Starting April 29, 2025, Gemini 1.5 Pro and Gemini 1.5 Flash models are not available in projects that have no prior usage of these models, including new projects. For details, see Model versions and lifecycle (/vertex-ai/generative-ai/docs/learn/model-versions#legacy-stable).

Include few-shot examples

You can include examples in the prompt that show the model what a good response looks like. The model attempts to identify patterns and relationships from the examples and applies them when generating a response. Prompts that contain examples are called *few-shot* prompts, while prompts that provide no examples are called *zero-shot prompts*. Few-shot prompts are often used to regulate the output formatting, phrasing, scoping, or general patterning of model responses. Use specific and varied examples to help the model narrow its focus and generate more accurate results.

Including few-shot examples in your prompts helps make them more reliable and effective. However, you should always accompany few-shot examples with clear instructions. Without clear instructions, models might pick up one unintended patterns or relationships from the examples, which can lead to poor results.

The key points to this strategy are as follows:

- Including prompt-response examples in the prompt helps the model learn how to respond.
- Use XML-like markup to markup the examples.
- Experiment with the number of prompts to include. Depending on the model, too few examples are ineffective at changing model behavior. Too many examples can cause the model to overfit.
- Use consistent formatting across examples

Zero-shot versus few-shot prompts

The following zero-shot prompt asks the model to extract the technical specifications from text and output it in JSON format:

:

```
ıct the technical specifications from the text below in JSON format.

.e Pixel 7, 5G network, 8GB RAM, Tensor G2 processor, 128GB of storage, Lemongrass
```

se:

```
{
  "Network": "5G",
  "RAM": "8GB",
  "Processor": "Tensor G2",
  "Storage": "128GB",
  "Color": "Lemongrass"
}
```

(gem

Suppose your use case requires specific formatting, such as lowecaser key names. You can include examples in the prompt that shows the model how to format the JSON. The following few-shot prompt demonstrates an output format where the JSON keys are lowercase:

:

```
ıct the technical specifications from the text below in a JSON format.

IPLE>
UT: Google Nest Wifi, network speed up to 1200Mpbs, 2.4GHz and 5GHz frequencies, WP3

PUT:

'product":"Google Nest Wifi",
'speed":"1200Mpbs",
'frequencies": ["2.4GHz", "5GHz"],
'protocol":"WP3"

MPLE>
```

```
ogle Pixel 7, 5G network, 8GB RAM, Tensor G2 processor, 128GB of storage, Lemongrass
```

---

se:

```
{
  "product": "Google Pixel 7",
  "network": "5G",
  "ram": "8GB",
  "processor": "Tensor G2",
  "storage": "128GB",
  "color": "Lemongrass"
}
```

Note that the example uses XML-like formatting to separate the components of the prompt. To learn more about how to optimally format few-shot prompts using XML-like formatting, see Structure prompts (/vertex-ai/generative-ai/docs/learn/prompts/structure-prompts).

Find the optimal number of examples

You can experiment with the number of examples to provide in the prompt for the most desired results. Models like Gemini can often pick up on patterns using a few examples, though you may need to experiment with what number of examples leads to the desired results. At the same time, if you include too many examples, the model might start to overfit (https://developers.google.com/machine-learning/glossary#overfitting) the response to the examples.

What's next

- Explore more examples of prompts in the Prompt gallery (/vertex-ai/generative-ai/docs/prompt-gallery).

Except as otherwise noted, the content of this page is licensed under the Creative Commons Attribution 4.0 License (https://creativecommons.org/licenses/by/4.0/), and code samples are licensed under the Apache 2.0 License

(https://www.apache.org/licenses/LICENSE-2.0). For details, see the Google Developers Site Policies (https://developers.google.com/site-policies). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2025-06-06 UTC.