

Documentation for the `Anthropic` Class

Overview and Introduction

The `Anthropic` class provides an interface to interact with the Anthropic large language models. This class encapsulates the necessary functionality to request completions from the Anthropic API based on a provided prompt and other configurable parameters.

Key Concepts and Terminology

- **Anthropic**: A large language model, akin to GPT-3 and its successors.
- **Prompt**: A piece of text that serves as the starting point for model completions.
- **Stop Sequences**: Specific tokens or sequences to indicate when the model should stop generating.
- **Tokens**: Discrete pieces of information in a text. For example, in English, a token can be as short as one character or as long as one word.

Class Definition

`Anthropic`

```
```python
```

```
class Anthropic:
```

```
 """Anthropic large language models."""
```

```
 ...
```

### **Parameters:**

- ``model (str)``: The name of the model to use for completions. Default is "claude-2".
- ``max_tokens_to_sample (int)``: Maximum number of tokens to generate in the output. Default is 256.
- ``temperature (float, optional)``: Sampling temperature. A higher value will make the output more random, while a lower value will make it more deterministic.
- ``top_k (int, optional)``: Sample from the top-k most probable next tokens. Setting this parameter can reduce randomness in the output.
- ``top_p (float, optional)``: Sample from the smallest set of tokens such that their cumulative probability exceeds the specified value. Used in nucleus sampling to provide a balance between randomness and determinism.
- ``streaming (bool)``: Whether to stream the output or not. Default is False.
- ``default_request_timeout (int, optional)``: Default timeout in seconds for API requests. Default is 600.

### **\*\*Methods and their Functionality\*\***

#### ``_default_params(self) -> dict``

- Provides the default parameters for calling the Anthropic API.

- **\*\*Returns\*\***: A dictionary containing the default parameters.

```
`generate(self, prompt: str, stop: list[str] = None) -> str`
```

- Calls out to Anthropic's completion endpoint to generate text based on the given prompt.

- **\*\*Parameters\*\***:

- `prompt (str)` : The input text to provide context for the generated text.

- `stop (list[str], optional)` : Sequences to indicate when the model should stop generating.

- **\*\*Returns\*\***: A string containing the model's generated completion based on the prompt.

```
`__call__(self, prompt: str, stop: list[str] = None) -> str`
```

- An alternative to the `generate` method that allows calling the class instance directly.

- **\*\*Parameters\*\***:

- `prompt (str)` : The input text to provide context for the generated text.

- `stop (list[str], optional)` : Sequences to indicate when the model should stop generating.

- **\*\*Returns\*\***: A string containing the model's generated completion based on the prompt.

## **\*\*Usage Examples\*\***

```

```python

# Import necessary modules and classes

from swarm_models import Anthropic


# Initialize an instance of the Anthropic class

model = Anthropic(anthropic_api_key="")


# Using the run method

completion_1 = model.run("What is the capital of France?")

print(completion_1)


# Using the __call__ method

completion_2 = model("How far is the moon from the earth?", stop=["miles", "km"])

print(completion_2)

...

## **Mathematical Formula**

```

The underlying operations of the `Anthropic` class involve probabilistic sampling based on token logits from the Anthropic model. Mathematically, the process of generating a token t from the given logits l can be described by the softmax function:

$$P(t) = \frac{e^{l_t}}{\sum_i e^{l_i}}$$

Where:

- $P(t)$ is the probability of token t .
- l_t is the logit corresponding to token t .
- The summation runs over all possible tokens.

The temperature, top-k, and top-p parameters are further used to modulate the probabilities.

Additional Information and Tips

- Ensure you have a valid `ANTHROPIC_API_KEY` set as an environment variable or passed during class instantiation.
- Always handle exceptions that may arise from API timeouts or invalid prompts.

References and Resources

- [Anthropic's official documentation](https://www.anthropic.com/docs)
- [Token-based sampling in Language Models](https://arxiv.org/abs/1904.09751) for a deeper understanding of token sampling.