PROMPTING

SR2024 - LLM group1

PROMPTING

- Crafting inputs or questions to guide AI systems to generate relevant and accurate responses.
- An effective prompt is clear, concise, and focused, ensuring that the AI understands the user's intent and delivers the desired output.

- Enhances model performance
- Handling ambiguity
- Improving efficiency
- Facilitating generalization
- Ensuring relevance of outputs

Best Practices

- Knowing the model’s strengths and weaknesses
- Being specific as possible
- Utilizing contextual prompts
- Providing examples
- Experimenting with prompts and personas

Multilingual Model Challenges

- Ambiguity (multiple meanings, be explicit as possible)
- Language specific grammar and structure (simpler sentences)
- Cultural context (include background or context when necessary)
- Low resource languages (data augmentation, prompt in both low and high resource)
- Mixing languages (clear switching)

→ To summarize, the challenges with multilingual models can be alleviated by providing explicit prompts, translations, and clarifications.

Zero-shot Prompting

- Prompt won’t contain any examples or demonstrations
- **Advantages:** requires minimal effort, no need to gather examples, more time and resource efficient
- **Limitations:** not as accurate as specifically fine-tuned models for particular tasks, struggles with complex tasks, performance dependent on its training

One-shot Prompting

- Prompt contains one example or demonstration
- **Advantages:** requires minimal effort but gives some context, better results than zero-shot for more specific and complex tasks
- **Limitations:** not as accurate as specifically fine-tuned models, may not capture complex patterns from a single example, dependent on the quality of one example

Few-shot Prompting

- Prompt contains a few examples or demonstrations
- **Advantages:** richer context, so improved accuracy and relevance, useful for complex tasks, more accurate than zero-shot and one-shot
- **Limitations:** requires more effort, performance can still be lower than fully fine-tuned models, longer prompts so harder to construct and use

Chain-of-thought Prompting

- Allows models to decompose multi-step problems into intermediate steps
- Since it decomposes into steps, it is easier to debug
- Can be readily elicited in sufficiently large off-the-shelf language models simply by including examples

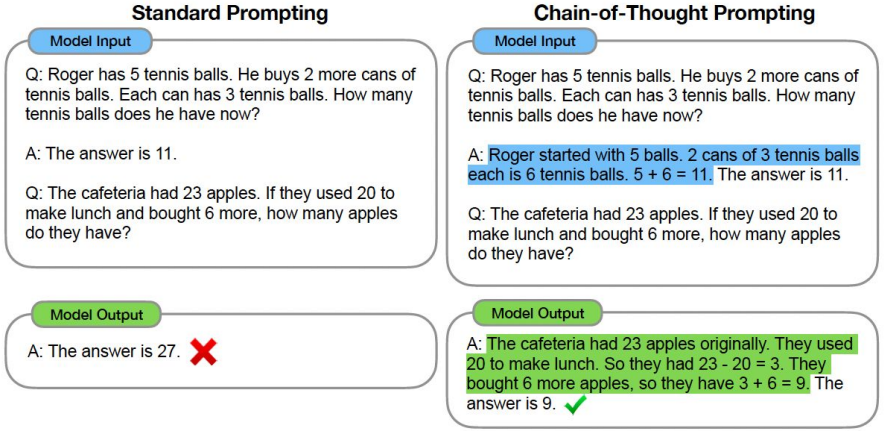
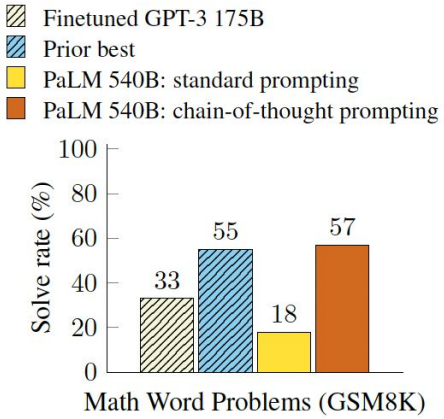


Figure 1: Chain-of-thought prompting enables large language models to tackle complex arithmetic, commonsense, and symbolic reasoning tasks. Chain-of-thought reasoning processes are highlighted.

Chain-of-thought Prompting

- Can be combined with few-shot prompting to get better results on more complex tasks
- Shown to be effective on:
  - arithmetic
  - commonsense
  - symbolic reasoning tasks
- Only yields performance gains when use ~100B parameters



Self-consistency

- Self-Consistency Prompting is a prompt engineering method that enhances the reasoning capabilities of Large Language Models (LLMs) by generating multiple outputs and selecting the most consistent answer among them.

- This approach leverages the idea that complex problems can be approached in various ways. By evaluating various paths and outputs, you can identify the most reliable and accurate solution, leading to improved performance.

Self-consistency

Benefits:

- Reduces incorrect predictions due to randomness in generation.
- Ensures more reliable and stable outputs.
- Encourages models to explore multiple ways to solve a problem, enhancing problem-solving depth.

Self-consistency

Optimal Strategies:

- Majority voting: tasks with clear-cut answers (math problems…)
- Weighted scoring: tasks where quality and coherence are essential (text generation…)
- Sampling diverse reasoning paths: complex tasks requiring reasoning (physics…)
- Explore different levels of granularity: verify before each step

Self-consistency

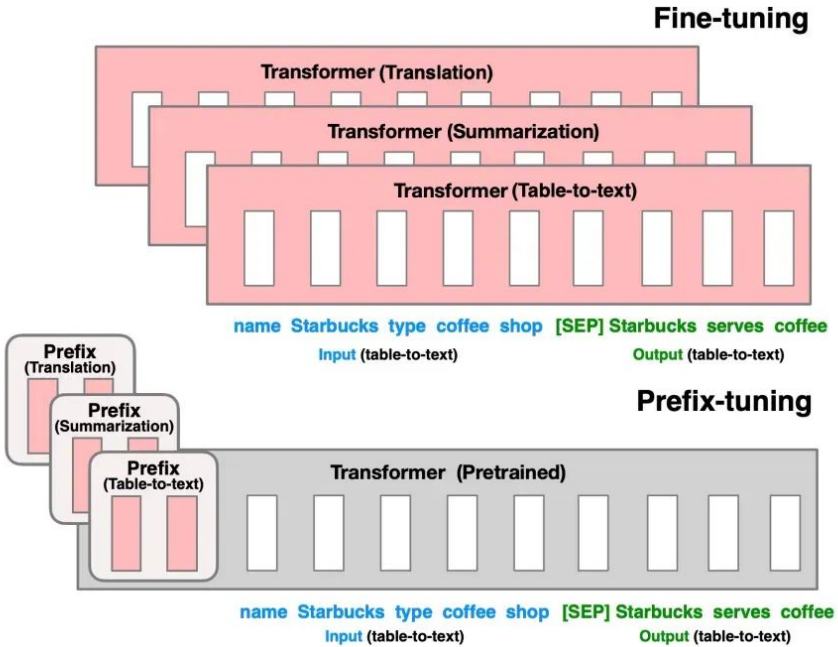
Combining with other techniques:

- Chain-of-thought: Chain-of-thought prompting generates explicit reasoning steps to solve a task. Self-consistency can then be applied to sample multiple reasoning paths and select the most consistent, logically sound answer.

- Few-shot prompting: Few-shot prompting provides the model with a few example tasks along with their solutions to guide it. By combining self-consistency, the model generates multiple solutions for the same input based on these examples and picks the most consistent output.

Prefix-Tuning

- Prefix-tuning is a lightweight fine-tuning technique where a trainable prefix is added to the input, allowing the model to adapt to new tasks without modifying its core parameters.
- This method addressed the challenges of efficiently adapting LMs to specific tasks while maintaining their generalisation capabilities and minimising the storage requirements for task-specific parameters.



Prefix-Tuning

Advantages:

- Parameter-Efficiency: Less computational and memory-intensive.
- Task-Specific Adaptation: Quickly adapts to different tasks.
- Better Generalization: Freezing the pretrained parameters enhances generalization to new datasets and topics.

Challenges:

- Architecture Dependency: Success depends on the model's structure.
- Optimization Complexity: Tuning can be challenging for complex tasks.

Automated Prompt Generation

- Automated Prompt Generation (APE) is an approach that automatically creates prompts through model feedback loops, optimizing the LLM's performance on specific tasks without human intervention.
- APE treats the instruction as a "program" that is optimized by exploring a set of instruction candidates generated by an LLM, aiming to maximize a predefined scoring function. The effectiveness of the selected instruction is then assessed based on the zero-shot performance of a different LLM that follows the given instruction.

Automated Prompt Generation

Effectiveness:

- Accuracy: AutoPrompt helps fine-tune the model's prompts for higher task accuracy by identifying more effective input patterns.
- Generalizability: By automatically adjusting prompts, AutoPrompt enables the model to generalize better to unseen tasks. This reduces the need for manual prompt engineering across different contexts.

Automated Prompt Generation

Key Success Factors:

- Model Architecture: A well-structured model makes prompts more efficient.
- Training Data Quality: High quality data leads to better prompts.
- Task Specificity: If the task definition is clear, APE will lead to better results.

Dynamic Prompting

- Dynamic/Contextual prompting is a method that adjusts prompts based on real-time context or user interactions. This allows the model to adapt to respond to changes in conversation flow, which ensures more accurate and relevant responses.
- It enhances the model’s ability to track and understand the evolving context in conversations, allowing for more coherent multi-turn exchanges.

Dynamic Prompting

Trade-offs Between Static and Dynamic Prompting:

- Static Prompting: Lower computational cost and simpler to implement, but it is less adaptable in scenarios where context shifts throughout the conversation.
- Dynamic Prompting: More computationally intensive due to continuous context updates, but it better maintains relevance in real-time applications and evolving dialogues.

Evaluation Metrics

- Accuracy
- Perplexity
- F1 Score
- ROUGE/BLEU
- Human Evaluation

Evaluation Metrics

Standardizing Metrics:

- Cross-Model Comparisons: Metrics should be consistent across models to allow for fair benchmarking and reliable comparisons of performance across different LLM architectures.
- Task-Specific Metrics: Metrics should be tailored to the specific task, such as using ROUGE for summarization or F1 for classification, ensuring evaluations focus on the most relevant aspects of performance.

Applications of Chain of Thought Prompting

Legal Reasoning:

- Case Analysis
- Legal Decision Support
- Drafting Legal Documents

Medical Diagnostics:

- Symptom Analysis
- Treatment Planning
- Medical Literature Review

Applications of Chain of Thought Prompting

Scientific Research:

- Data Analysis
- Literature Review
- Summarizing and Analysing Research Papers

Accuracy and Reliability of CoT

- **Define Clear Objectives:** When prompting the LLM, make sure that the chain of thought task is clear and specific.
- **Promote Transparency:** Ensure that in the input chain of thought prompt the reasoning and logical steps are clear.
- **Required Data:** Be sure that LLM has the necessary knowledge to do your domain specific task.

Retrieval Augmented Prompting

- It is a type of RAG.
- In RAG, the outputs are enhanced with additional information. In RAP, the prompts are augmented.

Applications:

- Roleplay AI
- Personal Assistants
- Customer Service Bots

Retrieval Augmented Prompting

Key Challenges:

- Quality of Retrieved Information
- Consistency and Bias
- Contextual Understanding
- Interpretability and Transparency

Role-Playing / Persona-Based Prompting

- It is kind of “Act as like …” pattern.
- LLM gets a role and acts according to that role.

Possible Applications:

- Medical Diagnostic and Treatment
- Psychologist
- Tutoring and Mentoring
- Travel Guide
- Language Learning

Role-Playing / Persona-Based Prompting

Key Challenges:

- Consistency
- Reliability
- Biases and Discriminations
- Complexity of Roles
- Model Limitations

Overcome Challenges:

- Clear and detailed prompts
- Limit the scope of the roleplaying
- Regularly remind the model its role and the context

Mitigate Ethical Concerns

- **Language Choices:** Please ensure that the language used in prompts avoids any words or phrases with negative historical connotations.
- **Inclusivity:** Be sure that the language you use in prompts is inclusive and do not reinforce stereotypes or biases.
- **Transparency:** Make sure that you know the logical process behind the output. Prompting techniques such as Chain Of Thought may help users to gain insight into the reasoning of the LLM.