How to create a custom example selector

Contents

In this tutorial, we’ll create a custom example selector that selects every alternate example from a given list of examples.

An `ExampleSelector` must implement two methods:

1. An `add_example` method which takes in an example and adds it into the ExampleSelector
2. A `select_examples` method which takes in input variables (which are meant to be user input) and returns a list of examples to use in the few shot prompt.

Let’s implement a custom `ExampleSelector` that just selects two examples at random.

> **Note**
>
> Take a look at the current set of example selector implementations supported in LangChain here.

Implement custom example selector

```
from langchain.prompts.example_selector.base import BaseExampleSelector
from typing import Dict, List
import numpy as np


class CustomExampleSelector(BaseExampleSelector):
```

Skip to main content

```
        self.examples = examples

    def add_example(self, example: Dict[str, str]) -> None:
        """Add new example to store for a key."""
        self.examples.append(example)

    def select_examples(self, input_variables: Dict[str, str]) -> List[dict]:
        """Select which examples to use based on the inputs."""
        return np.random.choice(self.examples, size=2, replace=False)
```

Use custom example selector

```
examples = [
    {"foo": "1"},
    {"foo": "2"},
    {"foo": "3"}
]

# Initialize example selector.
example_selector = CustomExampleSelector(examples)


# Select examples
example_selector.select_examples({"foo": "foo"})
# -> array([{'foo': '2'}, {'foo': '3'}], dtype=object)

# Add new example to the set of examples
example_selector.add_example({"foo": "4"})
example_selector.examples
# -> [{'foo': '1'}, {'foo': '2'}, {'foo': '3'}, {'foo': '4'}]

# Select examples
example_selector.select_examples({"foo": "foo"})
# -> array([{'foo': '1'}, {'foo': '4'}], dtype=object)
```