Paraphrase Types Elicit Prompt Engineering Capabilities

Jan Philip Wahle[*1]**, Terry Ruas**[1]**, Yang Xu**[2]**, Bela Gipp**[1]
[1]University of Göttingen, Germany
[2]University of Toronto, Canada
[*]wahle@uni-goettingen.de

Abstract

Much of the success of modern language models depends on finding a suitable prompt to instruct the model. Until now, it has been largely unknown how variations in the linguistic expression of prompts affect these models. This study systematically and empirically evaluates which linguistic features influence models through paraphrase types, i.e., different linguistic changes at particular positions. We measure behavioral changes for five models across 120 tasks and six families of paraphrases (i.e., morphology, syntax, lexicon, lexico-syntax, discourse, and others). We also control for other prompt engineering factors (e.g., prompt length, lexical diversity, and proximity to training data). Our results show a potential for language models to improve tasks when their prompts are adapted in specific paraphrase types (e.g., 6.7% median gain in Mixtral 8x7B; 5.5% in LLaMA 3 8B; cf. Figure 1). In particular, changes in morphology and lexicon, i.e., the vocabulary used, showed promise in improving prompts. These findings contribute to developing more robust language models capable of handling variability in linguistic expression. Code: https://github.com/jpwahle/emnlp24-prompt-paraphrase.

1 Introduction

It's not what you say it's how you say it.

— Albert Mehrabian

Large language models (LLMs) already mimic human interaction by receiving instructions through natural language prompts and responding in natural language (Radford et al., 2019; Ouyang et al., 2022; Touvron et al., 2023). The way prompts are designed has a marked impact on the value of a model's output, and current LLMs require some degree of prompt engineering to be successful (Liu et al., 2024; Lu et al., 2023; Leidinger et al., 2023). A key step in prompt engineering is understanding
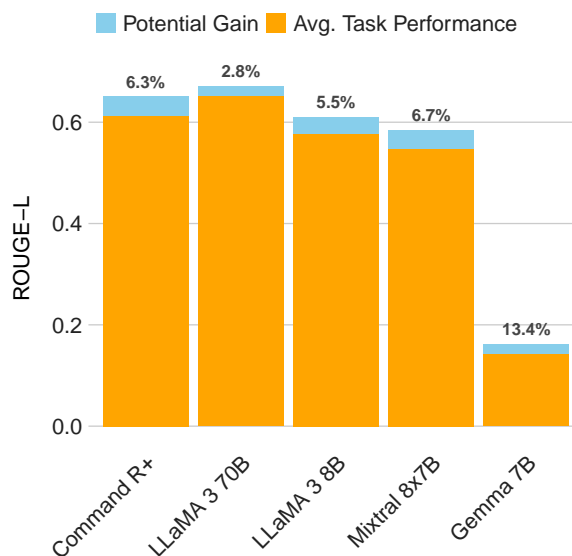

Figure 1: The potential median task performance gain (blue) over the model's baseline performance (orange) of five chat models across 120 tasks when their prompts were adjusted for specific paraphrase types (e.g., lexicon, syntax, morphology).

how humans can express similar meanings differently, also known as paraphrasing. For example, consider the following paraphrases of a prompt, which share no words and vary greatly in length but convey the same message:

Avoid procrastination.
Stop postponing what you have to do.

Humans understand and interpret the diversity in expressions, often without conscious effort. Arguably, LLMs should handle linguistic flexibility in a similar way to humans. Paraphrasing provides a window into the heart of prompt engineering; it gives us insight into what characteristics of the instructions language models value, what they understand, and where they lack capabilities. Ideally, LLMs should be robust to lexical, syntactic, morphological, inter alia, changes in the provided
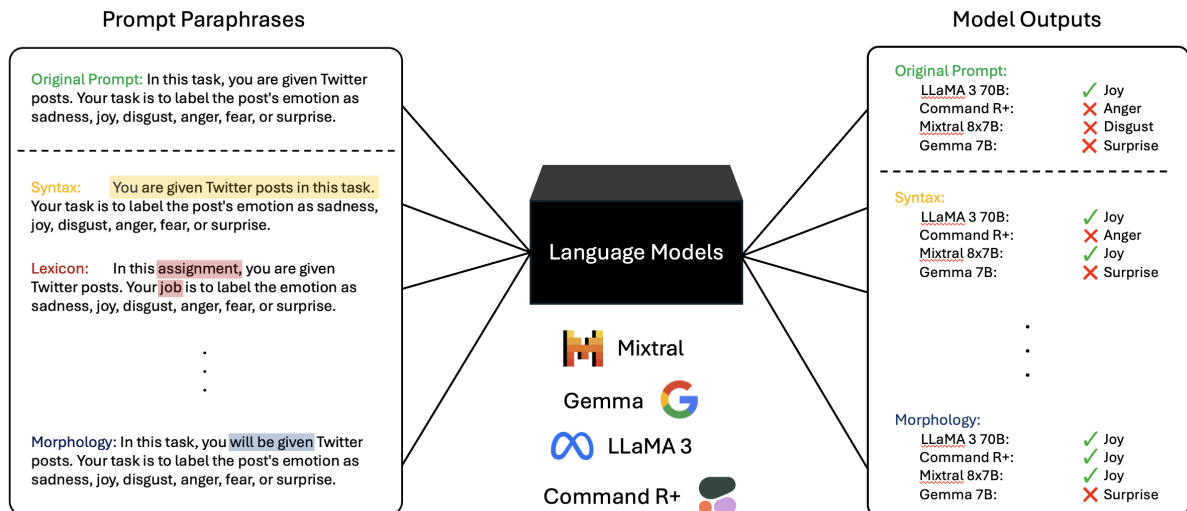
Figure 2: The main method of this paper. We paraphrase prompts of 120 tasks from 24 task families using 26 linguistic types of six categories (i.e., morphology, syntax, lexicon, lexico-syntax, discourse, and others) to analyze model inputs and outputs across different factors.

instruction — similar to how humans understand semantically identical texts written differently.

Through paraphrases, we can adjust prompts in these specific types to better understand which influence LLM's ability to solve tasks. Emerging tracks in conferences and workshops on how to design prompts underline a substantial interest in understanding which nuances of prompts affect models (Graux et al., 2024).

To determine the influence of linguistic types on prompts, we need a shared understanding of the variations between similar prompts. Vila et al. (2014) introduced *paraphrase types* which define linguistic changes at individual text positions. Wahle et al. (2023) adopted paraphrase types to train models that generate paraphrases with these capabilities. Paraphrase types can identify linguistic types, such as whether the lexicon has changed (by replacing words) or the syntax has changed (by altering grammatical structure).

So far, there has been little work to systematically evaluate how linguistic variations of prompts affect current language models. We fill this gap by evaluating hundreds of thousands of prompt variations across 120 tasks and five models (3.24m total prompt-example combinations). Our procedure is outlined in Figure 2. We measure how sensitive the models are to different prompt changes and what we need to improve their robustness and effectiveness in performing different tasks. Our work addresses the following key questions:

1. How sensitive are the models to specific paraphrase perturbations of the prompt?

2. Which linguistic changes in the prompt affect models the most?

3. What other factors play a role, such as the length and lexical diversity of the prompt or proximity to the training data?

4. How does the above evolve across models and tasks?

Our results show that adapting prompts to specific types (e.g., morphology) can yield significant gains in many models (cf. Figure 1) across different downstream tasks, such as summarization, text completion, or dialogue generation. We also control for various confounding factors, such as the length of prompts, lexical diversity, and proximity to training data, and show that performance gains can be observed independent of these additional factors. We recommend which paraphrase types to consider when adapting prompts for a particular task (e.g., polarity substitutions for sentiment analysis).

2   Related Work

Prompt-based learning has become a new trend, where pre-trained language models perform prediction using a template description of the intended task, and the model derives the necessary information without the need for gradient updates (Liu

et al., 2023). Prompt-based learning has advantages over traditional supervised learning, as the same pre-trained LLM can perform different unseen tasks without fine-tuning. Early prompt tuning methods focused on integrating trainable continuous prompt embeddings to perform various NLU tasks (Gu et al., 2022; Liu et al., 2021, 2024). Discrete prompts representing actual tokens have become more popular than continuous embeddings in prompt engineering, arguably because they are more accessible and linguistically interpretable.

Discrete methods used cloze-style phrases and differentiable prompts to improve few-shot learning capabilities (Schick and Schütze, 2021; Zhang et al., 2022). Shin et al. (2020) proposed Auto-Prompt, which uses a masked language model to find variations of prompts and has been used for various downstream tasks such as plagiarism detection (Wahle et al., 2022). Zhou et al. (2022) introduced Automatic Prompt Engineer, which searches over a pool of prompt candidates proposed by a second LLM. Other work focused on generating prompts for knowledge extraction from LLMs (Jiang et al., 2020) or investigated instruction induction and model tuning with minimal human intervention Honovich et al. (2022b,a). Recent work uses variations of Self-Instruct to improve model prompting (Wang et al., 2022a).

Reynolds and McDonell (2021); Hu and Levy (2023) evaluate the role of prompts in model control and the limitations of metalinguistic prompts for assessing linguistic knowledge in LLMs. Studies by Leidinger et al. (2023) and Lu et al. (2023) examined the influence of linguistic features of prompts on LLM performance with hand-selected prompt variations (e.g., tense, modality). Mizrahi et al. (2024) demonstrated through empirical analysis that LLM performance can vary based on how task instructions are phrased. Sorensen et al. (2022); Yang et al. (2023) explored the optimization of prompt selection using unified scoring frameworks and unsupervised techniques based on mutual information.

Our work contributes to previous work in several key aspects. While previous studies have explored hand-crafted linguistic features on LLM performance (Leidinger et al., 2023; Lu et al., 2023), we take a systematic approach by decomposing paraphrases into a set of six families of changes (i.e., syntax, lexicon, lexico-syntax, morphological, semantic, and others). Although other works have

proposed strategies for prompt selection (Yang et al., 2023; Sorensen et al., 2022) and methods for enhancing prompt tuning and generation (Liu et al., 2021; Shin et al., 2020; Zhou et al., 2022), our research quantitatively measures the effects of linguistic paraphrase types on model responses in a bottom-up method to apply successful types on different tasks. Other work focuses on domain-specific problems with few tasks. Our methodology spans five models, 120 tasks, and 24 task families. The main gap we address is the lack of empirical evaluation of how different linguistic manipulations of prompts affect LLM behavior and performance on a large scale. Our work controls for other confounding factors, such as prompt complexity, training data proximity, and lexical diversity.

3 Data & Models

Central to this study is a dataset of tasks, prompts, and variations of these prompts. We use the Super-NaturalInstructions dataset (Wang et al., 2022b), which contains more than 1 600 tasks with their respective prompts. We describe in detail how we construct different variations of these prompts in the experiments (Section 4). We sample 120 different tasks (each task has its own dataset) from 24 different task families (e.g., question answering, sentiment analysis) with the following conditions: the task family must have at least 10 tasks, the primary language is English, and each task has 200 examples to provide sufficient statistical power. We sample 5 tasks per domain, leaving us with 24 task families x 5 tasks per domain = 120 total datasets. See Appendix A for the task details.

We choose the five best non-proprietary models according to the LMSYS chatbot arena leaderboard in descending order of performance as of 1 May 2024[1]: LLaMA 3 Instruct (70B), Command R+ (104B), Mixtral 8x7B Instruct (47B), LLaMA 3 Instruct (8B), and Gemma Instruct (7B). Unless otherwise noted, we set the temperature to 0.2, the probability mass to sample words (top p) to 0.9, the penalty for repeatedly sampling the same sequence of tokens to 0.1, and the maximum number of tokens generated to the average of human references for that task. This number varies for different tasks but averages between 1 and 3154 tokens (see Appendix A for details). We use 40 NVIDIA A100 GPUs (40GB) with 16-bit precision (8 A100s for each model). Our experiments required a total

---

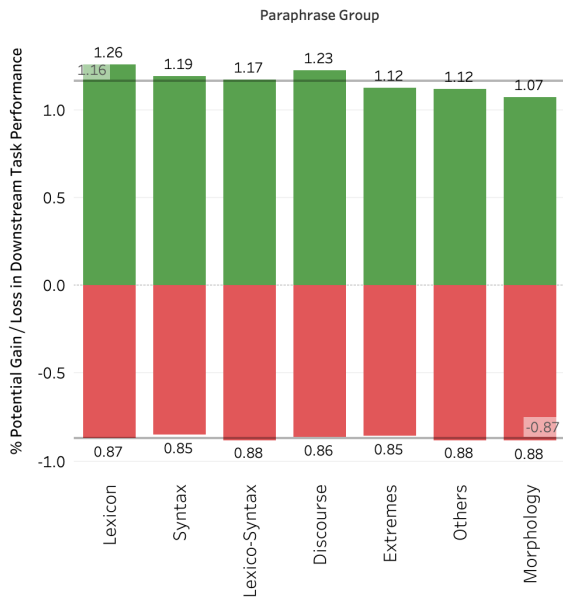[1] https://chat.lmsys.org/?leaderboard

Figure 3: The average downstream task performance gain or loss from applying specific paraphrase types to the prompt for all 120 tasks and five models.

computational budget of approximately 2880 A100 GPU hours, resulting in 311kg of CO_2 equivalent[2].

4 Experiments

Q1. *How do different linguistic changes in the prompt affect the performance of a language model? Which changes affect a model the most?*

Ans. We start with a seed prompt designed for a specific task, e.g., "In this task, you are given Twitter posts. Your task is to label the post's emotion as sadness, joy, disgust, anger, fear, or surprise." We paraphrase each prompt with one of 26 linguistic changes (e.g., syntax, lexicon) using the gpt-3.5-turbo-16k model, which we fine-tuned in the same way as Wahle et al. (2023) using the ETPC dataset (Kovatchev et al., 2018). See Table 9 and Figures 8 and 10 in Appendix A for an extensive list of types, full prompt template and examples respectively. For the original prompt and its 26 paraphrases, we test each model's few-shot settings. Depending on the task, the number of few-shot examples ranges from 3 to 6 (sum of positive and negative examples).

Results. As Figure 1 already revealed, we can observe marked changes when adjusting prompts. Command R+ experiences up to 6.3% median performance gain, and Gemma 7B up to 13.4%. We decompose these results by the different types in

Figure 3. Lexicon changes (+1.26%), closely followed by syntax changes (+1.19%), account for the largest median performance gain. The potential loss is comparable across changes. Overall, paraphrases had more upside than downside potential (+1.16% vs. -0.87% median change).

Discussion. Across all tested models, there seems to be marked upside potential when we adapt prompts in specific linguistic types. Morphology changes include changing modal verbs, which helps LLMs to follow instructions more clearly, e.g., "In this task, one ~~should~~ must detect the sentiment of the sentence." Lexicon changes have shown success across our experiments; in particular, we found examples in which more specific vocabulary benefits a task prompt, e.g., instead of "Determine how people feel about this text." a more precise version yielded better results: "Determine whether the sentiment expressed is positive, negative, or neutral",

Q2. *Are there prompt changes consistently improving performance on a particular set of tasks? Do these changes have a different impact on tasks in different domains?*

Ans. We decompose the results of Q1 into individual task families, i.e., a set of related problems, such as sentiment analysis or question answering. A task is a specific set of data and instructions for solving a problem in a task domain, e.g., classifying emotions on X (formerly known as Twitter).

Results. Figure 4 shows that title generation (+6.01%), text completion (+5.86%), and question answering (+5.60%) gain the most performance while having a low potential for a negative impact. Commonsense classification (-4.86%), sentiment analysis (-4.83%), and word semantics (-4.82%) have the highest loss potential.

Specific perturbations of the prompt affect the performance of models on tasks from varying domains differently. Morphology changes in the prompt have the largest gains in wrong candidate generation (+26.0%), question generation (+21.5%), and textual entailment (+17.5%). Lexical changes show consistent gains in summarization (10.8%), wrong candidate generation (+8.9%), and title generation (+7.1%). These tasks rely on semantic precision and vocabulary richness to interpret or generate nuanced responses. Positive effects with discourse changes can be observed in tasks that require an understanding of longer contexts or multi-sentence structures, such as summariza-

Text Completion 3.72 5.96
Title Generation 3.85 6.37
Question Generation 3.50 6.39
Answerability Classification 3.99 5.08
Question Rewriting 3.92 5.67
Sentence Composition 3.40 5.76
Summarization 3.37 6.29
Information Extraction 3.98 5.43
Dialogue Generation 3.47 5.47
Coreference Resolution 3.81 5.03
Named Entity Recognition 3.73 5.32
Text Categorization 3.68 4.07
Text Matching 4.24 4.70
Text to Code 3.66 4.88
Question Answering 2.93 4.26
Sentiment Analysis 3.96 4.43
Text Quality Evaluation 3.81 4.44
Commonsense Classification 3.85 4.17
Program Execution 2.78 5.53
Textual Entailment 3.00 3.65
Question Understanding 2.56 3.78
Word Semantics 2.47 3.17
Fill in The Blank 1.50 2.35
Wrong Candidate Generation 1.34 1.21

-5 -4 -3 -2 -1 0 1 2 3 4 5 6 7
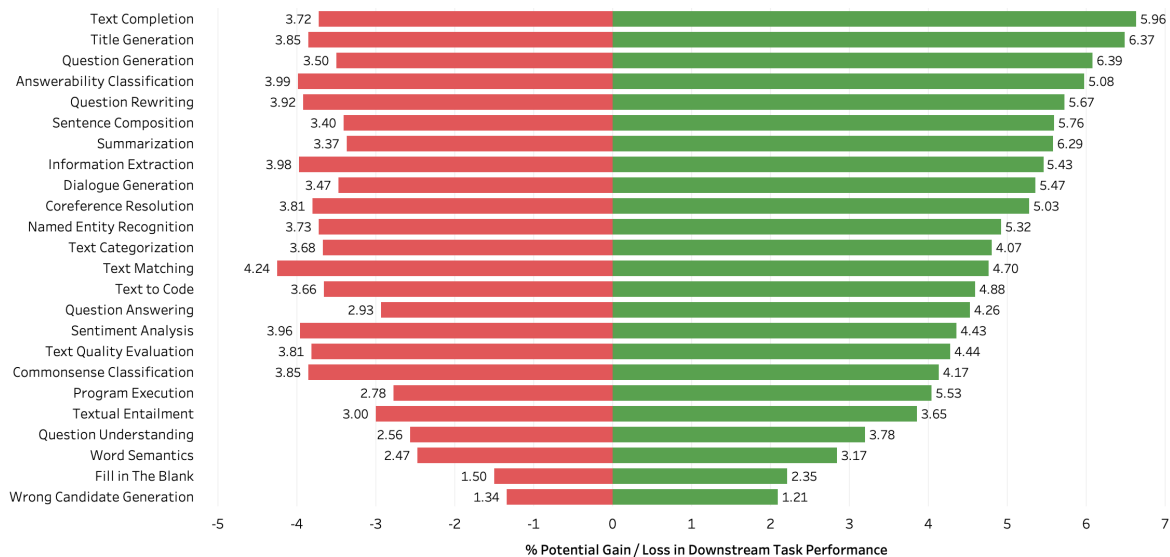
% Potential Gain / Loss in Downstream Task Performance

Figure 4: The avg. gain or loss in performance for all 120 tasks in 24 different task families across all five models.

tion (+7.8%). Detailed results for each model are available in Tables 10 to 15 in Appendix C.

Discussion. Some tasks are more sensitive to instructions than others. Tasks such as question generation and text completion may benefit from syntax changes because they can improve grammatical accuracy while reducing the flexibility of more creative tasks such as dialog generation. In particular, sentiment classification seems to benefit from polarity substitutions and is sensitive to negation, consistent with related work (Wiegand et al., 2010). Lexical changes improve performance in vocabulary-intensive tasks such as named entity recognition and text entailment. However, they may hinder performance in more general contexts such as question rewriting, which is often based on data from Wikipedia and knowledge bases (see Table 5 in Appendix A). In discourse, better context management, as well as indirect and direct style, improves clarity and coherence in tasks like summarization and long-form answer generation.

Sometimes, even the same changes can produce different results for different models. For example, larger models such as Command R+ benefit from morphology changes for summarization (+23.6%), while the same changes result in a loss of performance in the two small models LLaMA 3 8B (-16.4%) and Gemma 7B (-16.1%).

Q3. *How sensitive are different models and architectures to changes in the prompt? Are there differences between models of different sizes? Can a smaller model achieve better results than a larger model by paraphrasing its prompts?*

Ans. Related work has shown that model size and training scale play a central role in language model success, also known as "scaling laws" (Kaplan et al., 2020; Rae et al., 2021; Wei et al., 2022). What has not been shown so far is whether and how much lexical changes influence different model architectures and sizes when adapting prompts. Specifically, we investigate how to bring the performance of a weaker model up to the level of a larger model by tweaking its prompts.

Results. In Figure 1, note how the smaller 8B version of LLaMA 3 can gain up to 5.5% median task performance, while the larger 70B model can gain much less with up to 2.8%. LLaMA 3 8B has a lower baseline performance of 0.58, while the 70B model's baseline is at 0.65 (see Figure 12 in Appendix C for more details).

LLaMA 3 models are less sensitive to changes than other models (0.05 avg. std. in LLaMA models vs. 0.10 avg. std. over other models). Smaller models are more sensitive and have the highest possible gain potential when adjusting prompts (e.g., Gemma 7B: +10.2% gain through lexicon; LLaMA 8B +8.2% gain through morphology). The most sensitive model in our experiments is Command R+ (avg. std. 0.16).

Although Command R+ achieves an average score of 0.61, which is lower than LLaMA 3 70B's with 0.65, it can outperform LLaMA 3 70B by 0.08 when its prompts are tuned. The same applies to Mixtral 8x7B (0.55) and LLaMA 3 8B (0.58). Mixtral's prompts, when tuned, can score 0.06 higher than the previously better LLaMA model.

If we always choose the best (paraphrased) prompts, we can achieve even higher performance gains; for example, LLaMA 3 8B could gain 21.1%, making the model markedly better than its 70B counterpart by tweaking the prompts. We want to note that it is difficult to always find the best prompt. Therefore, we report median performance gains in the main body of the paper and report results for selecting the best prompt in Figure 9 in Appendix C.

Discussion. Models have different architectures and training processes, especially training data, which play a fundamental role in their behavior. It is encouraging to see that there is still upside potential to improve task results without relying on more computational resources for training. However, this raises the question of why such different behaviors can be achieved by changing the instructions. An interesting parallel is how humans can also sometimes produce different results depending on the instructions given. Our results suggest that smaller models can perform similarly to larger ones and are more sensitive to paraphrase changes.

Q4. *Do paraphrased prompts that increase a model's task performance also show greater similarity to the model's training data?*

Ans. We know that a confounding factor in prompt engineering is that prompts that are closer to a model's training data improve their likelihood of answering more confidently (Zhao et al., 2021). We use the FineWeb corpus with 350 billion tokens ($\approx$388GB) to search for examples close to paraphrased prompts and compute their similarity.

We build a BM25 index ($\approx$610GB), and for each prompt and its 26 variations, we query this index to find the closest examples. We measure the difference in similarity of the original prompt to the training data versus our paraphrased prompts to the training data by computing the following measure:

$$\Delta_{train} = RL(P, T_P) - RL(O, T_O) \qquad (1)$$

where RL is the ROUGE-L score, O is the original prompt and T_O is the nearest training example for O (as measured by the BM25 score), P is the paraphrased prompt, and T_P is the closest training example for P.

Results. The x-axis of Figure 5 shows Δ_{train}, which represents this difference in similarity to the training data between the paraphrased and original prompts. The y-axis then shows downstream

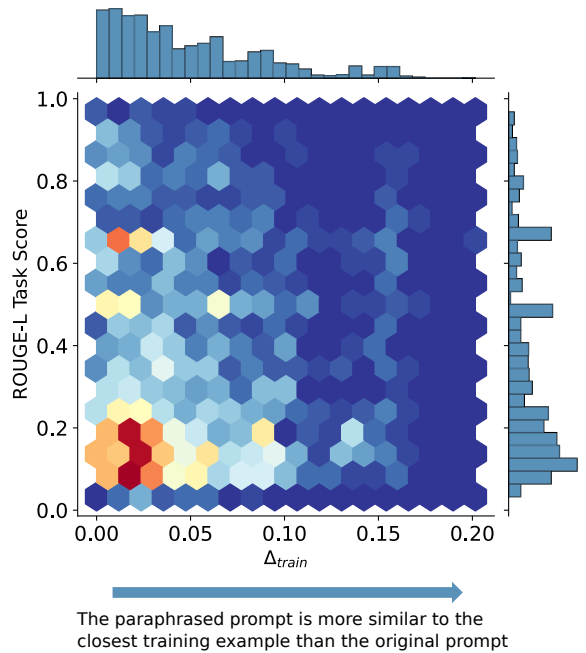The paraphrased prompt is more similar to the closest training example than the original prompt

Figure 5: The distribution of how much closer the paraphrased prompt is to the closest training example in FineWeb 350BT over the original prompt (x-axis) and the distribution of task performance (y-axis). Red colors mean high mass and blue colors mean low mass between Δ_{train} and task performance.

task performance. The results show few very close training examples for our tasks, i.e., the ROUGE-L between prompt and closest FineWeb example is typically below 0.5. Most successful paraphrased prompts with a downstream task performance greater than 0.8 do not show higher similarity to training than the original prompt. Sometimes, higher scores can be achieved when paraphrases are closer to the training (top right of the figure).

Discussion. Consistent with related work (Zhao et al., 2021), we show that prompts closer to training examples sometimes have higher accuracy than those without close training examples. However, this does not generally hold across our large set of tasks. We found no evidence that paraphrases were better because they had closer training examples.

Q5. *How do different prompt perturbations affect the lexical richness of language model responses in generative tasks?*

Ans. We have previously measured downstream performance as a key indicator of prompt quality. In classification tasks, this is directly measurable by the binary downstream metric (i.e., "yes" or "no"). In generative tasks, it is not easy to say whether a generative answer is correct using a human reference (Clark et al., 2021); in particular, an output
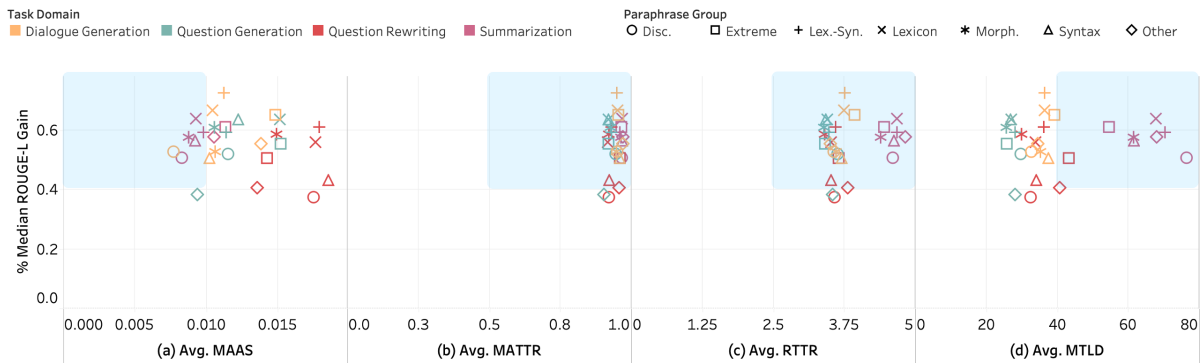
Figure 6: The percentage in task performance gain and lexical diversity as measured by four metrics in four tasks. The light blue quadrants show areas of high lexical diversity and strong task performance gains.

may be correct even without having any words in common with a reference (recall our example from the introduction). Therefore, we can evaluate complementary factors about responses (e.g., whether the response is creative or uses lexically diverse language). A key factor in complex responses in natural text is lexical richness (Van Hout and Vermeer, 2007; Jarvis, 2013; Kyle, 2019).

We measure lexical richness with four metrics that focus on the diversity of tokens and text segments (Guiraud, 1954; Mass, 1972; Covington and McFall, 2010; McCarthy and Jarvis, 2010): Root Type-Token Ratio (RTTR), Maas, Moving Average Type-Token Ratio (MATTR), and Measure of Lexical Diversity (MTLD) — more details in Appendix B. We select four task families with long responses to measure lexical richness: summarization, question generation, question rewriting, and dialogue generation (20 tasks in total).

Results. Summarization tasks have a particularly high diversity in responses as measured by Maas, RTTR, and MTLD (Figure 6 (a), (c), (d)). Overall, lexicon and lexico-syntactic changes often lead to higher performance at the expense of lexical richness. Models produce responses with lower lexical diversity in question rewriting tasks as measured by MAAS (Figure 6 (a)). Syntax and other changes lead to both lower lexical diversity and task performance gains compared to others measured by RTTR (Figure 6 (c)). Question rewriting achieves the highest lexical diversity as measured by MAAS, specifically with lexicon and lexico-syntactic changes (Figure 6 (a)), and question generation and question rewriting have overall lower diversity and performance gains, specifically with discourse and other changes (Figure 6 (c)).

Discussion. Unexpectedly, summarization and

question rewriting, which are arguably less open-ended and based more on extracting or paraphrasing information, lead to high lexical diversity. Purely generative and more open-ended tasks such as question generation and dialogue generation yield lower lexical diversity scores. Larger changes in the prompt markedly affect the lexical diversity of model responses. This is not surprising, as asking people to perform tasks in very different ways often leads to more diverse responses. What is surprising, however, is that marked performance gains can still be observed. Since these changes lead to substantial variance in the language model output, our results suggest that the complexity of the prompt (e.g., its length, word positions, or lexical shifts) can also be a confounding factor to task performance.

Q6. *What role does prompt complexity play in task performance outcomes? To what extent do variations in length, word position, and lexical variation correlate with downstream task performance?*

Ans. We use three markers to describe the complexity of a prompt paraphrase relative to the original instruction: the deviation in absolute number of tokens, word position deviation, and lexical deviation (Liu and Soh, 2022). We use Pearson correlation to measure a quantitative relationship between these three markers of prompt complexity and task performance. More details about the metrics and correlations can be found in Appendix B.

Results. Across tasks, there is no significant correlation between changes in prompt length, changes in word position, and lexical deviation, as all correlations are close to zero (see Table 2 in the Appendix C). We found no evidence that making prompts more or less complex is associated with higher or lower task performance.
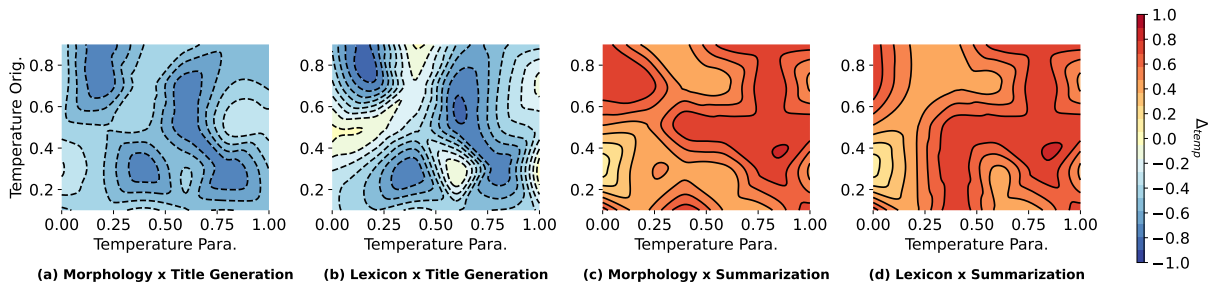
Figure 7: The performance difference between original and paraphrased prompts for temperatures from zero to one.

Discussion. Note that when paraphrasing prompts in certain lexical types, it is not new complexity added to the prompt that leads to different results (both gains and losses in task performance). This suggests that the actual linguistic perturbations make the prompt easier for the model to understand, leading to higher scores. However, there may be other factors that can also play a role.

Q7. *How much randomness is involved in these changes? Can the models' temperature lead to particularly good or bad task results that partially explain these gains or losses?*

Ans. When prompting models to perform tasks, we choose 11 temperature variations for LLaMA 3 70B from zero to one with steps of 0.1. For title generation and summarization, we run the original prompt and its paraphrased version using morphology and lexicon changes with these different temperature variations. We do this 11 times, with each run having a different temperature for the same paraphrased prompt, and then compute the difference between task performance of the original and paraphrased versions as:

$$\Delta_{temp} = RL(LM(O_i)) - RL(LM(P_j)) \quad (2)$$

where RL is the ROUGE-L score, O_i is the original prompt with temperature i, P is the paraphrased prompt with temperature j, and LM is the language model output.

Results. Figure 7 shows contour plots of different temperature choices. The coloring shows Δ_{temp} for different temperatures for the paraphrased (x-axis) and original (y-axis) prompts. Note how the distributions within the summarization have similar hills and valleys (see Figure 7 (c) and (d)). For title generation, with generally higher performance loss, the highest gains are achieved for low to medium temperatures (up to 0.6 for the paraphrased prompt). For summarization, with overall higher scores, low

temperature yields the highest performance difference, close to 0 for the paraphrased prompts and between 0.2 and 0.4 for the original.

Discussion. While we cannot fully exclude that randomness might improve results (i.e., there are some hills even in the high temperature ranges), most of the gains stem from low temperatures, suggesting that significant performance gains are due to paraphrase types. It may also be worth mentioning that additional randomness can come from the models trained by (Wahle et al., 2023), as they can only paraphrase with some degree of accuracy.

5 Conclusion

This study evaluated five language models across 120 tasks (including sentiment, question answering, commonsense reasoning, summarization, etc.) and showed that paraphrasing prompts can improve the performance of language models. We showed that language models have a marked upside potential to improve task results when their prompts are adapted in specific linguistic types (e.g., polarity substitutions for sentiment analysis). Other work can use our findings about which tasks benefit from which paraphrase types to design new prompts. We also controlled for prompt complexity, temperature, and proximity to training data.

Current model performance represents a lower performance bound for tasks as we showed that semantically identical instructions hold marked upside gain. While it is not entirely clear why language models are often sensitive to changes in instruction, we have systematically tested different lexical features and found that some have a larger positive impact (e.g., morphology) than others (e.g., syntax), depending on the tasks. Since humans understand tasks presented in different ways and are robust to small (or even complex) changes in instruction, language models should have a similarly robust interface to communication in the future.

We have contributed to this development of robust language interfaces by showing how specific types can benefit or harm models over a large set of tasks and prompts. One can also use our approach to create prompts the model does not understand to augment training data and increase its robustness.

Limitations

Our results show that the same paraphrase changes can potentially improve results for one model but harm another. The same is true for different changes in the same task. In a task like sentiment, sometimes the same polarity substitutions lead to improvements, and sometimes they do not. Variance across examples and models does play a role here but another possible reason is that models trained to generate paraphrase types only have a certain accuracy, leading to models sometimes confusing one type for another or generating an incorrect type as recent work shows (Meier et al., 2025). However, at a large aggregate level (3.24 million prompt-example combinations), our results provide important trends about prompts despite these margins of error. Better models for the controlled paraphrase generation will provide more accurate results in the future.

Our selection of paraphrase types is not a complete set of all flexibility in linguistic expression. There may be other variations, especially extremes, that we have not considered. However, the set of paraphrases across morphology, lexicon, lexico-syntax, syntax, and discourse covers many of the most common paraphrases encountered in texts. We also find that there is no single paraphrase type that improves a model's accuracy across tasks consistently. This is somewhat to be expected, as different tasks may benefit from different adaptations even though the overall domain is the same. The same is true for tasks in different domains; some modifications are successful in one domain but not in another. For example, polarity and negation play a larger role in sentiment, while lexical changes affect vocabulary intensive tasks (e.g., named entity recognition) or tasks that require specificity in vocabulary (e.g., example). Finding the most successful type of change in a given setting is non-trivial, and more research needs to be done on successfully perturbed prompts for new and unseen tasks.

References

Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A. Smith. 2021. All that's 'human' is not gold: Evaluating human evaluation of generated text. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7282–7296, Online. Association for Computational Linguistics.

Michael A Covington and Joe D McFall. 2010. Cutting the gordian knot: The moving-average type–token ratio (mattr). *Journal of quantitative linguistics*, 17(2):94–100.

Damien Graux, Sébastien Montella, Hajira Jabeen, Claire Gardent, and Jeff Z. Pan. 2024. [prompteng] first international workshop on prompt engineering for pre-trained language models. In *Companion Proceedings of the ACM on Web Conference 2024*, WWW '24, page 1311–1312, New York, NY, USA. Association for Computing Machinery.

Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2022. PPT: Pre-trained prompt tuning for few-shot learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8410–8423, Dublin, Ireland. Association for Computational Linguistics.

Pierre Guiraud. 1954. *Les caractères statistiques du vocabulaire: essai de méthodologie*.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2022a. Unnatural instructions: Tuning language models with (almost) no human labor. *ArXiv preprint*, abs/2212.09689.

Or Honovich, Uri Shaham, Samuel R Bowman, and Omer Levy. 2022b. Instruction induction: From few examples to natural language task descriptions. *ArXiv preprint*, abs/2205.10782.

Jennifer Hu and Roger Levy. 2023. Prompting is not a substitute for probability measurements in large language models. In *Proc. of EMNLP*, pages 5040–5060, Singapore. Association for Computational Linguistics.

Scott Jarvis. 2013. Capturing the diversity in lexical diversity. *Language learning*, 63:87–106.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

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. *ArXiv preprint*, abs/2001.08361.

Venelin Kovatchev, M. Antònia Martí, and Maria Salamó. 2018. ETPC - a paraphrase identification corpus annotated with extended paraphrase typology and negation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Kristopher Kyle. 2019. Measuring lexical richness. In *The Routledge handbook of vocabulary studies*, pages 454–476. Routledge.

Alina Leidinger, Robert Van Rooij, and Ekaterina Shutova. 2023. The language of prompting: What linguistic properties make a prompt successful? *ArXiv preprint*, abs/2311.01967.

Bill Yuchen Lin, Seyeon Lee, Rahul Khanna, and Xiang Ren. 2020. Birds have four legs?! NumerSense: Probing Numerical Commonsense Knowledge of Pre-Trained Language Models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6862–6868, Online. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, 55(9).

Timothy Liu and De Wen Soh. 2022. Towards better characterization of paraphrases. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8592–8601, Dublin, Ireland. Association for Computational Linguistics.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Lam Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2021. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *ArXiv preprint*, abs/2110.07602.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2024. Gpt understands, too. *AI open*, 5:208–215.

Sheng Lu, Hendrik Schuff, and Iryna Gurevych. 2023. How are prompts different in terms of sensitivity? *ArXiv preprint*, abs/2311.07230.

Heinz-Dieter Mass. 1972. Über den zusammenhang zwischen wortschatzumfang und länge eines textes. *Zeitschrift für Literaturwissenschaft und Linguistik*, 2(8):73.

Julian J. McAuley and Jure Leskovec. 2013. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, pages 897–908. International World Wide Web Conferences Steering Committee / ACM.

Philip M. McCarthy and Scott Jarvis. 2010. Mtld, vocd-d, and hd-d: A validation study of sophisticated approaches to lexical diversity assessment. *Behavior Research Methods*, 42(2):381–392.

Dominik Meier, Jan Philip Wahle, Terry Ruas, and Bela Gipp. 2025. Towards human understanding of paraphrase types in large language models.

Moran Mizrahi, Guy Kaplan, Dan Malkin, Rotem Dror, Dafna Shahaf, and Gabriel Stanovsky. 2024. State of what art? a call for multi-prompt llm evaluation. *Transactions of the Association for Computational Linguistics*, 12:933–949.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

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

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. *ArXiv preprint*, abs/2112.11446.

Laria Reynolds and Kyle McDonell. 2021. Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI EA '21, New York, NY, USA. Association for Computing Machinery.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Taylor Sorensen, Joshua Robinson, Christopher Rytting, Alexander Shaw, Kyle Rogers, Alexia Delorey, Mahmoud Khalil, Nancy Fulda, and David Wingate. 2022. An information-theoretic approach to prompt

engineering without ground truth labels. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 819–862, Dublin, Ireland. Association for Computational Linguistics.

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

Roeland Van Hout and Anne Vermeer. 2007. Comparing measures of lexical richness. *Modelling and assessing vocabulary knowledge*, 93:115.

Marta Vila, M. Antònia Martí, and Horacio Rodríguez. 2014. Is this a paraphrase? what kind? paraphrase boundaries and typology. *Open Journal of Modern Linguistics*, 04(01):205–218.

Jan Philip Wahle, Bela Gipp, and Terry Ruas. 2023. Paraphrase types for generation and detection. In *Proc. of EMNLP*, Singapore, Singapore. Association for Computational Linguistics.

Jan Philip Wahle, Terry Ruas, Frederic Kirstein, and Bela Gipp. 2022. How large language models are transforming machine-paraphrase plagiarism. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 952–963, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022a. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022b. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *ArXiv preprint*, abs/2206.07682.

Michael Wiegand, Alexandra Balahur, Benjamin Roth, Dietrich Klakow, and Andrés Montoyo. 2010. A survey on the role of negation in sentiment analysis. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, pages 60–68, Uppsala, Sweden. University of Antwerp.

Sohee Yang, Jonghyeon Kim, Joel Jang, Seonghyeon Ye, Hyunji Lee, and Minjoon Seo. 2023. Improving probability-based prompt selection through unified evaluation and analysis. *ArXiv preprint*, abs/2305.14877.

Ningyu Zhang, Luoqiu Li, Xiang Chen, Shumin Deng, Zhen Bi, Chuanqi Tan, Fei Huang, and Huajun Chen. 2022. Differentiable prompt makes pre-trained language models better few-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022. Large language models are human-level prompt engineers. *ArXiv preprint*, abs/2211.01910.

A Details on the Prompts & Tasks

Prompts. We use the template shown in Figure 8 to construct prompts for each of the 120 tasks. In Figure 10, we show an example for the numerical commonsense reasoning task NumerSense in which the model has to predict a blank token '_' (Lin et al., 2020). To paraphrase prompts, we use one of 26 paraphrase types in six groups shown in Table 9. Five examples of paraphrases applied to prompts can be seen in Table 1. In the first case, the added specification made the instruction more precise; in the second case, the model removed seemingly unnecessary context. The third case involved replacing words or phrases with synonyms or near-synonyms that have the same meaning in the given context, while the fourth case involved restructuring the sentence or changing the way the information is presented while maintaining the same overall meaning. Finally, the fifth case replaced a single word with a phrase (or vice versa) that conveys the same meaning. Further paraphrase type definitions and perturbation examples can be found in (Vila et al., 2014; Meier et al., 2025).

Tasks. We provide an overview of all 120 tasks together with the paraphrase type that, when applied to prompts in that task domain, had the most positive influence together with the potential downstream task performance gain over the original prompt in Tables 4 to 7. Further, we want to note an observation of our experiments showing that tasks containing content judged as unethical (such as toxic language detection on Twitter) led to many denied responses by the chat or instruction-finetuned models. Likely because of their post-training, models answered that they could not respond to these questions, although we did not ask them to produce new harmful content but to classify existing content. We removed these tasks from our final analysis. This observation raises questions about how many powerful models can be used in the future to classify toxic content when their guardrails prevent them from answering.

B Details on the Metrics

We provide the mathematical equations and interpretations of metrics we used to compute various aspects of this study, such as lexical diversity, prompt complexity, potential gains, etc. in the following subsections.

B.1 Prompt Complexity Metrics

Absolute Token Deviation. An intuitive heuristic that measures the total number of tokens added, removed, or changed between two text pairs. This metric reflects the overall extent of alteration in content and can be computed by:

$$tok(s1, s2) = |N_{s1} - N_{s2}| \qquad (3)$$

where N_s represents the number of tokens in sentence s. This metric captures the absolute difference in the length of the texts, offering a simple yet effective way to quantify the overall size of changes.

Word Position Deviation. This metric measures structural alterations by calculating the average shift in the positions of common words between two paraphrased sentences. The equation uses the mean of the maximum relative position shifts of all common words between two sentences $s1$ and $s2$, represented as

$$pos(s1, s2) = \qquad (4)$$
$$\frac{1}{N_\mathcal{C}} \sum max\{\delta_{s1,s2}(W), \delta_{s2,s1}(W)\} \quad (5)$$

where $\delta_{s1,s2}$ is the relative position shift of a word W with respect to sentence $s1$ in paraphrase pair $(s1, s2)$, and NC is the count of common words.

Lexical Deviation. This measure quantifies the difference in vocabulary used between two sentences. It is defined as the complement of the ratio of the number of common words to the total number of unique words in both sentences, given by the equation

$$lex(s1, s2) = 1 - \frac{N_\mathcal{C}}{N_\mathcal{A}} \qquad (6)$$

where $N_\mathcal{A}$ is the count of all unique words that occur in either occurs in either or both sentences and $N_\mathcal{C}$ is the set of common words that occur in both sentences.

B.2 Prompt Complexity Correlations

We calculate the Pearson correlation between each of the above prompt complexity metrics and the downstream performance in the following way.

P_0 represents the original prompt and $P_1, P_2, \ldots, P_{26}$ denote the paraphrased versions. The performance score for each paraphrased

Figure 8: The prompt template used to perform each of the 120 tasks. For Mixtral 8x7B Instruct and Gemma 7B Instruct, we prepend the system prompt to the user prompt because the models do not natively support system prompts.

Figure 9: The maximum performance gain models can experience if their prompts are adjusted in optimal ways as explained in Table 8.

prompt P_i is given by S_i, where S_i is ROUGE-L in this study. For each paraphrase P_i, the marker of change relative to P_0 is denoted by T_i (T_1 being deviation in absolute number of tokens, T_2 word position deviation, and T_3 lexical deviation). We calculate Pearson correlation r between T_i and task performance S_j:

$$Pearson(i,j) = \tag{7}$$

$$\frac{\sum_{i=1}^{n}(T_i - \bar{T})(S_i - \bar{S})}{\sqrt{\sum_{i=1}^{n}(T_i - \bar{T})^2}\sqrt{\sum_{i=1}^{n}(S_i - \bar{S})^2}} \tag{8}$$

where n represents the number of paired scores available (26 paraphrase-original pairs), $\bar{T}$ is the mean value of all n markers of change, and $\bar{S}$ is the mean value of all n performance scores.

B.3 Lexical Diversity Metrics

We use four main metrics to assess lexical richness for LLM generations: Root Type-Token Ratio (RTTR), Maas, Moving Average Type-Token Ratio (MATTR), and Measure of Lexical Diversity (MTLD). RTTR and Maas metrics consider the entire set of texts for a given category, focusing on the number of distinct words relative to the total number of words, with modifications to account for text length.

RTTR. Root Type-Token Ratio measures the proportion of unique words in a text, normalized by text length, indicating the lexical variety of the text (Guiraud, 1954). Computing the quotient of distinct words and the total number of tokens, the metric is defined as:

```
<system>
    You are a helpful assistant in solving various tasks. You should only output one answer to the task,
    nothing more, no explanations, and nothing around. Just read the instruction carefully,
    understand the positive and negative examples provided,
    and generate one single answer in the same way as the example's output.
</system>
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
<user>
    Instruction: Find the most appropriate number to replace the blank (indicated with _) and express it in words.

    Positive examples:
        Input: A lion has _ legs.
        Output: four

        Input: Numbers less than _ are called negative.
        Output: zero

        Input: There are _ hours in a day.
        Output: twenty four


    Negative examples:
        Input: A dog has _ legs.
        Output: 4   # Not expressed in words but numbers.

        Input: Numbers less than _ are called negative.
        Output: one   # Logically wrong.

    Input: Some plant varieties can grow up to _ feet tall.
    Output:
</user>
```

Figure 10: A prompt example for the first instance in the NumerSense dataset (Lin et al., 2020).

$$\text{RTTR} = \frac{T}{\sqrt{N}}$$

where T represents the number of unique types (distinct words), and N is the total number of tokens (words).

Maas. This metric measures the lexical richness by considering the relationship between the number of distinct words and the total word count, using a logarithmic transformation to reduce the impact of text length (Mass, 1972). The measure is calculated using the formula:

$$\log(V) = \log(N) + \alpha \log(N)$$

where V is the number of distinct words, N is the total number of words, and α is a parameter calculated as follows:

$$\alpha = \frac{\log(V) - \log(N)}{\log(N)}$$

MATTR. Moving Average Type-Token Ratio measures the stability of lexical diversity across different text segments, providing an average diversity score that accounts for variations within the text (Covington and McFall, 2010). Taking the average Type-Token Ratio (TTR) over a sliding window of size w, which in our experiments is 25 tokens, is calculated as:

$$\text{TTR}_i = \frac{T_i}{N_i}$$

Instruction Type	Instruction Text	ROUGE-L
Example 1: Addition/Deletion		
Original	Your task is to generate a headline (title) for this article.	0.60
Paraphrased	Your task is to generate a concise headline (title) for this article.	0.78
Example 2: Ellipsis		
Original	Your task is to generate a headline (title) for this article.	0.60
Paraphrased	Generate a headline (title) for this article.	0.67
Example 3: Same Polarity Substitution (contextual)		
Original	Summarize the main points of the given text in 3-4 sentences.	0.55
Paraphrased	Condense the key ideas of the provided passage into 3-4 sentences.	0.72
Example 4: Syntax/discourse structure changes		
Original	Explain the concept of gravity as if you were teaching a 10-year-old child.	0.63
Paraphrased	Imagine you're teaching a 10-year-old child. Explain the concept of gravity.	0.71
Example 5: Synthetic/analytic substitution		
Original	Translate the following English text into French.	0.58
Paraphrased	Convert the given English passage to its French equivalent.	0.70

Table 1: Comparison of original and paraphrased instructions for five types with downstream ROUGE-L scores.

where T_i is the number of unique types within the window i, and N_i is the number of tokens within the window i. The MATTR is then the average of these TTR values over all windows:

$$\text{MATTR} = \frac{1}{m} \sum_{i=1}^{m} \text{TTR}_i$$

where m is the number of windows.

MTLD. Measure of Lexical Diversity determines the text's lexical diversity by determining how many words are needed before the diversity drops below a set threshold, thereby capturing the consistency of word usage variety throughout the text (McCarthy and Jarvis, 2010). MTLD measures the text length until the Type-Token Ratio (TTR) reaches a threshold (0.72 in our experiments), then starting a new segment and continuing this process through the entire text measures MTLD. Mathematically:

$$\text{MTLD} = \frac{\text{Total tokens}}{\text{Number of segments}}$$

where a segment is defined as the number of tokens until the TTR reaches the threshold.

Higher RTTR (ranging from 0 to the square root of the total number of tokens), MATTR (0 to 1), and MTLD (typically 0 to infinity) values indicate greater lexical diversity, while lower values suggest less variety in word usage. The Maas metric, which typically ranges close to 0, operates inversely: lower values indicate higher lexical richness, while higher values suggest lower diversity.

B.4 Downstream Task Metrics

Because we evaluate on the benchmark suite of Wang et al. (2022b), we use the corresponding evaluation metric ROUGE-L for each task. Their study shows that ROUGE-L correlates well with human judgments for these tasks and yields comparable scores for all generative tasks. Further, this choice requires no postprocessing or judgment by another model, which could have its own biases.

B.5 Potential Downstream Gain

Figure 1 has shown the potential median gain that can be reached by paraphrasing prompts in specific types. In Table 8, we show how the median is calculated per example. For each example in a task dataset, we take the downstream task scores of paraphrased prompts that are higher than the original prompt and take the median. Then, for all examples in the dataset, we average that median.

Further, one can also compute the maximum performance that can be reached if the best paraphrase type would be chosen to paraphrase the prompt. As previous experiments have shown, it is challenging to devise a robust method to select the best paraphrase of a prompt. More research on the Monte-Carlo approach could bring us closer to this maximum. The median is a fair way to represent a gain that users could expect by selecting one of the multiple paraphrases of prompts that are better than

Task Family (↓)	Tok.	Pos.	Lex.
Answerability Class.	-0.06	-0.06	-0.06
Commonsense Class.	-0.01	-0.01	0.00
Coreference Resolution	0.00	0.00	0.00
Dialogue Generation	-0.02	-0.01	0.00
Fill in The Blank	0.04	0.04	0.04
Information Extr.	0.03	0.06	0.04
Named Entity Rec.	0.01	0.02	0.01
Program Execution	-0.02	-0.01	-0.03
Question Answering	0.14	0.13	0.13
Question Generation	0.06	0.06	0.06
Question Rewriting	0.03	0.04	0.04
Question Underst.	-0.01	-0.03	-0.02
Sentence Composition	0.10	0.11	0.11
Sentiment Analysis	0.03	0.03	0.03
Summarization	0.08	0.08	0.08
Text Categorization	-0.03	-0.02	-0.02
Text Completion	0.01	0.02	0.00
Text Matching	0.00	0.01	0.01
Text Quality Eval.	0.00	-0.01	-0.01
Text to Code	-0.02	-0.01	0.01
Textual Entailment	0.03	0.01	0.03
Title Generation	0.01	0.01	0.01
Word Semantics	-0.02	-0.03	-0.01
Wrong Candidate Gen.	0.01	0.03	0.00
Overall	0.02	0.02	0.02

Table 2: The average Pearson correlations between the number of tokens (Tok.) and task perf., word position deviation (Pos.) and task perf., and lexical deviation (Lex.) and task perf. for LLaMA 3 70B. Each task family has five downstream tasks. The average p-value is 0.05.

Task Family	Min	Max	Avg (↓)
Title Generation	24	53	36.0
Fill in The Blank	25	60	43.2
Summarization	21	123	52.0
Text Quality Eval.	42	67	52.6
Named Entity Rec.	38	77	52.8
Word Semantics	39	74	53.8
Sentiment Analysis	24	115	56.6
Sentence Composition	19	84	61.8
Answerability Class.	47	70	63.6
Text Completion	40	111	66.0
Dialogue Generation	27	186	67.8
Question Answering	45	106	72.0
Program Execution	43	110	75.4
Coreference Resolution	24	219	78.8
Information Extr.	70	91	79.0
Textual Entailment	61	111	81.2
Text Categorization	55	116	88.2
Text Matching	34	203	91.0
Question Generation	29	160	102.2
Wrong Candidate Gen.	105	192	129.4
Commonsense Class.	49	202	148.8
Question Rewriting	46	753	216.0
Question Understanding	125	581	291.8
Text to Code	121	638	390.6
Overall	19	753	102.1

Table 3: The number of tokens per input prompt (computed using the GPT-2 tokenizer). **Min** shows the shortest prompt, **Max** the longest prompt, and **Avg** the average prompt length for all five tasks.

the original. Compared to the mean, the median is not influenced by outliers that are unlikely to select (e.g., the two 1.0 scores in Table 8 are unlikely to reach; the most likely expectation is 0.67, which is also the median).

C Additional Results

Prompt Complexity Correlations. Table 2 shows the individual correlations between the three markers of prompt complexity and downstream task performance as described in Appendix B.2. All correlations are close to zero, and only very small correlations exist for question answering (0.13 - 0.14) and sentence composition (0.10 - 0.11) with $p < 0.05$. We could generally find no evidence that the complexity of the prompt contributes to gains or losses in performance outcomes.

Prompt Lengths. We conducted a post-hoc analysis of the number of input tokens across various task families using the GPT-2 tokenizer. As shown in Table 3, the average number of tokens varies significantly depending on the nature of the task, ranging from 36 tokens for title generation tasks to 390 tokens for text-to-code tasks. The overall distribution of input tokens is even broader when looking at individual tasks, with the shortest task, sentence composition, using as few as 19 tokens, and the longest, question rewriting, reaching up to 753 tokens. This wide spectrum of prompt lengths indicates that our study covers a diverse array of real-world use cases, from concise to extended prompts. By analyzing 120 tasks across 24 task families, we believe our conclusions hold across a broad range of prompt lengths, including those much shorter (e.g., fewer than 36 tokens) and much longer (e.g., more than 800 tokens).

Results by Task and Type. We show a detailed table of avgerage downstream task performance gain or loss over the original prompt of a task when paraphrasing the prompt using modifications of one of the six different paraphrase groups in Table 10. We decompose this table for each model in Tables 11 to 15.

Maximum Potential Gain. To understand how much models would possibly benefit from adjusting prompts in optimal ways as explained in Table 8, we also show the maximum potential gain in Figure 9 of the teaser on the first page of this paper (Figure 1). This gives an upper bound to what is possible if prompts were optimally adjusted with paraphrase types. Again, we note that although finding these optimal prompt adjustments is challenging, this task is not impossible and gives space for future work.

Model Scale Comparison. Figure 12 compares model size between LLaMA 3 8B and LLaMA 3 70B. Observe how the smaller 8B parameter model benefits more across tasks when compared to its larger 70B version. Only for program execution, information extraction, sentence completion, question rewriting, and question generation, the 70B model shows a larger gain.

Upon closer examination, the performance difference between the two models across various tasks highlights several trends. The smaller 8B model consistently outperforms the 70B model in tasks such as text categorization, text quality evaluation, named entity recognition, and sentiment analysis. This suggests that for these specific tasks, the architectural or training enhancements in the 8B model may be more efficiently leveraged, potentially due to better optimization or more effective use of parameter space.

In contrast, the 70B model demonstrates improvements over the 8B model in tasks like program execution, information extraction, and sentence completion. This indicates that for these more complex or nuanced tasks, the increased parameter count of the 70B model likely provides a richer representation space, allowing it to capture and utilize more intricate patterns and dependencies in the data.

Morphology			
	Derivational changes	-0.86	1.26
	Modal verb changes	-0.87	1.26
	Inflectional changes	-0.91	0.69
Lexicon	Same Pol. Sub. (contextual)	-0.86	1.32
	Same Pol. Sub. (named ent.)	-0.87	1.28
	Spelling changes	-0.87	1.25
	Change of format	-0.87	1.21
	Same Pol. Sub. (habitual)	-0.87	1.21
Lexico-Syntax	Opp. Pol. Sub. (contextual)	-0.88	1.21
	Opp. Pol. Sub. (habitual)	-0.89	1.19
	Synthetic/Analytic Sub.	-0.87	1.16
	Converse Sub.	-0.87	1.12
Discourse	Syntax/Discourse Structure	-0.85	1.35
	Punctuation Changes	-0.87	1.18
	Direct/Indirect Style Altern.	-0.87	1.15
Syntax	Subordination and nesting	-0.87	1.29
	Diathesis alternation	-0.88	1.25
	Ellipsis	-0.85	1.15
	Negation switching	-0.85	1.14
	Coordination changes	-0.81	1.13
Others	Addition/Deletion	-0.88	1.16
	Change of Order	-0.90	1.10
	Semantic-based	-0.86	1.09
Extremes	Entailment	-0.87	1.15
	Identity	-0.85	1.13
	Non-paraphrase	-0.84	1.09

Figure 11: The average percentage in downstream task performance gain or loss for applying different paraphrase types across all five models. This decomposes the groups of Figure 3 into their individual types.
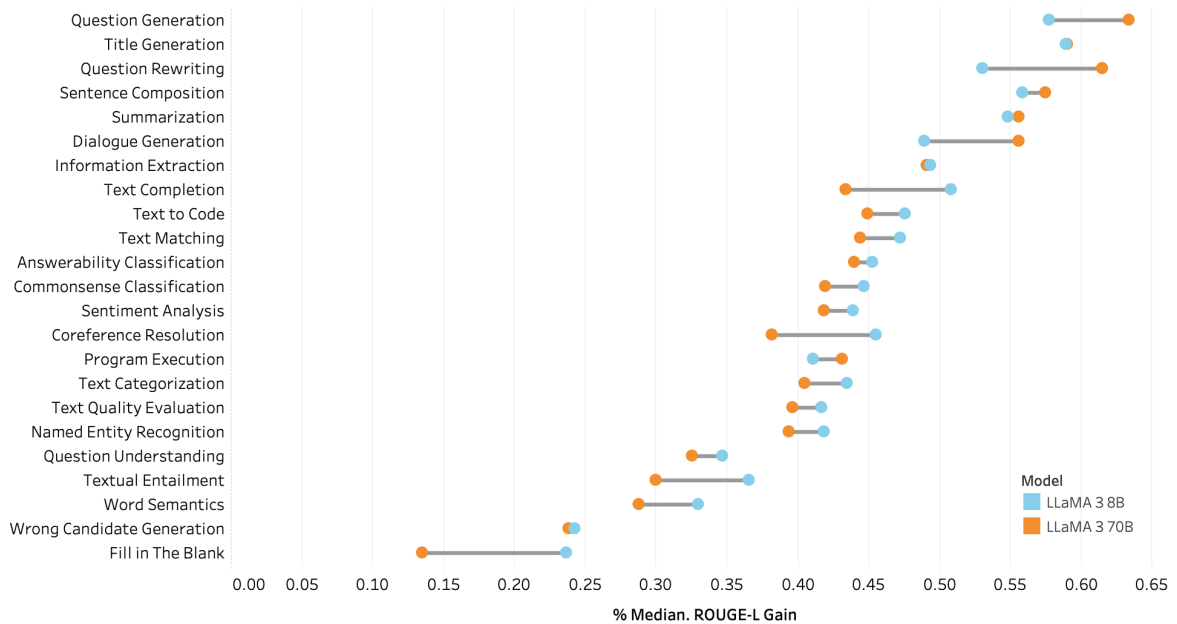

Figure 12: A comparison between model scale for LLaMA 3 8B (blue) and 70B (orange) for different tasks.

Task ID	Dataset Name	Domain	Inp. Tokens	Out. Tokens	URL
Text Completion - Morphology (+11.4% ↑), Syntax (-5.3% ↓)					
105	ROCStories	Story, Commonsense	39.3	9.0	Link
297	ROCStories	Story	56.1	1.0	Link
268	CaseHold	Law	269.1	1.0	Link
138	Detoxifying LMs	Social Media	59.6	2.0	Link
296	ROCStories	Story	56.1	1.0	Link
Question Answering - Extremes (+24.8% ↑), Lexicon (-3.0% ↓)					
1441	DoQA Movies	Movies, Dialogue	190.6	16.0	Link
2	Quoref	Wikipedia	356.6	1.7	Link
24	Cosmos QA	Personal Narratives	82.5	8.3	Link
332	TellMeWhy	Story	50.4	8.9	Link
310	RACE	English Exams	351.6	1.0	Link
Fill in The Blank - Extremes (+18.4% ↑), Discourse (-5.0% ↓)					
672	NumerSense	Commonsense	11.1	1.0	Link
277	StereoSet	Stereotypes	8.8	1.0	Link
965	LibriSpeech ASR	Books	42.1	1.0	Link
1217	ATOMIC	Sociology, Commonsense	5.3	1.1	Link
1360	NumerSense	Concepts and Relations	26.1	1.0	Link
Sentiment Analysis - Morphology (+15.6% ↑), Discourse (-2.1% ↓)					
420	PerSenT	News	352.6	1.0	Link
889	GoEmotions	Narrative, Dialogue	12.3	1.0	Link
1312	Amazon Review Polarity	Reviews	56.7	1.0	Link
1535	Daily Dialog	Dialogue	137.8	1.0	Link
284	IMDB	Movie Reviews	229.9	1.0	Link
Program Execution - Morphology (+1.9% ↑), Discourse (-4.2% ↓)					
208	Combinations of List	Mathematics	5.0	22.1	-
378	Reverse Words	Image Captions	21.4	10.4	-
1148	Maximum ASCII Value	Computer Science	1.0	1.0	-
94	Calculate Mean	Code	6.0	1.0	Link
99	Reverse Elements	Code	17.9	6.3	-
Question Generation - Morphology (+21.5% ↑), Discourse (+1.3% ↑)					
6	MCTACO	News, Wiki, Law, History	18.9	8.2	Link
1602	WebQuestions	Knowledge Base	9.6	6.6	Link
599	CUAD	Law	3153.9	43.3	Link
405	NarrativeQA	Books, Movies	575.7	8.6	Link
739	LhoestQ	Web	142.1	10.4	-

Table 4: Overview of the tasks in this study with their average input and output tokens across all examples. We also show the two paraphrase types with the highest and lowest gain or loss across all five sampled tasks within a task category. Table 1/4.

Task ID	Dataset Name	Domain	Inp. Tokens	Out. Tokens	URL
Text to Code - Morphology (+2.4% ↑), Lexico-Syntax (-5.5% ↓)					
869	CFQ MCD1	SQL	41.2	1.0	Link
956	LeetCode 420	Mathematics	3.0	1.0	Link
128	SCAN	Machine Learning	7.0	10.8	Link
107	SPLASH	SQL	13.3	17.5	Link
211	Logic2Text	Wikipedia, Logic	52.6	1.0	Link
Question Rewriting - Morphology (-4.7% ↓), Syntax (-17.2% ↓)					
402	GrailQA	Knowledge Base	125.5	11.2	Link
671	AmbigQA	Wikipedia	8.9	14.4	Link
34	WinoGrande	Commonsense	26.1	20.0	Link
1622	Disfl QA	Wikipedia	14.6	9.9	Link
670	AmbigQA	Wikipedia	8.9	12.1	Link
Summarization - Lexicon (+10.8% ↑), Syntax (+2.6% ↑)					
1290	XSum	News	371.8	21.1	Link
1357	XLSum	News	459.2	22.1	Link
668	SciTLDR	Scientific Papers	159.5	20.6	Link
589	Amazon Food Reviews	Reviews	79.6	4.2	Link
672	Amazon / Yelp Summ.	Reviews	404.0	50.7	Link
Commonsense Classification - Morphology (+14.6% ↑), Lexicon (-2.7% ↓)					
116	Com2Sense	Concepts and Relations	19.2	1.0	Link
1204	ATOMIC	Social Commonsense	10.4	1.0	Link
1209	ATOMIC	Physical Commonsense	7.0	1.0	Link
1199	ATOMIC	Social Commonsense	7.4	1.0	Link
1208	ATOMIC	Social Commonsense	7.0	1.0	Link
Text Matching - Syntax (+5.4% ↑), Discourse (-12.2% ↓)					
1288	GLUE MRPC	News, Web	45.9	1.0	Link
276	Enhanced WSC	Dialogue, Narrative	39.8	1.0	Link
910	Bianet	News	48.5	1.0	Link
148	AFS	Government and Politics	25.6	1.0	Link
624	OHSUMED	Scientific Papers	188.5	11.2	Link
Word Semantics - Morphology (+14.9% ↑), Lexico-Syntax (-6.5% ↓)					
1585	ROOT09	Misc.	1.0	1.0	Link
1582	BLESS	Misc.	1.0	1.0	Link
141	Odd-Man-Out	Card Game	9.0	1.1	Link
142	Odd-Man-Out	Card Game	5.3	1.1	Link
458	MATRES	News	49.2	1.0	Link

Table 5: Overview of the tasks in this study with their average input and output tokens across all examples. We also show the two paraphrase types with the highest and lowest gain or loss across all five sampled tasks within a task category. Table 2/4.

Task ID	Dataset Name	Domain	Inp. Tokens	Out. Tokens	URL
Question Understanding - Morphology (+15.1% ↑), Discourse (-4.1% ↑)					
19	MCTACO	News, Wiki, Law, History	27.8	1.0	Link
46	Misc.	Pop, Nat. Science, History	14.8	1.0	-
18	MCTACO	News, Wiki, Law, History	24.9	1.0	Link
1289	TREC	Misc.	10.2	1.0	Link
Text Quality Evaluation - Morphology (+7.7% ↑), Discourse (-8.6% ↑)					
616	CoLA	Linguistics	7.9	1.0	Link
1284	Human Ratings of NLG	Dialogue, Restaurants	23.8	1.0	Link
1623	Disfl QA	Wikipedia	12.2	1.0	Link
1283	Human Ratings of NLG	Dialogue, Restaurants	24.9	1.0	Link
1341	MSR Text Compression	News, Dialogue, Misc.	19.4	1.0	Link
Dialogue Generation - Morphology (+9.3% ↑), Discourse (-3.9% ↑)					
639	MultiWOZ v2.2	Dialogue	13.2	12.6	Link
1730	PersonaChat	Dialogue	143.0	9.9	Link
576	Curiosity Dialogs	Concepts and Relations	66.2	23.0	Link
361	Spolin	Dialogue	32.3	1.0	Link
1603	SMCalFlow	Dialogue	8.0	8.7	Link
Coreference Resolution - Others (+25.1% ↑), Lexico-Syntax (+1.6% ↑)					
1391	WinoGrande	Physical Commonsense	22.8	1.0	Link
648	Winograd WSC	Narrative	19.4	1.7	Link
133	WinoWhy	Concepts and Relations	43.8	1.0	Link
330	GAP	Wikipedia	73.8	1.4	Link
329	GAP	Wikipedia	81.2	1.0	Link
Answerability Classification - Lexico-Syntax (+11.8% ↑), Others (-3.5% ↑)					
290	TellMeWhy	Story	51.0	1.3	Link
50	MultiRC	News, Wiki, Law, History	23.3	1.0	Link
349	SQuAD 2.0	Wikipedia	140.5	1.0	Link
20	MCTACO	News, Wiki, Law, History	25.0	1.0	Link
1640	Adversarial QA	Wikipedia	127.3	1.0	Link
Wrong Candidate Generation - Morphology (+26.0% ↑), Syntax (-1.4% ↑)					
135	Winowhy	Concepts and Relations	26.0	12.5	Link
42	QASC	Nat. Science	22.5	1.6	Link
55	MultiRC	News, Wiki, Law, History	308.0	3.4	Link
11	MCTACO	News, Wiki, Law, History	27.6	4.9	Link
631	DBPedia 14	Wikipedia	53.4	1.4	Link

Table 6: Overview of the tasks in this study with their average input and output tokens across all examples. We also show the two paraphrase types with the highest and lowest gain or loss across all five sampled tasks within a task category. Table 3/4.

Task ID	Dataset Name	Domain	Inp. Tokens	Out. Tokens	URL
Sentence Composition - Discourse (+3.8% ↑), Morphology (-5.0% ↓)					
184	SNLI	Image Captions	23.4	8.3	Link
1613	SICK	Image & Video Captions	12.6	9.2	Link
1530	SciTailv1.1	Nat. Science	19.6	12.2	Link
1368	HealthFact	Healthcare	566.0	11.2	Link
1364	HANS	Movie Reviews	8.6	6.0	Link
Textual Entailment - Morphology (+17.5% ↑), Discourse (-6.2% ↓)					
640	e-SNLI	Misc.	23.1	1.0	Link
201	MultiNLI	History, Fiction, Dialogue, Law	54.9	1.0	Link
1387	ANLI R3	Misc.	67.8	1.0	Link
190	SNLI	Image Captions	24.2	1.0	Link
1612	SICK	Image & Video Captions	21.1	1.0	Link
Named Entity Recognition - Morphology (+9.2% ↑), Others (-15.3% ↓)					
1480	JNLPBA	Bioinformatics	29.5	2.5	Link
1486	ANEM	Clinical Knowledge	30.3	1.2	Link
959	E2E	Restaurants	27.1	2.1	Link
1483	ChemProt	Chemistry	14.8	1.2	Link
1481	BC2GM	Bioinformatics	30.5	2.4	Link
Text Categorization - Morphology (+9.2% ↑), Discourse (-7.8% ↓)					
1495	ADE Corpus V2	Clinical Knowledge, Healthcare	17.7	3.0	Link
1489	Sarcasm in Twitter	Social Media	17.7	1.0	Link
1308	Amazon Reviews	Reviews	64.2	1.0	Link
617	Amazon Reviews	Reviews	68.0	1.3	Link
1541	AG News	News	37.8	1.0	Link
Title Generation - Morphology (+12.1% ↑), Lexico-Syntax (+0.9% ↑)					
220	ROCStories	Narrative, Story	60.0	1.0	Link
288	GigaWord	News	29.9	8.8	Link
619	OHSUMED	Reviews	157.5	11.2	Link
1540	PeerRead	Computer Science	164.9	8.4	Link
1659	BillSum	Government and Politics	180.3	18.4	Link
Information Extraction - Morphology (+7.5% ↑), Others (-5.3% ↓)					
646	Winograd WSC	Narrative	16.4	1.7	Link
1413	DART	Wikipedia	8.9	1.7	Link
748	GLUCOSE	Story	55.4	11.5	Link
1411	DART	Wikipedia	9.1	2.2	Link
1506	Synthetic	Pop	22.5	3.0	-

Table 7: Overview of the tasks in this study with their average input and output tokens across all examples. We also show the two paraphrase types with the highest and lowest gain or loss across all five sampled tasks within a task category. Table 4/4.

Paraphrase Group	Paraphrase Type	Model Output	ROUGE-L
Morphology	Inflectional	Great Value	1.00
Extremes	Entailment	Great Value	1.00
Syntax	Negation Switching	Great Value Cat Food	0.67
Morphology	Modal Verb	Great Value Cat Food	0.67
Morphology	Derivational	Great Value Cat Food	0.67
Discourse	Direct/Indirect Style Altern.	Great Value Cat Food	0.67
Lexico-Syntax	Synthetic/Analytic Sub.	Great Value for Cats	0.67
Lexico-Syntax	Opp. Pol. Sub. (Habitual)	Great Value for Multiple Cats	0.57
Lexicon	Spelling	Great Value for Multiple Cats	0.57
Extremes	Non-Paraphrase	Good Value Cat Food	0.33
Original	Original	Good Value Cat Food	0.33
Discourse	Syntax/Discourse Structure	Good Value for Cats	0.33
Syntax	Subordination and Nesting	Purrfect Kitten Food	0.00
Others	Addition/Deletion	Kittens loved the food.	0.00
Discourse	Punctuation	Kittens loved the food	0.00
Others	Change of Order	Cats love the food.	0.00
Extremes	Identity	Cats loved the food	0.00
Syntax	Coordination	Kittens Loved the Food	0.00
Syntax	Ellipsis	Good for Kittens	0.00
Syntax	Diathesis Alternation	Positive	0.00
Lexico-Syntax	Opp. Pol. Sub. (Contextual)	Good Quality Cat Food	0.00
Lexico-Syntax	Converse Sub.	Cats Loved Food	0.00
Lexicon	Same Pol. Sub. (Named Ent.)	Kittens loved the food.	0.00
Lexicon	Same Pol. Sub. (Habitual)	Perfect for Multiple Cats	0.00
Lexicon	Same Pol. Sub. (Contextual)	Multiple Cat Food	0.00
Lexicon	Change of Format	Cats loved the food	0.00
Others	Semantic-Based	Cats loved the food	0.00

Table 8: How we calculate the potential performance gain for paraphrasing prompts using a selected example from the Amazon food review summarization task (McAuley and Leskovec, 2013) with LLaMA 3 8B. Median Gain. To calculate the median gain (as in Figure 1), we consider all paraphrases of the prompt that are better than the original prompt (green and yellow). Median Loss. To calculate the median loss (as in Figure 1), we consider all paraphrases of the prompt that are worse than the original prompt (red). Max. We calculate the maximum possible gain by aggregating only the best performing paraphrases of the prompts highlighted in yellow.

Paraphrase Type	# Examples Computed
Morphology	**360k**
Derivational Changes	
Inflectional Changes	
Modal Verb Changes	
Lexicon	**600k**
Spelling changes	
Change of format	
Same Polarity Substitution (contextual)	
Same Polarity Substitution (habitual)	
Same Polarity Substitution (named ent.)	
Lexico-syntactic	**480k**
Converse substitution	
Opposite polarity substitution (contextual)	
Opposite polarity substitution (habitual)	
Synthetic/analytic substitution	
Syntax	**600k**
Coordination changes	
Diathesis alternation	
Ellipsis	
Negation switching	
Subordination and nesting changes	
Discourse	**360k**
Direct/indirect style alternations	
Punctuation changes	
Syntax/discourse structure changes	
Others	**720k**
Addition/Deletion	
Change of order	
Semantic-based	
Entailment	
Identity	
Non-paraphrase	
Total	**3,24m**

Table 9: An overview of the considered types of paraphrase prompts, categorized into their six main groups. The numbers indicate how many task examples with instructions of that paraphrase type group were run in our experiments (this is a product of all models, tasks, and number of examples per task). The number of paraphrases within a group is equally balanced (i.e., derivational changes have occurred the same amount of time as inflectional changes).

Task Family	Morphology	Syntax	Lexicon	Lex.-Syn.	Discourse	Others
Answerability Classification	+7.0%	-3.0%	+6.2%	**+11.8%**	+0.9%	-3.5%
Commonsense Classification	+14.6%	+2.9%	-2.7%	-1.9%	+0.6%	-1.6%
Coreference Resolution	+8.3%	**+9.4%**	+4.6%	+1.6%	+3.3%	**+25.1%**
Dialogue Generation	+9.3%	+3.4%	+6.3%	+9.2%	-3.9%	+4.6%
Fill in The Blank	+10.0%	-2.0%	+4.7%	+2.7%	-5.0%	+18.4%
Information Extraction	+7.5%	-1.7%	-1.0%	-5.2%	+6.1%	-5.3%
Named Entity Recognition	+9.2%	-8.7%	-2.1%	+2.0%	**+8.7%**	-15.3%
Program Execution	+1.94%	-3.1%	-1.5%	-1.7%	-4.2%	-2.8%
Question Answering	+10.3%	+1.8%	-3.0%	+9.9%	+5.0%	+24.8%
Question Generation	+21.5%	+5.9%	+2.5%	+5.9%	+1.3%	+4.1%
Question Rewriting	-4.7%	-17.2%	-6.6%	-5.8%	-8.2%	-8.6%
Question Understanding	+15.1%	+3.5%	-2.7%	-2.8%	-4.1%	-3.3%
Sentence Composition	-5.0%	+3.4%	-2.9%	-2.7%	+3.8%	+2.9%
Sentiment Analysis	+15.6%	+1.6%	+1.7%	+1.9%	-2.1%	-1.7%
Summarization	+6.9%	+2.6%	**+10.8%**	+4.8%	+7.8%	+7.5%
Text Categorization	+9.2%	-5.3%	-7.0%	-1.6%	-7.8%	-2.4%
Text Completion	+11.4%	-5.3%	-3.0%	-2.6%	-3.1%	+1.7%
Text Matching	+0.5%	+5.4%	-1.1%	+0.3%	-12.2%	-1.9%
Text Quality Evaluation	+7.7%	-1.1%	+1.4%	+1.2%	-8.6%	+1.4%
Text to Code	+2.4%	-2.1%	-1.7%	-5.5%	-0.6%	+0.7%
Textual Entailment	+17.5%	-4.2%	+5.2%	+5.3%	-6.2%	-4.6%
Title Generation	+12.1%	+5.8%	+7.1%	+1.7%	+0.9%	+7.4%
Word Semantics	+14.9%	-0.5%	-2.7%	-6.5%	+3.1%	+2.6%
Wrong Candidate Generation	**+26.0%**	-1.4%	+8.9%	+7.5%	-2.9%	+5.9%

Table 10: The average downstream task performance gain or loss over the original prompt when paraphrased with a specific type from one of the six groups (columns) of tasks within a certain task family (rows) as an average over **all models**. We calculate the average across all paraphrase types in one of the six paraphrase groups and across all five tasks within one of the 24 categories. **Bold** indicates the highest score per column. Small changes between -3% and +3% are not colored.

Task Family	Morphology	Syntax	Lexicon	Lex.-Syn.	Discourse	Others
Answerability Classification	-9.0% ↓	-2.1%	**+21.9% ↑**	**+57.6% ↑**	0.0%	0.0%
Commonsense Classification	+6.4% ↑	0.0%	-3.1% ↓	-2.4%	0.0%	0.0%
Coreference Resolution	+6.4% ↑	**+35.0% ↑**	+10.8% ↑	+18.2% ↑	0.0%	+25.0% ↑
Dialogue Generation	+8.4% ↑	+1.6%	+4.5% ↑	+1.6%	-2.0%	+28.6% ↑
Fill in The Blank	-18.4% ↓	+4.5% ↑	+15.4% ↑	+3.3% ↑	-10.4% ↓	**+36.0% ↑**
Information Extraction	+8.7% ↑	-11.4% ↓	-13.6% ↓	-15.3% ↓	+17.9% ↑	-20.0% ↓
Named Entity Recognition	+9.0% ↑	-42.3% ↓	-16.2% ↓	+20.6% ↑	**+49.7% ↑**	-71.6% ↓
Program Execution	+9.7% ↑	-11.7% ↓	-7.9% ↓	-7.5% ↓	-18.9% ↓	-23.7% ↓
Question Answering	+7.6% ↑	+9.6% ↑	-6.6% ↓	+4.4% ↑	-4.9% ↓	+13.0% ↑
Question Generation	+19.4% ↑	-16.4% ↓	-10.8% ↓	-11.8% ↓	-10.7% ↓	-6.5% ↓
Question Rewriting	-8.3% ↓	-9.0% ↓	-5.8% ↓	-8.1% ↓	-26.1% ↓	-15.9% ↓
Question Understanding	-12.3% ↓	-14.6% ↓	-17.5% ↓	-21.3% ↓	-13.2% ↓	-34.8% ↓
Sentence Composition	-23.7% ↓	+6.3% ↑	+4.6% ↑	+4.7% ↑	+6.4% ↑	+1.7%
Sentiment Analysis	+3.5% ↑	0.0%	-2.5%	-5.6% ↓	0.0%	0.0%
Summarization	+14.6% ↑	-4.0% ↓	+7.9% ↑	+9.5% ↑	-5.3% ↓	+4.3% ↑
Text Categorization	+5.3% ↑	-19.5% ↓	-30.8% ↓	-13.2% ↓	-41.2% ↓	0.0%
Text Completion	+20.0% ↑	-33.7% ↓	-30.3% ↓	-18.8% ↓	-17.1% ↓	-10.0% ↓
Text Matching	+3.0%	+22.1% ↑	+5.5% ↑	-1.0%	-51.3% ↓	-22.7% ↓
Text Quality Evaluation	-5.0% ↓	-3.2% ↓	-4.0% ↓	-7.4% ↓	-65.4% ↓	0.0%
Text to Code	+8.0% ↑	+3.9% ↑	+10.0% ↑	-2.6%	-1.0%	0.0%
Textual Entailment	+16.3% ↑	-31.1% ↓	+17.0% ↑	+14.9% ↑	-35.5% ↓	-1.2%
Title Generation	+11.6% ↑	+9.9% ↑	+9.3% ↑	-13.1% ↓	-6.3% ↓	-1.8%
Word Semantics	+4.9% ↑	-7.8% ↓	-7.7% ↓	-19.6% ↓	+9.8% ↑	-0.1%
Wrong Candidate Generation	**+24.5% ↑**	+6.4% ↑	+8.4% ↑	+9.6% ↑	+8.9% ↑	+5.1% ↑

Table 11: The average downstream task performance gain or loss over the original prompt when paraphrased with a specific type from one of the six groups (columns) of tasks within a certain task family (rows) for **Mixtral 8x7B Instruct (47B)**. We calculate the average across all paraphrase types in one of the six paraphrase groups and across all five tasks within one of the 24 categories. **Bold** indicates the highest score per column. Small changes between -3% and +3% are not colored.

Task Family	Morphology	Syntax	Lexicon	Lex.-Syn.	Discourse	Others
Answerability Classification	+34.7% ↑	-10.5% ↓	+9.2% ↑	+5.0% ↑	+4.6% ↑	-17.6% ↓
Commonsense Classification	+44.9% ↑	+14.3% ↑	-10.4% ↓	+11.2% ↑	+2.8%	-7.9% ↓
Coreference Resolution	+23.1% ↑	-10.0% ↓	+11.1% ↑	-8.9% ↓	-4.8% ↓	+0.4%
Dialogue Generation	+15.5% ↑	-11.3% ↓	+11.4% ↑	+11.8% ↑	+0.9%	-8.9% ↓
Fill in The Blank	+30.7% ↑	-11.3% ↓	+12.5% ↑	-7.9% ↓	-16.7% ↓	+17.6% ↑
Information Extraction	+17.8% ↑	-4.7% ↓	+8.2% ↑	-8.3% ↓	+12.6% ↑	-6.5% ↓
Named Entity Recognition	+17.2% ↑	-5.2% ↓	+9.0% ↑	-9.5% ↓	-5.7% ↓	-5.1% ↓
Program Execution	+22.6% ↑	-7.1% ↓	+8.3% ↑	-9.1% ↓	+4.7% ↑	+9.6% ↑
Question Answering	+17.6% ↑	+4.7% ↑	**+15.9% ↑**	+8.0% ↑	+12.4% ↑	-8.3% ↓
Question Generation	+18.1% ↑	+11.1% ↑	+14.0% ↑	**+16.6% ↑**	-8.7% ↓	**+30.4% ↑**
Question Rewriting	-13.6% ↓	-4.0% ↓	-12.3% ↓	+6.8% ↑	-5.9% ↓	-7.5% ↓
Question Understanding	+35.3% ↑	**+17.7% ↑**	-11.3% ↓	-10.6% ↓	-9.6% ↓	-2.3%
Sentence Composition	+14.7% ↑	-6.2% ↓	-9.4% ↓	+6.5% ↑	+6.8% ↑	-4.5% ↓
Sentiment Analysis	+30.6% ↑	+8.1% ↑	+10.0% ↑	+10.1% ↑	-9.5% ↓	-8.5% ↓
Summarization	-16.1% ↓	-9.9% ↓	+11.3% ↑	+10.1% ↑	+16.9% ↑	+10.9% ↑
Text Categorization	+26.3% ↑	-10.9% ↓	-9.5% ↓	+8.0% ↑	-8.0% ↓	-12.1% ↓
Text Completion	+24.0% ↑	+11.7% ↑	+14.3% ↑	+10.4% ↑	-4.5% ↓	+8.8% ↑
Text Matching	-10.5% ↓	-4.4% ↓	-6.8% ↓	+4.1% ↑	-9.9% ↓	+13.8% ↑
Text Quality Evaluation	+28.2% ↑	-7.3% ↓	+11.6% ↑	+6.7% ↑	**+22.5% ↑**	+7.1% ↑
Text to Code	-11.6% ↓	-14.3% ↓	-11.4% ↓	-19.2% ↓	+10.8% ↑	+1.2%
Textual Entailment	**+43.2% ↑**	+9.1% ↑	+13.8% ↑	+11.8% ↑	+4.9% ↑	-23.3% ↓
Title Generation	+22.3% ↑	+9.6% ↑	+12.0% ↑	+7.7% ↑	-9.4% ↓	-7.3% ↓
Word Semantics	+38.5% ↑	+5.9% ↑	-9.0% ↓	-12.9% ↓	+1.7%	+8.3% ↑
Wrong Candidate Generation	+18.9% ↑	-7.8% ↓	+9.6% ↑	+8.2% ↑	-6.1% ↓	+6.7% ↑

Table 12: The average downstream task performance gain or loss over the original prompt when paraphrased with a specific type from one of the six groups (columns) of tasks within a certain task family (rows) for **Gemma 7B Instruct (7B)**. We calculate the average across all paraphrase types in one of the six paraphrase groups and across all five tasks within one of the 24 categories. **Bold** indicatess the highest score per paraphrase group (column). Small changes between -3% and +3% are not colored.

Task Family	Morphology	Syntax	Lexicon	Lex.-Syn.	Discourse	Others
Answerability Classification	+3.2% ↑	-2.6%	-0.1%	-1.7%	0.0%	0.0%
Commonsense Classification	+10.4% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Coreference Resolution	+5.8% ↑	0.0%	-1.3%	0.0%	0.0%	0.0%
Dialogue Generation	+11.2% ↑	**+19.4% ↑**	+5.6% ↑	**+18.4% ↑**	-8.5% ↓	0.0%
Fill in The Blank	+13.3% ↑	-3.6% ↓	0.0%	0.0%	0.0%	0.0%
Information Extraction	+5.5% ↑	+6.6% ↑	+3.9% ↑	-2.8%	-0.4%	0.0%
Named Entity Recognition	+11.5% ↑	0.0%	-3.3% ↓	-0.7%	0.0%	0.0%
Program Execution	-9.9% ↓	-4.7% ↓	-3.5% ↓	-3.2% ↓	-5.9% ↓	0.0%
Question Answering	+10.6% ↑	-10.7% ↓	-6.8% ↓	+16.2% ↑	+8.3% ↑	**+30.0% ↑**
Question Generation	**+38.3% ↑**	+13.4% ↑	+13.1% ↑	-8.9% ↓	+7.3% ↑	+12.7% ↑
Question Rewriting	-6.5% ↓	-9.1% ↓	-6.9% ↓	-5.9% ↓	-2.4%	-4.2% ↓
Question Understanding	+10.9% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Sentence Composition	-14.5% ↓	+2.1%	+4.8% ↑	-4.8% ↓	-0.7%	+3.8% ↑
Sentiment Analysis	+7.5% ↑	0.0%	-1.2%	0.0%	0.0%	0.0%
Summarization	+23.6% ↑	+6.6% ↑	**+15.5% ↑**	-9.2% ↓	**+15.7% ↑**	-11.1% ↓
Text Categorization	+3.6% ↑	0.0%	-0.2%	0.0%	0.0%	0.0%
Text Completion	+5.5% ↑	+1.6%	+2.7%	+3.3% ↑	-0.4%	-0.1%
Text Matching	+3.5% ↑	0.0%	0.0%	-0.2%	0.0%	0.0%
Text Quality Evaluation	+6.0% ↑	0.0%	-0.4%	0.0%	0.0%	0.0%
Text to Code	+4.8% ↑	-0.7%	-1.4%	-1.0%	-4.3% ↓	0.0%
Textual Entailment	+11.0% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Title Generation	+12.8% ↑	+8.2% ↑	+14.2% ↑	+12.0% ↑	+5.6% ↑	+10.5% ↑
Word Semantics	+14.1% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Wrong Candidate Generation	+37.8% ↑	-3.1% ↓	+10.2% ↑	+10.9% ↑	-11.7% ↓	-2.0%

Table 13: The average downstream task performance gain or loss over the original prompt when paraphrased with a specific type from one of the six groups (columns) of tasks within a certain task family (rows) for **Command R+ (104B)**. We calculate the average across all paraphrase types in one of the six paraphrase groups and across all five tasks within one of the 24 categories. **Bold** indicatess the highest score per paraphrase group (column). Small changes between -3% and +3% are not colored.

Task Family	Morphology	Syntax	Lexicon	Lex.-Syn.	Discourse	Others
Answerability Classification	+4.2% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Commonsense Classification	+7.8% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Coreference Resolution	+8.0% ↑	0.0%	+2.2%	-0.7%	0.0%	0.0%
Dialogue Generation	+6.9% ↑	+4.9% ↑	+6.0% ↑	+7.7% ↑	-9.6% ↓	+3.3% ↑
Fill in The Blank	+19.7% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Information Extraction	+3.1% ↑	-0.6%	-2.4%	+1.2%	-0.5%	0.0%
Named Entity Recognition	+5.8% ↑	-1.4%	+0.6%	-0.2%	0.0%	0.0%
Program Execution	-8.5% ↓	+5.5% ↑	+7.3% ↑	+6.1% ↑	-1.1%	0.0%
Question Answering	+11.5% ↑	+5.4% ↑	+9.5% ↑	+10.7% ↑	-0.6%	**+37.7% ↑**
Question Generation	+19.4% ↑	+13.6% ↑	**+14.1% ↑**	**+15.7% ↑**	+6.5% ↑	-15.0% ↓
Question Rewriting	+9.1% ↑	-10.0% ↓	-9.2% ↓	-6.6% ↓	-7.5% ↓	-8.2% ↓
Question Understanding	+9.6% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Sentence Composition	+7.3% ↑	+10.3% ↑	-7.9% ↓	-8.5% ↓	**+11.0% ↑**	-6.8% ↓
Sentiment Analysis	+4.2% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Summarization	-16.4% ↓	+10.4% ↑	+11.2% ↑	+12.7% ↑	+9.5% ↑	+30.9% ↑
Text Categorization	+5.0% ↑	0.0%	-0.1%	0.0%	0.0%	0.0%
Text Completion	+4.9% ↑	**+15.9% ↑**	-4.3% ↓	-7.1% ↓	0.0%	0.0%
Text Matching	+4.1% ↑	0.0%	-0.3%	0.0%	0.0%	0.0%
Text Quality Evaluation	+5.3% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Text to Code	+10.1% ↑	-4.1% ↓	-3.8% ↓	-3.2% ↓	-6.5% ↓	+4.5% ↑
Textual Entailment	+13.4% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Title Generation	+10.9% ↑	+13.1% ↑	-9.0% ↓	-14.0% ↓	+3.6% ↑	+34.5% ↑
Word Semantics	+9.6% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Wrong Candidate Generation	**+22.1% ↑**	-6.6% ↓	+6.1% ↑	+12.4% ↑	-3.1% ↓	-1.7%

Table 14: The average downstream task performance gain or loss over the original prompt when paraphrased with a specific type from one of the six groups (columns) of tasks within a certain task family (rows) for **LLaMA 3 Instruct (8B)**. We calculate the average across all paraphrase types in one of the six paraphrase groups and across all five tasks within one of the 24 categories. **Bold** indicatess the highest score per paraphrase group (column). Small changes between -3% and +3% are not colored.

Task Family	Morphology	Syntax	Lexicon	Lex.-Syn.	Discourse	Others
Answerability Classification	+1.7%	0.0%	0.0%	0.0%	0.0%	0.0%
Commonsense Classification	+3.6% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Coreference Resolution	+1.1%	+2.2%	0.0%	0.0%	0.0%	0.0%
Dialogue Generation	+4.7% ↑	+2.2%	+4.1% ↑	+0.5%	+0.6%	+3.0% ↑
Fill in The Blank	+4.8% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Information Extraction	+2.2%	-8.7% ↓	-3.3% ↓	-0.8%	+0.8%	+3.4% ↑
Named Entity Recognition	+3.2% ↑	0.0%	-0.5%	0.0%	0.0%	0.0%
Program Execution	-6.0% ↓	+1.5%	-4.6% ↓	+5.1% ↑	0.0%	0.0%
Question Answering	+4.3% ↑	+5.6% ↑	-7.1% ↓	+10.3% ↑	+9.9% ↑	**+30.4%** ↑
Question Generation	+12.3% ↑	+7.8% ↑	-8.1% ↓	+6.5% ↑	+10.2% ↑	-3.0%
Question Rewriting	-4.4% ↓	-53.7% ↓	-5.7% ↓	-13.4% ↓	-4.8% ↓	-4.6% ↓
Question Understanding	+9.3% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Sentence Composition	-8.6% ↓	+2.6%	+2.8%	-0.9%	-8.7% ↓	+10.6% ↑
Sentiment Analysis	+2.2%	0.0%	0.0%	0.0%	0.0%	0.0%
Summarization	+11.7% ↑	**+9.9%** ↑	**+8.1%** ↑	+7.7% ↑	-1.8%	+3.6% ↑
Text Categorization	+1.5%	-6.3% ↓	-0.8%	**+20.0%** ↑	0.0%	0.0%
Text Completion	+2.7%	-1.1%	-4.4% ↓	-0.2%	**+15.2%** ↑	-0.2%
Text Matching	+1.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Text Quality Evaluation	+4.1% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Text to Code	+1.9%	-4.9% ↓	-2.0%	-2.4%	0.0%	-1.4%
Textual Entailment	+3.8% ↑	0.0%	0.0%	0.0%	0.0%	0.0%
Title Generation	+7.4% ↑	+4.1% ↑	+7.3% ↑	-7.5% ↓	+5.0% ↑	0.0%
Word Semantics	+2.5%	0.0%	0.0%	0.0%	0.0%	0.0%
Wrong Candidate Generation	**+26.8%** ↑	-6.6% ↓	+8.0% ↑	-4.7% ↓	+1.6%	+21.4% ↑

Table 15: The average downstream task performance gain or loss over the original prompt when paraphrased with a specific type from one of the six groups (columns) of tasks within a certain task family (rows) for **LLaMA 3 Instruct (70B)**. We calculate the average across all paraphrase types in one of the six paraphrase groups and across all five tasks within one of the 24 categories. **Bold** indicatess the highest score per paraphrase group (column). Small changes between -3% and +3% are not colored.