

## Prompt Engineering / Meta Prompts

Generate diverse, realistic input-output examples for a prompt to demonstrate desired behavior.

Difficulty: Intermediate

Model: GPT-4 / Claude / Gemini

Use Case: Prompt Improvement, Few-Shot Learning, Model Fine-Tuning Prep

Updated: May 2026

Why This Prompt Exists

Few-shot examples are the most powerful way to improve prompt reliability — but writing good examples is hard.

You get:

- examples that are too similar (don't cover edge cases)
- examples that confuse the model (inconsistent formatting)
- examples that don't match real-world inputs
- no examples for failure cases (what NOT to do)
- spending hours writing examples instead of iterating

But good examples have patterns:

- typical case: what most inputs look like
- edge case: unusual but possible input
- difficult case: input that might confuse the model
- negative case: what NOT to do (anti-example)
- boundary case: input just on the edge of the rules

Without good examples, few-shot fails.

This prompt generates diverse, realistic examples for any task.

The Prompt

Assume the role of a prompt engineer who generates few-shot examples.

Your task is to create diverse input-output examples for a given task.

Generate:

#### 1. TASK SUMMARY

- What the model should do
- Input format
- Output format

#### 2. EXAMPLE SET (5-10 examples covering diversity)

- Include: Input → Output
- Cover these categories:
  - \* Typical case (2-3 examples)
  - \* Edge case (1-2 examples)
  - \* Difficult/corner case (1 example)
  - \* Negative/anti-example (what NOT to do – 1 example)
  - \* Boundary case (just inside/outside rules – 1 example)

#### 3. FORMAT FOR PROMPT INSERTION

- How to paste these examples into your prompt
- XML tags or markdown formatting

#### 4. EXAMPLE DIVERSITY CHECK

- Are inputs sufficiently different?
- Do outputs demonstrate the full range of possible responses?
- Are edge cases covered?

## 5. GAPS TO FILL YOURSELF

- What real-world examples you should add (not generated)

### INPUTS:

#### Task description:

[E.G., "Classify customer support emails as urgent, normal, or low priority"]

#### Input format:

[E.G., "Email subject and body text"]

#### Output format:

[E.G., "One word: URGENT / NORMAL / LOW"]

#### Example real inputs (optional, for realism):

[PASTE 2-3 REAL EXAMPLES IF AVAILABLE]

#### Model:

[GPT-4 / CLAUDE / GEMINI]

### RULES:

- Examples must be realistic (could actually occur)
- Outputs must be correct (no errors in the examples)
- Vary input length, structure, and complexity

- Include at least one edge case that tests the task's boundaries
- Anti-examples should show common mistakes (e.g., classifying based on wrong signal)
- Use consistent formatting across all examples

## How To Use It

- Run this before writing any few-shot prompt — start with good examples.
- Use real inputs from your data when available (paste them into the “example real inputs” field).
- Include at least one anti-example — models learn from negative examples too.
- Test your examples on the model before deploying — bad examples hurt performance.
- Update examples as you discover new edge cases in production.

## Example Input

### **Task description:**

“Extract action items from meeting notes”

### **Input format:**

“Raw meeting transcript or notes”

### **Output format:**

“Bulleted list: [Action item] — Owner: [person] — Due: [date if mentioned]”

### **Example real inputs (optional):**

“Sarah: we need to update the pricing page by Friday. John: I’ll handle the SEO review. No deadline given.”

## Why It Works

Most few-shot examples are written by the prompt author — which means they reflect what the author thinks is important, not what the model needs.

This framework improves outcomes by forcing:

- task summary (clarity on what you're doing)
- example diversity (typical, edge, difficult, negative, boundary)
- format consistency (model needs predictable structure)
- realism check (examples must be plausible)
- gap identification (what you still need to add)

Great few-shot generation doesn't just give examples — it gives the right examples for robust performance.

## **Build Better AI Systems**

Subscribe for advanced prompt engineering, AI coding tools, debugging frameworks, and practical strategies for developers and engineers.

Carefully engineered prompts for people doing real work.

### **Share this:**

- [Share on Facebook \(Opens in new window\) Facebook](#)
- [Share on X \(Opens in new window\) X](#)

See also [Prompt Compression Engine](#)