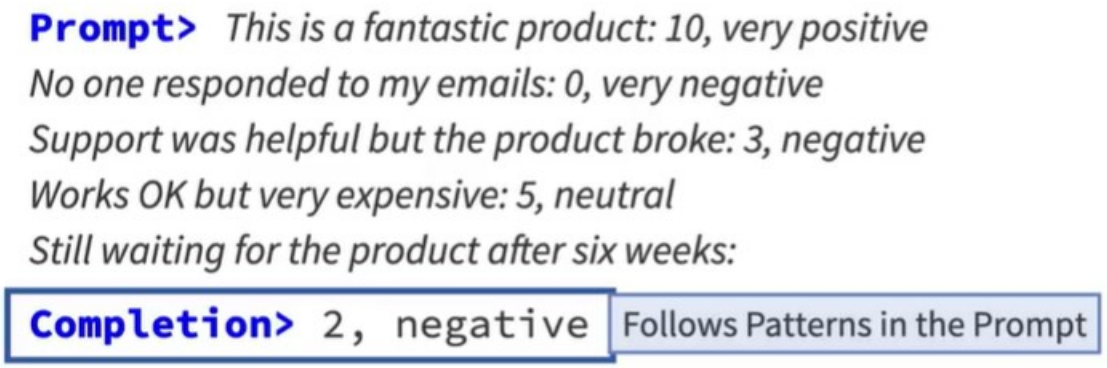
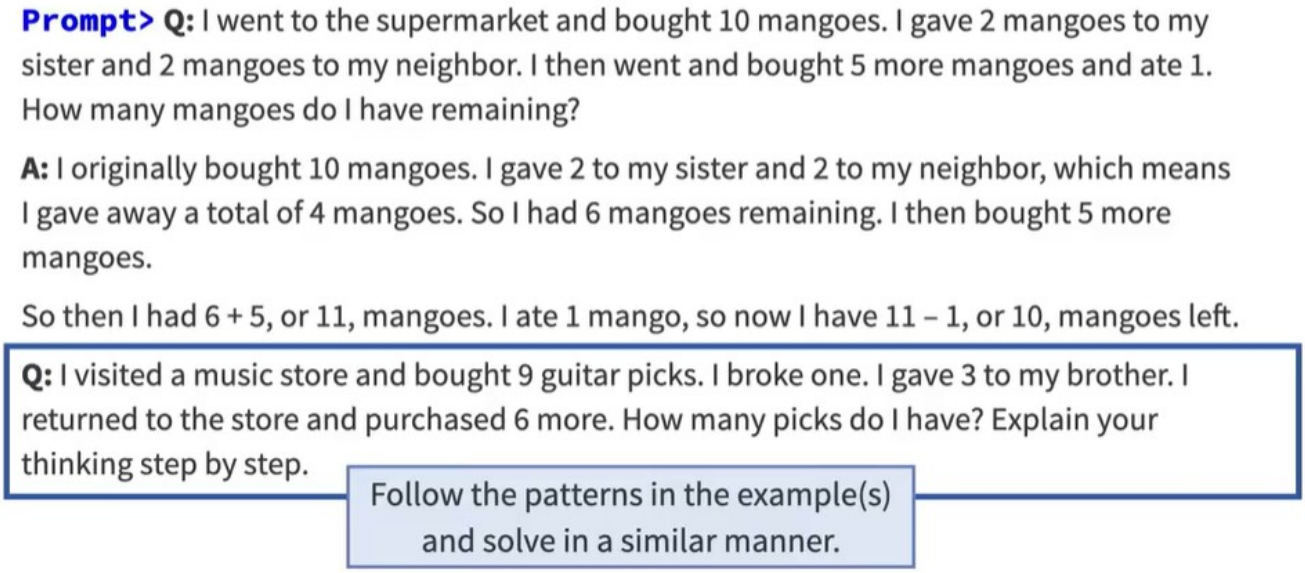
Zero shot:

A short prompt technique that asks the LLM for a response without any examples or demonstrations. There is no additional context for the model. We are relying purely on what the model already knows. Zero shot prompts are simple and quick, good for general summarizations, language translation, and basic classification. Not effective when needing something new, creative thinking, or something for a specific scenario.

Few shot prompting:
A prompt technique that gives a small number of examples or demonstrations to the LLM. These are the patterns for the LLM to follow. Few shot prompting is a type of in-context learning. By providing additional patterns, that's more context, for the language model to follow, we have helped guide it to give us a helpful response. The more context in your prompt, the better.



Chain of thought (CoT) prompting:
A prompt technique that demonstrates how to solve sub-problems of a larger problem for a final response. We show the language model, the pattern for breaking-down a problem into a series of sub-problems. Another type of in-context learning. We are guiding the language model by providing with more patterns.



I solve a problem for the language model by providing patterns and breaking it down into a series of sub-problems, then I provide a similar problem to the language model and then I request it to solve it. Language model solves the problem in the similar manner using CoT.

CoT is the basis of newer “reasoning” models. Reasoning models understand the CoT technique inherently. Non-reasoning models might give different results. That means you got to track and monitor your results.

Importance of context: extra information that guides the LLM to generate more relevant responses. The more specific and detailed context, better the LLM can produce accurate and appropriate completions. Context is a set of strings that you provide along with the core prompt. It can be inserted by the user’s input, it can be generated by your app, it can be retrieved from a file system, it can be scrapped from a webpage, it can be extracted from a database or any other data source.

A Neural network inside of an LLM has billions of nodes and each node has a set of parameters, most importantly the next node probability. During training process, NN learns which paths are more likely than other paths based upon those probabilities. In a trained NN, it knows some paths are likely to occur while some paths are not as likely. When we provide the context, we are widening the probability cloud, thats how the context affects LLM’s choices. Depending on the context, LLM could go into a different path, it is upto you to provide proper context to guide the LLM in giving you the effective response.