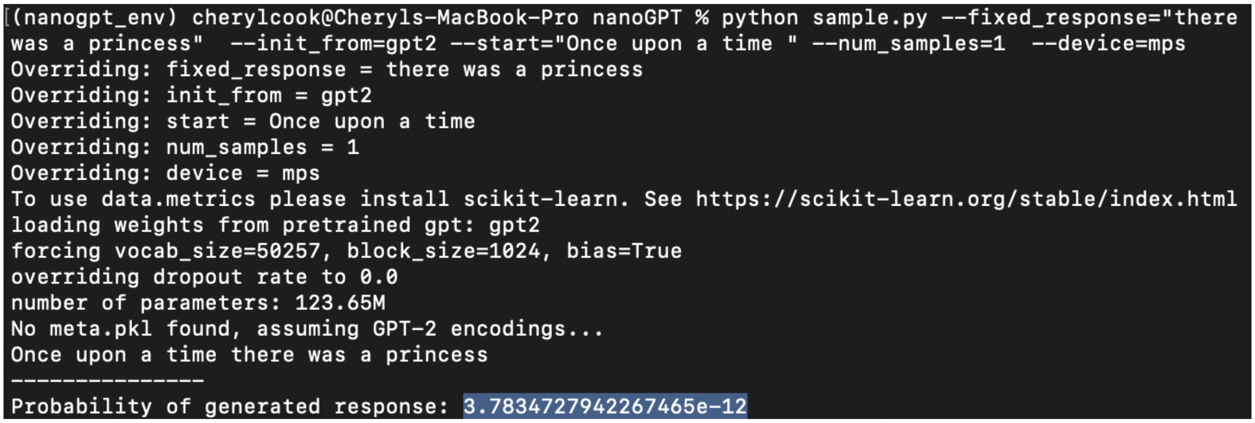
Effect of sequence length on sequence probability:

All arguments are kept constant across runs, except for the length of the fixed continuation of the prompt.

Initial response (4 tokens):

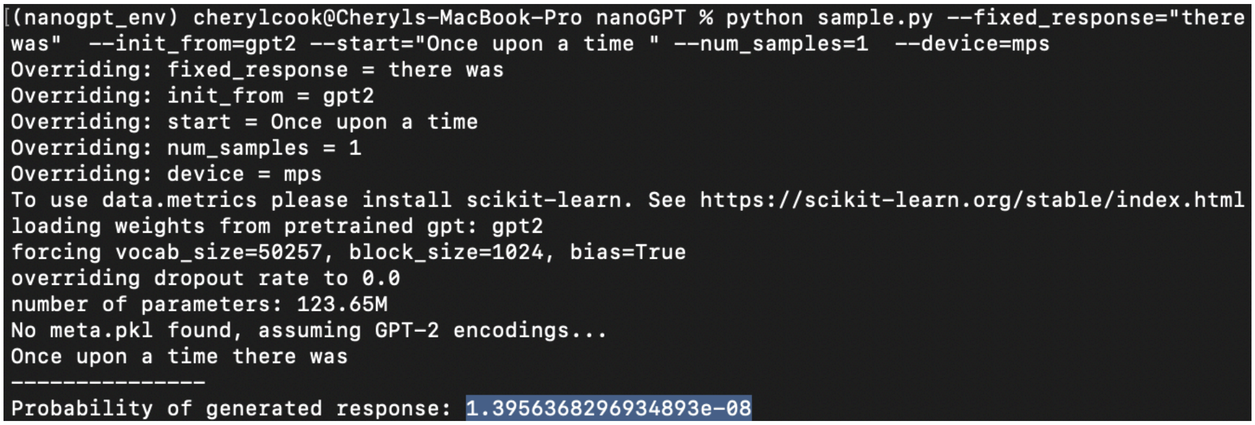
Start: “Once upon a time ” | Fixed response: “there was a princess”
Probability: 3.78e-12



```
[(nanogpt_env) cherylcook@Cheryls-MacBook-Pro nanoGPT % python sample.py --fixed_response="there
was a princess"  --init_from=gpt2 --start="Once upon a time " --num_samples=1  --device=mps
Overriding: fixed_response = there was a princess
Overriding: init_from = gpt2
Overriding: start = Once upon a time
Overriding: num_samples = 1
Overriding: device = mps
To use data.metrics please install scikit-learn. See https://scikit-learn.org/stable/index.html
loading weights from pretrained gpt: gpt2
forcing vocab_size=50257, block_size=1024, bias=True
overriding dropout rate to 0.0
number of parameters: 123.65M
No meta.pkl found, assuming GPT-2 encodings...
Once upon a time there was a princess
---------------
Probability of generated response: 3.7834727942267465e-12
```

Decreasing the sequence length to two tokens:

Start: “Once upon a time ” | Fixed response: “there was”
Probability: 1.34e-08

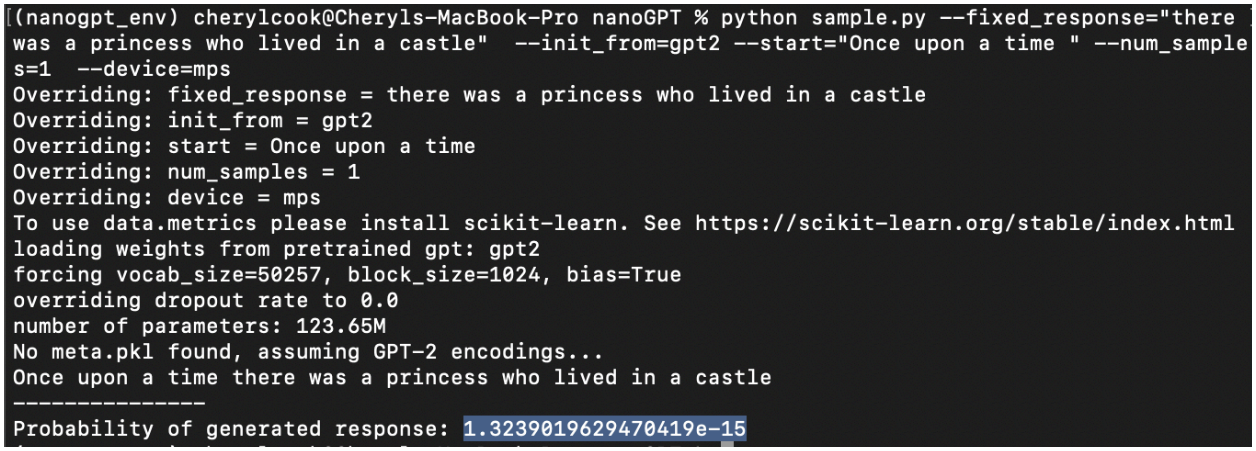


```
[(nanogpt_env) cherylcook@Cheryls-MacBook-Pro nanoGPT % python sample.py --fixed_response="there
was"  --init_from=gpt2 --start="Once upon a time " --num_samples=1  --device=mps
Overriding: fixed_response = there was
Overriding: init_from = gpt2
Overriding: start = Once upon a time
Overriding: num_samples = 1
Overriding: device = mps
To use data.metrics please install scikit-learn. See https://scikit-learn.org/stable/index.html
loading weights from pretrained gpt: gpt2
forcing vocab_size=50257, block_size=1024, bias=True
overriding dropout rate to 0.0
number of parameters: 123.65M
No meta.pkl found, assuming GPT-2 encodings...
Once upon a time there was
---------------
Probability of generated response: 1.3956368296934893e-08
```

A shorter sequence yields a higher overall sequence probability because fewer log probabilities (negative values) are being summed.

Increasing the sequence length to nine tokens:

Start: “Once upon a time ” | Fixed response: “there was a princess who lived in a castle”
Probability: 1.32e-15



```
[(nanogpt_env) cherylcook@Cheryls-MacBook-Pro nanoGPT % python sample.py --fixed_response="there 
was a princess who lived in a castle"  --init_from=gpt2 --start="Once upon a time " --num_sample
s=1  --device=mps
Overriding: fixed_response = there was a princess who lived in a castle
Overriding: init_from = gpt2
Overriding: start = Once upon a time
Overriding: num_samples = 1
Overriding: device = mps
To use data.metrics please install scikit-learn. See https://scikit-learn.org/stable/index.html
loading weights from pretrained gpt: gpt2
forcing vocab_size=50257, block_size=1024, bias=True
overriding dropout rate to 0.0
number of parameters: 123.65M
No meta.pkl found, assuming GPT-2 encodings...
Once upon a time there was a princess who lived in a castle
---------------
Probability of generated response: 1.3239019629470419e-15
```

A longer sequence yields a lower overall sequence probability since more token log probabilities are being summed.

Takeaway:

Shorter sequences are more probable than longer ones because there are fewer token probabilities contributing to the total.