

```

1 from transformers import AutoTokenizer, AutoModelForCausalLM
2 model_id = "meta-llama/Meta-Llama-3.1-8B"
3 tokenizer = AutoTokenizer.from_pretrained(model_id)
4 lm = AutoModelForCausalLM.from_pretrained(model_id)

5 mlp = lm.model.layers[16].mlp.down_proj
6 neurons = [394, 5490, 8929]
7 prompt = "The truth is the"

8 def pre_hook_fn(module, input):
9     input[0][:, -1, neurons] = 10

10 hook = mlp.register_forward_pre_hook(pre_hook_fn)
11 inputs = tokenizer(prompt, return_tensors="pt")
12 out = lm(**inputs)
13 hook.remove()

14 last = out["logits"][:, -1].argmax()
15 prediction = tokenizer.decode(last)
16 print(prediction)

```

(a)

```

1 from nnsight import LanguageModel
2 model_id = "meta-llama/Meta-Llama-3.1-8B"
3 lm = LanguageModel(model_id)

4 mlp = lm.model.layers[16].mlp.down_proj
5 neurons = [394, 5490, 8929]
6 prompt = "The truth is the"

7 with lm.trace(prompt, remote=True):
8     mlp.input[:, -1, neurons] = 10
9     out = lm.output.save()

10 last = out["logits"][:, -1].argmax()
11 prediction = lm.tokenizer.decode(last)
12 print(prediction)

```

(b)