Prompt Generation Specification

Investigation Prompt Engineering in Large Language Models

Zero-shot

Purpose: Test baseline ability to solve problems without examples or reasoning guidance.

Structure:

“Solve the following math problem. Return ONLY the final line in the form: Answer: <number>.

Problem: {QUESTION}

Answer:”

One-shot

Purpose: Show the model a single example before the test problem, to enforce format.

Structure:

“Solve the following math problems. Each answer must end with: Answer: <number>.

Example:
Q: {EXAMPLE_QUESTION}
A: {EXAMPLE_REASONING} Answer: {EXAMPLE_NUMBER}

Now solve:
Q: {QUESTION}
A:”

Few-shot

Purpose: Provide multiple examples (1–3) to strengthen pattern imitation.

Structure:

“Solve the following math problems. Each answer must end with: Answer: <number>.

Examples:
Q: {EXAMPLE_1_QUESTION}
A: {EXAMPLE_1_REASONING} Answer: {EXAMPLE_1_NUMBER}

Q: {EXAMPLE_2_QUESTION}
A: {EXAMPLE_2_REASONING} Answer: {EXAMPLE_2_NUMBER}

...

Now solve:
Q: {QUESTION}
A:”

Chain-of-Thought (CoT)

Purpose: Force explicit step-by-step reasoning before the final numeric answer.

Structure:

“Solve the following problem step by step, showing your reasoning briefly.
Then give the final line as: Answer: <number>.

Problem: {QUESTION}

Solution:”

Generated-Knowledge

Purpose: Encourage the model to recall relevant “facts” or intermediate values first, then reason.

Structure:

“First list any facts or helpful intermediate values that might be needed.
Then solve the problem step by step.
Finally, return the last line as: Answer: <number>.

Problem: {QUESTION}

Solution:”

Common Rules Across All Prompts

- Answer format: Always ends with a line “Answer: <number>”.
- Parsing: Evaluator extracts the last number after “Answer:”. If missing, falls back to the last numeric token.
- Shots: Drawn from a fixed example pool from the train dataset (to avoid leakage).