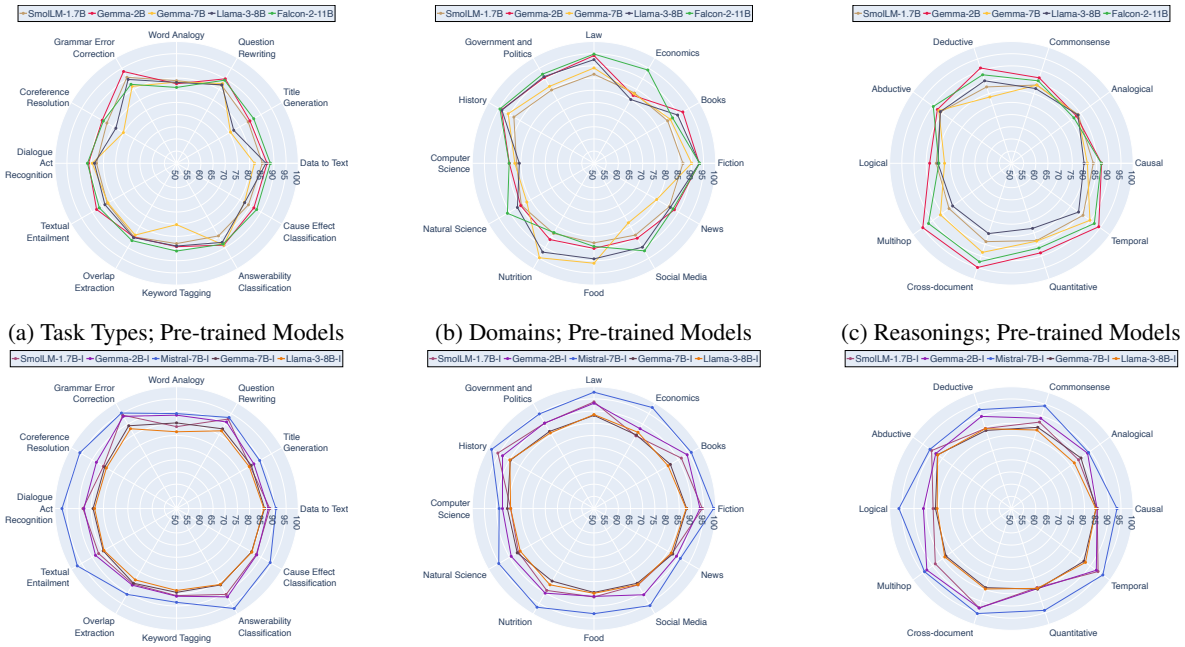
Group-level behavior is more prominent in this aspect, highlighting the importance of our three-tier framework. Even in case of analyzing a new domain that is not present here, the performance of the group that domain would belong to can give an idea of baseline performance.

3.4 Comparison Across Reasoning Types

52 out of 119 task definitions in the dataset don't have a reasoning type as not all tasks require reasoning. For the remaining, the performance of different pre-trained LMs are shown in Figure 5c and for all IT models in Figure 5f.

In the pre-trained models, we find that where reasoning is involved, Gemma-2B marginally outperforms Falcon-2-11B in all types of reasoning except Abductive reasoning, where it comes second by a small margin. It shows that Gemma-2B is a great choice where reasoning is involved, having advantage in both performance and model size. Llama-3-8B proves to be the best in analogical reasoning. In general, it is observed that the performance of all pre-trained LMs is the least for Comparative and Relational reasoning types, highlighting a potential common limitation of ability in these types of task in zero-shot. Therefore, adapting the LMs might become crucial in this case.

With IT models, behavior remains similar to the previous two aspects for all the five models, with Mistral-7B-I coming out to be a clear choice. The difference between Mistral-7B-I and Gemma-2B-I is minimum in complex inference & analysis types, and maximum for types like logical and quantitative reasoning. SmolLM-1.7B-I also depicts weaknesses in some reasoning types. This shows that while choosing a pre-trained model has its complexities, for IT models, the choice is relatively



(a) Task Types; Pre-trained Models (b) Domains; Pre-trained Models (c) Reasonings; Pre-trained Models

(d) Task Types; IT Models (e) Domains; IT Models (f) Reasonings; IT Models

Figure 5: Mean BERTScore recall across various task types, domains, and reasoning types, segmented by pre-trained vs. instruction-tuned models (Note that the range doesn't start from 0 for better visibility).

simpler after considering external constraints.

The quantified performance of each entity of all three aspects in the dataset (even ones not included in Fig 3) with each LM is given in Appendix B.

3.5 Comparison with State-of-the-art LLMs

We compare small, open LMs Gemma-2B, Falcon-11B, Mistral-7B-I and Gemma-2B-I (overall best two from each category) with recent SOTA LLMs like GPT-4o-mini, GPT-4o (OpenAI et al., 2024; OpenAI, 2023), and Gemini-1.5-Pro (Team et al., 2024a). GPT-4o, Gemini-1.5-Pro and GPT-4o-mini are costly, large, closed models accessible using APIs. We use 8 examples with task definition for SOTA models, and report results in Figure 6.

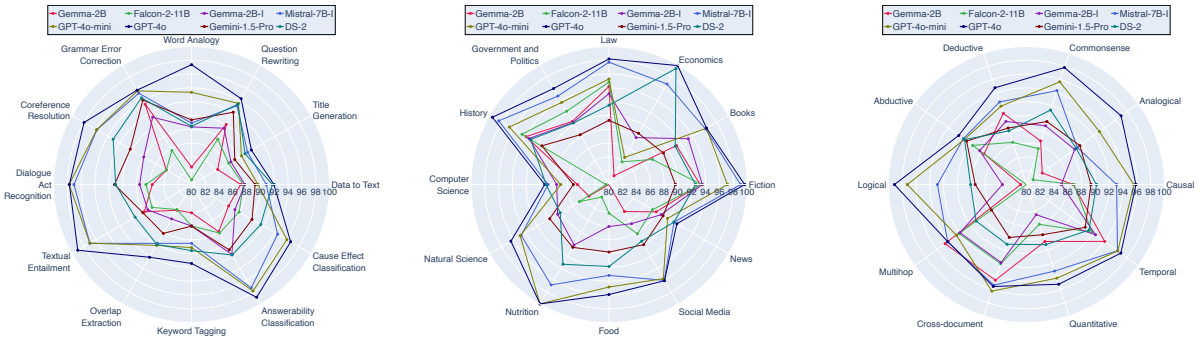
We witness that Mistral-7B-I matches closely with all SOTA models globally. It's even very close to GPT-4o in some groups like Generation tasks, Art and Literature, and Media and Entertainment domains. All the 4 models outperform GPT-4o-mini, Gemini-1.5-Pro and DS-2 in many categories where they are strong, proving them to be a very strong choice. In application domains like in Social Sciences and Humanities group and Art and Literature group, Gemma-2B and Gemma-2B-I outperform Gemini-1.5-Pro as well. Being the open-sourced variant of a close family, this is commendable and shows that open LMs can be better choices than large or expensive ones in some usage scenarios. Many inferences can be drawn from the graph based a reader's need through this evaluation framework. From the average global % decrease in performance reported in Table 1, these models are globally competitive with the SOTA LLMs, proving their readiness in being utilized for practical applications with their other advantages as discussed previously. The gaps of pre-trained models are higher than IT models, but aligning them further for specific use can improve results. We also evaluate these SOTA LMs for all entities of each aspect in Appendix B.

LM	Gem-1.5	G-4o-m	G-4o
Gemma-2B	3.28%	8.12%	9.78%
Falcon-2-11B	3.54%	8.37%	10.02%
Gemma-2B-I	1.44%	6.38%	8.07%
Mistral-7B-I	-4.94%	0.32%	2.12%

Table 1: Avg. Percentage decrease in mean BERTScore recall of open LMs compared to Gemini-1.5-Pro (Gem-1.5), GPT-4o-mini (G-4o-m) and GPT-4o (G-4o).

3.6 Comparison Across Prompt Styles

Language models' behavior depends significantly on the prompts. Writing good task descriptions and in-context examples requires time, good understanding of subtle variations, sufficient domain



(a) Task Types (b) Domains (c) Reasonings

Figure 6: Mean BERTScore recall across various task types, domains, and reasoning types, compared against SOTA (Note the change in range and interval gaps for better visibility of small differences).

knowledge, etc., which is not straightforward. So, we analyze how the performance varies for each entity of each aspect with changing instruction, focusing on the best performing IT model - Mistral-7B-I, since it can directly be used if prompted correctly.

We visualize the results in Figure 7. Using this, users can analyze the trade-offs of crafting instructions versus its possible impact on performance.

On initial analysis, using chat-style definitions proves better, but the performance increase looks small after 2 examples. So, using 2 examples can suffice. This trend is consistent for most entities across all three aspects. However, adding definition impacts different entities differently. For example, dialogue act recognition's performance on zero examples increases from 80.37 to 88.77 just by including task definition. But, for keyword tagging, the change is from 82.73 to only 82.81. We also see behaviors like Word Analogy, for which more examples negatively impact the output if definition is not provided. It may be because in absence of clear instruction, the model fails to comprehend the task from examples. Further, taking 'Social Media', adding task definition increases performance from 82.27 to 91.58 without examples, but, adding 2 examples without definition also improves score to 93.17. So, a choice is available between definition and examples. The rate of improvement with adding examples is also different for different entities. Some tasks don't have 8 examples in the dataset, so 4 to 8 example transition should be inferred accordingly.

Using these graphs, one can determine a prompt style for an application within other constraints of ability, cost, need, etc. in crafting instructions. These trends are different for different LMs. So, we have included these line graphs for all other LMs in Appendix D.2. This will also help in analyzing best prompt style and studying relative performance difference of each entity of each aspect.

3.7 Task Definition v/s Paraphrased Definition

To evaluate dependency of models to the provided task definition, we also evaluate them with their paraphrases. These are generated using `gpt-3.5-turbo` (Brown et al., 2020; OpenAI, 2023), and used with best in-context example count as per Table 7. Then, results are evaluated using the same pipeline, and reported in Table 2 for the two-best performing LMs in each category.

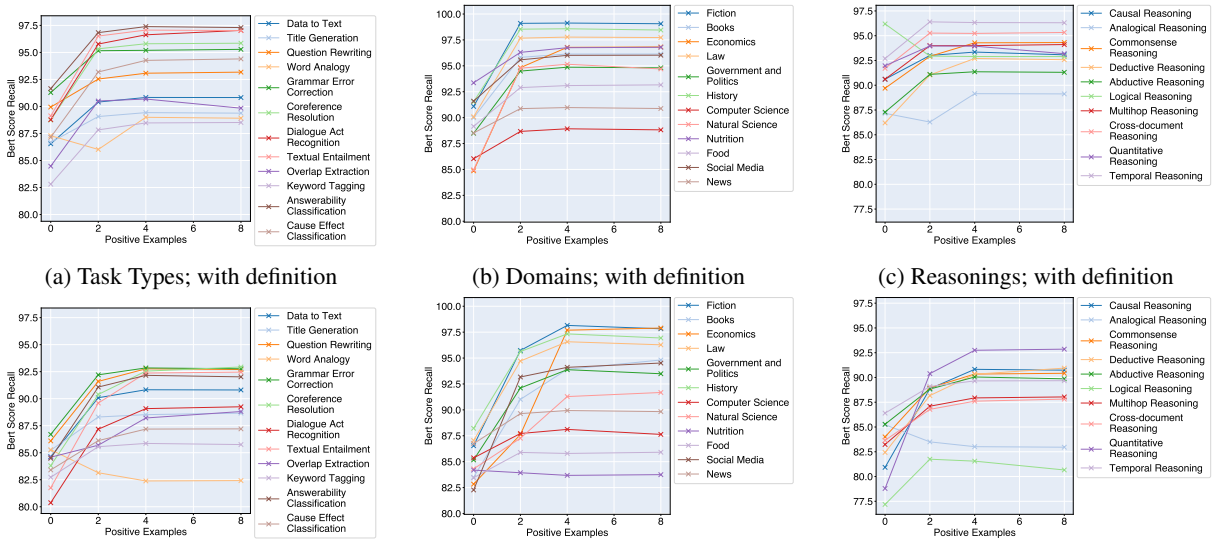
Model Name	Ex.	Def	Par. Def.
Gemma-2B	4	86.41	85.77
Falcon-2-11B	8	86.18	86.00
Gemma-2B-I	4	87.96	87.67
Mistral-7B-I	8	93.76	93.22

Table 2: Mean BERTScore recall values of outputs with actual task definition (Def) versus paraphrased definitions (Par. Def) using 'Ex.' in-context examples.

The median decrease in performance for all 10 LMs also is only 0.35%., which can be attributed to some loss of information during paraphrasing. But, most of the models prove robust to perturbations in task definitions, as long as a prompt can reasonably explain the task. Appendix D.3 has more details on obtaining paraphrases and results on all LMs.

4 High-level Takeaways

We find that recent, open and small-scale Language Models (LMs) are very effective. Detailed recommendations on LMs and their performance trends in different groups and entities are discussed in



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

Figure 7: Mean BERTScore recall for Mistral-7B-I for task types, domains, and reasoning types by varying in-context examples, segmented by with and without using task definitions.

depth in Sections 3.2, 3.3 and 3.4, but we summarize them in the below paragraphs too. Although it is visible that no single LM is a global solution, but, if selected and used appropriately with an effective prompt style for a task type, domain, or reasoning type, they can perform within 10% (worst case) of SOTA LLMs like GPT-4o, and outperform DS-2, GPT-4o-mini, and Gemini-1.5-Pro with advantages in efficiency, control and cost.

For the LMs we experimented with, among pre-trained models, we recommend using Gemma-2B and Falcon-2-11B based on different aspects and entities, but sometimes, Gemma-7B, Llama-3-8B can be great choices (also detailed in Appendix B). The performance of pre-trained models can be taken as a measure of their knowledge of different use-cases. Based on other factors like availability, compliance, size, right LM can be selected and customized as needed. Limitations of some pre-trained models are discussed in Appendix F.2.

For IT models, Mistral-7B-I is a clear best in all aspects, and Gemma-2B-I and SmolLM-1.7B-I come second in most cases. Since these models are IT, they can be used directly with chat-style description and examples. We recommend a model in these three (and other models), based on other factors like size, licensing, etc. Some qualitative outputs of Mistral-7B-I are given in Appendix F.1.

We also study the performance trade-off for multiple prompt styles and recommend the best one for these models. As noted in Section 3.6, having a chat-style task description to guide the LM is recommended. But, having more in-context examples is not always better, and considering use-case and LMs, the right number of example can vary. The models are also robust to changes in task definitions, if it can provide all (or most) information to complete the task. They are also reasonably robust to subtle intended/unintended incorrectness in definitions, which is analyzed in Appendix D.4. In appendix B, we also compare and show that the small LMs even outperform DeepSeek-v2 (DS-2) (DeepSeek-AI, 2024) in many categories.

5 Conclusion

We identify some limitations of using SOTA, proprietary LLMs and show that open LMs with 1.7B–11B parameters can be effective for applications. We create a three-tier evaluation framework and analyze semantic correctness of output of 10 LMs across multiple hierarchical umbrellas. Using this framework, we demonstrate that while these models don't work best in every scenario, if selected properly, they are effective and can compete with and outperform models like Gemini-1.5-Pro, GPT-4o-mini and GPT-4o despite being 10-20 folds smaller in size. We also provide a guide in Appendix A on how one can this work to select an LM for one's specific needs. We hope that our contributions will enable the community to make a confident shift towards considering using these small, open LMs for their need.

6 Limitations

Limitations of Dataset: We derive our experimental dataset from the test-set of Super Natural Instructions (Wang et al., 2022) and take the labels of aspects from there itself. We therefore assume that they are labeled correctly for task types, domains and reasoning types. There are many other task types, domains and reasoning types not available in its test set, which we were not able to consider. However, if an entity of an aspect is not present, one can leverage the performance of the groups that we created in Figure 3, or choose the nearest entity (from Section 3/ Appendix B) and roughly estimate the performance. We were also bounded in prompts by the examples and task definitions available. But, we did experiment by paraphrasing definitions in Section 3.7, Appendix D.3 to limit this to some extent. Using this dataset however may not bring significant dataset bias, as this is a meta-dataset curated using several NLP benchmark datasets.

Limitations of LMs: There are many LMs proposed by the research community, and it is not practically feasible to experiment with all of them. Further, the definition of a small LM is also relative. We selected the LMs based on the ones which have come out recently and promise strong capabilities. Although we capped our LMs at 11B parameters, we didn't find the performance to be a direct function of number of parameters, so we believe this decision should not have too drastic effects. We also didn't evaluate biases (Gallegos et al., 2024) and other factors other than semantic correctness of generated results of these models, but the models' technical reports (Allal et al., 2024; Team et al., 2024b; AI@Meta, 2024; Almazrouei et al., 2023; TIIUAE, 2024) provide more details on those.

Limitations of Prompts: We experimented with 8 prompt styles, apart from using paraphrased definitions, adversarial definitions. But, all of them (excluding paraphrased and adversarial definitions) were built using the elements of the dataset available. We acknowledge that there may be some tasks where another prompt style or using more domain-adapted prompts perform better. Additionally, if the LM is adapted/fine-tuned in any way, the best prompt style can change based on the data and technique used for it. However, to keep a standard and common features across tasks, we intentionally chose this approach. This study should provide an initial idea of whether descriptions are needed and the number of examples required when using the LM without any changes.

Assumptions in Reporting of Results: We are considering the impact of each aspect one at a time when reporting results. For example, in pre-trained models, we see that Gemma-2B is best for Grammar Error Correction, and Falcon-2-11B is best for Economics domain. But what if there is a task instance that involves grammar error correction for an Economics article? This can sometimes give a dual outcome, with one LM recommended for task type (grammar error correction), and one for domain (Economics). To eliminate this, we tried to do a pairwise aspect analysis, but in the dataset, 86.86% of task type-domain and 88.25% of domain-reasoning type pairs had no task instances. We could have generated labels of aspect entities using other techniques, or could've generated artificial data to fill these gaps, but we didn't want conflicting sources of experimental data as they could created additional undesired biases/variations of sources, type of data. Therefore, due to lack of sufficient labels, we didn't report those results. However, despite this independent assumption, this work can still help in narrowing down to 2-3 models which will be helpful. We also assume that the results reported by our experimental data represents the actual performance of that LM on that entity type. This may not be correct but considering Super Natural Instructions to be a meta-dataset of many other datasets, we believe it is a reasonable assumption.

7 Ethical Considerations

This work evaluates performance of Language Models in terms of semantic correctness of outputs on various task types, application domains and reasoning types using different prompt styles. While we only included the entities that help the community, one can utilize/extrapolate the conclusions of this work for applications that are harmful. Further, one can create prompts using task definitions, in-context examples to extract negative behavior from the LMs, or attempt adversarial attacks on these LMs. We strictly discourage utilizing the results of this work or LMs in general in such ways. We also didn't evaluate these LMs on Bias and Fairness as it was out of scope of this paper. This work (Gallegos et al., 2024) discusses different types of biases and mitigation strategies.

References

AI@Meta. 2024. Llama 3 model card.

Falwah Alhamed, Julia Ive, and Lucia Specia. 2024. Using large language models (LLMs) to extract evidence from pre-annotated social media data. In *Proceedings of the 9th Workshop on Computational Linguistics and Clinical Psychology (CLPsych 2024)*, pages 232–237, St. Julians, Malta. Association for Computational Linguistics.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Leandro von Werra, and Thomas Wolf. 2024. Smollm - blazingly fast and remarkably powerful.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. The falcon series of open language models. *Preprint*, arXiv:2311.16867.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

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

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Preprint*, arXiv:2205.14135.

DeepSeek-AI. 2024. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *Preprint*, arXiv:2405.04434.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Preprint*, arXiv:2305.14314.

Tianyu Ding, Tianyi Chen, Haidong Zhu, Jiachen Jiang, Yiqi Zhong, Jinxin Zhou, Guangzhi Wang, Zhihui Zhu, Ilya Zharkov, and Luming Liang. 2024. The efficiency spectrum of large language models: An algorithmic survey. *Preprint*, arXiv:2312.00678.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. 2023. A survey on in-context learning. *Preprint*, arXiv:2301.00234.

Shangbin Feng, Chan Young Park, Yuhan Liu, and Yulia Tsvetkov. 2023. From pretraining data to language models to downstream tasks: Tracking the trails of political biases leading to unfair NLP models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11737–11762, Toronto, Canada. Association for Computational Linguistics.

Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K. Ahmed. 2024. Bias and fairness in large language models: A survey. *Preprint*, arXiv:2309.00770.

Zishan Guo, Renren Jin, Chuang Liu, Yufei Huang, Dan Shi, Supryadi, Linhao Yu, Yan Liu, Jiaxuan Li, Bojian Xiong, and Deyi Xiong. 2023. Evaluating large language models: A comprehensive survey. *Preprint*, arXiv:2310.19736.

Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. 2024. Parameter-efficient fine-tuning for large models: A comprehensive survey. *Preprint*, arXiv:2403.14608.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Jingang Wang, Juanzi Li, Wei Wu, and Maosong Sun. 2022. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2225–2240, Dublin, Ireland. Association for Computational Linguistics.

Jie Huang and Kevin Chen-Chuan Chang. 2023. Towards reasoning in large language models: A survey. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1049–1065, Toronto, Canada. 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.

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

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *Preprint*, arXiv:2001.08361.

Onur Keles and Omer Turan Bayraklı. 2024. LLaMA-2-econ: Enhancing title generation, abstract classification, and academic Q&A in economic research. In *Proceedings of the Joint Workshop of the 7th Financial Technology and Natural Language Processing, the 5th Knowledge Discovery from Unstructured Data in Financial Services, and the 4th Workshop on Economics and Natural Language Processing @ LREC-COLING 2024*, pages 212–218, Torino, Italia. ELRA and ICCL.

Alapan Kuila and Sudeshna Sarkar. 2024. Deciphering political entity sentiment in news with large language models: Zero-shot and few-shot strategies. In *Proceedings of the Second Workshop on Natural Language Processing for Political Sciences @ LREC-COLING 2024*, pages 1–11, Torino, Italia. ELRA and ICCL.

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

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

Pingchuan Ma, Rui Ding, Shuai Wang, Shi Han, and Dongmei Zhang. 2023. InsightPilot: An LLM-empowered automated data exploration system. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 346–352, Singapore. Association for Computational Linguistics.

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

OpenAI. 2023. Gpt-3.5 turbo. OpenAI API. Available from OpenAI: `https://platform.openai.com/docs/models/gpt-3.5-turbo`.

OpenAI. 2023. Gpt-4o. `https://www.openai.com/`. Accessed: 2024-06-06.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever,

Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Gray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*.

Liangming Pan, Wenhu Chen, Wenhan Xiong, Min-Yen Kan, and William Yang Wang. 2021. Unsupervised multi-hop question answering by question generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5866–5880, Online. Association for Computational Linguistics.

Yao Qiang, Subhrangshu Nandi, Ninareh Mehrabi, Greg Ver Steeg, Anoop Kumar, Anna Rumshisky, and Aram Galstyan. 2024. Prompt perturbation consistency learning for robust language models. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1357–1370, St. Julian's, Malta. Association for Computational Linguistics.

Pawan Kumar Rajpoot, Ashvini Jindal, and Ankur Parikh. 2024. Adapting LLM to multi-lingual ESG impact and length prediction using in-context learning and fine-tuning with rationale. In *Proceedings of the Joint Workshop of the 7th Financial Technology and Natural Language Processing, the 5th Knowledge Discovery from Unstructured Data in Financial Services, and the 4th Workshop on Economics and Natural Language Processing @ LREC-COLING 2024*, pages 274–278, Torino, Italia. ELRA and ICCL.

Ananya B. Sai, Tanay Dixit, Dev Yashpal Sheth, Sreyas Mohan, and Mitesh M. Khapra. 2021. Perturbation CheckLists for evaluating NLG evaluation metrics. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7219–7234, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, Soroosh Mariooryad, Yifan Ding, Xinyang Geng, Fred Alcober, Roy Frostig, Mark Omernick, Lexi Walker, Cosmin Paduraru, Christina Sorokin, Andrea Tacchetti, Colin Gaffney, Samira Daruki, Olcan Sercinoglu, Zach Gleicher, Juliette Love, Paul Voigtlaender, Rohan Jain, Gabriela Surita, Kareem Mohamed, Rory Blevins, Junwhan Ahn, Tao Zhu, Kornraphop Kawintiranon, Orhan Firat, Yiming Gu, Yujing Zhang, Matthew Rahtz, Manaal Faruqui, Natalie Clay, Justin Gilmer, JD Co-Reyes, Ivo Penchev, Rui Zhu, Nobuyuki Morioka, Kevin Hui, Krishna Haridasan, Victor Campos, Mahdis Mahdieh, Mandy Guo, Samer Hassan, Kevin Kilgour, Arpi Vezer, Heng-Tze Cheng, Raoul de Liedekerke, Siddharth Goyal, Paul Barham, DJ Strouse, Seb Noury, Jonas Adler, Mukund Sundararajan, Sharad Vikram, Dmitry Lepikhin, Michela Paganini, Xavier Garcia, Fan Yang, Dasha Valter, Maja Trebacz, Kiran Vodrahalli, Chulayuth Asawaroengchai, Roman Ring, Norbert Kalb, Livio Baldini Soares, Siddhartha Brahma, David Steiner, Tianhe Yu, Fabian Mentzer, Antoine He, Lucas Gonzalez, Bibo Xu, Raphael Lopez Kaufman, Laurent El Shafey, Junhyuk Oh, Tom Hennigan, George van den Driessche, Seth Odoom, Mario Lucic, Becca Roelofs, Sid Lall, Amit Marathe, Betty Chan, Santiago Ontanon, Luheng He, Denis Teplyashin, Jonathan Lai, Phil Crone, Bogdan Damoc, Lewis Ho, Sebastian Riedel, Karel Lenc, Chih-Kuan Yeh, Aakanksha Chowdhery, Yang Xu, Mehran Kazemi, Ehsan Amid, Anastasia Petrushkina, Kevin Swersky, Ali Khodaei, Gowoon Chen, Chris Larkin, Mario Pinto, Geng Yan, Adria Puigdomenech Badia, Piyush Patil, Steven Hansen, Dave Orr, Sebastien M. R. Arnold, Jordan Grimstad, Andrew Dai, Sholto Douglas, Rishika Sinha, Vikas Yadav, Xi Chen, Elena Gribovskaya, Jacob Austin, Jeffrey Zhao, Kaushal Patel, Paul Komarek, Sophia Austin, Sebastian Borgeaud, Linda Friso, Abhimanyu Goyal, Ben Caine, Kris Cao, Da-Woon Chung, Matthew Lamm, Gabe Barth-Maron, Thais Kagohara, Kate Olszewska, Mia Chen, Kaushik Shivakumar, Rishabh Agarwal, Harshal Godhia, Ravi Rajwar, Javier Snaider, Xerxes Dotiwalla, Yuan Liu, Aditya Barua, Victor Ungureanu, Yuan Zhang, Bat-Orgil Batsaikhan, Mateo Wirth, James Qin, Ivo Danihelka, Tulsee Doshi, Martin Chadwick, Jilin Chen, Sanil Jain, Quoc Le, Arjun Kar, Madhu Gurumurthy, Cheng Li, Ruoxin Sang, Fangyu Liu, Lampros Lamprou, Rich Munoz, Nathan Lintz, Harsh Mehta, Heidi Howard, Malcolm Reynolds, Lora Aroyo, Quan Wang, Lorenzo Blanco, Albin Cassirer, Jordan Griffith, Dipanjan Das, Stephan Lee, Jakub Sygnowski, Zach Fisher, James Besley, Richard Powell, Zafarali Ahmed, Dominik Paulus, David Reitter, Zalan Borsos, Rishabh Joshi, Aedan Pope, Steven Hand, Vittorio Selo, Vihan Jain, Nikhil Sethi, Megha Goel, Takaki Makino, Rhys May, Zhen Yang, Johan Schalkwyk, Christina Butterfield, Anja Hauth, Alex Goldin, Will Hawkins, Evan Senter, Sergey Brin, Oliver Woodman, Marvin Ritter, Eric Noland, Minh Giang, Vijay Bolina, Lisa Lee, Tim Blyth, Ian Mackinnon, Machel Reid, Obaid Sarvana, David Silver, Alexander Chen, Lily Wang, Loren Maggiore, Oscar Chang, Nithya At-

taluri, Gregory Thornton, Chung-Cheng Chiu, Oskar Bunyan, Nir Levine, Timothy Chung, Evgenii Eltyshev, Xiance Si, Timothy Lillicrap, Demetra Brady, Vaibhav Aggarwal, Boxi Wu, Yuanzhong Xu, Ross McIlroy, Kartikeya Badola, Paramjit Sandhu, Erica Moreira, Wojciech Stokowiec, Ross Hemsley, Dong Li, Alex Tudor, Pranav Shyam, Elahe Rahimtoroghi, Salem Haykal, Pablo Sprechmann, Xiang Zhou, Diana Mincu, Yujia Li, Ravi Addanki, Kalpesh Krishna, Xiao Wu, Alexandre Frechette, Matan Eyal, Allan Dafoe, Dave Lacey, Jay Whang, Thi Avrahami, Ye Zhang, Emanuel Taropa, Hanzhao Lin, Daniel Toyama, Eliza Rutherford, Motoki Sano, HyunJeong Choe, Alex Tomala, Chalence Safranek-Shrader, Nora Kassner, Mantas Pajarskas, Matt Harvey, Sean Sechrist, Meire Fortunato, Christina Lyu, Gamaleldin Elsayed, Chenkai Kuang, James Lottes, Eric Chu, Chao Jia, Chih-Wei Chen, Peter Humphreys, Kate Baumli, Connie Tao, Rajkumar Samuel, Cicero Nogueira dos Santos, Anders Andreassen, Nemanja Rakićević, Dominik Grewe, Aviral Kumar, Stephanie Winkler, Jonathan Caton, Andrew Brock, Sid Dalmia, Hannah Sheahan, Iain Barr, Yingjie Miao, Paul Natsev, Jacob Devlin, Feryal Behbahani, Flavien Prost, Yanhua Sun, Artiom Myaskovsky, Thanumalayan Sankaranarayana Pillai, Dan Hurt, Angeliki Lazaridou, Xi Xiong, Ce Zheng, Fabio Pardo, Xiaowei Li, Dan Horgan, Joe Stanton, Moran Ambar, Fei Xia, Alejandro Lince, Mingqiu Wang, Basil Mustafa, Albert Webson, Hyo Lee, Rohan Anil, Martin Wicke, Timothy Dozat, Abhishek Sinha, Enrique Piqueras, Elahe Dabir, Shyam Upadhyay, Anudhyan Boral, Lisa Anne Hendricks, Corey Fry, Josip Djolonga, Yi Su, Jake Walker, Jane Labanowski, Ronny Huang, Vedant Misra, Jeremy Chen, RJ Skerry-Ryan, Avi Singh, Shruti Rijhwani, Dian Yu, Alex Castro-Ros, Beer Changpinyo, Romina Datta, Sumit Bagri, Arnar Mar Hrafnkelsson, Marcello Maggioni, Daniel Zheng, Yury Sulsky, Shaobo Hou, Tom Le Paine, Antoine Yang, Jason Riesa, Dominika Rogozinska, Dror Marcus, Dalia El Badawy, Qiao Zhang, Luyu Wang, Helen Miller, Jeremy Greer, Lars Lowe Sjos, Azade Nova, Heiga Zen, Rahma Chaabouni, Mihaela Rosca, Jiepu Jiang, Charlie Chen, Ruibo Liu, Tara Sainath, Maxim Krikun, Alex Polozov, Jean-Baptiste Lespiau, Josh Newlan, Zeyncep Cankara, Soo Kwak, Yunhan Xu, Phil Chen, Andy Coenen, Clemens Meyer, Katerina Tsihlas, Ada Ma, Juraj Gottweis, Jinwei Xing, Chenjie Gu, Jin Miao, Christian Frank, Zeynep Cankara, Sanjay Ganapathy, Ishita Dasgupta, Steph Hughes-Fitt, Heng Chen, David Reid, Keran Rong, Hongmin Fan, Joost van Amersfoort, Vincent Zhuang, Aaron Cohen, Shixiang Shane Gu, Anhad Mohananey, Anastasija Ilic, Taylor Tobin, John Wieting, Anna Bortsova, Phoebe Thacker, Emma Wang, Emily Caveness, Justin Chiu, Eren Sezener, Alex Kaskasoli, Steven Baker, Katie Millican, Mohamed Elhawaty, Kostas Aisopos, Carl Lebsack, Nathan Byrd, Hanjun Dai, Wenhao Jia, Matthew Wiethoff, Elnaz Davoodi, Albert Weston, Lakshman Yagati, Arun Ahuja, Isabel Gao, Golan Pundak, Susan Zhang, Michael Azzam, Khe Chai Sim, Sergi Caelles, James Keeling, Abhanshu Sharma, Andy Swing, YaGuang Li, Chenxi Liu, Carrie Grimes Bostock, Yamini Bansal, Zachary Nado, Ankesh Anand, Josh Lipschultz, Abhijit Karmarkar, Lev Proleev, Abe Ittycheriah, Soheil Hassas Yeganeh, George Polovets, Aleksandra Faust, Jiao Sun, Alban Rrustemi, Pen Li, Rakesh Shivanna, Jeremiah Liu, Chris Welty, Federico Lebron, Anirudh Baddepudi, Sebastian Krause, Emilio Parisotto, Radu Soricut, Zheng Xu, Dawn Bloxwich, Melvin Johnson, Behnam Neyshabur, Justin Mao-Jones, Renshen Wang, Vinay Ramasesh, Zaheer Abbas, Arthur Guez, Constant Segal, Duc Dung Nguyen, James Svensson, Le Hou, Sarah York, Kieran Milan, Sophie Bridgers, Wiktor Gworek, Marco Tagliasacchi, James Lee-Thorp, Michael Chang, Alexey Guseynov, Ale Jakse Hartman, Michael Kwong, Ruizhe Zhao, Sheleem Kashem, Elizabeth Cole, Antoine Miech, Richard Tanburn, Mary Phuong, Filip Pavetic, Sebastien Cevey, Ramona Comanescu, Richard Ives, Sherry Yang, Cosmo Du, Bo Li, Zizhao Zhang, Mariko Iinuma, Clara Huiyi Hu, Aurko Roy, Shaan Bijwadia, Zhenkai Zhu, Danilo Martins, Rachel Saputro, Anita Gergely, Steven Zheng, Dawei Jia, Ioannis Antonoglou, Adam Sadovsky, Shane Gu, Yingying Bi, Alek Andreev, Sina Samangooei, Mina Khan, Tomas Kocisky, Angelos Filos, Chintu Kumar, Colton Bishop, Adams Yu, Sarah Hodkinson, Sid Mittal, Premal Shah, Alexandre Moufarek, Yong Cheng, Adam Bloniarz, Jaehoon Lee, Pedram Pejman, Paul Michel, Stephen Spencer, Vladimir Feinberg, Xuehan Xiong, Nikolay Savinov, Charlotte Smith, Siamak Shakeri, Dustin Tran, Mary Chesus, Bernd Bohnet, George Tucker, Tamara von Glehn, Carrie Muir, Yiran Mao, Hideto Kazawa, Ambrose Slone, Kedar Soparkar, Disha Shrivastava, James Cobon-Kerr, Michael Sharman, Jay Pavagadhi, Carlos Araya, Karolis Misiunas, Nimesh Ghelani, Michael Laskin, David Barker, Qiujia Li, Anton Briukhov, Neil Houlsby, Mia Glaese, Balaji Lakshminarayanan, Nathan Schucher, Yunhao Tang, Eli Collins, Hyeontaek Lim, Fangxiaoyu Feng, Adria Recasens, Guangda Lai, Alberto Magni, Nicola De Cao, Aditya Siddhant, Zoe Ashwood, Jordi Orbay, Mostafa Dehghani, Jenny Brennan, Yifan He, Kelvin Xu, Yang Gao, Carl Saroufim, James Molloy, Xinyi Wu, Seb Arnold, Solomon Chang, Julian Schrittwieser, Elena Buchatskaya, Soroush Radpour, Martin Polacek, Skye Giordano, Ankur Bapna, Simon Tokumine, Vincent Hellendoorn, Thibault Sottiaux, Sarah Cogan, Aliaksei Severyn, Mohammad Saleh, Shantanu Thakoor, Laurent Shefey, Siyuan Qiao, Meenu Gaba, Shuo yiin Chang, Craig Swanson, Biao Zhang, Benjamin Lee, Paul Kishan Rubenstein, Gan Song, Tom Kwiatkowski, Anna Koop, Ajay Kannan, David Kao, Parker Schuh, Axel Stjerngren, Golnaz Ghiasi, Gena Gibson, Luke Vilnis, Ye Yuan, Felipe Tiengo Ferreira, Aishwarya Kamath, Ted Klimenko, Ken Franko, Kefan Xiao, Indro Bhattacharya, Miteyan Patel, Rui Wang, Alex Morris, Robin Strudel, Vivek Sharma, Peter Choy, Sayed Hadi Hashemi, Jessica Landon, Mara Finkelstein, Priya Jhakra, Justin Frye, Megan Barnes, Matthew Mauger, Dennis Daun, Khuslen Baatarsukh, Matthew Tung, Wael Farhan, Henryk Michalewski, Fabio Viola, Felix de Chaumont Quitry, Charline Le Lan, Tom Hud-

son, Qingze Wang, Felix Fischer, Ivy Zheng, Elspeth White, Anca Dragan, Jean baptiste Alayrac, Eric Ni, Alexander Pritzel, Adam Iwanicki, Michael Isard, Anna Bulanova, Lukas Zilka, Ethan Dyer, Devendra Sachan, Srivatsan Srinivasan, Hannah Muckenhirn, Honglong Cai, Amol Mandhane, Mukarram Tariq, Jack W. Rae, Gary Wang, Kareem Ayoub, Nicholas FitzGerald, Yao Zhao, Woohyun Han, Chris Alberti, Dan Garrette, Kashyap Krishnakumar, Mai Gimenez, Anselm Levskaya, Daniel Sohn, Josip Matak, Inaki Iturrate, Michael B. Chang, Jackie Xiang, Yuan Cao, Nishant Ranka, Geoff Brown, Adrian Hutter, Vahab Mirrokni, Nanxin Chen, Kaisheng Yao, Zoltan Egyed, Francois Galilee, Tyler Liechty, Praveen Kallakuri, Evan Palmer, Sanjay Ghemawat, Jasmine Liu, David Tao, Chloe Thornton, Tim Green, Mimi Jasarevic, Sharon Lin, Victor Cotruta, Yi-Xuan Tan, Noah Fiedel, Hongkun Yu, Ed Chi, Alexander Neitz, Jens Heitkaemper, Anu Sinha, Denny Zhou, Yi Sun, Charbel Kaed, Brice Hulse, Swaroop Mishra, Maria Georgaki, Sneha Kudugunta, Clement Farabet, Izhak Shafran, Daniel Vlasic, Anton Tsitsulin, Rajagopal Ananthanarayanan, Alen Carin, Guolong Su, Pei Sun, Shashank V, Gabriel Carvajal, Josef Broder, Iulia Comsa, Alena Repina, William Wong, Warren Weilun Chen, Peter Hawkins, Egor Filonov, Lucia Loher, Christoph Hirnschall, Weiyi Wang, Jingchen Ye, Andrea Burns, Hardie Cate, Diana Gage Wright, Federico Piccinini, Lei Zhang, Chu-Cheng Lin, Ionel Gog, Yana Kulizhskaya, Ashwin Sreevatsa, Shuang Song, Luis C. Cobo, Anand Iyer, Chetan Tekur, Guillermo Garrido, Zhuyun Xiao, Rupert Kemp, Huaixiu Steven Zheng, Hui Li, Ananth Agarwal, Christel Ngani, Kati Goshvadi, Rebeca Santamaria-Fernandez, Wojciech Fica, Xinyun Chen, Chris Gorgolewski, Sean Sun, Roopal Garg, Xinyu Ye, S. M. Ali Eslami, Nan Hua, Jon Simon, Pratik Joshi, Yelin Kim, Ian Tenney, Sahitya Potluri, Lam Nguyen Thiet, Quan Yuan, Florian Luisier, Alexandra Chronopoulou, Salvatore Scellato, Praveen Srinivasan, Minmin Chen, Vinod Koverkathu, Valentin Dalibard, Yaming Xu, Brennan Saeta, Keith Anderson, Thibault Sellam, Nick Fernando, Fantine Huot, Junehyuk Jung, Mani Varadarajan, Michael Quinn, Amit Raul, Maigo Le, Ruslan Habalov, Jon Clark, Komal Jalan, Kalesha Bullard, Achintya Singhal, Thang Luong, Boyu Wang, Sujeevan Rajayogam, Julian Eisenschlos, Johnson Jia, Daniel Finchelstein, Alex Yakubovich, Daniel Balle, Michael Fink, Sameer Agarwal, Jing Li, Dj Dvijotham, Shalini Pal, Kai Kang, Jaclyn Konzelmann, Jennifer Beattie, Olivier Dousse, Diane Wu, Remi Crocker, Chen Elkind, Siddhartha Reddy Jonnalagadda, Jong Lee, Dan Holtmann-Rice, Krystal Kallarackal, Rosanne Liu, Denis Vnukov, Neera Vats, Luca Invernizzi, Mohsen Jafari, Huanjie Zhou, Lilly Taylor, Jennifer Prendki, Marcus Wu, Tom Eccles, Tianqi Liu, Kavya Kopparapu, Francoise Beaufays, Christof Angermueller, Andreea Marzoca, Shourya Sarcar, Hilal Dib, Jeff Stanway, Frank Perbet, Nejc Trdin, Rachel Sterneck, Andrey Khorlin, Dinghua Li, Xihui Wu, Sonam Goenka, David Madras, Sasha Goldshtein, Willi Gierke, Tong Zhou, Yaxin Liu, Yannie Liang, Anais White, Yunjie Li, Shreya Singh, Sanaz Bahargam, Mark Epstein, Sujoy Basu, Li Lao, Adnan Ozturel, Carl Crous, Alex Zhai, Han Lu, Zora Tung, Neeraj Gaur, Alanna Walton, Lucas Dixon, Ming Zhang, Amir Globerson, Grant Uy, Andrew Bolt, Olivia Wiles, Milad Nasr, Ilia Shumailov, Marco Selvi, Francesco Piccinno, Ricardo Aguilar, Sara McCarthy, Misha Khalman, Mrinal Shukla, Vlado Galic, John Carpenter, Kevin Villela, Haibin Zhang, Harry Richardson, James Martens, Matko Bosnjak, Shreyas Rammohan Belle, Jeff Seibert, Mahmoud Alnahlawi, Brian McWilliams, Sankalp Singh, Annie Louis, Wen Ding, Dan Popovici, Lenin Simicich, Laura Knight, Pulkit Mehta, Nishesh Gupta, Chongyang Shi, Saaber Fatehi, Jovana Mitrovic, Alex Grills, Joseph Pagadora, Dessie Petrova, Danielle Eisenbud, Zhishuai Zhang, Damion Yates, Bhavishya Mittal, Nilesh Tripuraneni, Yannis Assael, Thomas Brovelli, Prateek Jain, Mihajlo Velimirovic, Canfer Akbulut, Jiaqi Mu, Wolfgang Macherey, Ravin Kumar, Jun Xu, Haroon Qureshi, Gheorghe Comanici, Jeremy Wiesner, Zhitao Gong, Anton Ruddock, Matthias Bauer, Nick Felt, Anirudh GP, Anurag Arnab, Dustin Zelle, Jonas Rothfuss, Bill Rosgen, Ashish Shenoy, Bryan Seybold, Xinjian Li, Jayaram Mudigonda, Goker Erdogan, Jiawei Xia, Jiri Simsa, Andrea Michi, Yi Yao, Christopher Yew, Steven Kan, Isaac Caswell, Carey Radebaugh, Andre Elisseeff, Pedro Valenzuela, Kay McKinney, Kim Paterson, Albert Cui, Eri Latorre-Chimoto, Solomon Kim, William Zeng, Ken Durden, Priya Ponnapalli, Tiberiu Sosea, Christopher A. Choquette-Choo, James Manyika, Brona Robenek, Harsha Vashisht, Sebastien Pereira, Hoi Lam, Marko Velic, Denese Owusu-Afriyie, Katherine Lee, Tolga Bolukbasi, Alicia Parrish, Shawn Lu, Jane Park, Balaji Venkatraman, Alice Talbert, Lambert Rosique, Yuchung Cheng, Andrei Sozanschi, Adam Paszke, Praveen Kumar, Jessica Austin, Lu Li, Khalid Salama, Wooyeol Kim, Nandita Dukkipati, Anthony Baryshnikov, Christos Kaplanis, XiangHai Sheng, Yuri Chervonyi, Caglar Unlu, Diego de Las Casas, Harry Askham, Kathryn Tunyasuvunakool, Felix Gimeno, Siim Poder, Chester Kwak, Matt Miecnikowski, Vahab Mirrokni, Alek Dimitriev, Aaron Parisi, Dangyi Liu, Tomy Tsai, Toby Shevlane, Christina Kouridi, Drew Garmon, Adrian Goedeckemeyer, Adam R. Brown, Anitha Vijayakumar, Ali Elqursh, Sadegh Jazayeri, Jin Huang, Sara Mc Carthy, Jay Hoover, Lucy Kim, Sandeep Kumar, Wei Chen, Courtney Biles, Garrett Bingham, Evan Rosen, Lisa Wang, Qijun Tan, David Engel, Francesco Pongetti, Dario de Cesare, Dongseong Hwang, Lily Yu, Jennifer Pullman, Srini Narayanan, Kyle Levin, Siddharth Gopal, Megan Li, Asaf Aharoni, Trieu Trinh, Jessica Lo, Norman Casagrande, Roopali Vij, Loic Matthey, Bramandia Ramadhana, Austin Matthews, CJ Carey, Matthew Johnson, Kremena Goranova, Rohin Shah, Shereen Ashraf, Kingshuk Dasgupta, Rasmus Larsen, Yicheng Wang, Manish Reddy Vuyyuru, Chong Jiang, Joana Ijazi, Kazuki Osawa, Celine Smith, Ramya Sree Boppana, Taylan Bilal, Yuma Koizumi, Ying Xu, Yasemin Altun, Nir Shabat, Ben Bariach, Alex Korchemniy, Kiam Choo, Olaf Ronneberger, Chimezie Iwuanyanwu, Shubin Zhao,

David Soergel, Cho-Jui Hsieh, Irene Cai, Shariq Iqbal, Martin Sundermeyer, Zhe Chen, Elie Bursztein, Chaitanya Malaviya, Fadi Biadsy, Prakash Shroff, Inderjit Dhillon, Tejasi Latkar, Chris Dyer, Hannah Forbes, Massimo Nicosia, Vitaly Nikolaev, Somer Greene, Marin Georgiev, Pidong Wang, Nina Martin, Hanie Sedghi, John Zhang, Praseem Banzal, Doug Fritz, Vikram Rao, Xuezhi Wang, Jiageng Zhang, Viorica Patraucean, Dayou Du, Igor Mordatch, Ivan Jurin, Lewis Liu, Ayush Dubey, Abhi Mohan, Janek Nowakowski, Vlad-Doru Ion, Nan Wei, Reiko Tojo, Maria Abi Raad, Drew A. Hudson, Vaishakh Keshava, Shubham Agrawal, Kevin Ramirez, Zhichun Wu, Hoang Nguyen, Ji Liu, Madhavi Sewak, Bryce Petrini, DongHyun Choi, Ivan Philips, Ziyue Wang, Ioana Bica, Ankush Garg, Jarek Wilkiewicz, Priyanka Agrawal, Xiaowei Li, Danhao Guo, Emily Xue, Naseer Shaik, Andrew Leach, Sadh MNM Khan, Julia Wiesinger, Sammy Jerome, Abhishek Chakladar, Alek Wenjiao Wang, Tina Ornduff, Folake Abu, Alireza Ghaffarkhah, Marcus Wainwright, Mario Cortes, Frederick Liu, Joshua Maynez, Andreas Terzis, Pouya Samangouei, Riham Mansour, Tomasz Kępa, François-Xavier Aubet, Anton Algymr, Dan Banica, Agoston Weisz, Andras Orban, Alexandre Senges, Ewa Andrejczuk, Mark Geller, Niccolo Dal Santo, Valentin Anklin, Majd Al Merey, Martin Baeuml, Trevor Strohman, Junwen Bai, Slav Petrov, Yonghui Wu, Demis Hassabis, Koray Kavukcuoglu, Jeffrey Dean, and Oriol Vinyals. 2024a. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *Preprint*, arXiv:2403.05530.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. 2024b. Gemma: Open models based on gemini research and technology. *Preprint*, arXiv:2403.08295.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozińska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucińska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjoesund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, Lilly McNealus, Livio Baldini Soares, Logan Kilpatrick, Lucas Dixon, Luciano Martins, Machel Reid, Manvinder Singh, Mark Iverson, Martin Görner, Mat Velloso, Mateo Wirth, Matt Davidow, Matt Miller, Matthew Rahtz, Matthew Watson, Meg Risdal, Mehran Kazemi, Michael Moynihan, Ming Zhang, Minsuk Kahng, Minwoo Park, Mofi Rahman, Mohit Khatwani, Natalie Dao, Nenshad Bardoliwalla, Nesh Devanathan, Neta Dumai, Nilay Chauhan, Oscar Wahltinez, Pankil Botarda, Parker Barnes, Paul Barham, Paul Michel, Pengchong Jin, Petko Georgiev, Phil Culliton, Pradeep Kuppala, Ramona Comanescu, Ramona Merhej, Reena Jana, Reza Ardeshir Rokni, Rishabh Agarwal, Ryan Mullins, Samaneh Saadat, Sara Mc Carthy, Sarah Perrin, Sébastien M. R. Arnold, Sebastian Krause, Shengyang Dai, Shruti Garg, Shruti Sheth, Sue Ronstrom, Susan Chan, Timothy Jordan, Ting Yu, Tom Eccles, Tom Hennigan, Tomas Kocisky, Tulsee Doshi, Vihan Jain, Vikas Yadav, Vilobh Meshram, Vishal Dharmadhikari, Warren Barkley, Wei Wei, Wenming Ye, Woohyun Han, Woosuk Kwon, Xiang Xu, Zhe Shen, Zhitao Gong, Zichuan Wei, Victor Cotruta, Phoebe Kirk, Anand Rao,

Minh Giang, Ludovic Peran, Tris Warkentin, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, D. Sculley, Jeanine Banks, Anca Dragan, Slav Petrov, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Sebastian Borgeaud, Noah Fiedel, Armand Joulin, Kathleen Kenealy, Robert Dadashi, and Alek Andreev. 2024c. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

TIIUAE. 2024. Falcon 11B. `https://huggingface.co/tiiuae/falcon-11B`.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. *arXiv preprint arXiv:2204.07705*.

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. 2022a. Emergent abilities of large language models. *Preprint*, arXiv:2206.07682.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022b. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Thilini Wijesiriwardene, Ruwan Wickramarachchi, Bimal Gajera, Shreeyash Gowaikar, Chandan Gupta, Aman Chadha, Aishwarya Naresh Reganti, Amit Sheth, and Amitava Das. 2023. ANALOGICAL - a novel benchmark for long text analogy evaluation in large language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3534–3549, Toronto, Canada. Association for Computational Linguistics.

Ziang Xiao, Susu Zhang, Vivian Lai, and Q. Vera Liao. 2023. Evaluating evaluation metrics: A framework for analyzing NLG evaluation metrics using measurement theory. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10967–10982, Singapore. Association for Computational Linguistics.

Zhongqi Yang, Elahe Khatibi, Nitish Nagesh, Mahyar Abbasian, Iman Azimi, Ramesh Jain, and Amir M. Rahmani. 2024. Chatdiet: Empowering personalized nutrition-oriented food recommender chatbots through an llm-augmented framework. *Preprint*, arXiv:2403.00781.

Xinli Yu, Zheng Chen, and Yanbin Lu. 2023. Harnessing LLMs for temporal data - a study on explainable financial time series forecasting. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 739–753, Singapore. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Yang Zhao, Tetsuya Nasukawa, Masayasu Muraoka, and Bishwaranjan Bhattacharjee. 2023. A simple yet strong domain-agnostic de-bias method for zero-shot sentiment classification. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3923–3931, Toronto, Canada. Association for Computational Linguistics.

Appendices

A Guide to LM Selection for Your Application Using this Work

Before coming to this paper, finalize other constraints of your solution - resource availability, data availability, system constraints, economic parameters, expectation of results, etc. These are outside the scope of this work, but will help in choosing LMs based on this work.

Then, check the relative performance of LMs for your task type/domain/reasoning type (or a combination). Find the closest available entity, and look up the performance of LMs of interest from Tables 4, 6, 5. From there, consider some options based on other constraints. For example, if you are planning to further align LMs on your task using any technique, choose from pre-trained models, if not, utilizing IT models will likely yeild better results. If you are bounded by resources, consider using smaller models that fit the requirements, or if you are bound by business/regulatory constraints, choose accordingly.

Next, look-up those LMs and entities in Figure 8–17 to find the prompt style that gives best results. This will be less important if you are planning to fine-tune your LM or use a more domain-adapted prompt. But if not, this will be beneficial. Decide if you can use the best prompt style, and if not, what is the performance trade-off with styles you can use. Finalize the feasible prompt style from here.

Based on these selections and other design constraints, implement your solution.

This work is accompanied by a GitHub repository linked in the first page of the paper as a utility which will allow evaluating any LM as per this framework and generating visualizations. It supports evaluation and generation of visualizations on other evaluation metrics that are discussed in Table 7, and on a different set of task types, application domain and reasoning types as needed with minor configuration changes. No code change will be needed for utilizing HuggingFace implemented models. Usage guidelines are available in the README of the repository.

B Aspect-level Analyses

In this appendix, we report results of all 14 LMs (5 pre-trained, 5 IT and 4 SOTA models that we compared our work to) on all entities of all three aspects present in the test set of the dataset. It includes the ones not covered in Section 2.3, but were available in the test-set of Super-Natural Instructions (Wang et al., 2022), with English as the input and output languages. Note, we also provide the results SOTA models for comparisons. Table 4 reports the results for all task types, Table 6 reports the results on all application domains and Table 5 for all reasoning types. Note, we abbreviate the model names at some places in the columns of these tables. The abbreviations and full model names can be found in Table 3.

In all our analyses, each domain has been considered independent, which is not always the case. There can be some tasks which can be classified into two aspects, like title generation for News articles will belong to title generation task type and News domain. However, in the dataset, there are many such pairwise aspects that do not contain any tasks, and for most of the ones that were present, Mistral-7B-I was the best model. Thus, we are not reporting the tabulated results for aspects considered pairwise considering the sparsity and repetitiveness of such a dense table. This is also discussed in Section 6.

Abbreviation	Model name
S-1.7B	SmolLM-1.7B
G-2B	Gemma-2B
G-7B	Gemma-7B
L-3-8B	Meta-Llama-3-8B
F-2-11B	Falcon-2-11B
S-1.7B-I	SmolLM-1.7B-I
G-2B-I	Gemma-2B-I
M-7B-I	Mistral-7B-I-v0.3
G-7B-I	Gemma-7B-I
L-3-8B-I	Meta Llama-3-8B-I
GPT-4o-m	GPT-4o-mini
GPT-4o	GPT-4o
DS-2	DeepSeek-v2
Gem-1.5-Pro	Gemini-1.5-Pro

Table 3: Abbreviation for model names.

Task Type	# Inst.	Pre-trained Models					Instruction-tuned Models					SOTA Models			
		S-1.7B	G-2B	G-7B	L-3-8B	F-2-11B	G-2B-I	S-1.7B-I	M-7B-I	G-7B-I	L-3-8B-I	GPT-4o-m	GPT-4o	DS-2	Gem-1.5-Pro
Answerability Classification	1300	84.42	**88.80**	88.76	87.50	88.29	91.87	90.70	**97.39**	86.21	85.98	97.82	98.87	91.78	90.93
Cause Effect Classification	700	84.09	86.65	82.16	82.30	**88.00**	88.03	87.79	**94.40**	85.56	85.73	95.96	96.59	91.58	90.12
Coreference Resolution	1400	83.03	**85.26**	75.20	78.76	84.70	88.00	84.68	**95.85**	84.08	83.27	95.88	97.99	93.12	90.25
Data to Text	826	85.25	87.19	82.01	86.71	**88.53**	87.72	88.30	**90.83**	86.14	85.87	91.55	89.62	91.73	89.22
Dialogue Act Recognition	700	82.87	86.16	84.61	83.70	**86.60**	88.04	88.42	**97.04**	84.28	83.66	97.74	97.70	91.00	91.16
Grammar Error Correction	100	90.75	**93.52**	86.47	89.76	87.45	93.72	94.29	**95.29**	89.25	87.84	95.67	95.79	94.51	94.13
Keyword Tagging	500	82.97	84.23	75.17	83.97	**86.03**	86.01	85.75	**88.52**	84.42	83.64	89.13	91.43	89.57	86.01
Overlap Extraction	200	84.91	85.03	84.03	85.27	**86.66**	86.34	85.82	**90.69**	85.40	83.83	90.18	92.16	90.04	88.18
Question Rewriting	1100	87.01	**90.05**	87.81	87.21	89.41	91.05	92.43	**93.17**	87.80	86.85	93.58	94.37	93.46	92.11
Textual Entailment	2400	82.83	**87.91**	82.52	83.96	86.61	88.43	86.93	**97.07**	84.76	84.41	97.00	99.04	89.44	88.15
Title Generation	1784	83.97	84.81	75.57	77.12	**86.61**	86.67	84.88	**89.43**	85.44	84.45	88.35	90.01	88.95	87.23
Word Analogy	800	**83.93**	82.67	83.35	83.22	81.15	88.35	83.63	**89.00**	85.17	81.53	93.37	97.37	88.44	89.37

Table 4: Mean BERTScore recall values for all task types for all models (Column abbreviations in Table 3, **BOLD** values represent best pre-trained and instruction-tuned models).

Reasoning Type	# Inst.	Pre-trained Models					Instruction-tuned Models					SOTA Models			
		S-1.7B	G-2B	G-7B	L-3-8B	F-2-11B	S-1.7B-I	G-2B-I	M-7B-I	G-7B-I	L-3-8B-I	GPT-4o-m	GPT-4o	DS-2	Gem-1.5-Pro
Abductive	200	85.91	87.58	86.86	86.08	**89.67**	90.51	88.40	**91.36**	87.54	87.43	91.12	92.11	91.08	90.71
Analogical	900	**83.92**	83.02	83.71	83.76	81.73	84.05	88.65	**89.15**	85.32	81.92	93.07	96.95	88.91	89.58
Causal	800	83.62	**87.00**	81.19	79.85	86.90	85.21	85.18	**93.35**	84.78	84.65	95.58	95.88	90.16	89.35
Commonsense	3000	83.80	**86.91**	83.75	82.26	85.61	87.31	88.96	**94.31**	85.07	83.89	95.65	97.82	91.35	89.60
Cross-document	200	83.82	**94.98**	88.45	80.34	92.52	92.88	92.94	**95.33**	84.15	84.73	96.24	95.53	89.10	88.06
Deductive	200	82.97	**91.14**	78.63	85.61	88.22	84.58	89.78	**92.70**	83.73	84.49	91.94	94.75	88.18	88.60
Discrete	200	83.82	**94.98**	88.45	80.34	92.52	92.88	92.94	**95.33**	84.15	84.73	96.23	95.53	89.10	88.06
Logical	100	80.87	**83.58**	77.46	80.16	79.78	82.16	86.13	**96.19**	81.19	80.50	97.26	99.12	88.09	87.42
Multihop	226	81.65	**94.98**	86.04	79.84	92.07	88.62	92.94	**94.11**	83.22	83.82	92.52	94.08	89.01	86.26
Numerical	200	83.82	**94.98**	88.45	80.34	92.52	92.88	92.94	**95.33**	84.15	84.73	96.24	95.53	89.10	88.06
Quantitative	300	83.38	**88.67**	83.60	78.10	86.59	84.11	84.59	**93.96**	84.80	84.46	94.26	95.19	89.15	87.64
Reasoning on Actions	300	84.96	85.03	82.07	86.80	**89.73**	85.94	87.18	**92.31**	86.03	85.12	89.95	92.78	92.48	89.37
Reasoning on Objects	100	88.44	90.79	86.00	91.66	**92.66**	92.44	91.81	**93.08**	87.16	87.68	93.95	93.64	93.12	93.25
Social Interactions	700	82.44	84.63	83.36	**86.33**	86.14	86.44	87.61	**93.91**	84.52	84.33	95.57	96.55	92.32	91.24
Relational	1100	83.79	83.31	**83.53**	84.22	82.43	83.94	87.98	**88.80**	85.08	82.36	92.40	95.56	88.97	89.57
Temporal	300	86.34	**94.32**	89.83	84.07	92.07	93.98	93.10	**96.40**	86.78	87.61	96.36	96.91	91.22	90.54
Textual Entailment	2400	83.35	**88.42**	82.27	83.87	87.41	87.45	88.59	**97.07**	85.20	85.01	96.64	98.43	89.48	88.71

Table 5: Mean BERTScore recall for all reasoning types for all models (Column abbreviations in Table 3, **BOLD** values represent best pre-trained and instruction-tuned models).

Domain	# Inst.	Pre-trained Models					Instruction-tuned Models					SOTA Models			
		S-1.7B	G-2B	G-7B	L-3-8B	F-2-11B	S-1.7B-I	G-2B-I	M-7B-I	G-7B-I	L-3-8B-I	GPT-4o-m	GPT-4o	DS-2	Gem-1.5-Pro
Anthropology	200	91.68	92.87	91.88	**93.18**	92.51	95.67	95.36	**97.22**	92.75	93.21	96.62	99.21	97.23	95.73
Books	300	84.92	**92.14**	86.18	89.65	87.28	91.42	94.19	**96.14**	86.23	85.14	96.30	96.33	91.09	89.12
Captions	700	83.94	**92.01**	87.52	86.04	86.05	90.18	93.16	**94.11**	85.93	84.90	98.08	98.58	89.54	89.62
Code	100	82.36	85.24	86.76	88.73	84.31	86.23	98.87	**99.10**	85.26	83.26	99.84	99.87	89.79	89.39
Commonsense	2500	84.33	**85.00**	80.50	81.26	84.29	85.82	88.77	**94.03**	85.54	83.60	95.86	97.82	91.51	90.84
Computer Science	100	81.80	84.76	82.50	80.65	**84.83**	84.21	87.58	**88.92**	85.56	84.23	86.99	89.30	89.12	85.16
Debatepedia	100	84.44	**91.13**	76.56	87.98	87.75	83.90	86.02	**99.18**	84.16	85.73	92.95	99.99	87.49	86.06
Dialogue	1900	82.97	**89.15**	84.72	86.33	88.17	88.49	89.21	**96.53**	84.47	84.24	96.33	98.00	90.56	89.25
Economics	100	83.31	82.17	82.92	80.23	**94.22**	84.46	87.85	**97.90**	84.88	86.14	84.54	99.91	99.41	88.57
English Exams	100	90.75	**93.52**	86.47	89.76	87.45	94.29	93.72	**95.29**	89.25	87.84	95.67	95.79	94.51	94.13
Fiction	700	86.53	**93.22**	90.11	93.25	**93.39**	94.50	93.53	**99.13**	88.06	87.86	97.15	99.66	92.54	89.63
Food	200	82.67	84.96	**91.03**	89.24	84.15	86.30	86.07	**93.16**	84.44	84.82	94.84	95.94	91.87	89.78
Formal logic	100	83.43	82.49	83.44	**89.21**	83.58	83.34	93.98	**99.84**	84.36	84.64	99.89	99.86	85.37	89.79
Government and Politics	800	84.78	90.65	86.48	91.09	**92.27**	90.46	90.52	**94.87**	86.56	85.86	93.74	96.06	90.30	88.34
History	800	87.98	**93.89**	90.68	93.66	**94.60**	95.58	93.36	**98.58**	89.66	89.89	96.67	99.51	93.08	91.22
Justice	200	91.68	92.87	91.88	**93.18**	92.51	95.67	95.36	**97.22**	92.75	93.21	96.63	99.21	97.23	95.73
Knowledge Base	100	85.91	**89.95**	84.38	85.13	87.31	91.83	90.15	**92.69**	87.73	86.26	91.96	92.10	92.23	92.17
Law	700	86.59	94.23	89.15	92.57	**94.85**	93.79	93.24	**97.77**	88.26	95.29	96.52	98.20	91.49	
Linguistics	100	82.36	85.24	86.76	88.73	84.31	86.23	98.87	**99.10**	85.26	83.26	99.84	99.87	89.79	89.39
Miscellaneous	800	84.23	**88.27**	85.86	82.09	85.67	85.66	87.06	**96.43**	85.44	84.83	96.64	97.52	91.03	89.71
Movies	100	84.71	**97.80**	94.69	91.53	91.76	**98.77**	95.14	95.86	83.95	87.34	99.97	99.97	95.89	90.49
Narrative	800	82.90	85.40	81.93	75.63	**85.82**	85.98	87.21	**91.35**	83.81	84.12	91.21	93.74	89.33	88.12
Natural Science	400	84.44	84.77	81.85	86.30	**91.05**	85.84	89.25	**95.15**	86.39	84.98	94.86	96.41	88.16	89.92
News	726	85.69	**88.13**	79.73	86.21	87.47	86.99	89.14	**90.97**	87.35	86.59	89.80	91.36	90.89	89.06
Nutrition	100	83.24	86.16	**94.79**	92.12	82.84	88.81	90.11	**96.79**	84.39	86.16	99.97	99.98	93.33	90.50
Professions	100	83.19	92.14	87.32	83.65	82.81	87.89	94.68	**94.87**	85.49	82.86	91.16	99.25	96.74	90.35
Public Places	300	86.32	**88.21**	83.44	87.58	87.28	88.08	88.96	**90.07**	86.43	85.63	91.35	94.92	90.18	90.94
Reviews	300	85.04	86.18	79.74	85.11	**87.55**	87.39	87.51	**89.07**	86.13	85.34	88.19	89.59	89.51	87.78
School Science Textbooks	200	91.68	92.87	91.88	**93.18**	92.51	95.67	95.36	**97.22**	92.75	93.21	96.63	99.21	97.23	95.73
Scientific Research Papers	400	82.41	84.26	81.28	80.06	**85.72**	86.13	**86.89**	86.84	85.22	83.94	89.29	89.88	89.77	85.34
Social Media	200	84.04	85.52	78.21	89.79	**91.51**	86.19	90.87	**96.03**	85.43	86.06	95.75	96.09	89.48	90.06
Sports	26	79.10	78.82	80.50	82.61	82.65	77.75	80.01	**89.41**	81.89	84.15	85.08	88.52	88.67	82.65
Statistics	26	79.10	78.82	80.50	82.61	82.65	77.75	80.01	**89.41**	81.89	84.15	85.08	88.52	88.67	82.65
Story	500	83.73	**89.49**	82.21	67.79	89.23	84.78	88.29	**92.66**	85.25	85.18	90.67	93.81	89.98	88.25
Web	400	86.67	87.18	83.53	82.49	**90.95**	89.79	90.17	**94.90**	86.65	86.78	95.22	97.31	91.62	90.46
Wikipedia	2184	84.60	86.74	80.90	84.41	**88.66**	88.75	89.03	**95.12**	85.72	85.78	94.46	96.03	92.27	90.45

Table 6: Mean BERTScore recall values for all domains for all models (Column abbreviations in Table 3, **BOLD** values represent best pre-trained and instruction-tuned models).

C LM-level Results

In Table 7, we report the best prompt style at the LM-level, abstracting all analyses at aspect-level with different performance metrics like ROUGE 1/2/L (Lin, 2004), METEOR (Banerjee and Lavie, 2005) and BERTScore P/R/F1 (Zhang et al., 2019) for reference.

Here we also include the results on Gemma-2-2B, Gemma-2-2B-I (Team et al., 2024c) and Mistral-7B-v0.3 (Jiang et al., 2023). From the results, it is visible why we ignored these models from main analysis. The Gemma-2 family is performing less compared to Gemma, and we wanted to keep a wide LM family for analysis. For Mistral, it was just underperforming. So, for brevity, we excluded them.

Among pre-trained models, Gemma-2B, the smallest of all models, gives best results. In IT models, Mistral-7B-I significantly outperforms others, despite its pre-trained version under-performing. This can be due of extensive fine-tuning of Mistral using several conversational datasets.

D Prompt Analyses

This appendix aims to analyze the performance of LMs on various prompts, offering an extension of the ideas discussed in the main paper.

D.1 Best Prompt Style at LM Level

We report BERTScore recall values for all prompt styles used in this work at Language Model level without going into the aspects in Table 8. These are scores on the entire experimental dataset.

From the table, we see that the differences with increasing examples are less prominent as compared to the aspect-level analyses of prompt style in Section 3.6 and Appendix D.2. This highlights the importance of conducting the prompt style analysis at aspect level. It is important to determine the prompt style that serves the best for a given use-case.

D.2 Variation of Performance with Different Prompt Styles for all Language Models

This is a continuation from Section 3.6 where we analyzed how performance of Mistral-7B-I varied for different task types, domains and reasoning types with the 8 different prompt styles that we use. In this section, we will provide similar visualizations for all other models. Using these graphs, one can determine the best prompt style for that particular task type, domain, or reasoning type. Additionally, the performance trade-off of using any other prompt style can also be analyzed. The visualizations are provided in Figures 8 – 17. From these, it is clear that for each LM, the variation in performance is different for each entity of task type, application domain and reasoning type. Therefore, the prompt style should be carefully selected by examining the trend.

D.3 Paraphrasing Definitions

In Section 2.2 and Section 3.7, we discussed about paraphrasing the task definitions. Here, we give more details around how we did the paraphrasing. We also reported results for only four LMs in the main paper, but here, we will provide the performance change for all LMs. We use the following prompt to paraphrase task definitions with GPT-3.5-Turbo (Brown et al., 2020; OpenAI, 2023) to generate paraphrases. Some paraphrases generated are given in Table 10.

```
You are an AI assistant designed to
paraphrase a definition of a task.
You will be provided with a paragraph
that defines a particular task to be
done. Your task is to paraphrase
the given definition so that it is
interpretable by another AI assistant
to fulfill the task. Make sure to
not omit any information from the
paragraph. It might be necessary to
complete the task. Only paraphrase
it.
{task_definition}
```

The mean BERTScore recall values of the performance of all the 10 models with actual and paraphrased definitions are given in Table 9. This will support the arguments in Section 3.7.

D.4 Adversarial Definitions

This experiment aims to identify how robust the LMs are when they are asked to complete a task instance with a task definition that has subtle differences capable confuse it, or are provided to elicit a response that is not desired. These subtle differences can both be intentional and non-intentional.

To perform this, similar to Appendix D.3, we generate adversarial task definitions for all the task definitions available in the dataset using `gpt-3.5-turbo` (Brown et al., 2020; OpenAI, 2023) using a pre-determined prompt which in-

Model	# Params	Def	Ex.	R-1	R-2	R-L	MET.	B-Score P/R/F1
SmolLM-1.7B	1.71B	✓	2	2.50	1.07	2.27	4.92	67.34/83.71/74.56
Gemma-2B	2.51B	✓	4	**22.04**	**7.88**	**21.23**	**18.12**	78.22/**86.41**/81.88
Gemma-2-2B	2.61B	✓	0	7.56	2.18	7.21	9.43	70.29/83.66/76.23
Mistral-7B	7.25B	✓	8	1.17	0.54	1.08	1.99	49.25/58.41/53.40
Gemma-7B	8.54B	✓	0	18.17	5.89	17.49	16.14	71.86/81.06/75.94
Llama-3-8B	8.03B	✓	0	16.38	5.35	15.30	14.96	75.52/82.73/78.80
Falcon-2-11B	11.1B	✓	8	16.88	6.46	16.01	16.45	79.65/86.18/82.72
SmolLM-1.7B-I	1.71B	✓	2	20.22	7.59	19.03	18.78	80.34/86.66/83.24
Gemma-2B-I	2.51B	✓	2	27.56	8.08	26.24	20.62	84.56/88.06/86.19
Gemma-2-2B-I	2.61B	×	0	3.45	1.35	2.99	5.60	73.91/82.76/78.05
Mistral-7B-I	7.25B	✓	8	**51.96**	**14.67**	**50.12**	**35.55**	91.29/**93.76**/92.39
Gemma-7B-I	8.54B	✓	0	8.64	3.23	7.96	12.57	78.18/85.14/81.48
Llama-3-8B-I	8.03B	×	8	4.68	2.19	4.23	8.31	74.23/84.33/78.89

Table 7: Mean Performance Metrics of Models with **# Params** parameters. **Def** (✓/×) indicates task definition presence. **Ex.** is the example count in the best prompt style. R-X (X=1,2,L) denotes Rouge scores, MET. is METEOR, and B-Score P/R/F1 represents BERTScore Precision, Recall, and F1 (**BOLD** indicates best results).

Model Name	With Definition				Without Definition			
	0	2	4	8	0	2	4	8
SmolLM-1.7B	83.33	83.71	83.66	83.69	82.68	83.30	83.28	83.30
Gemma-2B	84.69	86.15	86.41	86.34	82.13	81.79	81.14	81.17
Gemma-7B	81.06	68.29	67.87	68.10	65.72	72.90	71.67	71.48
Meta-Llama-3-8B	82.73	52.43	52.13	52.45	77.98	56.17	54.18	53.30
Falcon-2-11B	84.27	86.06	86.05	86.18	83.46	85.61	86.06	86.09
SmolLM-1.7B-I	84.61	86.66	86.55	86.44	83.34	86.45	85.80	85.84
Gemma-2B-I	87.79	88.06	87.96	88.05	84.70	86.03	86.24	86.28
Mistral-7B-I	88.29	93.04	93.75	93.76	83.82	88.88	90.20	90.28
Gemma-7B-I	85.14	84.71	84.76	84.82	83.58	83.96	84.08	84.05
Meta-Llama-3-8B-I	84.11	84.11	84.04	83.96	82.79	84.30	84.25	84.33

Table 8: Mean BERTScore recall values of all LMs with different prompt styles on the entire experimental dataset (2nd-level column denotes number of examples of the prompt style).

Model Name	Ex.	Def.	Par. Def.	% Dec
SmolLM-1.7B	2	83.71	83.17	0.54
Gemma-2B	4	86.410	85.771	0.74
Gemma-7B	0	81.055	80.998	0.07
Meta-Llama-3-8B	0	82.727	82.501	0.27
Falcon-2-11B	8	86.184	86.000	**0.21**
SmolLM-1.7B-I	0	86.66	86.26	0.46
Gemma-2B-I	4	87.959	87.671	0.33
Mistral-7B-I-v0.3	8	93.755	93.219	0.57
Gemma-7B-I	0	85.142	84.825	0.37
Meta-Llama-3-8B-I	0	84.112	84.217	**0.12**

Table 9: Mean BERTScore recall values of outputs with actual task definition (Def) and paraphrased definitions (Par. Def), along with percentage decrease in value (% Dec) when paraphrased definitions are used using 'Ex.' in-context examples for all models (**BOLD** values indicate least decrease in percentage in the two types of LMs).

Given Task Definition	Paraphrased Task Definition
The task is to write a full sentence or two using all of the information given. The sentence(s) will be a brief review of a restaurant. Use all of the information provided.	Create a concise statement that summarizes the overall experience and offering of a dining establishment, utilizing all the details provided.
In this task you are given a story and a question regarding that story. You must judge whether the question is answerable based on the info given to you. Label the instances as "Answerable" or "Not Answerable" based on your judgment. the story and the question are separated by a new line character.	Your task is to determine whether a given question, presented alongside a corresponding story, is answerable. You will need to classify each instance as either "Answerable" or "Not Answerable" based on the information provided. The story and the question are presented separately, with a new line character separating them.
In this task, you are given a context tweet, a question and corresponding answer of given question. Your task is to classify given passage into two categories: (1) "yes" if the given context is useful in answering the question, and (2) "no" if the given context is not useful.	Your job is to determine if a provided passage, including a context tweet, a question, and its answer, is helpful in answering the question by classifying it as either "yes" if the context is useful or "no" if it is not.

Table 10: Comparison between actual and paraphrased task definitions.

structs it to generate the adversarial definition. The prompt is given below, and some examples are given in Table 11.

```
Consider the following task description: {task_definition}.
Now, create an adversarial version of this task description. The adversarial task should appear legitimate but include subtle changes that could potentially mislead a language model or alter its output significantly. Focus on altering keywords, injecting ambiguities, or rephrasing the instructions in a way that shifts the expected response without being overtly misleading.
```

Then, we use the prompt style with definition and 0 examples, but replace the definition with the adversarial definition of the task. At last, we calculate the BERTScore recall values for adversarial versus actual task definition, and report the results in Table 12. The reason to choose 0 examples was to avoid the scenario of the model recovering by learning from in-context examples.

From the table, we see that most models are robust to adversarial change in task definition, with 7/10 models suffering less than 3% of decrease in performance. For the pre-trained model, Falcon-2-11B is most robust with only 0.613% decrease, and Gemma-2B ranks second with 0.88% decrease. For IT models, Gemma-2B-I is still one of the best, suffering only 1.2% decrease in BERTScore recall values only, but is outperformed by Llama-3-8B-I. Mistral-7B-I, the best performing IT model on true definitions is also not very sensitive to this change. Ph-3-mini-128k-I suffers the most amongst all models. We have seen sensitivity to be a general trend in this model with all varying parameters.

E Comparison of Decoding Techniques

In all experiments, we used greedy decoding. However, sometimes using top-k or top-p sampling (Holtzman et al., 2020) can offer better results. To analyze the impact of these sampling techniques, we generate and evaluate outputs with both these for each LM using the best instruction as per Table 7. The results are given in Table 13. We also tried different values of k and p, but did not find any meaningful change in the results.

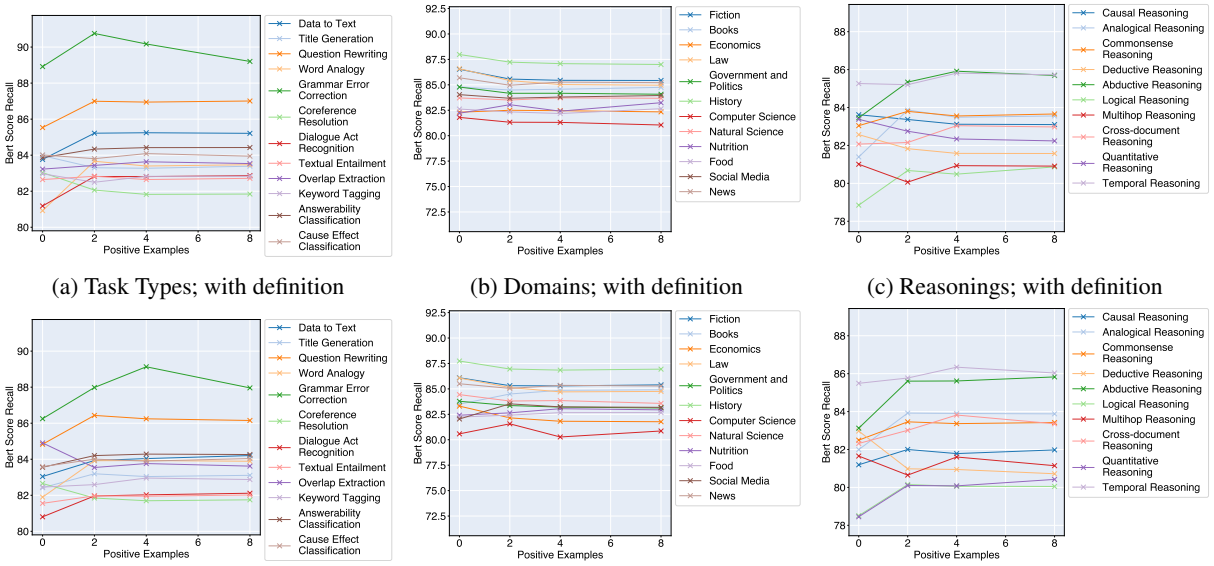
From the table, we can see that the performance doesn't change significantly at the LM level. In general, the performance of all LMs remains roundabout similar. We didn't observe a significant change in performance at aspect and entity level

Given Task Definition	Adversarial Task Definition	Difference
The task is to write a full sentence or two using all of the information given. The sentence(s) will be a brief review of a restaurant. Use all of the information provided.	Write a concise summary of a dining experience at a restaurant by incorporating all the details provided. Craft a couple of sentences that capture the essence of the establishment. Be sure to utilize all the data given effectively.	Dining experience is not always the complete review of the restaurant.
In this task you are given a story and a question regarding that story. You must judge whether the question is answerable based on the info given to you. Label the instances as "Answerable" or "Not Answerable" based on your judgment. the story and the question are separated by a new line character.	In this task, you will be provided with a narrative followed by an inquiry. Your task is to determine the question's answerability based on the given information. You are required to classify the statements as either "Answerable" or "Not Answerable" based on your assessment. Please note that paragraphs containing the narrative and question shall be separated by a newline character.	Narrative and inquiry are not same as story and question. Additionally, it is not specified that inquiry is related to the narrative.

Table 11: Comparison between actual and adversarial task definitions.

Model Name	Def.	Adv. Def.	% Dec.
SmolLM-1.7B	83.33	82.21	1.34
Gemma-2B	84.68	83.94	0.88
Gemma-7B	81.06	78.67	2.94
Llama-3-8B	82.73	78.01	5.70
Falcon-2-11B	84.27	83.75	**0.61**
SmolLM-1.7B-I	84.61	83.38	1.46
Gemma-2B-I	87.79	86.74	1.20
Mistral-7B-I	88.29	86.90	1.58
Gemma-7B-I	85.14	83.87	1.50
Llama-3-8B-I	84.11	83.57	**0.65**

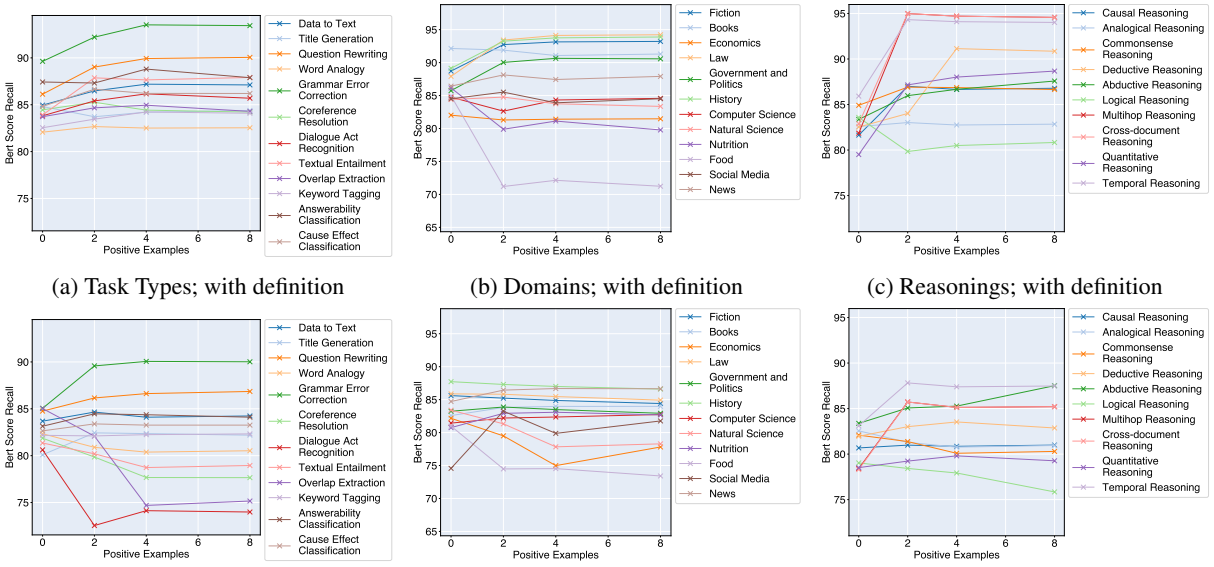
Table 12: Mean BERTScore recall values of outputs using actual task definition (Def.) versus adversarial definitions (Adv Def.) using 0 in-context examples for all models with percentage decrease (% Dec.) in performance with adversarial definitions (**BOLD** values indicate least decrease in percentage in the two types of LMs).



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

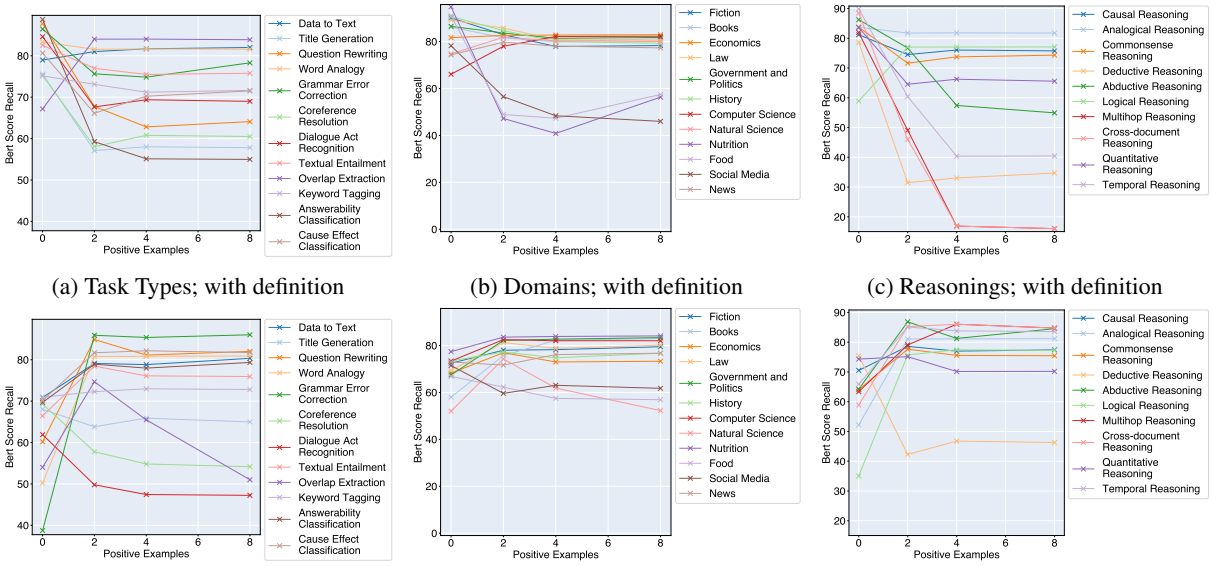
Figure 8: Mean BERTScore recall variation for **SmolLM-1.7B** across various task types, domains, and reasoning types by varying number of in-context examples, segmented by with and without using task definitions (Columns: different aspects, row 1: with task definition, row 2: without task definitions).



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

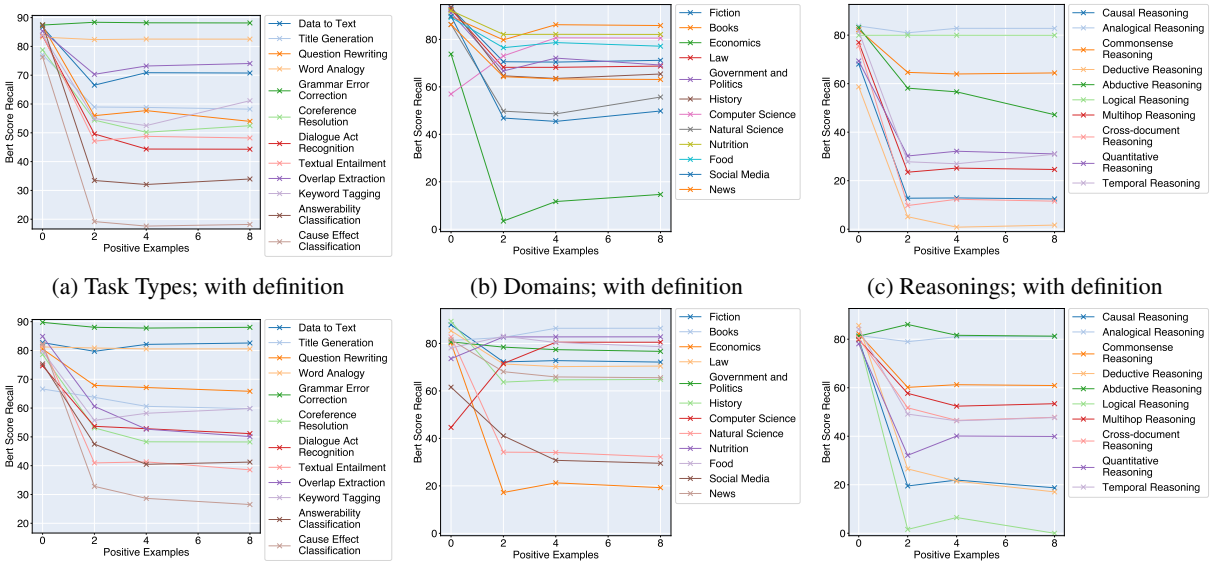
Figure 9: Mean BERTScore recall variation for **Gemma-2B** across various task types, domains, and reasoning types by varying number of in-context examples, segmented by with and without using task definitions (Columns: different aspects, row 1: with task definition, row 2: without task definitions).



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

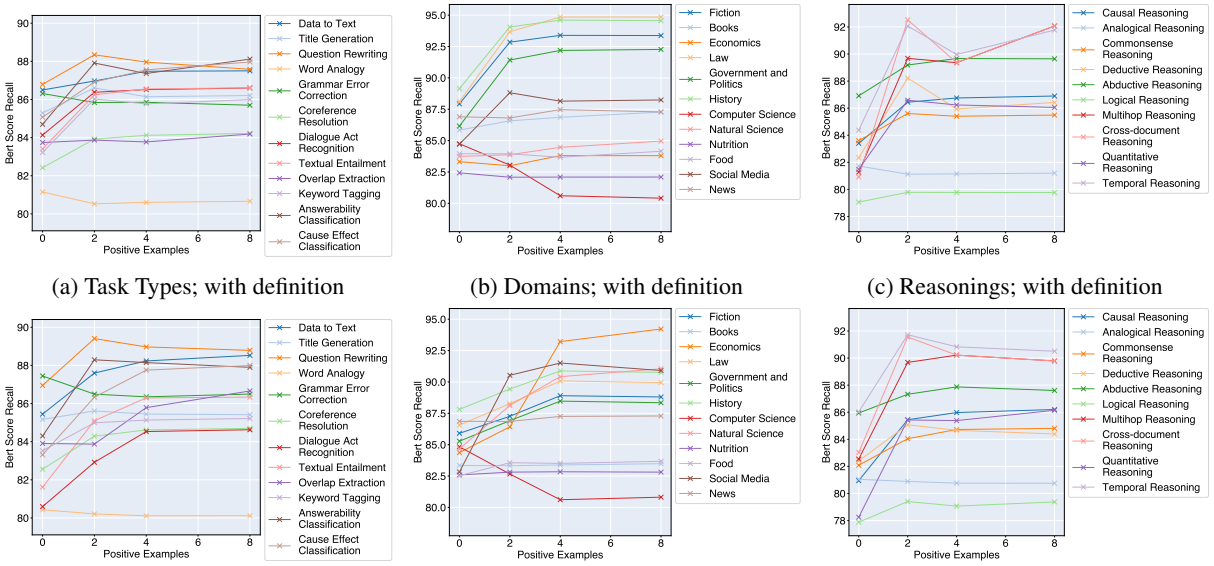
Figure 10: Mean BERTScore recall variation for **Gemma-7B** across various task types, domains, and reasoning types by varying number of in-context examples, segmented by with and without using task definitions (Columns: different aspects, row 1: with task definition, row 2: without task definitions).



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

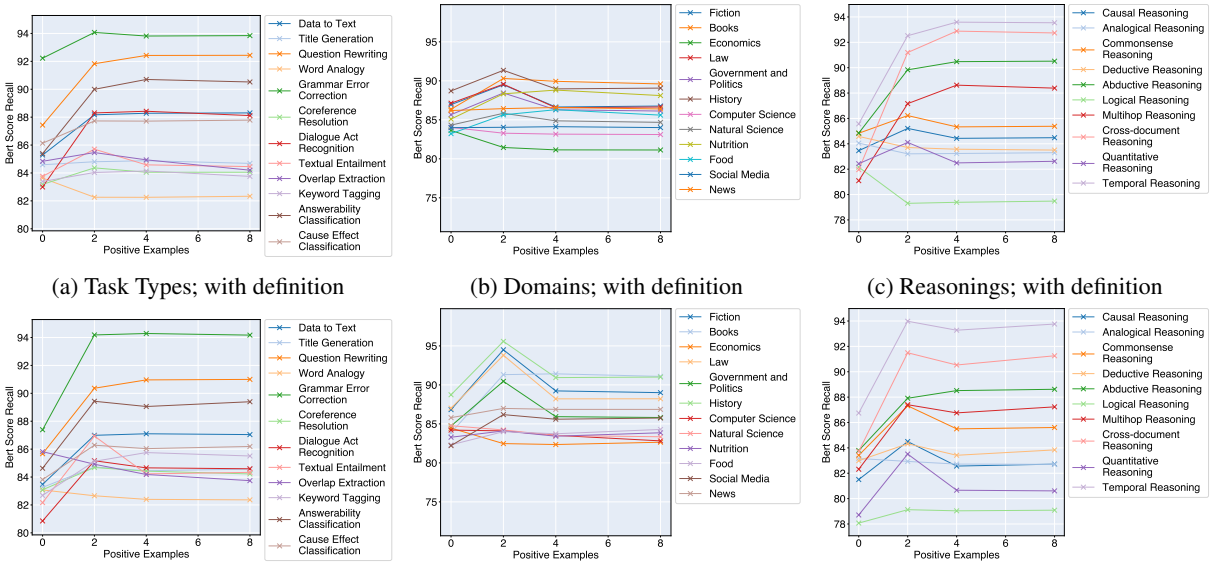
Figure 11: Mean BERTScore recall variation for **Llama-3-8B** across various task types, domains, and reasoning types by varying number of in-context examples, segmented by with and without using task definitions (Columns: different aspects, row 1: with task definition, row 2: without task definitions).



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

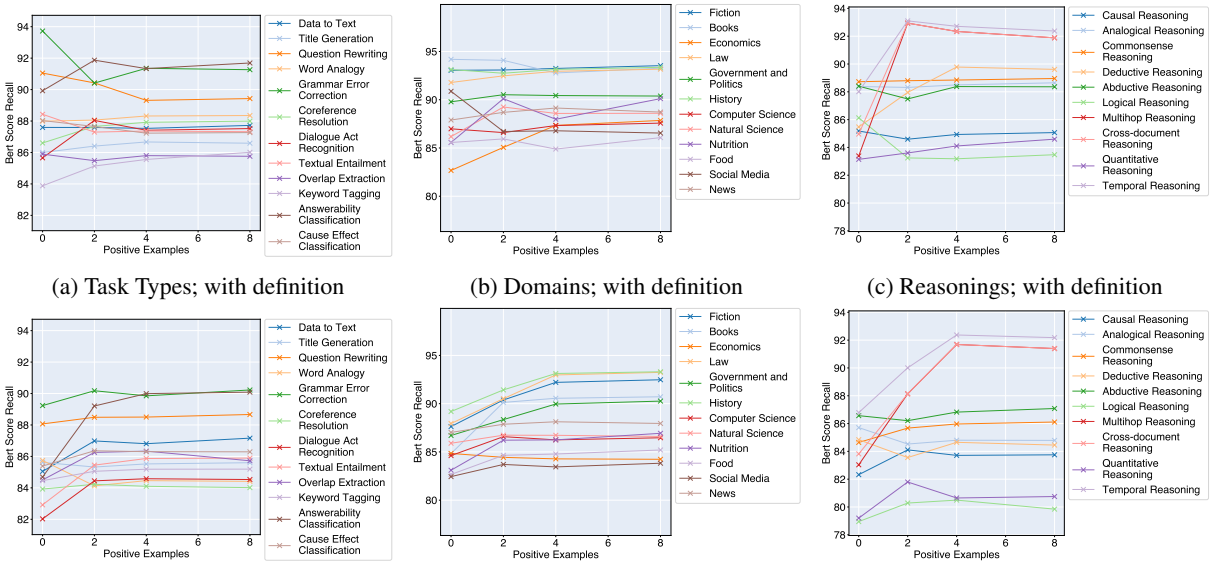
Figure 12: Mean BERTScore recall variation for **Falcon-2-11B** across various task types, domains, and reasoning types by varying number of in-context examples, segmented by with and without using task definitions (Columns: different aspects, row 1: with task definition, row 2: without task definitions).



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

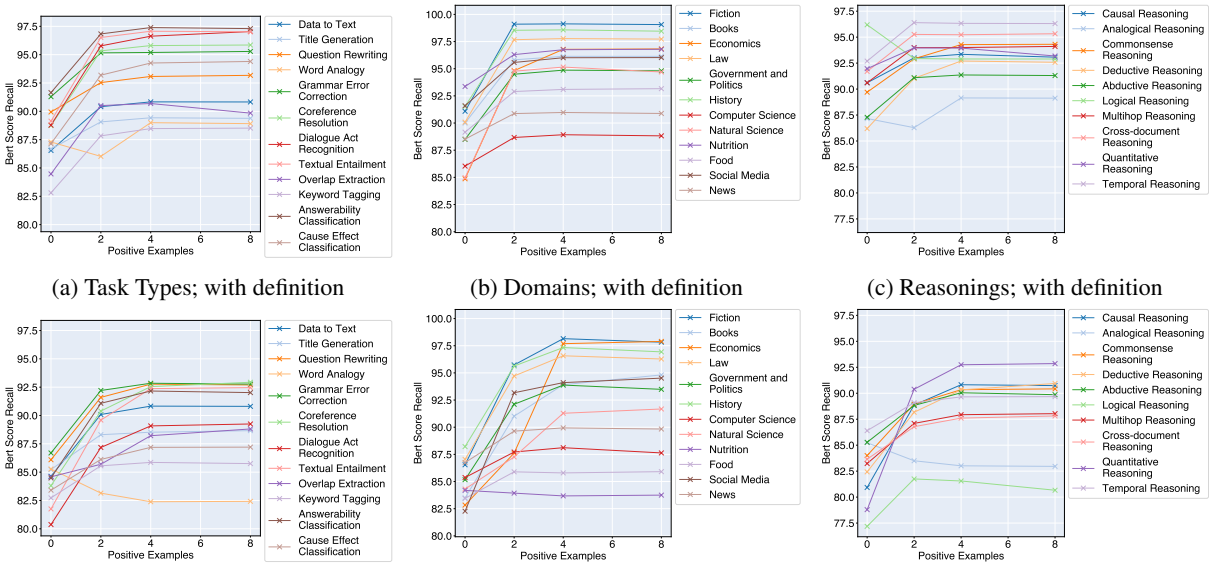
Figure 13: Mean BERTScore recall variation for **SmolLM-1.7B-I** across various task types, domains, and reasoning types by varying number of in-context examples, segmented by with and without using task definitions (Columns: different aspects, row 1: with task definition, row 2: without task definitions).



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

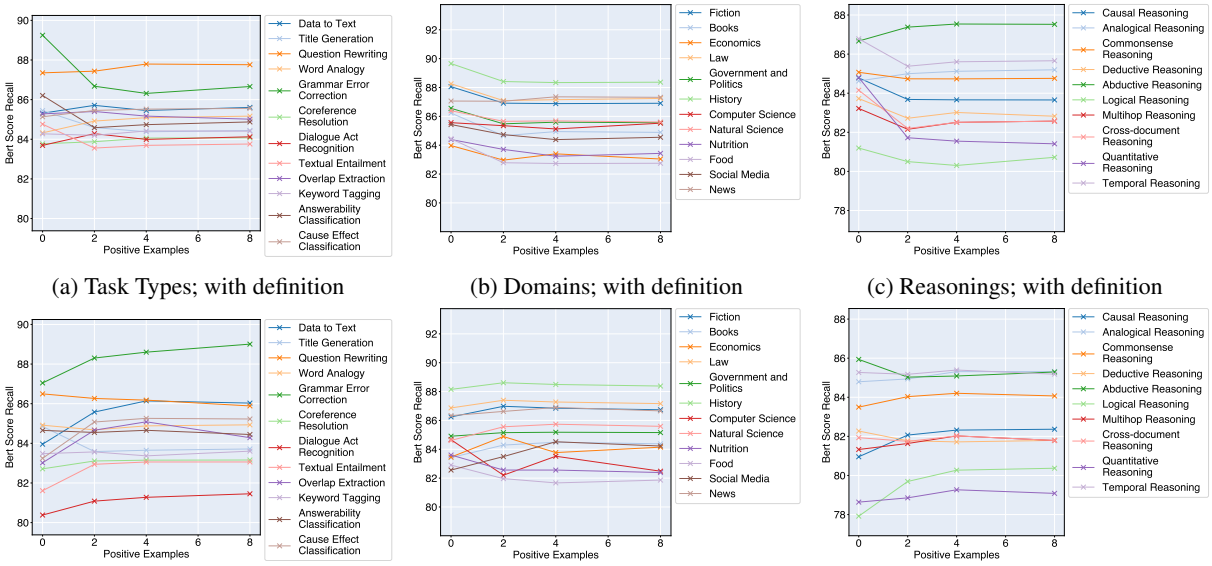
Figure 14: Mean BERTScore recall variation for **Gemma-2B-I** across various task types, domains, and reasoning types by varying number of in-context examples, segmented by with and without using task definitions (Columns: different aspects, row 1: with task definition, row 2: without task definitions).



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

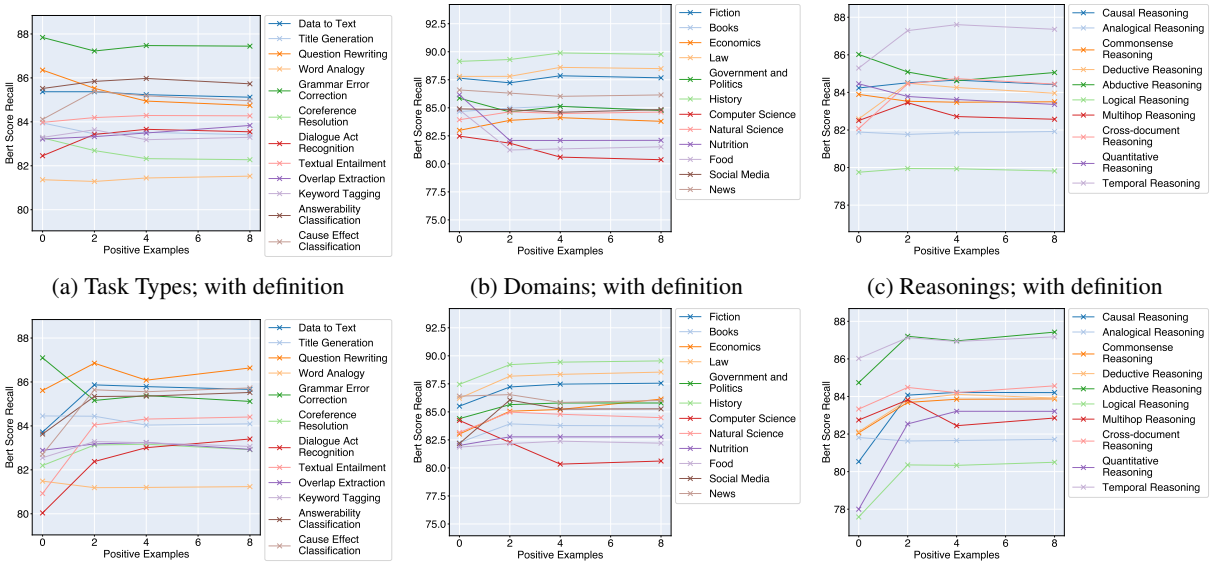
Figure 15: Mean BERTScore recall variation for **Mistral-7B-I** across various task types, domains, and reasoning types by varying number of in-context examples, segmented by with and without using task definitions (Columns: different aspects, row 1: with task definition, row 2: without task definitions).



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

Figure 16: Mean BERTScore recall variation for **Gemma-7B-I** across various task types, domains, and reasoning types by varying number of in-context examples, segmented by with and without using task definitions (Columns: different aspects, row 1: with task definition, row 2: without task definitions).



(a) Task Types; with definition (b) Domains; with definition (c) Reasonings; with definition

(d) Task Types; without definition (e) Domains; without definition (f) Reasonings; without definition

Figure 17: Mean BERTScore recall variation for **Llama-3-8B-I** across various task types, domains, and reasoning types by varying number of in-context examples, segmented by with and without using task definitions (Columns: different aspects, row 1: with task definition, row 2: without task definitions).

also. Given these factors, we preferred greedy decoding since it offers other advantages such as efficiency and reproducibility.

F Qualitative Analyses of Generated Outputs

In this appendix section, we will do some qualitative analyses of the generated outputs by Language Models.

F.1 Qualitative Examples for Mistral-7B-I

We have a wide number of varying parameters. Showing outputs for 10 LMs (+3 SOTA), 8 prompt styles, 12 task types, 12 domains, 10 reasoning types is not practically feasible. However, since the dataset is public and we are using openly available LMs, we think any desired output is fairly reproducible. We still show some of the qualitative examples in Table 14 for reference for Mistral-7B-I-v0.3 on the prompt style with 8 examples and added task definition. We have only included the task instance, and removed the full prompt for brevity.

We see that in general, the outputs of the model are aligned and can be used directly. This is probably expected since it has a BERTScore recall value of 93.76, and Rouge-L value of 35.55 with the gold-standard label. For classification tasks also, it is generating the response that is perfectly aligned. We still have tried to find and outline some cases where the output is not perfect. This highlights that the model is instruction-tuned on a wide variety of dataset and is very powerful to use directly.

F.2 Explaining Incorrect Responses of Pre-trained models

In Section 3.5 and Appendix B, we observed that even the best pre-trained models are not able to match the performance of IT models on SOTA models. While there was a theoretical reasoning, in this appendix, we will take some qualitative observation of outputs and focus on explainability of errors for the two best performing pre-trained models - Gemma-2B and Falcon-2-11B using the best prompt style.

Table 15 outlines four different types of common errors observed in Gemma-2B with the best prompt style (4 examples with definition). We can see that in the second and fourth example, the model is able to answer the question. But, in the second example, it is adding extra HTML tags. In the fourth, instead of answering yes and no, it is generating '100%'. Particularly, we found significant instances where outputs had extra HTML tags of `<strong>`, `<b>`, etc., despite the model getting 4 in-context examples to understand desired response. So, it can be inferred that Gemma-2B has a limitation of not being able to generate aligned responses learning from examples, and adding extra HTML tags to it. This is not observed for Gemma-2B-I; therefore, adapting the model for a specific application can eliminate such issues. In case 1, the model didn't generate any valid answer, and in case 3, it gave a wrong answer. In these scenarios, it is not certain if these issues can be resolved or are just limitations of the model's knowledge.

The generated outputs for Falcon-2-11B, as given in Table 16 was found to have other kinds of differences. First, no HTML tags were witnessed, which also confirms that it was specific to Gemma-2B. In Falcon-2, the outputs were often given as sentences, like Example 1 and Example 3 from the table. Example 1 has a correct answer, but it does not match the reference. However, while the output is misaligned, it is not wrong. For Example 3, the output is both misaligned and incorrect. There were several outputs that were like this. But, there were even more cases like the second example, where the model generated a sequence of steps for itself before giving the result, something like COT prompting (Wei et al., 2022b). The result was correct ultimately. This case can be easily handled by aligning the output, or post-processing it to extract desired text. We observed that ignoring these differences, the outputs of Falcon-2-11B were generally correct, making it a very powerful model if used appropriately. We couldn't compare it to the IT version, as it is not available yet.

G Implementation Details

We used a publicly available dataset Super Natural Instructions (Wang et al., 2022) for this work. It dataset is a meta-dataset created using multiple datasets. The paper reports its creation steps and multi-stage quality control process including automatic and manual processes, which were sufficient to eliminate the risks of personal or offensive content. We thoroughly went through the dataset paper, its collection process, and manually examined few samples of the dataset to verify this.

We use a single Nvidia A-40 GPU with 48 GB GPU memory to conduct all our experiments on a GPU cluster for each run. We define one run as a

Model Name	Greedy	top-k $k = 10$	top-p $p = 0.9$
SmolLM-1.7B	83.71	83.72	83.71
Gemma-2B	86.41	86.16	86.38
Gemma-7B	81.06	81.02	81.02
Meta-Llama-3-8B	82.73	83.12	83.05
Falcon-2-11B	86.18	86.12	86.12
SmolLM-1.7B	86.66	86.64	86.67
Gemma-2B-I	87.96	88.01	87.91
Mistral-7B-I-v0.3	88.29	88.29	88.29
Gemma-7B-I	85.14	85.15	85.15
Meta-Llama-3-8B-I	84.11	84.12	84.18

Table 13: Mean BERTScore recall values of outputs using greedy, top-k and top-p decoding with best instructions.

single forward pass on one model using a single prompt style. The batch sizes used are different and range from 2-8 for different models based on their sizes (2 for 11B model, 4 for 7B models, 8 for 2B and 3B models). Each run varied from approximately 80 minutes (for Gemma-2B-I) to approximately 60 hours (for Falcon-2-11B).

All model implementations are used from HuggingFace [3], except the SOTA models for which we use the OpenAI APIs [4]. We have used all artifacts as per their intended use. The implementations used and license details are provided in Table 17.

All inputs were tokenized using respective model tokenizers and left-padded to match the context size of each model, and 512 max new tokens were generated during inference by the model.

We perform all inferences with 4-bit quantized (Dettmers et al., 2023) versions of all models using Huggingface BitsAndBytes, along with Flash Attention 2 (Dao et al., 2022).

As discussed before, we are also sharing a GitHub repository of our implementation (link available on page 1 footnote) as a utility which will allow evaluating any LM using this dataset and generating these visualizations. It also supports doing this using other evaluation metrics discussed in Table 7 if required.

[3] https://huggingface.co

[4] https://openai.com/index/openai-api/

Task Type	Domain	R. Type	Instance	Candidate	Reference
Question Rewriting	Government and Politics	–	What did this president major in during their college studies? How did the stock market perform while this president was in office?	Did the stock market perform well while this president was in office, despite their college major being in economics?	Did this president major in economics or business?
Word Analogy	Common sense	Common sense, Analogical	cut : scissors. crack : ?	hammer	nutcracker
Grammar Error Correction	English Exams	–	She know just that what teacher has told her but no more .	She knows just what the teacher told her, but no more.	She know just what her teacher has told her but no more .
Overlap Extraction	Natural Science	–	Sentence1: satellites are used to take images from space. Sentence2: satellites are used to take Meteorologists.	satellites.	satellites
Title Generation (choosing best title here)	Narrative, Story	Deductive	Sentence 1: Crystal decided to learn how to sew. Sentence 2: She worked hard at cutting out the pattern. Sentence 3: Then she lined it up and sewed the edges up. Sentence 4: She had created a beautiful pillowcase! Sentence 5: Crystal decided to try a quilt next. Choices: a. The Sewing Machine. b. The Ram.	a	a

Table 14: Examples of generated responses (candidate) by **Mistral-7B-I** along with reference and task type, domain, reasoning type (R. Type). We have only included the instance (not the complete prompt) for brevity.

Task Type /Domain/ Reasoning Type	Task Instance	Candidate	Reference	Identified Reason
Coreference Resolution/ Commonsense/ Commonsense Reasoning	You must fill in the blank in a provided question with one of the two objects mentioned in the question. **...(truncated)** Input: Bob finished eating his zuchinnis but couldn't eat all his watermelons, because there was a small amount of the _ .	the answer is:	zuchinnis	not able to answer
Word Analogy/ Commonsense/ Relational	mainland : boat. portland : ?	car	car	output not aligned
Coreference Resolution/ Commonsense/ Commonsense Reasoning	Your job is to assess if a hypothesis sentence implies, contradicts, or is neutral with a given premise sentence. Respond with ""Contradiction"", ""Neutral"", or ""Entailment"" based on the relationship between the premise and hypothesis. Input: Premise: TORONTO, March 7 (Reuters) - The Canadian dollar weakened to a session low against the greenback after data showed the domestic economy unexpectedly shed jobs in February. **...(truncated)** <sep> Hypothesis: Toronto is the most populous city in Canada.	Neutral	Entailment	incorrect output
Answerability Classification/ Nutrition/ Reasoning on Social Interactions	Given a paragraph about cooking, and a set of conversational questions and answers about the paragraph, say whether the passage contains sufficient information to answer the follow-up question. **...(truncated)** Input: CONTEXT: Yes that's exactly what it means, apply salt to the steak. You shouldn't coat it, but you should apply salt very liberally. **...(truncated)** <sep> Q: What does it mean to 'salt' a steak prior to frying?	100%	YES	answering but differently

Table 15: Examples of some responses different from target by **Gemma-2B** using 4 examples with def. We have only included the instance and some description (not the complete prompt), and truncated the output for brevity.

Task Type /Domain/ Reasoning Type	Task Instance	Candidate	Reference	Identified Reason
Natural Science/Overlap Extraction	Generate an overlapping word between the given two sentences. When you find the overlapping words, they don't have to match exactly, **...(truncated)**. Input: Sentence1: a warm front causes cloudy and rainy weather. Sentence2: Cloudy, cool, or rainy weather can cause some algae scums to die.	The overlapping word between the given two sentences is "weather".	cloudy	produces a different but correct output, but is not aligned
Answerability Classification/ News	You are given a sentence and a question in the input. If the information provided in the sentence is enough to answer the question, label ""Yes"", otherwise label ""No"". Do not use any facts other than those provided in the sentence while labeling ""Yes"" or ""No"". There are only two types of valid responses: Yes and No. Sentence: They didn't scream if they saw a dog, cat or chicken. Question: What was Bobby's favorite thing?	To complete this task, I will analyze the given sentence and question to determine if the information provided in the sentence is enough to answer the question. Sentence: They didn't scream if they saw a dog, cat or chicken. Question: What was Bobby's favorite thing? Step 1: Identify the relevant information in the sentence. **...(truncated)** Output: No.	No	output not aligned
Coreference Resolution/ Commonsense/ Commonsense Reasoning	You need to answer a given question containing a blank (_). Your answer must be one of the two objects mentioned in the question, **...(truncated)** After the bike accident, the handle bar needed to be repaired, but the rim was fine, because the _ misshapen.	The handle bar needed to be repaired, but the rim was fine, because the _ misshapen. The blank is ""rim""."	handle bar	incorrect, not aligned

Table 16: Examples of some responses different from target by **Falcon-2-11B** using 8 examples with def. We have only included the instance and some description (not the complete prompt), and truncated the output for brevity.

Artifact	Implementation Link	License
Super Natural Instructions	Page (v2.8 used)	Apache 2.0 License
SmolLM-1.7B	Model Card	Apache 2.0 License
Gemma-2B	Model Card	Apache 2.0 License
Gemma-2-2B	Model Card	Apache 2.0 License
Mistral-7B-v0.3	Model Card	Apache 2.0 License
Gemma-7B	Model Card	Apache 2.0 License
Meta-Llama-3-8B	Model Card	Meta Llama-3 Community License
Falcon-2-11B	Model Card	Falcon 2 11B TII License
SmolLM-1.7B-I	Model Card	Apache 2.0 License
Gemma-2B-I	Model Card	Apache 2.0 License
Gemma-2-2B-I	Model Card	Apache 2.0 License
Mistral-7B-I-v0.3	Model Card	Apache 2.0 License
Gemma-7B-I	Model Card	Apache 2.0 License
Meta-Llama-3-8B-I	Model Card	Meta Llama-3 Community License
METEOR	Doc	Apache 2.0 License
ROUGE	Doc	Apache 2.0 License
BERTScore	Doc (using Roberta Large)	MIT License

Table 17: Details of artifacts used with implementation links and license details.