BeyondChats Data Processor

Overview:

This script is built to automatically fetch and analyze data from the BeyondChats API, focusing on extractin

Prerequisites:

The script runs with Python version 3.10 and several Python libraries which include requests, pandas, json

Installing Required Libraries:

The necessary libraries for this script can be installed via pip, Python's package installer. The commands a

```
pip install requests pandas spacy
```

After installing the libraries, you must download the spaCy language model which aids in text processing:

```
python -m spacy download en_core_web_md
```

Environment Configuration:

You should ensure that Python 3.10 is installed on your computer. You can verify this by executing the follo

```
python --version
```

If you do not have Python 3.10, you should install it from the Python official website.

Running the Script:

To get the script running:

1. Clone the repository to your local machine using:

```
git clone <repository-url>
```

2. To execute the script, navigate to the directory containing the script and run:

python beyondchats_processor.py

How the Script Works:

Data Retrieval:

The script uses the requests library to make API calls to the BeyondChats endpoint specified. It manages p

Data Storage:

Once data is fetched, it is stored locally in a JSON format to facilitate ease of access and manipulation.

Citation Identification:

The script uses the spaCy library and its language model en_core_web_md for text processing. The operat

- Tokenization: Breaking down the text into manageable pieces or tokens.

- Semantic Analysis: It compares the fetched text data semantically to predefined citation contexts to find re

- Output Generation: It then formats these matches into a structured document that links each piece of text

Outputs Produced by the Script:

The script outputs two files:

- complete_fetched_data.json: Contains all the data retrieved from the API.

- citation_results.json: Contains the citations found in the text, formatted with links to their sources.

Technologies Used:

The script primarily uses the medium-sized English model from spaCy for natural language tasks, capable