Analyzing the Impact of CO2 Emissions on Temperature Changes

Main Question

How do CO2 emissions correlate with temperature changes globally?

Data Sources

Datasource 1: FAO Temperature Change Dataset

- **Why Chosen:** This dataset provides comprehensive and reliable annual temperature change data for various countries, essential for analyzing global temperature trends.
- **Source:** Food and Agriculture Organization (FAO)
- **Data Contains:** Annual temperature change data for various countries.
- **Metadata URL:** FAO Temperature Change Metadata
- **Data URL:** FAO Temperature Change Data
- **Data Type:** CSV
- **Data Structure and Quality:** Structured in CSV format with columns for country, year, and temperature change values. High-quality data maintained by the FAO.
- **License:** CC BY-NC-SA 3.0 IGO
- **Citation:** Food and Agriculture Organization of the United Nations. (2023). FAOSTAT: Temperature Change. Retrieved from FAOSTAT Temperature Change
- **License Compliance:** The data can be used, shared, and adapted for non-commercial purposes with appropriate attribution. Obligations include providing proper credit and sharing any derivative works under the same license.

Datasource 2: World Bank CO2 Emissions Dataset

- **Why Chosen:** This dataset offers extensive CO2 emissions data, crucial for analyzing the relationship between emissions and temperature changes.
- **Source:** World Bank
- **Data Contains:** CO2 emissions data (in kilotons) for various countries.
- **Metadata URL:** World Bank CO2 Emissions Metadata
- **Data URL:** World Bank CO2 Emissions Data
- **Data Type:** CSV
- **Data Structure and Quality:** CSV format with columns for country, year, and CO2 emission values. High-quality data from a reputable source.
- **License:** Attribution-NonCommercial 4.0 International (CC BY-NC 4.0)
- **Citation:** World Bank. (2023). CO2 emissions (kt). Retrieved from World Bank CO2 Emissions
- **License Compliance:** Data usage is permitted for non-commercial purposes with appropriate attribution. Obligations include providing a link to the license and indicating if changes were made.

Loading Data and Initial Inspection

Displaying the data sets after applying transformations on it

In [39]:
```
import pandas as pd
# get data from temperature table
temp_df = pd.read_sql_table('temperature', 'sqlite:///../data/pipelineDB.
temp_df.head(5)
```

Out[39]:

	Area	Year	Change

0	Afghanistan	1961	0.023667
1	Afghanistan	1962	-0.282250
2	Afghanistan	1963	0.854000
3	Afghanistan	1964	-1.003250
4	Afghanistan	1965	0.011833

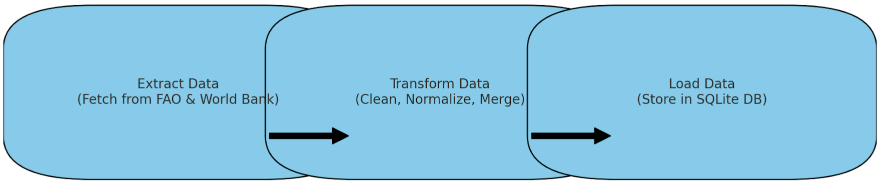
In [40]:

```
# get data from CO2 emissions table
CO2_df = pd.read_sql_table('CO2_emissions', 'sqlite:///../data/pipelineDB
CO2_df.head()
```

Out[40]:

	Area	country_code	Year	co2_emissions
0	Africa Eastern and Southern	AFE	1990	304614.720181
1	Afghanistan	AFG	1990	2046.870000
2	Africa Western and Central	AFW	1990	97190.345000
3	Angola	AGO	1990	6564.200000
4	Albania	ALB	1990	6060.500000

DATA PIPELINE



Data Pipeline Description

The data pipeline for this project automates the extraction, transformation, and loading (ETL) process to analyze the impact of CO2 emissions on temperature changes.

Technologies Used

- **Extraction:** Python scripts using the `requests` library for downloading datasets from FAO and World Bank.
- **Transformation:** Pandas library for data cleaning, normalization, and transformation.
- **Loading:** SQLite database for storing the processed data.

Steps:

1. **Extract Data**

    - **Description:** The pipeline extracts data from FAO and World Bank sources. Both datasets are downloaded as a zip file and extracted.
    - **Output:** Raw data files (`fao_data.zip` , `world_bank_data.zip`) are stored locally.

2. **Transform Data**

   - **Description:** The raw data is cleaned and transformed. The FAO dataset is renamed for clarity, and both datasets are merged based on common keys such as 'year' and 'country'. Any necessary data cleaning operations, such as handling missing values and normalizing data formats, are performed.
   - **Output:** A transformed table (`tempCO2`) is created, containing the merged and cleaned data.
   - **Transformation Steps:**
     - **Transform FAO Temperature Data:**
       - Keep only the rows that contain temperature change.
       - Drop irrelevant columns such as 'Area Code', 'Area Code (M49)', 'Element Code', 'Months Code', 'Unit', and 'Element'.
       - Filter the data to keep only the desired months.
       - Drop forecast columns for each year and keep only the rows with estimated values.
       - Reshape the data from wide to long format using `pd.melt()` .
       - Remove 'Y' from each year and convert the datatype to int.
       - Aggregate to yearly data by taking the mean temperature change for each year.
     - **Transform World Bank CO2 Data:**
       - Rename columns for consistency (e.g., 'Country Name' to 'Area').
       - Drop unnecessary columns such as 'Indicator Name' and 'Indicator Code'.
       - Reshape the data from wide to long format using `pd.melt()` .
       - Drop rows with missing CO2 emission values.
       - Convert data types for 'year' and 'co2_emissions'.
       - Rename columns to maintain consistency.
3. **Load Data**

   - **Description:** The transformed data is loaded into an SQLite database. This step ensures that the data is stored in a structured and queryable format, facilitating further analysis.
   - **Output:** An SQLite database (`pipelineDB.sqlite`) with a tables (`tempCO2`) (`temperature`) and (`CO2_emissions`) containing the transformed data.

Problems Encountered and Solutions

- **Problem:** Inconsistent data formats across datasets.
  - **Solution:** Standardized data formats during the transformation step to ensure consistency.
- **Problem:** Missing values in critical columns.
  - **Solution:** Applied imputation techniques to fill missing values where possible, or removed records with insufficient data.

Error Handling and Changing Input Data

- **Error Handling:**
  - **Logging:** Implemented comprehensive logging to track errors at each step of the pipeline.
  - **Retries:** Configured Airflow to automatically retry failed tasks up to a specified number of times.
- **Changing Input Data:**

- **Flexible Parsing:** Used flexible data parsing techniques to accommodate minor changes in data format without breaking the pipeline.

Results and Limitations

Output Data

- **Description:** The output data from the pipeline is a merged dataset containing CO2 emissions and temperature change data.
- **Format:** The data is stored in an SQLite database (`pipelineDB.sqlite`) with a table named `tempCO2` .
- **Chosen Format:** SQLite database was chosen because it is lightweight, easy to use, and suitable for handling structured data. It allows for efficient querying and data manipulation, which is ideal for analysis tasks.

In [41]:

```
# display merged table
merged_df = pd.read_sql_table('tempCO2', 'sqlite:///../data/pipelineDB.sq
merged_df.head()
```

Out[41]:

	Area	Year	Change	country_code	co2_emissions
0	Afghanistan	1990	0.714000	AFG	2046.87
1	Afghanistan	1991	0.138333	AFG	1941.37
2	Afghanistan	1992	-0.185917	AFG	1525.47
3	Afghanistan	1993	0.163000	AFG	1527.89
4	Afghanistan	1994	0.469667	AFG	1493.59

Data Quality

- **Consistency:** The data has been cleaned and standardized to ensure consistency.
- **Completeness:** Missing values have been handled appropriately, either by imputation or removal.
- **Accuracy:** The data is sourced from reputable organizations (FAO and World Bank), ensuring high accuracy.

Critical Reflection and Potential Issues

- **Data Availability:** The datasets may not cover all countries or all years uniformly, leading to potential gaps in the analysis. This could affect the comprehensiveness of the results.
- **Data Quality:** Despite cleaning, some data inaccuracies may remain due to original source errors. There might be inconsistencies in how data was recorded or reported by different countries.
- **Temporal Resolution:** The data is annual, which may not capture short-term variations or trends. More granular data could provide better insights but is not available in this case.
- **Licensing and Usage:** While the data is publicly available and used in accordance with licensing agreements, there may be limitations on its commercial use or redistribution.
- **Data Integration:** Combining datasets from different sources introduces challenges in ensuring that the merged data is harmonized correctly. Differences in data collection methods, definitions, and units can cause integration issues.