**🧩 Task 5: Few-Shot Prompting for Structured Feedback Analysis**

In this task, we will practice the **few-shot prompting** technique and also revisit techniques from previous tasks, such as **Chain-of-Thought (CoT) reasoning** and **output formatting**.

**📚 Theory**

**AI Techniques:**

* Information Extraction
* Classification
* Few-Shot Prompting
* Chain-of-Thought Reasoning
* Structured Output Generation

**Few-shot learning** is an approach in machine learning where a model is trained (or prompted) to perform a new task using only a **small number of examples**.

Instead of requiring thousands of labeled data points, **few-shot learning** allows generalization from just a few instances—sometimes as few as **one or two**.

In the context of **large language models (LLMs)** like ChatGPT, few-shot learning typically involves providing **a few examples directly in the prompt**.  
This helps the model understand the desired **format** or **behavior** for generating the correct response.

**🛠️ When is Few-Shot Learning Useful?**

* You don’t have enough training data
* You want to **quickly adapt** a general model to a specific task
* You need to **prototype without fine-tuning** a model

Few-shot learning is ideal for **challenges**, **rapid prototyping**, and **personalized AI interactions**.

**💬 Example**

In few-shot learning, the model is shown a few examples to learn the task pattern.

**Prompt:**  
Classify the sentiment of the following customer feedback as Positive, Neutral, or Negative:

* "The app is super intuitive and runs smoothly!" — **Positive**
* "It’s okay, but sometimes it lags." — **Neutral**
* "I hate the latest update. Everything is broken!" — **Negative**
* "I like the new design." —

**Expected Model Output:**  
**Positive**

**🛠️ Task**

The goal of this task is to practice **advanced prompt engineering** for **classifying unstructured user feedback** and extracting **detailed insights**.

Many services lack an effective **feedback loop system** — systems that allow teams to improve their product using insights from **millions of posts and comments**.  
A key challenge is building a reliable **feedback sorting mechanism**.

🎯 **Your Task:**Design a prompt that enables AI to analyze **product feedback comments** and extract **key insights** in a **structured JSON format** that can be integrated with product team workflows.

📄 **Given Feedbacks** (from Reddit):

**Sample 1:**  
*"Sensor just stops tracking for like a half second kinda often even at 8000hz. I've also tried it plugged in and still the same problem. First one I got I had to return also because the dongle just didnt work, $150 mouse btw"*

**Sample 2:**  
*"Is it worth it? It is a product with no flaws in my opinion, if you love it go for it, but its not worth the price since you'll be able to perform the same with a cheaper product with half the specs."*

🔧 **Your prompt must include the following techniques and structure:**

**🧪 Few-Shot Prompting**

Use the **3 provided examples** to demonstrate the **expected format and reasoning**.

**🧠 Chain-of-Thought Reasoning**

Instruct the AI to analyze the feedback **only if it is relevant to the product**.  
If not, set "isRelevant": false and skip further analysis.

**📦 JSON Output Format**

Ensure the AI’s response uses the following structure:

{  
 "sentiment": "string (Positive|Negative|Neutral|Mixed)",  
 "isRelevant": boolean,  
 "mainSubject": "string or null",  
 "positives": ["array of strings"],  
 "painPoints": ["array of strings"],  
 "improvementSuggestions": ["array of strings"],  
 "featuresMentioned": ["array of strings"],  
 "userExpertise": "string (Experienced|Novice|Unknown)"  
}

**📌 Requirements**

* Your prompt must use **Few-Shot Prompting**
* It must incorporate **Chain-of-Thought (CoT) reasoning**
* The AI's output must include:
  + The **specified JSON structure**
  + **Sentiment analysis**
  + **CoT-based decision-making**, especially for determining "isRelevant"