Hierarchical Prompting Taxonomy: A Universal Evaluation Framework for Large Language Models Aligned with Human Cognitive Principles

Devichand Budagam[1], **Ashutosh Kumar**[2], **Mahsa Khoshnoodi**[3], **Sankalp KJ**[4], **Vinija Jain**[5, 7 *], **Aman Chadha**[6, 7 †]

[1]Indian Institute of Technology Kharagpur, India [2]Rochester Institute of Technology, USA
[3]Researcher, Fatima Fellowship [4]AI Institute, University of South Carolina, USA
[5]Meta, USA [6]Amazon GenAI, USA [7]Stanford University, USA

Abstract

Assessing the effectiveness of large language models (LLMs) in performing different tasks is crucial for understanding their strengths and weaknesses. This paper presents Hierarchical Prompting Taxonomy (HPT), grounded on human cognitive principles and designed to assess LLMs by examining the cognitive demands of various tasks. The HPT utilizes the Hierarchical Prompting Framework (HPF), which structures five unique prompting strategies in a hierarchical order based on their cognitive requirement on LLMs when compared to human mental capabilities. It assesses the complexity of tasks with the Hierarchical Prompting Index (HPI), which demonstrates the cognitive competencies of LLMs across diverse datasets and offers insights into the cognitive demands that datasets place on different LLMs. This approach enables a comprehensive evaluation of an LLM's problem-solving abilities and the intricacy of a dataset, offering a standardized metric for task complexity. Extensive experiments with multiple datasets and LLMs show that HPF enhances LLM performance by 2%-63% compared to baseline performance, with GSM8k being the most cognitively complex task among reasoning and coding tasks with an average HPI of 3.20 confirming the effectiveness of HPT. To support future research and reproducibility in this domain, the implementations of HPT and HPF are available here.

Code — https://github.com/devichand579/HPT

1 Introduction

Large Language Models (LLMs) have revolutionized natural language processing (NLP), enabling significant advancements in a wide range of applications. Conventional evaluation frameworks often apply a standard prompting approach to assess different LLMs, regardless of the complexity of the task, which may result in biased and suboptimal outcomes. Moreover, applying the same prompting approach across all samples within a dataset without considering each sample's relative complexity adds to the unfair situation. To achieve a more balanced evaluation framework, it is essential to account for both the task-solving ability of LLMs and the varying cognitive complexities of the dataset samples. This limitation highlights the need for more sophisticated evaluation methods that can adapt to varying levels of task complexity. Within this study, complexity is defined as the cognitive

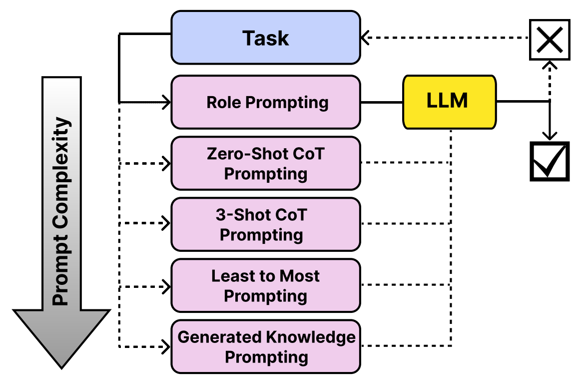

Figure 1: Hierarchical Prompting Framework includes five distinct prompting strategies, each designed for different levels of task complexity to ensure the appropriate prompt is selected for the given task. A ✓ indicates task completion, while a × signifies task incompletion.

demands associated with solving a task or the cognitive load introduced by a prompting strategy on LLMs. Henceforth, the term "complexity" will be applied solely in this context. Task complexity, within the realm of human cognition, pertains to the cognitive requirements that a task imposes, which includes the diverse levels of mental effort necessary for processing, analyzing, and synthesizing information. According to Sweller (1988), tasks become more complex as they require greater cognitive resources, engaging working memory in more demanding processes such as reasoning and problem-solving. Similarly, Anderson et al. (2014) highlights that human cognitive abilities span a continuum from basic recall to higher-order thinking, with increasing difficulty correlating to tasks that demand analysis, synthesis, and evaluation. When applied to LLMs, the complexity of prompting strategies can be systematically evaluated by mapping them onto this human cognitive hierarchy. This alignment allows for an assessment of how LLMs perform tasks that reflect varying degrees of cognitive load, thereby providing a structured framework for understanding the cognitive demands

*Work does not relate to position at Meta
†Work does not relate to position at Amazon

associated with various tasks. By imposing this cognitive complexity framework on LLMs, this paper establishes a universal evaluation method, grounded in human cognitive principles, that enables more precise comparisons of model performance across tasks with varying levels of difficulty.

This paper introduces the HPT, a set of rules that maps the human cognitive principles for assessing the complexity of different prompting strategies. It employs the HPF shown in Figure 1, a prompt selection framework that selects the prompt with the optimal cognitive load on LLM required in solving the task. HPF enhances the interaction with LLMs, and improves performance across various tasks by ensuring prompts resonate with human cognitive principles. The main contributions of this paper are as follows:

1. **Hierarchical Prompting Taxonomy (HPT)**: The paper proposes HPT, a set of rules that maps prompting strategies onto human cognitive principles, thereby enabling a universal measure of task complexity for LLMs.
2. **Hierarchical Prompting Framework (HPF)**: The HPF is a framework designed to select the most effective prompt from five distinct prompting strategies, which optimizes the cognitive load on LLMs during task completion. This framework not only facilitates a more accurate evaluation of LLMs but also enhances their performance, delivering more transparent insights.
3. **Hierarchical Prompting Index (HPI)**: HPI[1] quantitatively evaluates the task complexity of LLMs across various datasets, offering insights into the cognitive demands that each task imposes on different LLMs.

HPF can be effectively compared to an "Open Book" examination as shown in Figure 2, where questions represent tasks and textbooks serve as prompting strategies. In this analogy, the exam questions vary in complexity, from simple factual recall to intricate analytical problems, analogous to the tasks in HPT that are evaluated based on their cognitive demands. Similarly, textbooks provide structured guidance for solving these questions, much like the HPF organizes prompts in increasing levels of complexity to support LLMs. For example, a straightforward glossary lookup corresponds to a low-complexity task, while solving a multi-step analytical problem requiring synthesis of concepts represents a high-complexity task. The effort a student invests in answering a question mirrors the HPI, which measures the cognitive load placed on the LLM. Just as students perform better with structured resources like textbooks, LLMs improve with well-designed hierarchical prompting strategies, enabling them to tackle progressively complex tasks effectively.

The remainder of the paper is structured as follows: Section 2 reviews the related work on prompting and evaluation in LLMs. Section 3 details the HPT and its associated frameworks. Section 4 outlines the experimental setup, results, and ablation studies. Section 5 concludes the paper. Section 6 discusses the ethical impact of the work.

[1]HPI can be quantitatively assessed to analyze the cognitive abilities of an LLM and the cognitive demands imposed by datasets on LLMs, as both factors are interchangeably related to the complexity of tasks.

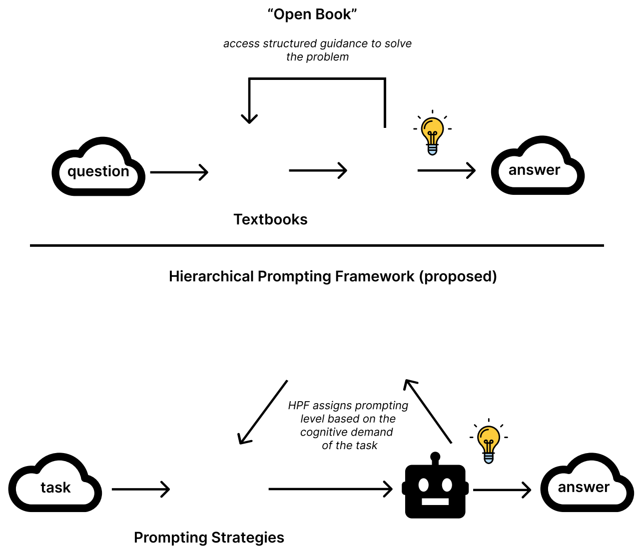

Figure 2: Analogical framework comparing the HPF with "Open Book" examination methodology. The diagram illustrates how HPF components (below) mirror traditional educational assessment elements (above), with parallel relationships between task complexity levels, resource utilization (prompts/textbooks), and performance metrics (HPI/student effort). This comparison demonstrates how LLM task complexity scales similarly to educational assessment complexity, from simple lookup tasks to complex synthesis problems

2 Related Work

The advent of LLMs has revolutionized NLP by demonstrating significant improvements in few-shot and zero-shot learning capabilities. Brown et al. (2020) introduced GPT-3, a 175 billion parameter autoregressive model, showcasing its ability to perform a wide range of tasks such as question-answering, reading comprehension, translation, and natural language inference without fine-tuning. This study highlighted the potential of very large models for in-context learning while also identifying limitations in commonsense reasoning and specific comprehension tasks. Similarly, Liu et al. (2021) surveyed prompt-based learning, emphasizing the role of prompt engineering in leveraging pre-trained models for few-shot and zero-shot adaptation to new tasks with minimal labeled data.

2.1 Prompt Engineering

Prompting plays a vital role in unlocking the full potential of LLMs. By designing specific input prompts, the LLM's responses can be guided, significantly influencing the quality and relevance of the output. Effective prompting strategies have enhanced LLM performance on tasks ranging from simple question-answering to complex reasoning and problem-solving. Recent research has explored various approaches to prompting and reasoning evaluation in LLMs. Chain-of-Thought (CoT) prompting (Wei et al. 2022b) elicits step-by-step reasoning, improving performance on complex tasks. Specializing smaller models (Fu et al. 2023) and using large models as reasoning teachers (Ho, Schmid, and Yun 2022) have demonstrated the potential for enhancing reasoning capabilities. Emergent abilities in LLMs, which appear suddenly at certain scale thresholds, have also been a topic of interest. Wei et al. (2022a) examined these abilities in few-shot prompting, discussing the underlying factors and implications for future scaling. Complementing this, Kojima et al. (2022) demonstrated that LLMs could exhibit multi-step reasoning capabilities in a zero-shot setting by simply modifying the prompt structure, thus highlighting their potential as general reasoning engines. Yao et al. (2023) introduced the Tree-of-Thoughts framework, enabling LLMs to deliberate over coherent text units and perform heuristic searches for complex reasoning tasks. This approach generalizes over chain-of-thought prompting and has shown significant performance improvements in tasks requiring planning and search, such as creative writing and problem-solving games. Kong et al. (2024) introduced role-play prompting to improve zero-shot reasoning by constructing role-immersion interactions, which implicitly trigger chain-of-thought processes and enhance performance across diverse reasoning benchmarks. Progressive-hint prompting (Zheng et al. 2023) has been proposed to conceptualize answer generation and guide LLMs toward correct responses. Metacognitive prompting (Wang and Zhao 2024) incorporates self-aware evaluations to enhance understanding abilities.

These works collectively highlight the advancements in leveraging LLMs through innovative prompting techniques, addressing their emergent abilities, reasoning capabilities, interaction strategies, robustness, and evaluation methodologies. Despite significant advancements, the current LLM research reveals several limitations, particularly in terms of prompt design, handling complex reasoning tasks, and evaluating model performance across diverse scenarios. While promising, the emergent abilities of LLMs often lack predictability and control, and the robustness of these LLMs in the face of misleading prompts remains a concern.

2.2 Prompt Optimization and Selection

The challenge of optimizing prompts for LLMs has been addressed in several key studies, each contributing unique methodologies to enhance model performance and efficiency. Shen et al. (2023) introduce PFLAT, a metric utilizing flatness regularization to quantify prompt utility, which leads to improved results in classification tasks. Do et al. (2024) propose a structured three-step methodology that contains data clustering, prompt generation, and evaluation, effectively balancing generality and specificity in prompt selection. ProTeGi (Pryzant et al. 2023) offers a non-parametric approach inspired by gradient descent, leveraging natural language "gradients" to iteratively refine prompts. Wang et al. (2024) present PromISe, which transforms prompt optimization into an explicit chain of thought, employing self-introspection and refinement techniques. Zhou et al. (2023b) proposed DYNAICL, a framework for efficient prompting that dynamically allocates in-context examples based on a meta-controller's predictions, achieving better performance-efficiency trade-offs compared to uniform example allocation.

These studies aim to automate prompt design, moving away from traditional manual trial-and-error methods while emphasizing efficiency and scalability across various tasks and models. They report significant improvements in LLMs performance, with enhancements ranging from 5% to 31% across different benchmarks. This body of work underscores the increasing importance of prompt optimization and selection in unlocking the potential of LLMs and points toward future research avenues, such as exploring theoretical foundations, integrating multiple optimization techniques, and distinguishing between task-specific and general-purpose strategies.

2.3 Evaluation Benchmarks

To facilitate the evaluation and understanding of LLM capabilities, Zhu et al. (2024) introduced PromptBench, a unified library encompassing a variety of LLMs, datasets, evaluation protocols, and adversarial prompt attacks. This modular and extensible tool aims to support collaborative research and advance the comprehension of LLM strengths and weaknesses. Further exploring reasoning capabilities, Qiao et al. (2023) categorized various prompting methods and evaluated their effectiveness across different model scales and reasoning tasks, identifying key open questions for achieving robust and generalizable reasoning. (Wang et al. 2021) introduced a multi-task Benchmark for robustness Evaluation of LLMs extends the original GLUE (Wang et al. 2018) benchmark to assess model robustness against adversarial inputs. It incorporates perturbed versions of existing GLUE tasks, such as paraphrasing, negation, and noise, to test models' abilities with challenging data. The study highlights that despite their success on clean datasets, state-of-the-art models often struggle with adversarial examples, underscoring the importance of robustness evaluations in model development.

3 Hierarchical Prompting Taxonomy

3.1 Governing Rules

Figure 3 illustrates the HPT, a taxonomy that systematically reflects human cognitive functions as outlined in Bloom (1956). Each rule embodies complex cognitive processes based on established principles from learning and psychology.

1. **Basic Recall and Reproduction**: This reflects the fundamental cognitive process of remembering and reproducing factual information without analysis or interpretation, which involves mere recognition or retrieval of knowledge from memory (Anderson et al. 2014).
2. **Understanding and Interpretation**: This corresponds to the second cognitive rule of (Bloom 1956), where individuals must not only recall information but also explain it in their own words, summarize key points or clarify the meaning of content. This rule demands an intermediate cognitive load involving information processing rather than retrieving it.
3. **Analysis and Reasoning**: This aligns with the analysis stage of (Bloom 1956), which involves higher cognitive functions such as comparison, contrast, and deep understanding of the underlying principles. It is more complex

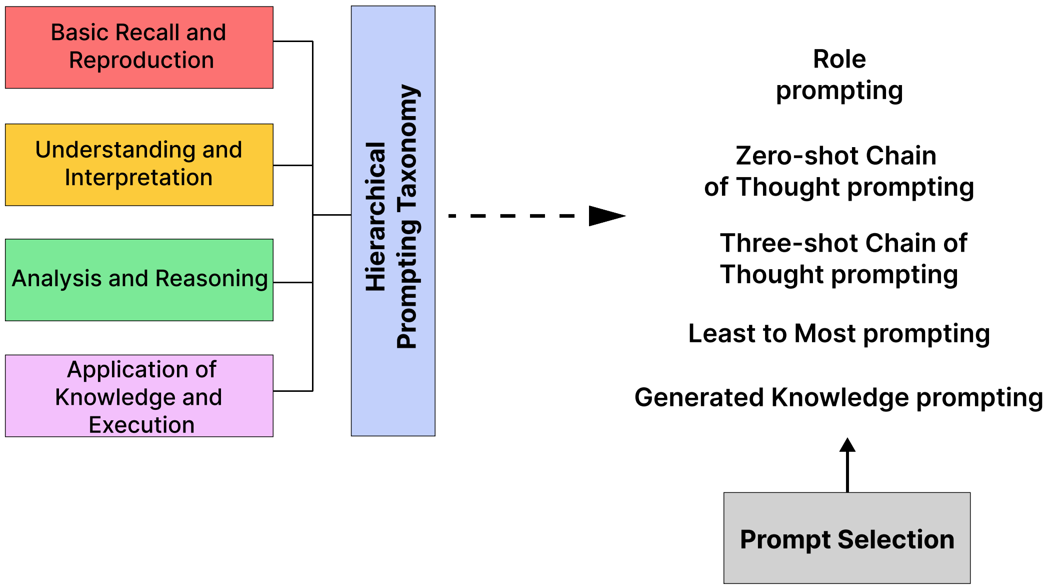

Figure 3: Hierarchical Prompting Taxonomy: A taxonomy designed to assess the complexity of prompting strategies based on the criteria: Basic Recall and Reproduction, Understanding and Interpretation, Analysis and Reasoning, and Application of Knowledge and Reasoning.

than mere understanding because it requires examining structure and identifying patterns and connections.

4. **Application of Knowledge and Execution**: This mirrors the application and evaluation stages of (Bloom 1956), where individuals must not only understand and analyze but also use knowledge to perform multi-step tasks, solve complex problems, and execute decisions. It represents the most cognitively complex tasks, which require synthesis of information and practical decision-making, highlighting the critical leap from understanding theory to executing it in practice.

In HPT, the progression from basic recall to application of knowledge reflects increasing cognitive complexity, consistent with educational and cognitive frameworks, where more advanced cognitive processes build on foundational ones, demanding deeper engagement and mental effort.

3.2 Hierarchical Prompting Framework

The HPF consists of five prompting strategies, each assigned a complexity level. These levels are determined by the degree to which the strategies are shaped by the four principles of the HPT. The complexity levels of the prompting strategies are assigned based on human assessment of their relative cognitive loads over a set of 7 different tasks, guaranteeing that the cognitive abilities of LLMs are in harmony with those of humans. This approach enables the assessment of tasks in terms of their complexity and the cognitive load they impose on both humans and LLMs by utilizing HPI. Section 4.4 examines the hierarchical structure of the HPF in conjunction with the LLM-as-a-Judge framework, validating that the cognitive demands on LLMs can be aligned with those of humans.

The set of five prompting strategies were chosen from a diverse range of existing strategies to populate the framework, guided by a human judgment policy, prioritizing comprehensiveness in cognitive demands rather than the sheer number of strategies. See Appendix A for more details. Consequently, the HPF can be replicated or expanded with other relevant prompting strategies that exhibit similar cognitive demands, making the framework adaptable. The five prompting strategies, listed from least to most complex, are as follows:

1. **Role Prompting** (Kong et al. 2024): Prompts that specify the role of the LLM in task resolution represent the lowest level of complexity within the framework, as they exhibit minimal influence from the four principles of the HPT.
2. **Zero-Shot Chain-of-Thought Prompting (Zero-CoT)** (Kojima et al. 2022): Prompts that use the phrase "Let's think step by step" without providing prior examples aim to encourage critical thinking and problem-solving. This strategy implicitly activates chain-of-thought reasoning, demonstrating moderate influence from rule 3 of the HPT, while exerting a weaker influence from the other rules.
3. **Three-Shot Chain-of-Thought Prompting (3-CoT)** (Wei et al. 2022b): Prompts that present three examples to guide the LLM's reasoning process exert a strong influence from rules 1 and 2, requiring recall and interpretation of the examples to complete the task. This strategy also demonstrates moderate influence from rule 3, increasing its cognitive demands.
4. **Least-to-Most Prompting** (Zhou et al. 2023a): A sequence of prompts that breaks the task into sub-problems exerts significant influence from all three rules 1, 2, and 3.

This approach requires recalling prior prompts, interpreting previous responses, and analyzing them to effectively solve the task, resulting in a highly cognitively demanding strategy.

5. **Generated Knowledge Prompting (GKP)** (Liu et al. 2022): Prompts that require integrating external knowledge to generate relevant information represent the most complex and cognitively demanding strategy. This approach is strongly influenced by rules 2, 3, and 4, as it involves correlating knowledge with the prompt and applying and analyzing external information, making it the most cognitively demanding within the HPT framework. In the experiments, Llama-3 8B is used to generate external knowledge.

3.3 Hierarchical Prompting Index

HPI is an evaluation metric for assessing the task complexity of LLMs over different datasets, which is influenced by the HPT rules. A lower HPI for a dataset suggests that the corresponding LLM is more adept at solving the task with fewer cognition processes. For each dataset instance, we begin with the least complex prompting strategy and progressively move through the HPF prompting strategies until the instance is resolved. The HPI corresponds to the complexity level of the prompting strategy where the LLM first tackles the instance.

Algorithm 1: HPI Metric

```
HPI_List = []
for sample i in the evaluation dataset do
    for level x in the HPF do
        if LLM resolves the task then
            HPI_List[i] = x
            break
        end if
    end for
    if LLM failed to resolve the task then
        HPI_List[i] = m + HPI_Dataset
    end if
end for
HPI = (1/n) Σ_{j=1}^{n} HPI_List[j]
```

m is the total number of levels in the HPF, and n is the total number of samples in the evaluation dataset. $\mathtt{HPI}_{Dataset}$ represents the penalty introduced into the framework by human assessments. For further details on human annotation, see Appendix A.

4 Results

4.1 Experimental Setup

Datasets

The experiments utilized a diverse set of datasets, including MMLU, GSM8k, HumanEval, BoolQ, CSQA, SamSum, and IWSLT en-fr covering areas such as reasoning, coding, mathematics, question-answering, summarization, and machine translation, to evaluate the framework's robustness and applicability. For further details on evaluation dataset sizes, see Appendix A.

Reasoning: MMLU (Hendrycks et al. 2021) includes multiple-choice questions across 57 subjects, covering areas like humanities, social sciences, physical sciences, basic mathematics, U.S. history, computer science, and law. CommonSenseQA (CSQA) (Talmor et al. 2019) contains 12,000 questions to assess commonsense reasoning.
Coding: HumanEval (Chen et al. 2021a) features 164 coding challenges, each with a function signature, docstring, body, and unit tests, designed to avoid training data overlap with LLMs.
Mathematics: Grade School Math 8K (GSM8k) (Cobbe et al. 2021) comprises 8.5K diverse math problems for multi-step reasoning, focusing on basic arithmetic and early Algebra.
Question-Answering: BoolQ (Clark et al. 2019) consists of 16,000 True/False questions based on Wikipedia passages for binary reading comprehension.
Summarization: SamSum (Gliwa et al. 2019) features 16,000 human-generated chat logs with summaries for dialogue summarization.
Machine Translation: IWSLT-2017 en-fr (IWSLT) (Cettolo et al. 2017) is a parallel corpus with thousands of English-French sentence pairs from TED Talks for translation tasks.

Large Language Models

For the evaluation, LLMs with parameter sizes ranging from 7 billion to 12 billion from top open-source models and top proprietary models were selected to determine the effectiveness of the proposed framework across varied parameter scales and architectures.
Proprietary Models: GPT-4o (OpenAI 2024), Claude 3.5 Sonnet (Anthropic 2024)
Open-Source models: Gemma 7B (Team et al. 2024a), Mistral 7B (Jiang et al. 2023), Llama-3 8B (AI@Meta 2024), Gemma-2 9B (Team et al. 2024b) and Mistral-Nemo 12B (Mistral AI and NVIDIA 2024).

Additional Evaluation Metrics

- **Coding**: The Pass@k (Chen et al. 2021b) metric measures the probability of at least one correct solution among the top k outputs, used for evaluating code generation.
- **Summarization**: ROUGE-L (Lin 2004) evaluates the longest common subsequence between generated text and reference, focusing on sequence-level similarity for summaries and translations.
- **Machine Translation**: BLEU (Papineni et al. 2002) is a precision-based metric that assesses machine-generated text quality by comparing n-grams with reference texts.

In the experiments, thresholds of 0.15, and 0.20 were established for summarization and machine translation tasks to define the conditions required for task completion at each complexity level of the HPF. These thresholds allowed for iterative refinement of HPF prompting strategies.

4.2 Results on Standard Benchmarks: MMLU, GSM8K, and Humaneval

The evaluation of HPF effectiveness as shown in Figure 4 spans three standard benchmarks: MMLU, GSM8k, and HumanEval. On the MMLU benchmark, which tests general knowledge across multiple domains, all models showed notable improvements over their baseline performance. Mistral-Nemo 12B demonstrated the most substantial MMLU enhancement (+21.8%), while Claude 3.5 Sonnet achieved a consistent improvement of 3.5%. In mathematical reasoning, assessed through GSM8k, the results revealed a correlation with the model scale. Larger models like GPT-4 and Claude 3.5 Sonnet showed modest gains (+4.4% and +1.3% respectively), while smaller models exhibited more variable performance. The HumanEval benchmark, which assesses code generation capabilities, revealed the most dramatic improvements across all models. Mistral 7B achieved an exception 62.5% improvement in HumanEval scores, followed by Mistral-Nemo 12B with an impressive 51.4% improvement, and Gemma-2 9B with a 50.8% enhancement. These findings indicate that HPF improves performance across all benchmarks for most of the LLMs, its impact is particularly pronounced in programming tasks, suggesting that the technique may be especially valuable for enhancing code-related capabilities.

Table 1 highlights the improved performance of various LLMs on MMLU, with all models showing an HPI index below three. This indicates that reasoning over most MMLU samples requires minimal cognitive effort for these models, compared to baseline multi-shot CoT methods (5 shot), which typically require more than five examples and are more cognitively demanding according to HPT. Interestingly, while Claude 3.5 Sonnet achieves the highest MMLU accuracy, GPT-4o records the best HPI score, showing that minimal cognitive effort does not necessarily equate to the best performance. The enhancement in GSM8k is relatively smaller compared to MMLU, with decreased performances for both Mistral 7B and Gemma 7B. The high HPI values for Gemma 7B and Mistral 7B indicate that none of the five prompting strategies in HPF posed significant cognitive challenges for these LLMs, highlighting a limitation of the HPF. As shown in Table 2, Claude 3.5 Sonnet achieves a perfect pass@1 of 1.00 with low HPI values, outperforming GPT-4o, which scores 0.95 but has a higher HPI. Gemma 7B struggles with the lowest pass@1 of 0.79 and the highest HPI of 3.71, indicating a need for more complex prompting strategy.

Interestingly, HPF significantly enhanced the performance of most LLMs across three benchmark datasets, even when the HPI difference was less than 1 relative to the best performing LLMs. This highlights that tailoring the prompting strategy to align with the complexity of each dataset instance can lead to substantial improvements, achieving performance levels comparable to state-of-the-art LLMs such as GPT-4o and Claude 3.5 Sonnet on these benchmarks.

4.3 Results on Other Datasets

Table 1 presents the performance of LLMs on the BoolQ and CSQA datasets. Notably, no significant insights emerge from the results, aside from GPT-4o performing unexpectedly poorly, which contrasts with its typical performance. With most LLMs achieving near-perfect scores, the BoolQ dataset appears to lack the complexity needed to serve as an effective benchmark for modern LLMs, as they perform

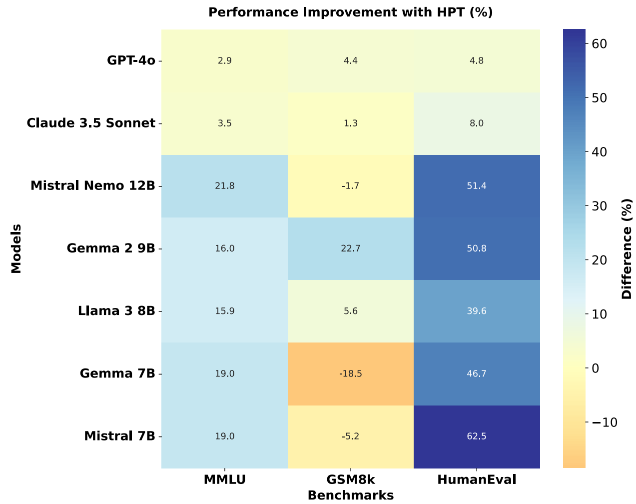

Figure 4: **Performance Comparison of HPT-based Evaluation vs. Standard Evaluation**: Performance improvements (in %) when using HPT-based evaluation compared to standard evaluation across three benchmarks: MMLU, GSM8k, and HumanEval. Positive values indicate performance gains with HPT, while negative values indicate performance decreases. The baseline standard evaluation scores are sourced from Hugging Face leaderboard and official research reports.

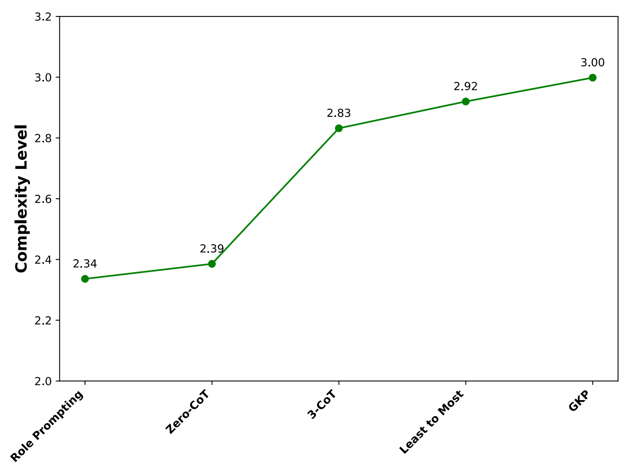

Figure 5: Hierarchy of prompting strategies with LLM-as-a-Judge framework with GPT-4o as the judge.

exceptionally well even with minimal cognitive prompting strategies. This underscores the utility of HPF in evaluating dataset complexities relative to an LLM, offering researchers valuable insights for designing more challenging and robust benchmarks.

Table 3 presents the performance of LLMs on IWSLT and SamSum datasets at varying thresholds. GPT-4o consistently achieved the highest scores across all thresholds, while most models, except Gemma 7B, performed similarly. Interestingly, Claude 3.5 Sonnet, which excelled in reasoning tasks,

DATASETS	MMLU		GSM8k		BoolQ		CSQA	
Models	**HPI**	**Accuracy**	**HPI**	**Accuracy**	**HPI**	**Accuracy**	**HPI**	**Accuracy**
GPT-4o	1.81	91.61	1.71	96.43	1.32	96.82	**1.65**	**92.54**
Claude 3.5 Sonnet	1.84	92.16	**1.35**	**97.72**	1.20	99.81	2.01	86.15
Mistral-Nemo 12B	2.45	89.75	3.01	86.80	1.75	99.87	2.06	90.17
Gemma-2 9B	2.34	87.28	2.17	91.28	1.30	98.28	1.94	88.86
Llama-3 8B	2.84	82.63	2.34	86.20	1.37	99.30	2.43	84.76
Gemma 7B	2.93	83.31	6.70	27.88	1.45	99.42	2.50	83.78
Mistral 7B	2.89	81.45	5.11	46.93	1.41	98.07	2.49	82.06

Table 1: HPI (lower is better) and accuracy of LLMs across MMLU, GSM8K, BoolQ, and CSQA datasets. Blue indicates datasets where the LLM with the best HPI does not achieve the best performance. Green indicates the LLM with the best performance over the maximum number of datasets.

DATASET	HumanEval	
Models	**HPI**	**Pass@1**
GPT-4o	2.25	0.95
Claude 3.5 Sonnet	1.04	1.00
Mistral-Nemo 12B	2.07	0.96
Gemma-2 9B	1.01	0.91
Llama-3 8B	1.03	1.00
Gemma 7B	3.71	0.79
Mistral 7B	1.10	0.93

Table 2: HPI (lower is better) and Pass@1 of LLMs on the HumanEval dataset. Blue indicates datasets where the LLM with the best HPI does not achieve the best performance. Green indicates the LLMs with the best performance over the dataset.

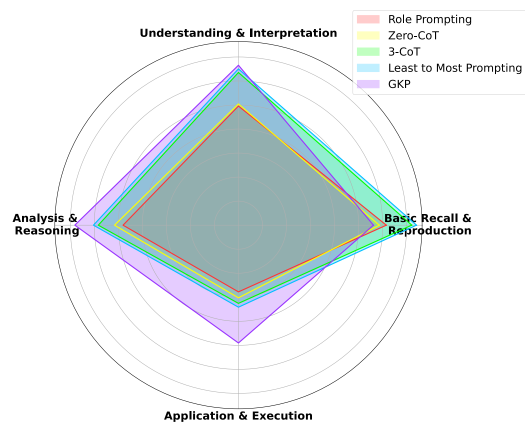

Figure 6: Scoring distribution for each of the four rules of the HPT for the prompting strategies in the HPF.

did not perform as strongly in summarization and translation tasks. The threshold selection is guided by the observed performance plateau across most LLMs as the threshold increases. For a detailed explanation of the threshold selection process, please refer to Appendix B.

4.4 Complexity Levels with LLM-as-a-Judge

This study evaluated prompting strategies by assessing how GPT-4o, as the LLM judge, replicates the hierarchical complexity levels of these strategies using a systematic scoring approach across tasks. Figure 5 shows a consistent hierarchy with less variability than human judges, indicating a strong alignment between LLM and human judgment. These results validate the proposed framework and demonstrate the correspondence between human cognitive principles and LLM behavior. Figure 6 shows the scoring distribution across the four HPT rules for each strategy. Further details related to dataset specifications and scoring method are in Appendix C.

4.5 Parallels with System 1 and System 2 Thinking

HPF align closely with the principles of System 1 and System 2 thinking from dual-process cognitive theories (Booch et al. 2021). HPT categorizes tasks and HPF structures prompts based on their cognitive complexity, mirroring how humans allocate cognitive resources. For tasks with low cognitive demands, HPF employs simple prompts that parallel System 1 thinking. These tasks, like fact recall or basic classification, require minimal reasoning, allowing the LLM to respond quickly and efficiently without extensive computation. For instance, asking an LLM to *"identify the capital of a country"* is analogous to a person retrieving a familiar fact using System 1.

In contrast, tasks with high cognitive demands involve prompts that guide the LLM through complex reasoning, abstraction, or multi-step problem-solving—analogous to System 2 thinking. Examples include generating logical arguments or solving intricate problems, where deliberate and resource-intensive processes are necessary. Just as System 2 engages when a problem exceeds the capacity of System 1, higher levels of HPF are invoked for tasks requiring deeper analysis.

HPF explicitly measures this transition with HPI, assessing the cognitive load required for each task. By tailoring prompting strategies to task complexity, HPF optimizes LLM

DATASETS	IWSLT				SamSum			
	HPI		BLEU		HPI		ROUGE-L	
Models	**0.15**	**0.20**	**0.15**	**0.20**	**0.15**	**0.20**	**0.15**	**0.20**
GPT-4o	**2.66**	**3.08**	**0.32**	**0.32**	**1.11**	**1.21**	**0.30**	**0.29**
Claude 3.5 Sonnet	4.63	4.87	0.20	0.20	1.25	1.60	0.23	0.23
Mistral-Nemo 12B	2.87	3.40	0.27	0.27	1.19	1.47	0.23	0.24
Gemma-2 9B	4.40	4.75	0.21	0.20	1.30	1.86	0.22	0.22
Llama-3 8B	3.40	3.92	0.24	0.23	1.30	1.72	0.22	0.22
Gemma 7B	5.39	5.84	0.08	0.09	3.31	5.03	0.11	0.10
Mistral 7B	3.52	4.14	0.22	0.22	1.26	1.68	0.21	0.22

Table 3: HPI (lower is better), BLEU score for IWSLT, and ROUGE-L score for SamSum, of LLMs with thresholds.

performance, much like humans adaptively switch between System 1 and System 2 based on the situation. This parallel highlights how HPT bridges computational strategies with human-like cognitive models, enabling more nuanced task evaluation and resource allocation.

4.6 Adaptive HPF

The Adaptive HPF automates the selection of the optimal complexity level in the HPF using a *prompt-selector*, Llama-3 8B in a zero-shot setting, bypassing iterative steps. Figure 7 shows that Adaptive HPF yields higher HPI but lower evaluation scores than the standard HPF. This suggests that Adaptive HPF struggles to select the optimal complexity level, likely due to hallucinations by the *prompt-selector* when choosing the prompting strategy. For more results and ablation studies, see Appendix E.

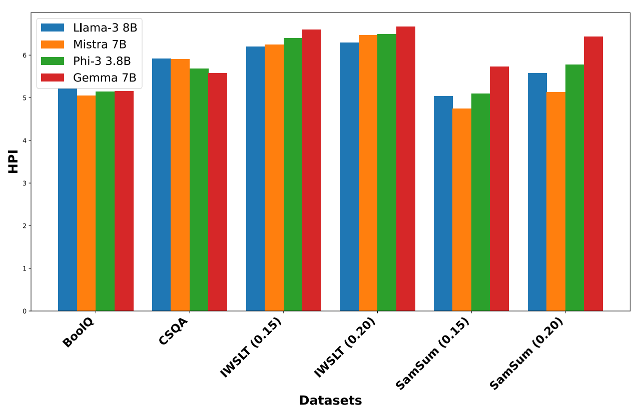

Figure 7: HPI of datasets for LLMs in Adaptive HPF.

5 Conclusion

The HPT provides a strong and efficient approach for assessing LLMs by focusing solely on the cognitive demands of different tasks. The results emphasize that the HPF is effective in evaluating diverse datasets, using the most cognitively effective prompting strategies tailored to task complexity, which results in improved LLM performance across multiple datasets. This method not only offers an in-depth insight into LLM's problem-solving abilities but also suggests that dynamically choosing suitable prompting strategies can enhance LLM performance, setting the stage for future improvements in LLM evaluation methods. This study paves the way for formulating and designing evaluation methods grounded in human cognitive principles, aligning them with the problem-solving capabilities of LLMs. Additionally, it facilitates the development of more robust benchmarks and in-context learning methodologies to effectively assess LLM performance across various tasks.

6 Ethical Statement

The $\mathtt{HPI}_{Dataset}$ assigned by experts to the datasets: MMLU, GSM8k, Humaneval, BoolQ, CSQA, IWSLT, and SamSum may introduce bias into the comparative analysis. This potential bias stems from the subjective nature of expert scoring, which can be influenced by individual experience and perspective. Despite this, the datasets utilized in this study are publicly available and widely recognized in the research community, thereby minimizing the risk of unanticipated ethical issues. Nevertheless, it is crucial to acknowledge the possibility of scoring bias to ensure transparency and integrity in the analysis.

References

AI@Meta. 2024. Llama 3 Model Card. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

Anderson, L.; Krathwohl, D.; Cruikshank, K.; Airasian, P.; Raths, J.; Pintrich, P.; Mayer, R.; and Wittrock, M. 2014. *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's*. Always learning. Pearson. ISBN 9781292042848.

Anthropic. 2024. Claude 3.5 Sonnet. https://www.anthropic.com/claude-3-5-sonnet. Accessed: 2024-09-16.

Bloom, B. 1956. *Taxonomy of Educational Objectives: The Classification of Educational Goals*. Number v. 1 in Taxonomy of Educational Objectives: The Classification of Educational Goals. Longmans, Green. ISBN 9780679302094.

Booch, G.; Fabiano, F.; Horesh, L.; Kate, K.; Lenchner, J.; Linck, N.; Loreggia, A.; Murgesan, K.; Mattei, N.; Rossi, F.; et al. 2021. Thinking fast and slow in AI. In *Proceedings of* *the AAAI Conference on Artificial Intelligence*, volume 35, 15042–15046.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Cettolo, M.; Federico, M.; Bentivogli, L.; Niehues, J.; Stüker, S.; Sudoh, K.; Yoshino, K.; and Federmann, C. 2017. Overview of the IWSLT 2017 Evaluation Campaign. In *Proceedings of the 14th International Conference on Spoken Language Translation*, 2–14. Tokyo, Japan: International Workshop on Spoken Language Translation.

Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; de Oliveira Pinto, H. P.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; Ray, A.; Puri, R.; Krueger, G.; Petrov, M.; Khlaaf, H.; Sastry, G.; Mishkin, P.; Chan, B.; Gray, S.; Ryder, N.; Pavlov, M.; Power, A.; Kaiser, L.; Bavarian, M.; Winter, C.; Tillet, P.; Such, F. P.; Cummings, D.; Plappert, M.; Chantzis, F.; Barnes, E.; Herbert-Voss, A.; Guss, W. H.; Nichol, A.; Paino, A.; Tezak, N.; Tang, J.; Babuschkin, I.; Balaji, S.; Jain, S.; Saunders, W.; Hesse, C.; Carr, A. N.; Leike, J.; Achiam, J.; Misra, V.; Morikawa, E.; Radford, A.; Knight, M.; Brundage, M.; Murati, M.; Mayer, K.; Welinder, P.; McGrew, B.; Amodei, D.; McCandlish, S.; Sutskever, I.; and Zaremba, W. 2021a. Evaluating Large Language Models Trained on Code. arXiv:2107.03374.

Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; de Oliveira Pinto, H. P.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; Ray, A.; Puri, R.; Krueger, G.; Petrov, M.; Khlaaf, H.; Sastry, G.; Mishkin, P.; Chan, B.; Gray, S.; Ryder, N.; Pavlov, M.; Power, A.; Kaiser, L.; Bavarian, M.; Winter, C.; Tillet, P.; Such, F. P.; Cummings, D.; Plappert, M.; Chantzis, F.; Barnes, E.; Herbert-Voss, A.; Guss, W. H.; Nichol, A.; Paino, A.; Tezak, N.; Tang, J.; Babuschkin, I.; Balaji, S.; Jain, S.; Saunders, W.; Hesse, C.; Carr, A. N.; Leike, J.; Achiam, J.; Misra, V.; Morikawa, E.; Radford, A.; Knight, M.; Brundage, M.; Murati, M.; Mayer, K.; Welinder, P.; McGrew, B.; Amodei, D.; McCandlish, S.; Sutskever, I.; and Zaremba, W. 2021b. Evaluating Large Language Models Trained on Code. arXiv:2107.03374.

Clark, C.; Lee, K.; Chang, M.-W.; Kwiatkowski, T.; Collins, M.; and Toutanova, K. 2019. BoolQ: Exploring the Surprising Difficulty of Natural Yes/No Questions. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2924–2936. Minneapolis, Minnesota: Association for Computational Linguistics.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*.

Do, V.-T.; Hoang, V.-K.; Nguyen, D.-H.; Sabahi, S.; Yang, J.; Hotta, H.; Nguyen, M.-T.; and Le, H. 2024. Automatic Prompt Selection for Large Language Models. arXiv:2404.02717.

Fu, Y.; Peng, H.; Ou, L.; Sabharwal, A.; and Khot, T. 2023. Specializing Smaller Language Models towards Multi-Step Reasoning. In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 10421–10430. PMLR.

Gliwa, B.; Mochol, I.; Biesek, M.; and Wawer, A. 2019. SAMSum Corpus: A Human-annotated Dialogue Dataset for Abstractive Summarization. In Wang, L.; Cheung, J. C. K.; Carenini, G.; and Liu, F., eds., *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, 70–79. Hong Kong, China: Association for Computational Linguistics.

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2021. Measuring Massive Multitask Language Understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Ho, N.; Schmid, L.; and Yun, S.-Y. 2022. Large language models are reasoning teachers. *arXiv preprint arXiv:2212.10071*.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; de las Casas, D.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; Lavaud, L. R.; Lachaux, M.-A.; Stock, P.; Scao, T. L.; Lavril, T.; Wang, T.; Lacroix, T.; and Sayed, W. E. 2023. Mistral 7B. arXiv:2310.06825.

Kojima, T.; Gu, S. S.; Reid, M.; Matsuo, Y.; and Iwasawa, Y. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213.

Kong, A.; Zhao, S.; Chen, H.; Li, Q.; Qin, Y.; Sun, R.; Zhou, X.; Wang, E.; and Dong, X. 2024. Better Zero-Shot Reasoning with Role-Play Prompting. arXiv:2308.07702.

Lin, C.-Y. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, 74–81. Barcelona, Spain: Association for Computational Linguistics.

Liu, J.; Liu, A.; Lu, X.; Welleck, S.; West, P.; Le Bras, R.; Choi, Y.; and Hajishirzi, H. 2022. Generated Knowledge Prompting for Commonsense Reasoning. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 3154–3169. Dublin, Ireland: Association for Computational Linguistics.

Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; and Neubig, G. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Mistral AI and NVIDIA. 2024. Mistral NeMo 12B. https://mistral.ai/news/mistral-nemo/. Accessed: 2024-09-16.

OpenAI. 2024. GPT-4o. https://openai.com/gpt-4. Accessed: 2024-09-16.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, 311–318. USA: Association for Computational Linguistics.

Pryzant, R.; Iter, D.; Li, J.; Lee, Y. T.; Zhu, C.; and Zeng, M. 2023. Automatic prompt optimization with" gradient descent" and beam search. *arXiv preprint arXiv:2305.03495*.

Qiao, S.; Ou, Y.; Zhang, N.; Chen, X.; Yao, Y.; Deng, S.; Tan, C.; Huang, F.; and Chen, H. 2023. Reasoning with Language Model Prompting: A Survey. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 5368–5393. Toronto, Canada: Association for Computational Linguistics.

Shen, L.; Tan, W.; Zheng, B.; and Khashabi, D. 2023. Flatness-aware prompt selection improves accuracy and sample efficiency. *arXiv preprint arXiv:2305.10713*.

Sweller, J. 1988. Cognitive load during problem solving: Effects on learning. *Cognitive Science*, 12(2): 257–285.

Talmor, A.; Herzig, J.; Lourie, N.; and Berant, J. 2019. CommonsenseQA: A Question Answering Challenge Targeting Commonsense Knowledge. arXiv:1811.00937.

Team, G.; Mesnard, T.; Hardin, C.; Dadashi, R.; Bhupatiraju, S.; Pathak, S.; Sifre, L.; Rivière, M.; Kale, M. S.; Love, J.; Tafti, P.; Hussenot, L.; Sessa, P. G.; Chowdhery, A.; Roberts, A.; Barua, A.; Botev, A.; Castro-Ros, A.; Slone, A.; Héliou, A.; Tacchetti, A.; Bulanova, A.; Paterson, A.; Tsai, B.; Shahriari, B.; Lan, C. L.; Choquette-Choo, C. A.; Crepy, C.; Cer, D.; Ippolito, D.; Reid, D.; Buchatskaya, E.; Ni, E.; Noland, E.; Yan, G.; Tucker, G.; Muraru, G.-C.; Rozhdestvenskiy, G.; Michalewski, H.; Tenney, I.; Grishchenko, I.; Austin, J.; Keeling, J.; Labanowski, J.; Lespiau, J.-B.; Stanway, J.; Brennan, J.; Chen, J.; Ferret, J.; Chiu, J.; Mao-Jones, J.; Lee, K.; Yu, K.; Millican, K.; Sjoesund, L. L.; Lee, L.; Dixon, L.; Reid, M.; Mikuła, M.; Wirth, M.; Sharman, M.; Chinaev, N.; Thain, N.; Bachem, O.; Chang, O.; Wahltinez, O.; Bailey, P.; Michel, P.; Yotov, P.; Chaabouni, R.; Comanescu, R.; Jana, R.; Anil, R.; McIlroy, R.; Liu, R.; Mullins, R.; Smith, S. L.; Borgeaud, S.; Girgin, S.; Douglas, S.; Pandya, S.; Shakeri, S.; De, S.; Klimenko, T.; Hennigan, T.; Feinberg, V.; Stokowiec, W.; hui Chen, Y.; Ahmed, Z.; Gong, Z.; Warkentin, T.; Peran, L.; Giang, M.; Farabet, C.; Vinyals, O.; Dean, J.; Kavukcuoglu, K.; Hassabis, D.; Ghahramani, Z.; Eck, D.; Barral, J.; Pereira, F.; Collins, E.; Joulin, A.; Fiedel, N.; Senter, E.; Andreev, A.; and Kenealy, K. 2024a. Gemma: Open Models Based on Gemini Research and Technology. arXiv:2403.08295.

Team, G.; Riviere, M.; Pathak, S.; Sessa, P. G.; Hardin, C.; Bhupatiraju, S.; Hussenot, L.; Mesnard, T.; Shahriari, B.; Ramé, A.; et al. 2024b. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In Linzen, T.; Chrupała, G.; and Alishahi, A., eds., *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 353–355. Brussels, Belgium: Association for Computational Linguistics.

Wang, B.; Xu, C.; Wang, S.; Gan, Z.; Cheng, Y.; Gao, J.; Awadallah, A. H.; and Li, B. 2021. Adversarial GLUE: A Multi-Task Benchmark for Robustness Evaluation of Language Models. In *Advances in Neural Information Processing Systems*.

Wang, M.; Xu, N.; Zhao, J.; Luo, Y.; and Mao, W. 2024. PromISe: Releasing the Capabilities of LLMs with Prompt Introspective Search. In Calzolari, N.; Kan, M.-Y.; Hoste, V.; Lenci, A.; Sakti, S.; and Xue, N., eds., *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 13120–13130. Torino, Italia: ELRA and ICCL.

Wang, Y.; and Zhao, Y. 2024. Metacognitive Prompting Improves Understanding in Large Language Models. arXiv:2308.05342.

Wei, J.; Tay, Y.; Bommasani, R.; Raffel, C.; Zoph, B.; Borgeaud, S.; Yogatama, D.; Bosma, M.; Zhou, D.; Metzler, D.; Chi, E. H.; Hashimoto, T.; Vinyals, O.; Liang, P.; Dean, J.; and Fedus, W. 2022a. Emergent Abilities of Large Language Models. *Transactions on Machine Learning Research*. Survey Certification.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022b. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Yao, S.; Yu, D.; Zhao, J.; Shafran, I.; Griffiths, T. L.; Cao, Y.; and Narasimhan, K. R. 2023. Tree of Thoughts: Deliberate Problem Solving with Large Language Models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Zheng, C.; Liu, Z.; Xie, E.; Li, Z.; and Li, Y. 2023. Progressive-Hint Prompting Improves Reasoning in Large Language Models. arXiv:2304.09797.

Zhou, D.; Schärli, N.; Hou, L.; Wei, J.; Scales, N.; Wang, X.; Schuurmans, D.; Cui, C.; Bousquet, O.; Le, Q. V.; and Chi, E. H. 2023a. Least-to-Most Prompting Enables Complex Reasoning in Large Language Models. In *The Eleventh International Conference on Learning Representations*.

Zhou, W.; Jiang, Y. E.; Cotterell, R.; and Sachan, M. 2023b. Efficient Prompting via Dynamic In-Context Learning. arXiv:2305.11170.

Zhu, K.; Zhao, Q.; Chen, H.; Wang, J.; and Xie, X. 2024. PromptBench: A Unified Library for Evaluation of Large Language Models. arXiv:2312.07910.

A Human Annotation and Judgement Policy

A.1 Human Annotation Policy

$\texttt{HPI}_{Dataset}$ is introduced to penalize the HPI of tasks or samples unsolvable by the LLM, aligning the framework more closely with human cognitive demands and enhancing its comprehensiveness. We implemented a rigorous human annotation process to ensure the quality of $\texttt{HPI}_{Dataset}$ scored by human experts for the datasets. Human annotators were tasked with calculating the HPI for each sample in a given dataset. The HPI quantifies the cognitive demands imposed on human expert proficiency in completing a task, based on the HPT, where higher values indicate greater cognitive demands. Each sample was scored on a scale from *1* (lowest complexity level) to *5* (highest complexity level) for the following criteria:

1. **Basic Understanding and Reproduction**: This criterion evaluates the annotator's ability to comprehend and accurately reproduce the content.
2. **Understanding and Interpretation**: This criterion assesses the annotator's depth of understanding and the ability to interpret the information correctly.
3. **Analysis and Reasoning**: This criterion measures the annotator's ability to analyze the information and apply logical reasoning.
4. **Application of Knowledge and Execution**: This criterion evaluates the annotator's practical application of knowledge and the execution of tasks based on the relevant knowledge.

Higher scores for the four rules signify a stronger influence of the respective rules, indicating that completing the task requires greater cognitive effort. The $\texttt{HPI}_{Dataset}$ for each dataset, as shown in Table 4, was calculated by taking the mean of the values from these four criteria, acknowledging the challenge of estimating or computing the individual weights of the influence of each rule.

The `Representative Set Size` in Table 4 refers to the subset of the dataset evaluated by human annotators, ensuring that the assessment reflects the overall task. Human annotation, while time-consuming and costly, provides a gold standard for calibrating the evaluation process of this paper. Selecting 5% of the dataset as the representative set size balances quality assessment and feasibility, capturing the dataset's diversity and ensuring that human annotations encompass a broad range of cases without needing to annotate every sample.

Dataset	Evaluation Set Size	Representative Set Size	$\texttt{HPI}_{Dataset}$
MMLU	14500	725	3.03
GSM8k	1320	66	2.14
Humaneval	160	8	4.68
BoolQ	3270	162	1.71
CSQA	1221	60	2.52
IWSLT	890	45	1.92
SamSum	819	40	2.23

Table 4: **$\texttt{HPI}_{Dataset}$ scores across datasets evaluated by human annotators**. The table lists the evaluation set size, representative set size, and $\texttt{HPI}_{Dataset}$ for various datasets. $\texttt{HPI}_{Dataset}$ scores provide a measure of task complexity relative to human annotators.

A.2 Human Judgement Policy

To populate the HPF with relevant prompting strategies across a wide range of strategies, human annotators who adhered to the annotation policy for assessing $\texttt{HPI}_{Dataset}$ were instructed to follow a judgment policy for a predefined set of prompting strategies. They were instructed to evaluate the influence of the four rules of the HPT on solving the annotated tasks using each prompting strategy, rating their influence as High (H), Moderate (M), or Low (L). It's important to note that a high rating on rule 4 has a greater influence than a high rating on rule 3, and similarly for the other two rules. Considering the rating as shown in Table 5 and varying influences of these rules, five prompting strategies that prioritize comprehensive coverage of cognitive demands while ensuring the set optimally widens the variation across complexity levels were selected for populating the HPF.

Prompting Strategy	Rule 1	Rule 2	Rule 3	Rule 4
Role Prompting	L	L	L	L
Emotion Prompting	L	L	M	L
Zero-shot CoT	L	L	M	L
Meta Prompting	M	H	M	L
Three-shot CoT	H	H	M	L
Five-shot CoT	H	H	H	L
Chain-of-Verification	H	H	H	H
Least-to-Most Prompting	H	H	H	L
Self-Consistency	H	H	H	M
GKP	L	H	H	H

Table 5: Human judgment of influence of the rules of taxonomy on different prompting strategies in solving the tasks of the representative set. The ratings are provided based on a voting system involving all human annotators. Green represents the prompting strategies selected for populating the complexity levels of the HPF.

B Threshold Selection for Summarization and Translation Tasks

In addition to the 0.15 and 0.20 thresholds presented in the main paper, extended evaluations were conducted on the IWSLT and SamSum datasets using thresholds of 0.25 and 0.30 with GPT-4o, Mistral-Nemo 12B, and Llama-3 to assess the impact of varying thresholds on LLM performance.

B.1 Summarization Task

In the summarization task, increasing the threshold evaluates an LLM's ability to condense content while retaining key information. Higher thresholds like 0.25 and 0.30 reveal the trade-offs between conciseness and informativeness. However, as shown in Figure 8, there was no significant improvement in ROUGE-L, except for a slight increase with GPT-4o. The experiments showed a sharp rise in HPI, reflecting the increased task complexity. These results suggest that LLM performance has plateaued, with no further gains at higher thresholds. This validates the use of 0.15 and 0.20 thresholds in the main paper are sufficient for optimal LLM performance.

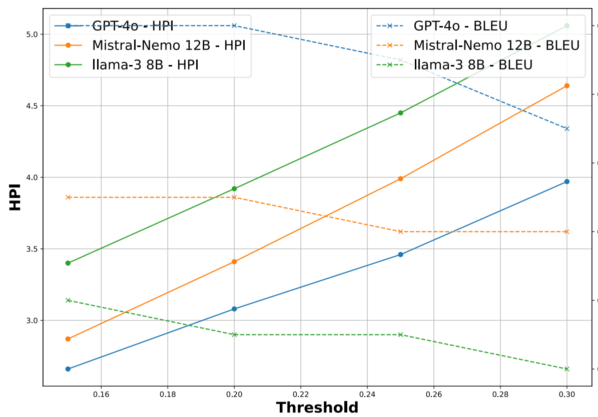

Figure 8: Comparison of HPI and ROUGE-L score across different threshold values in summarization task.

B.2 Translation Task

In machine translation, higher thresholds (0.25 and 0.30) impose stricter evaluations, assessing how well models capture the nuances of the source text. Lower thresholds (0.15 and 0.20) focus on general adequacy, while higher ones test performance under more challenging conditions. As shown in Figure 9, no BLEU improvements were observed across any LLMs, with models either reaching saturation or showing decreased performance alongside a rapid rise in HPI. This validates the selection of 0.15 and 0.20 thresholds in the main paper as sufficient for optimal LLM performance.

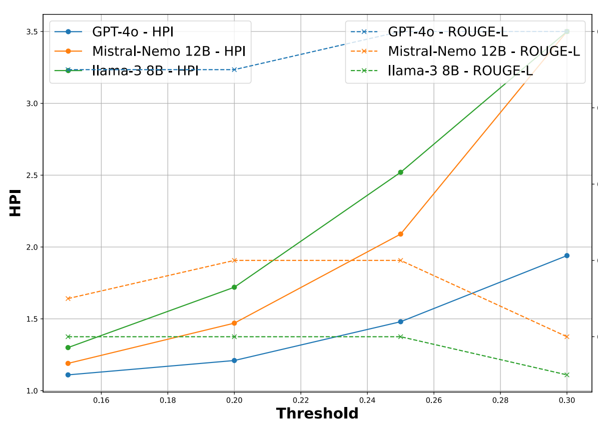

Figure 9: Comparison of HPI and BLEU score across different threshold values in the translation task.

C LLM-as-a-Judge

C.1 Scoring Prompt Template

The system prompt is designed to guide the LLM judge in evaluating different prompting strategies based on four specific criteria: Basic Recall and Reproduction, Understanding and Interpretation, Analysis and Reasoning, and Application of Knowledge and Execution. Each criterion is scored on a scale of 1-5. The evaluation uses GPT-4o as a judge, with the following system prompt:

```
You are a judge evaluating different prompting strategies and you need to score these prompting strategies based on pre-defined criteria. Different prompting strategies leverage varied amounts of intelligence from the model to achieve the required answer. So, assign the scores very carefully based on your analysis of the prompt and its effect on your intelligence to achieve the given answer as well as the number of multi-step prompts which increases the complexity of execution.

score1: Basic Recall and Reproduction
Definition: The need of the model to remember and reproduce factual information without interpretation or analysis to answer the prompt
Range: 1-5

score2: Understanding and Interpretation
Definition: The need of the model to comprehend and explain the meaning of information, summarizing or clarifying content to answer the prompt
Range: 1-5

score3: Analysis and Reasoning
Definition: The need for the model to break down complex information, understand relationships, and solve problems using logical reasoning to answer the prompt
Range: 1-5

score4: Application of Knowledge and Execution
Definition: The need for the model to apply knowledge in practical situations, execute multi-step processes, and solve complex tasks to answer the prompt
Range: 1-5
```

C.2 Hybrid Dataset

The hybrid dataset is composed of 1106 samples uniformly distributed over seven different task-specific datasets, covering a wide range of language understanding and generation tasks. This diversity allows for a comprehensive evaluation of the prompting strategies across various problem types. The evaluation uses a hybrid dataset composed of samples from various task-specific datasets and each dataset contributes specific types of tasks:

1. MMLU (Massive Multitask Language Understanding)
2. HumanEval (Code Generation and Completion)
3. GSM8K (Grade School Math 8K)
4. BoolQ (Boolean Questions)
5. CSQA (Commonsense Question Answering)
6. IWSLT (International Workshop on Spoken Language Translation)
7. SamSum (Dialogue Summarization)

C.3 Scoring Method

For each prompting strategy (Role Prompting, Zero-shot CoT, Three-shot CoT, Least to Most Prompting, Generated Knowledge Prompting), the system:

1. Applies the prompting strategy to each sample in the hybrid dataset
2. Generates an answer using GPT-4o
3. Presents the prompt, generated answer, and correct answer to the LLM judge
4. Collects scores for each of the four criteria and the system calculates average scores for each criterion across all tasks and datasets.

This study ensured that both the human judge and the LLM judge utilized the same scoring methodology to eliminate any potential bias in the comparison.

D Limitations

Human Annotation Constraints: A limitation of this study is the reliance on human evaluation for inducing the $\texttt{HPI}_{Dataset}$ penalty into the HPF. While this study assessed 5% of the datasets, expanding coverage would offer a more comprehensive analysis. However, due to constraints in human resources for manual annotation, we could not include a larger portion. Future work could address this by increasing manpower or automating parts of the evaluation process.

HPF Optimization: The effectiveness of the HPF heavily relies on the quality of the prompts used at each level of the taxonomy. Crafting high-quality prompts that accurately reflect the subtleties of each level demands considerable expertise and repeated refinement. This study only investigated a limited set of prompting strategies within the HPF, indicating a need for further research into creating diverse structural frameworks and incorporating additional prompting strategies.

Zero-shot Prompt Selection: HPF dynamically determines the optimal cognitive complexity level by iterating through the framework's levels, which leads to increased inference time. While this study investigated Adaptive HPF for zero-shot prompt selection, it faced considerable hallucinations. Future research should focus on automating HPF using fine-tuning or reinforcement learning-based approaches to select the appropriate complexity level without manual iteration. This strategy would optimize inference time and improve overall performance.

E Adaptive HPF

E.1 HPI for Adaptive HPF

The *prompt-selector* can dynamically select the most suitable prompting strategy for a given task's complexity from the HPF's hierarchy of complexity levels. To determine the most effective prompting strategy to complete the task, the *prompt-selector* was given a maximum number of iterations equivalent to the number of levels in the manual HPF. The score for *i*th iteration is *i* + *x*, where *x* is the complexity level by the *prompt-selector*. If the LLM fails to complete the task after all iterations, it is assigned a penalty, $\texttt{HPI}_{Dataset}$. x represents the level of the HPF selected by *prompt-selector*

Algorithm 2: Prompt-Selector for Adaptive HPF

```
HPI_List = []
for sample j in the evaluation dataset do
    solved = False
    for iteration i = 1 to m do
        Select prompting strategy at level x
        if LLM completes the task at iteration i then
            HPI_List[j] = x + i
            solved = True
            break
        end if
    end for
    if solved = False then
        HPI_List[j] = m + HPI_Dataset
    end if
end for
```

$\texttt{HPI}_{\texttt{Adaptive}} = \frac{1}{n}\sum_{j=1}^{n} \texttt{HPI\_List}[j]$

at ith iteration at which the task is addressed, m represents the total number of levels in the HPF, and n denotes the total number of samples in the evaluation set.

E.2 Hallucination in Adaptive HPF

Hallucinations in *prompt-selector* refer to instances where the LLM generates incorrect or misleading prompting levels or nonsensical information that is not supported by the HPF. These hallucinations can occur across various tasks, including question answering, multiple-choice questions, translation, and summarization.

For the BoolQ task, the *prompt-selector* initially struggles, indicated by the iterations where it reaches Level 4 with hallucinations (represented by '...'). However, by the fourth iteration, Llama-3 8B manages to answer correctly at Level 2. For the CSQA task, *prompt-selector* exhibits hallucinations initially, shown by Level 4 and Level 0 (not included in HPF) responses. Eventually, it corrects itself by the third iteration, providing the correct answer at Level 2. For the IWSLT task, *prompt-selector* demonstrates a consistent pattern of hallucinations across multiple iterations. Even though Llama-3 8B attempts the translation at Level 2 multiple times, it ultimately fails to provide a correct translation, indicating a persistent hallucination. For the SamSum task, *prompt-selector* shows initial hallucinations in the first three iterations (Level 4). However, by the fourth and fifth iterations, the *prompt-selector* starts producing lower levels. Finally, Llama-3 8B achieves the correct answer at Level 2 in the last iteration .

The results in Table 6 and Table 7 indicate that the *prompt-selector* exhibits hallucinations in selecting complexity levels across various tasks and iterations resulting in higher HPI for Adaptive HPF, with performance varying significantly. While the LLM can eventually produce correct answers, as seen in the BoolQ and SamSum tasks, it often requires multiple attempts and may still fail in tasks like IWSLT translation.

Model	BoolQ	CSQA	IWSLT (0.15)	IWSLT (0.20)	SamSum (0.15)	SamSum (0.20)
Llama-3 8B	5.2173	5.9136	**6.2006**	**6.2841**	5.0316	5.5756
Mistra 7B	**5.0483**	5.9073	6.2478	6.4604	**4.7423**	**5.1336**
Phi-3 3.8B	5.1386	**5.6793**	6.3955	6.4936	5.0961	5.7778
Gemma 7B	5.1514	5.5771	6.5947	6.6605	5.7229	6.4347

Table 6: HPI (lower is better) of LLMs across datasets (with thresholds) for Adaptive HPF.

Dataset	Metric	Threshold	Llama-3 8B	Phi-3 3.8B	Mistral 7B	Gemma 7B
BoolQ	Accuracy	-	0.88577	0.91115	**0.91752**	0.91166
CSQA	Accuracy	-	0.59451	0.68019	0.60111	**0.68549**
IWSLT	BLEU	0.15	**0.21140**	0.15557	0.20000	0.08447
		0.2	**0.21146**	0.15354	0.20568	0.07730
SamSum	ROUGE-1	0.15	0.24407	0.20586	**0.26910**	0.16023
		0.2	0.24981	0.21580	**0.28335**	0.16179

Table 7: Performance scores of LLMs across datasetsfor Adaptive HPF.

E.3 Prompt Template for Prompt-Selector

The *prompt-selector* in adaptive HPF selects the prompting level based on the task complexity to address the task. Llama-3 8B serves as the *prompt-selector* in the experiments. The prompt template was meticulously designed to ensure maximum clarity, aiming to reduce hallucinations and select the most effective prompting strategy.

Prompt Template: Choose the most effective prompting strategy among five available strategies for the task. Begin with the lowest indexed strategy and progress to higher indexed strategies if the earlier ones are not effective. For a given task, the prompting strategies are:

- **Role Prompting:** Defines a role for the model in solving the task.
- **Zero-shot Chain of Thought prompting:** Stimulates reasoning and problem-solving by including the phrase 'Let's think step by step' without offering previous examples related to the task.
- **Three-shot Chain of Thought prompting:** Offers three examples related to the task to guide the model's reasoning process.
- **Least-to-most prompting:** Uses a sequential method to derive essential insights from the task to solve it.
- **Generated Knowledge Prompting:** Integration and application of external knowledge to accomplish the task. The external knowledge is generated using some other model based on the task.

Select only the index (do not provide the name) of the most effective prompting strategy.

F Computational Budget

All evaluation experiments and ablation studies were conducted on V100 GPUs (16GB and 32GB variants), utilizing a total of around 9,000 computation hours, this equates to approximately 1.125 petaflop-hours of computational resources.

G Large Language Models Used for Evaluation

The HPF supports leading open source and proprietary LLMs and includes mechanisms for optimizing performance through advanced quantization techniques. The experiments were conducted on the following instruction-tuned LLMs, and the model description and licenses are discussed in Table 8.

The LLMs were loaded in 4-bit precision format, with a maximum generation limit of 1024 tokens per run to ensure concise outputs. The temperature was set to 0.6 to control prediction randomness, while top-p sampling (p=0.9) enabled the exploration of diverse continuations. Additionally, a repetition penalty was applied to discourage the generation of repeated phrases, promoting coherent and varied text output.

H Prompt Templates

H.1 Level 1: Role Prompting

Role prompting represents the most basic interaction with an LLM, assigning it a specific role or task without additional context or examples. This approach relies solely on the initial instruction to guide responses. For instance, asking the LLM to *"act as a translator"* prompts it to translate text based on its training data. While straightforward, this method may lack depth, resulting in less accurate or nuanced outputs. Table 9 shows the prompt templates used for role prompting across all datasets in the experiments.

H.2 Level 2: Zero-shot Chain-of-Thought Prompting

Zero-shot Chain-of-Thought (CoT) prompting enhances basic role prompting by requiring the LLM to generate a reasoning process for a task, despite not being explicitly trained on similar examples. This method encourages the LLM to break down the problem and solve it step-by-step using its internal knowledge, improving response quality through logical progression and coherence. Table 10 displays the prompt

Model	License Type	Usage Restrictions
GPT-4o	Proprietary	Commercial use requires paid API access, subject to OpenAI's terms of service
Claude 3.5 Sonnet	Proprietary	Commercial use requires paid API access, subject to Anthropic's terms of service
Mistral-Nemo 12B	Proprietary	Usage likely restricted to authorized partners or specific use cases
Gemma-2 9B	Research License	Non-commercial use only, research purposes
Llama-3 8B	Research License	Specific restrictions may apply, typically for non-commercial research use
Mistral 7B	Open-source	Broad use allowed, must include original license and notices
Gemma 7B	Open-source/Research	Depending on the license, may have non-commercial restrictions or broad use allowed
Phi-3 3.8B	Open-source	Broad use allowed, must include original license and notices

Table 8: License information for LLMs used in the experiments.

Dataset	Prompt
BoolQ	`Based on the passage: ‘‘`**`passage`**`”, answer True/False to the question: ‘‘`**`question`**`” as an Omniscient person.`
CSQA	`Choose the answer: ‘‘`**`question`**`”,A. ‘‘`**`option 1`**`”,B. ‘‘`**`option 2`**`”,C. ‘‘`**`option 3`**`”,D. ‘‘`**`option 4`**`”,E. ‘‘`**`option 5`**`” as a critical thinker.`
IWSLT	`Translate ‘‘`**`english text`**`” to french as a Translator.`
SamSum	`Summarize the Dialogue: ‘‘`**`dialogue`**`” as a Summarizer.`
GSM8k	`Based on the question: "`**`question`**`”, calculate the numerical answer to the question as an expert mathematician.`
HumanEval	`Complete the given code based on the mentioned constraints: "`**`code`**`” as an expert programmer.`
MMLU	`Choose the answer: ‘‘`**`question`**`”,A. ‘‘`**`option 1`**`”,B. ‘‘`**`option 2`**`”,C. ‘‘`**`option 3`**`”,D. ‘‘`**`option 4`**`” as a critical thinker.`

Table 9: Prompt templates of different datasets for Role Prompting.

templates used for Zero-CoT across all datasets in the experiments.

H.3 Level 3: Three-Shot Chain-of-Thought Prompting

Three-shot Chain-of-Thought (CoT) prompting builds on the zero-shot approach by providing the LLM with three task examples, including the reasoning steps used to reach the solution. These examples help the LLM grasp the required structure and logic, enabling it to better replicate the problem-solving process and produce more accurate, contextually relevant responses. Table 11 shows the prompt templates used for 3-CoT across all datasets in the experiments.

H.4 Level 4: Least-to-Most Prompting

Least-to-most prompting is an advanced technique that gradually increases prompt complexity, starting with simpler tasks and progressing to more complex challenges. This method allows the LLM to build confidence and leverage insights from easier prompts to tackle harder ones, enhancing its ability to generalize from straightforward examples to intricate scenarios. Table 12 displays the prompt templates used for Least-to-Most Prompting across all datasets in the experiments.

H.5 Level 5: Generated Knowledge Prompting

Generated Knowledge prompting is one of the most complex techniques in HPF, where the LLM not only addresses the task but also integrates relevant additional information to enhance its response. This method prompts another LLM to produce auxiliary knowledge, creating a richer context for understanding and solving the problem. By leveraging self-generated insights, the LLM can deliver more detailed, accurate, and nuanced answers. Table 13 shows the prompt templates used for Generated Knowledge Prompting across all datasets in the experiments.

Dataset	**Prompt**
BoolQ	`Based on the passage: ‘‘`**`passage`**`”, answer True/False to the question: ‘‘`**`question`**`”. Let’s think step by step.`
CSQA	`Choose the answer: A. ‘‘`**`option 1`**`”,B. ‘‘`**`option 2`**`”,C. ‘‘`**`option 3`**`”,D. ‘‘`**`option 4`**`”,E. ‘‘`**`option 5`**`”. Let’s think step by step.`
IWSLT	`Translate ‘‘`**`english text`**`” to french. Let’s think step by step.`
SamSum	`Summarize the Dialogue: ‘‘`**`dialogue`**`”. Let’s think step by step.`
GSM8k	`Based on the question: "`**`question`**`", calculate the numerical answer to the question. Let’s think step by step.`
HumanEval	`Complete the given code based on the mentioned constraints: "`**`code`**`". Let’s think step by step.`
MMLU	`Choose the answer: ‘‘`**`question`**`”,A. ‘‘`**`option 1`**`”,B. ‘‘`**`option 2`**`”,C. ‘‘`**`option 3`**`”,D. ‘‘`**`option 4`**`”. Let’s think step by step.`

Table 10: Prompt templates of different datasets for Zero-shot Chain-of-Thought Prompting.

Dataset	Prompt
BoolQ	Based on the passage: "**passage1**", answer True/False to the question: "**question1**". Answer: "**answer1**". Explanation: "**explaination1**". Based on the passage: "**passage2**", Answer True/False to the question: "**question2**". Answer: "**answer2**". Explanation: "**explaination2**". Based on the passage: "**passage3**", Answer True/False to the question: "**question3**". Answer: "**answer3**". Explanation: "**explaination3**". Based on the passage: "**passage**", Answer True/False to the question: "**question**".
CSQA	Choose the answer: "**question1**",A. "**option1-1**",B. "**option2-1**",C. "**option3-1**",D. "**option4-1**",E. "**option5-1**", Answer: "**answer1**", Explanation: "**explaination1**". Choose the answer: "**question2**", A. "**option1-2**",B. "**option2-2**",C. "**option3-2**",D. "**option4-2**",E. "**option5-2**", Answer: "**answer2**", Explanation: "**explainatio n2**". Choose the answer: "**question3**", A. "**option1-3**",B. "**option2-3**",C. "**option3-3**",D. "**option4-3**",E. "**option5-3**", Answer: "**answer3**", Explanation: "**explaination3**". Choose the answer: "**question**", ‘‘**question**",A. ‘‘**option 1**",B. ‘‘**option 2**",C. ‘‘**option 3**",D. ‘‘**option 4**",E. ‘‘**option 5**".
IWSLT	Translate "**english text1**" to French. French: "**french text1**". Translate "**english text2**" to French. French: "**french text2**". Translate "**english text3**" to French. French: "**french text3**". Translate "**english text**" to French.
SamSum	Summarize the Dialogue: "**dialogue1**". Summary: "**summary1**". Summarize the Dialogue: "**dialogue2**". Summary: "**summary2**". Summarize the Dialogue: "**dialogue3**". Summary: "**summary3**". Summarize the Dialogue: "**dialogue**".
GSM8k	Based on the question:"**gsm8k_question1**", calculate the numerical answer to the question.Answer: "**gsm8k_ans1**".Based on the question:"**gsm8k_question2**", calculate the numerical answer to the question.Answer: "**gsm8k_ans2**". Based on the question:"**gsm8k_question3**", calculate the numerical answer to the question.Answer: "**gsm8k_ans3**". Based on the question: "**question**", calculate the numerical answer to the question.
HumanEval	Complete the given code based on the mentioned constraints: "**humaneval_code1**", Code: "**humaneval_sol1**". Complete the given code based on the mentioned constraints: "**humaneval_code2**", Code: "**humaneval_sol1**".Complete the given "**code**" based on the mentioned constraints: "**humaneval_code3**", Code: "**humaneval_sol3**".
MMLU	Choose the answer for the question: "**mmlu_ques1**" A. [AND, NOT] B. [NOT, OR] C. [AND, OR] D. [NAND] Answer: C. Explanation: "**mmlu_exp1**". Choose the answer for the question "**mmlu_ques2**" A. The defendant’s statement was involuntary. B. The defendant’s statement was voluntary. C. The defendant was not in custody when the statement was made. D. The statement was not made in response to a known police interrogation. Answer: A, Explanation: "**mmlu_exp2**". Choose the answer for the question: "**mmlu_ques3**" . A. Wrong, Wrong. B. Wrong, Not wrong C. Not wrong, Wrong D. Not wrong, Not wrong. Answer: B Explanation: "**mmlu_exp3**". Choose the answer."**question**" ‘‘**question**",A. ‘‘**option 1**",B. ‘‘**option 2**",C. ‘‘**option 3**",D. ‘‘**option 4**".

Table 11: Prompt templates of different datasets for Three-Shot Chain-of-Thought Prompting.

Dataset	Prompt
BoolQ	**prompt 1**: Summarize the main points of this passage: "**passage**". **prompt 2**: Analyze this question to identify its key components: "**question**". **prompt 3**: Find the part of the passage that relates to this question: "**question**", Passage: "**passage**". **prompt 4**: Based on the passage, what is the answer to this question: "**question**", Relevant Information: "**previous response**".
CSQA	**prompt 1**: Analyze this question: "**question**". **prompt 2**: Elaborate about each option for the question: "**question**", options: A. ‘‘**option 1**",B. ‘‘**option 2**",C. ‘‘**option 3**",D. ‘‘**option 4**",E. ‘‘**option 5**". **prompt 3**: Based on the analysis: "**previous response**", discard wrong answers among the options: A. ‘‘**option 1**",B. ‘‘**option 2**",C. ‘‘**option 3**",D. ‘‘**option 4**",E. ‘‘**option 5**". **prompt 4**: Choose the correct answer from the options: A. ‘‘**option 1**",B. ‘‘**option 2**",C. ‘‘**option 3**",D. ‘‘**option 4**",E. ‘‘**option 5**".
IWSLT	**prompt 1**: What is the main idea or theme of this text? "**english text**". **prompt 2**: Identify and list the key phrases or terms in this text: "**english text**". **prompt 3**: Translate the following key phrases into French: "**previous response**". **prompt 4**: Translate "**english text**" into French, incorporating the translations of the key phrases: "**previous response**".
SamSum	**prompt 1**: List the main points or key ideas present in this dialogue: "**dialogue**". **prompt 2**: Elaborate on the following key points, providing additional details or context: "**previous response**". **prompt 3**: Using the listed key points and their elaborations, draft a concise summary of this text: "**dialogue**". **prompt 4**: Refine this draft summary to make it more concise and coherent, ensuring it captures the essence of the text: "**dialogue**".
GSM8k	Analyze the question: "**question**". Break the question into sub-problems: "**question**". Calculate answers for the subproblems of the question: "**pred**". Calculate the numerical answer to this question: "**question**" based on the previous calculations: "**pred**"
HumanEval	Analyze the code: "**code**". Break the question into sub-problems: "**code**". Complete code for the subproblems of the code: "**pred**". Complete the code based on the mentioned constraints: "**code**" based on the previous calculations: "**pred**"
MMLU	Analyze the question: "**question**". Elaborate about each option for the question: "**question**", options: A. "**option 1**" B. "**option 2**" C. "**option 3**" D. "**option 4**". Based on the analysis: "**question**", Discard wrong answers among the options: A. "**option 1**" B. "**option 2**" C. "**option 3**" D. "**option 4**".

Table 12: Prompt templates of different datasets for Least-to-Most Prompting.

Dataset	Prompt
BoolQ	**inference prompt**: Based on the passage:"**passage**", answer True/False to the question: '**question**' using knowledge of the passage:"**knowledge**" **knowledge generation prompt**: Generate Knowledge about the passage: "**passage**".
CSQA	**inference prompt**: Choose the answer:"**question**", A. ''**option 1**",B. ''**option 2**",C. ''**option 3**",D. ''**option 4**",E. ''**option 5**" using knowledge of the question:"**knowledge**" **knowledge generation prompt**: Generate Knowledge about the question: "**question**".
IWSLT	**inference prompt**: Translate "**english text**": to French using definitions of the keywords:"**knowledge**" **knowledge generation prompt**: Generate definitions in french of each word in the text: "**english text**".
SamSum	**inference prompt**: Summarize the Dialogue: "**dialogue**" using the interpretation of the dialogue:"**knowledge**" **knowledge generation prompt**: Generate interpretation about the dialogue: "**dialogue**".
GSM8k	Based on the question:"**question**", calculate the numerical answer to the question using an interpretation of the question:"**pred**"
HumanEval	Complete the code based on the mentioned constraints:"**code**" using knowledge of the constraints:"**pred**"
MMLU	Choose the answer. "**question**", options: A. "**option 1**" B. "**option 2**" C. "**option 3**" D. "**option 4**" using knowledge of the question:"**pred**"

Table 13: Prompt templates of different datasets for Generated Knowledge Prompting.