# UNIT 4

**Prompt Engineering for LLMs:**

Deep dive into the nuances of prompting LLMs for optimal and precise outputs. Crafting, refining, and testing prompts for various application needs

# ZERO SHOT and FEW SHOT

- The terms zero-shot and few-shot refer to the number of examples that the model is given before being asked to perform a task.
- **Definition:** Zero-shot learning refers to a model's ability to generalize and **perform tasks it hasn't explicitly been trained on.** It leverages large-scale pre-trained models and prompts them in natural language to infer new tasks.
- **Mechanism**: The model is trained on a broad dataset containing diverse types of knowledge. At inference, the model relies on its pre-trained knowledge and context provided in the query (prompt) to perform the task. No specific task-related examples are given during inference.

**Example:** Suppose you ask a language model, "Classify the following review as positive or negative: 'The product was amazing!'" Here, the model has not seen explicit examples of sentiment classification for this query but can infer the task using its general understanding.

**Strengths:** Useful for handling scenarios where labeled data is unavailable or impractical to collect.

**Challenges:** May struggle with domain-specific or highly nuanced tasks.

- In the few-shot case, the model is given a few examples of the task in the prompt , the text that the model takes as input to determine what output it should generate.
- **Definition:** Few-shot learning involves training or **fine-tuning a model** with only a small number of task-specific examples (e.g., a few-shot prompt). It allows the model to learn quickly from minimal data.
- **Mechanism**: Alongside the task description, the model is provided with a few labeled examples (also known as in-context learning). These examples guide the model on how to interpret and solve the problem.

```
"Classify the sentiment of these reviews:
- Review: 'I love this movie!' Sentiment: Positive
- Review: 'It was boring.' Sentiment: Negative
- Review: 'I enjoyed the visuals.' Sentiment: Positive
- Review: 'The plot was dull.' Sentiment: Negative
- Review: 'Amazing soundtrack!' Sentiment: ?"
```

The model uses the provided examples to infer sentiment for last review.

**Strengths**: Enhances performance by providing relevant task-specific context with minimal data.

**Challenges**: Model performance can depend on the quality, clarity, and relevance of the few examples

# Example: restaurant's menu

- For instance, let's say that a restaurateur wants to add visual indicators for vegetarian dishes on their restaurant's menu. Using ChatGPT, they might write something like,

  "**Please rewrite this menu and put an asterisk next to all dishes that do not contain any meat,**" and then copy and paste the menu.

- This might seem like a trivial task for a human, but the model must first interpret the request, then classify each written menu item according to whether or not it contains meat, and finally produce the output in the corresponding format.

# Example: Freelance writer

- Let's say a freelance writer is working on 3 different pieces—one about dog breeding, one about exoplanets, and one about Pittsburgh—and wants to organize a list of articles by topic.
- In this case, they might write something like: Each of the following articles is related to one of "dog breeding," "exoplanets," or "Pittsburgh". If the response wasn't exactly what the writer wanted, they might try to provide additional guidance:

**Example: "The latest discovery of space telescopes": Exoplanets;**

**Example: "Why pugs have breathing problems": Dog breeding;**

- In addition to zero-shot and few-shot examples in the model's prompts, other changes to the model's prompt have uncovered additional emergent abilities.
- A technique called chain-of-thought prompting, or directing the model to break apart challenging problems into multiple steps, has been shown to improve model performance.
- People have also tested detailed instructions on zero-shot tasks, as well as asking the model about its level of confidence in its own response, each of which can improve responses in certain settings.

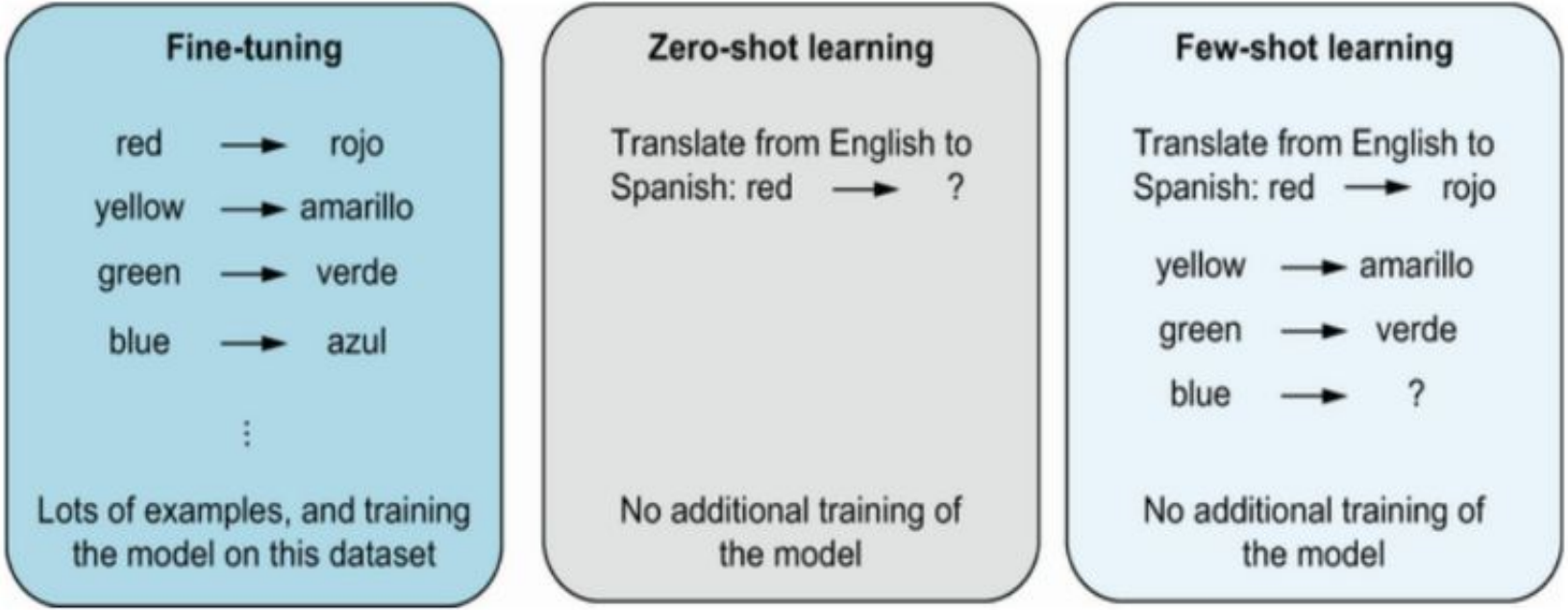


**Figure 2.2 A comparison of fine-tuning, zero-shot learning, and one-shot learning on a machine translation task**

## Comparison: Zero-Shot vs. Few-Shot

| Aspect | Zero-Shot | Few-Shot |
|---|---|---|
| **Input** | Task description only | Task description + few examples |
| **Labeled Examples** | None | Few (usually 1-5 examples) |
| **Training Data** | Broad pre-trained knowledge | Same as zero-shot + small task-specific context |
| **Use Case** | Generalization to completely new tasks | Slightly adapted for specific tasks |
| **Dependence on Context** | High | Higher due to examples provided |

Both methods are widely applied in the field of GenAI for tasks such as text classification, translation, summarization, and question answering, making them pivotal for modern NLP research and applications

# Zero shot LLM prompt for use case “online shopping support”

**Customer Query:**

"Hi, I’m looking for a gift for my sister’s birthday. She loves reading and enjoys scented candles. My budget is around $50. Can you help me find something she’d like?"

**Response Requirements:**

1. Understand the customer's needs and preferences.
2. Suggest relevant products or categories that align with the provided budget.
3. Include links (or placeholders) for the recommended products.
4. Offer additional tips or suggestions for a thoughtful gift selection.

**Response Example:**

"Sure! Here are a few ideas within your budget:

1. A book from her favorite genre or a bestseller. If she likes fiction, I recommend [Book Title], available for $15.
2. A set of premium scented candles, like [Brand Name], priced at $35.
3. Combine the two for a complete gift set! Let me know if you'd like links to these products or need more suggestions."

# Two examples for Few shot prompt LLM

## Example 1: Summarization Task

```
Summarize the following passages into one concise sentence.

Example 1:
Input: Climate change is a pressing global issue that has led to rising temperatures,
Output: Climate change is causing global warming and extreme weather.

Example 2:
Input: Advances in artificial intelligence, such as deep learning and natural language
Output: AI advancements are revolutionizing various industries.
```

# Example 2: Sentiment Analysis

```
Classify the sentiment of the following statements as Positive, Negative, or Neutral.

Example 1:
Statement: "I absolutely love the new design of this phone. It's sleek and user-friendly."
Sentiment: Positive

Example 2:
Statement: "The delivery was late, and the product was damaged. I'm very disappointed."
Sentiment: Negative

Example 3:
Statement: "The presentation was average, but it conveyed the necessary information."
Sentiment: Neutral
```

# Effective Prompt LLM :

Effective prompting refers to **crafting well-structured, concise, and clear instructions** to elicit the desired response from the LLM.

**Characteristics:**

1. **Clarity**: The prompt should clearly state what the LLM is expected to do.
2. **Specificity**: Provide enough context and details for the LLM to generate a relevant response.
3. **Focus on the Objective**: Avoid ambiguity and irrelevant details.
4. **Examples**: Include examples or guidelines if possible.

# Use Case: Online Shopping Support

An **effective prompt** might be structured to assist customers with product searches, comparisons, or troubleshooting.

**Example:**

"You are an online shopping assistant. A customer is looking for a laptop under $800 with at least 8GB RAM and a touchscreen. Suggest three suitable options available in the store, including their prices and key features."

# Responsive Prompt LLM :

Responsive prompting **focuses on creating prompts** that adapt to the user's queries or feedback dynamically, enabling conversational flow and context retention.

**Characteristics:**

Context-Aware: Prompts should maintain continuity in a conversation.

Interactive: Encourage follow-up questions or provide tailored responses.

Personalized: Adapt responses based on the user's preferences or history.

# Use Case: Online Shopping Support

A responsive prompt enhances the user experience by addressing specific needs or resolving issues seamlessly.

**Example:**

User: "I ordered a phone case, but it doesn't fit my phone. What should I do?"

Prompt for LLM: "Apologize for the inconvenience and provide a return/exchange policy. Offer to help find a phone case that fits the user's phone model."

The LLM can respond: "I'm sorry to hear about the issue with your phone case. You can return or exchange it using our simple return policy. Could you share your phone model so I can help you find the right case?"

## Key Differences in the Context of Online Shopping Support:

| Aspect | Effective Prompt LLM | Responsive Prompt LLM |
| --- | --- | --- |
| **Focus** | Crafting precise and detailed instructions. | Adapting to user queries in real-time. |
| **Objective** | Provide targeted solutions or suggestions. | Facilitate interactive, context-aware conversations. |
| **Use Case Examples** | Product recommendations, comparing features. | Handling complaints, guiding returns/exchanges. |

# Steps to Craft, Refine, and Test Prompts

1. **Understand the Context and Objective**

   Clearly define the application or use case for which the prompt is being designed. Are you targeting tasks like summarization, classification, text generation, or decision support?

2. **Start with a Basic Prompt**

   Begin with a straightforward prompt aligned with your use case. For example:

   - **Summarization**: "Summarize the following article in three sentences:"
   - **Data Extraction**: "Extract the key entities and their roles from the given text:"
   - **Creative Writing**: "Write a short story about overcoming challenges in an unfamiliar environment."

3. **Incorporate Specificity**

   Refine the prompt to include details about the expected output. For example:

   - Instead of "Summarize the article," use "Summarize the following article in three concise sentences that focus on the main argument."

4. **Experiment with Variants**

   Modify the wording, style, or tone to test how these changes affect the results. For example:

   - Change tone: "Write an engaging short story for children about..."
   - Rephrase: "Provide a detailed explanation of the following concept suitable for a beginner."

5. **Test the Prompt with Diverse Inputs**

   Use a range of input examples to evaluate how well the prompt performs in different scenarios.

6. **Evaluate Output Quality**

   Assess the results based on relevance, completeness, and alignment with your goals. Use metrics like precision, recall, or subjective ratings from users.

7. **Iterate and Refine**

   Use feedback to adjust the prompt. Introduce constraints, remove ambiguity, or rephrase as necessary.

8. **Document the Process**

   Record the evolution of your prompts, noting what changes improved or worsened the results.

## Example for Your Course Project

If your project involves creating a mini demo, you could:

1. Choose a specific GenAI application like **customer support automation**.
2. Draft initial prompts such as:
    - "Generate a polite and helpful response to the following customer query about late delivery."
3. Refine it based on output:
    - "Generate a polite, empathetic, and professional response to this customer complaint about late delivery, including an apology and next steps."