Automated Data Summary Script

Introduction

This documentation provides an overview of an automated script developed in Python to send periodic data summaries to a Slack channel or an email address. The script utilizes Python's data manipulation libraries and Slack's webhooks to generate and deliver monthly trend analysis of COVID-19 deaths from the top 3 states.

Functionality

The script performs the following tasks:

- Retrieves COVID-19 death data from a dataset.
- Analyses the data to determine the monthly trend analysis of deaths from the top 3 states
- Sends periodic data summaries to Slack using a predefined webhook URL.
- Ensures the messages are sent at fixed intervals for the months of March, April, May, and June.

Dependencies

The script requires the following Python packages:

- time : used to introduce a time delay or interval between sending messages to Slack.
- requests: Used for making HTTP requests to the Slack API.
- pandas: Used for data manipulation and analysis.
- calendar: Used for retrieving month names.
- numpy: Used for numerical calculations.

Configuration

Before running the script, the user needs to perform the following configurations:

- Set up a Slack account and create a webhook URL to post messages to a desired channel
- Install the necessary Python packages listed in the dependencies section.
- Ensure the dataset containing COVID-19 state-level data is available and provide the correct file path.

covid_data_analysis

The 'Covid_data_analysis.ipynb' notebook provides insights into how the top 3 states' data on COVID-19 deaths is extracted from the original DataFrame. The notebook presents a step-by-step analysis, demonstrating the process of filtering and aggregating the data to determine the states with the highest number of COVID-19 deaths.

Functions in main.py

send_slack_message(message)

This function sends a message to Slack using the provided webhook URL. It takes a message as input, formats it, and sends a POST request to the webhook URL using the requests library. If the message fails to be sent, an appropriate error message is displayed.

creating_and_sending_message()

This function generates and sends periodic messages to Slack. It performs the following tasks:

- 1. Reads the COVID-19 state-level data from the dataset.
- 2. Filters the data for the months of March, April, May, and June.
- 3. Calculates the monthly trend analysis by summing the deaths and selecting the top 3 states for each month.
- 4. Calculates the percentage of total deaths for each state in a specific month.
- 5. Constructs a message string for each month, including the month name, top 3 states, number of deaths, and percentage of total deaths.
- 6. Sends the message to Slack using the send slack message function.
- 7. Waits for a fixed interval of 2 minutes before sending the next message.

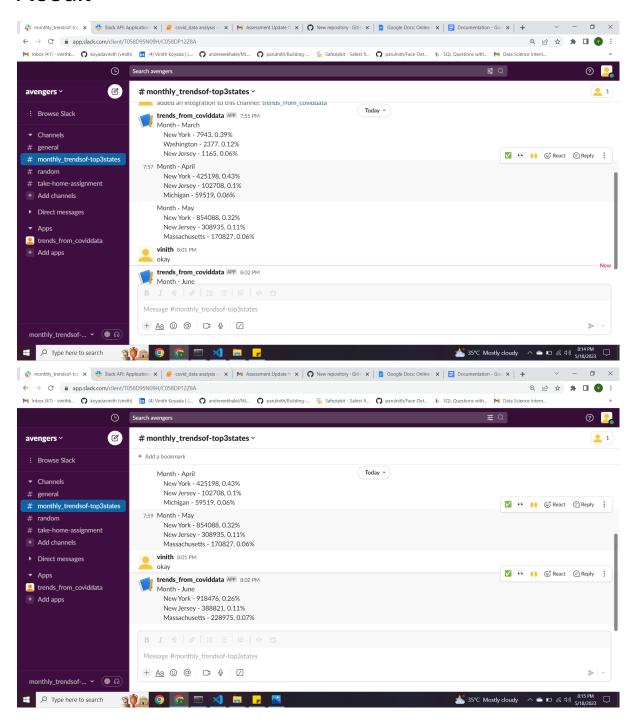
Execution

To run the script and send periodic data summaries:

- 1. Make sure the required Python packages are installed.
- 2. Update the WEBHOOK_URL variable with the appropriate Slack webhook URL.
- 3. Provide the correct file path of the COVID-19 state-level data in the pd.read_excel() function.
- 4. Execute the creating_and_sending_message() function.

5. The script will generate and send four messages, each containing the monthly trend analysis of COVID-19 deaths from the top 3 states for the months of March, April, May, and June.

Result



Conclusion

The automated script described in this documentation simplifies the process of generating and delivering periodic data summaries to a Slack channel or an email address. By utilising Python's data manipulation capabilities and Slack's webhooks, the script provides a convenient solution for monitoring important metrics, such as the monthly trend analysis of COVID-19 deaths from the top 3 states.