gpt4allpandas

July 21, 2023

[1]:
```
import subprocess

package_name = "gpt4pandas"

try:
    # Attempt to install the package using pip
    subprocess.check_call(["pip", "install", package_name])
    print(f"Successfully installed {package_name}.")
except subprocess.CalledProcessError as e:
    # Handle the error if the installation fails
    print(f"Error: Failed to install {package_name}.")
else:
    # Code to be executed if the installation is successful
    pass
```

```
Successfully installed gpt4pandas.
```

[2]:
```
import pandas as pd
from gpt4pandas import GPT4Pandas
from pathlib import Path
from tqdm import tqdm
import urllib
import sys

# If there is no model, then download one
# These models can be automatically downloaded, uncomment the model you want to␣
 ↪use
# url = "https://huggingface.co/ParisNeo/GPT4All/resolve/main/
 ↪gpt4all-lora-quantized-ggml.bin"
# url = "https://huggingface.co/ParisNeo/GPT4All/resolve/main/
 ↪gpt4all-lora-unfiltered-quantized.new.bin"
# url = "https://huggingface.co/eachadea/legacy-ggml-vicuna-7b-4bit/resolve/
 ↪main/ggml-vicuna-7b-4bit-rev1.bin"
url = "https://huggingface.co/eachadea/ggml-vicuna-13b-4bit/resolve/main/
 ↪ggml-vicuna-13b-4bit-rev1.bin"
model_name  = url.split("/")[-1]
folder_path = Path("models/")
```

```
model_full_path = (folder_path / model_name)

# ++++++++++++++++++++ Model downloading␣
 ↪++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
# Check if file already exists in folder
if model_full_path.exists():
    print("File already exists in folder")
else:
    # Create folder if it doesn't exist
    folder_path.mkdir(parents=True, exist_ok=True)
    progress_bar = tqdm(total=None, unit="B", unit_scale=True,␣
 ↪desc=f"Downloading {url.split('/')[-1]}")
    # Define callback function for urlretrieve
    def report_progress(block_num, block_size, total_size):
        progress_bar.total=total_size
        progress_bar.update(block_size)
    # Download file from URL to folder
    try:
        urllib.request.urlretrieve(url, folder_path / url.split("/")[-1],␣
 ↪reporthook=report_progress)
        print("File downloaded successfully!")
    except Exception as e:
        print("Error downloading file:", e)
        sys.exit(1)
#␣
 ↪++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

# Load a sample dataframe
data = {
    "Name": ["Alice", "Bob", "Charlie"],
    "Age": [25, 30, 35],
    "City": ["New York", "Paris", "London"],
    "Salary": [50000, 60000, 70000],
}
df = pd.DataFrame(data)

# Initialize the GPT4Pandas model
model_path = "models/"+model_name
gpt = GPT4Pandas(model_path, df, verbose=False)

print("Dataframe")
print(df)
# Ask a question about the dataframe
question = "What is the average salary?"
print(question)
answer = gpt.ask(question)
```

```
print(answer)  # Output: "mean(Salary)"

# Ask another question
question = "Which person is youngest?"
print(question)
answer = gpt.ask(question)
print(answer)  # Output: "max(Age)"

# Set a new dataframe and ask a question
new_data = {
    "Name": ["David", "Emily"],
    "Age": [40, 45],
    "City": ["Berlin", "Tokyo"],
    "Salary": [80000, 90000],
}
new_df = pd.DataFrame(new_data)
print("Dataframe")
print(new_df)

gpt.set_dataframe(new_df)
question = "What is salary in Tokyo?"
print(question)
answer = gpt.ask(question)
print(answer)
```

```
Downloading ggml-vicuna-13b-4bit-rev1.bin: 100%|      | 8.13G/8.14G
[01:08<00:00, 138MB/s]

File downloaded successfully!
Dataframe
      Name  Age      City  Salary
0    Alice   25  New York   50000
1      Bob   30     Paris   60000
2  Charlie   35    London   70000
What is the average salary?

Downloading ggml-vicuna-13b-4bit-rev1.bin: 8.14GB [05:09, 138MB/s]

The average salary is 55,000.
Which person is youngest?
Alice is the youngest.
Dataframe
    Name  Age    City  Salary
0  David   40  Berlin   80000
1  Emily   45   Tokyo   90000
What is salary in Tokyo?
90000.
```