How to work with Lisette

An Introduction to Answer.ai's Lisette Library

2024-01-15

This is how to write some text including some citations as sidenotes

Quarto (abbreviated Qto, 4to or 4º) is the format of a book or pamphlet produced from full sheets printed with eight pages of text, four to a side, then folded twice to produce four leaves. The leaves are then trimmed along the folds to produce eight book pages. Each printed page presents as one-fourth size of the full sheet.[1]

[1] If this is working that is really nice.

The earliest known European printed book is a quarto, the Sibyllenbuch, believed to have been printed by Johannes Gutenberg in 1452–53, before the Gutenberg Bible, surviving only as a fragment. Quarto is also used as a general description of size of books that are about 12 inches (30 cm) tall, and as such does not necessarily indicate the actual printing format of the books, which may even be unknown, as is the case for many modern books. These terms are discussed in greater detail in book sizes.

```
# this is a sample code block to reverse a list
def reverse_list(list):                              ①
    if something_happens:
        print('this happened')
    else:
        print('this happened becuase the other did not')
```

① will this code annotation work ? hardly

This is how to embed an image

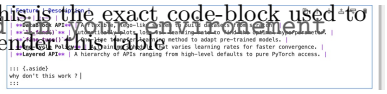
Pro Tip: This dataset was cleaned using the Pandas library before ingestion.

What can we do with lisette lib

Overview

Feature	Description
DataBlock API	A flexible, lego-like system to build datasets from scratch.
`lr_find()`	Automatically plots loss vs. learning rate to find the optimal hyperparameter.
`fine_tune()`	One-line transfer learning method to adapt pre-trained models.
One-Cycle Policy	A training schedule that varies learning rates for faster convergence.
Layered API	A hierarchy of APIs ranging from high-level defaults to pure PyTorch access.

```
!uv pip list # this is how you look for what is installed in your environment
```



this is the exact code-block used to render this table

Imagine you need to write a simple tools(or fuctions).How can you do that ?. Here is how and it is so easy.

```
def adder(a:float,b:float):
    '''This adds two number a and b'''                    ①
    return a + b
```

① you always need to add a docstring.

```
!uv pip install lisette -qq
```

```
from lisette import *
```

```
import os
from dotenv import load_dotenv
```

```
load_dotenv()
```

```
True
```

```
model = 'gemini/gemini-2.5-pro'
```

```
c = Chat(model)
```

```
c('hey')
```

Hey there! How can I help you today?

- id: `f7kwaZ22HP3w4-EPrqXEgQ4`
- model: `gemini-2.5-pro`
- finish_reason: `stop`
- usage: `Usage(completion_tokens=777, prompt_tokens=2, total_tokens=779, completion_tokens_details=CompletionTokensDetailsWrapper(accepted_predic audio_tokens=None, reasoning_tokens=767, rejected_prediction_tokens=None, text_tokens=10, image_tokens=None), prompt_tokens_details=PromptTokensDetailsWrapper(audio_tokens=None, cached_tokens=None, text_tokens=2, image_tokens=None))`