ClimateSet: A Large-Scale Climate Model Dataset for Machine Learning

Course: MC460303 - Machine Learning

BRANCH: MCA (DS & I)

Submitted by:

Aditya Raj(2446072)

Submitted to:

Dr Akshay Deepak

Authors: Julia Kaltenborn et al. (NeurIPS 2023, Datasets and Benchmarks Track)

Problem Statement:
Building a unified, ML-ready climate dataset (36 CMIP6 models) combining physical simulations and emission forcings to benchmark machine-learning climate emulators.

Goal:

Build a scalable, standardized pipeline to process CMIP6 + Input4MIPs data
→ enabling ML models to learn **mapping from emissions → temperature & precipitation**.

Challenges addressed:

- Huge diversity across CMIP6 models
- Harmonizing calendars, grids, and variables
- Computational bottlenecks (multi-TB data)

- **Input4MIPs** (Inputs for Model Intercomparison Projects) is given by **ESGF (Earth System Grid Federation)** — provides standardized *forcing fields* for models Greenhouse gases,Aerosols etc.

  CMIP6 (Coupled Model Intercomparison Project Phase 6) : It's the **official global framework** through which climate research centers share simulations of past, present, and future climate.

- Time span: **Historical (1850–2014)** + **SSP scenarios (2015–2100)**
- **SSPs (Shared Socioeconomic Pathways)-**Define how **greenhouse gases evolve (2015–2100)** and drive CMIP6 climate projections.(ex-SSP1-2.6 Low emissions,SSP5-8.5 High fossil use)
- 36 models × 4 SSPs = ~2 TB of preprocessed data

Dataset Construction Pipeline

- **Main modules:**
- Downloader (fetch CMIP6/Input4MIPs from ESGF)
- Checker (file integrity & metadata)
- Raw Processor (time/calendar fix)
- Resolution Processor (CDO-based remap to 250 km grid)
- Structure Processor (standardized coords, XMIP)

Output: Cleaned, ML-ready dataset → CO_2/CH_4/SO_2/BC → TAS/PR

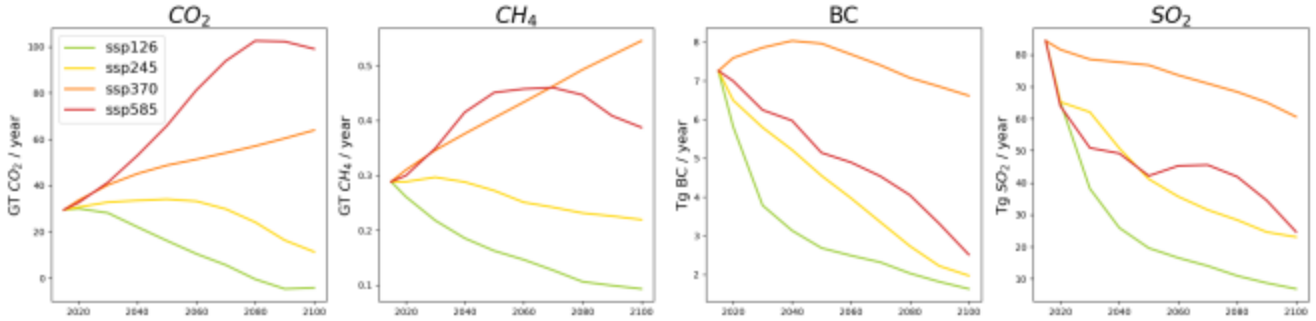


Figure 2: Forcing agent trajectories. Greenhouse gas (CO_2, CH_4), aerosol (BC), and aerosol precursor (SO_2) emission trajectories from 2015 – 2100 for different Shared Socio-economic pathways (SSPs).

Project Workflow:

- **Goal:** Build ML emulator predicting Temperature & Precipitation.
- **Steps:**
- Download & extract **NorESM2-LM (historical)** dataset
- Preprocess inputs/outputs → CO_2, CH_4, SO_2, BC → TAS, PR
- Normalize & align data (1850–2014)
- Train **MiniClimaX** (Transformer-based model)
- Predict **future SSP245 (2015–2035)**
- Evaluate with RMSE, MAE, Correlation
- *Tools:* `PyTorch`, `xarray`, `numpy`, `matplotlib`, `tqdm`

Data Preprocessing Steps :

1. **Data Extraction:**
   Unzipped `NORESM2_LM_full.zip` (CMIP6 model) → structured folders.
2. **File Cleanup:**
   Fixed invalid filenames & directory names (replaced backslashes, etc.)..
3. **Variable Selection:**
   Inputs → CO_2, CH_4, SO_2, BC *(from Input4MIPs)*
   Outputs → `Temperature (tas)`, `Precipitation (pr)` *(from CMIP6)*
4. **Unit Conversion:**
   Temperature: : **Kelvin → °C**
   Precipitation: **kg m^{-2} s^{-1} → mm/day**

5.Time Fixing:

- Created continuous monthly timeline **1850–2014** using `pandas.date_range()`.

6.Spatial Alignment:

- Interpolated all variables to the **same lat–lon grid** (96 × 144).

7.Dataset Combination:

- Merged inputs + outputs → single `xarray.Dataset` (`NORESM2_LM_historical_4in2out.nc`).

8.Normalization:

- Applied **min–max scaling (0–1)** separately for each variable.
- Saved arrays → `X_hist_norm.npy`, `Y_hist_norm.npy`.
- **Final data shape:** `(1980, 4, 96, 144)` → months × variables × lat × lon
- **Format:** NetCDF (`.nc`) — ready for model training.

Model Architecture – MiniClimaX (Transformer-based)

- “Transformer-based models use *self-attention* to learn global relationships in data. In climate science, they capture spatial and temporal dependencies between atmospheric variables —enabling faster and more accurate emulation of Earth System Models.”
- Transformer-based models like **ClimaX** or your **MiniClimaX**:
- Treat the **Earth’s grid** (latitude × longitude) as a sequence of “tokens”.
- Use **attention** to learn how global regions affect one another.
- Can generalize across **different models (ESMs)** and **scenarios (SSPs)**.

How the Transformer Works in My Project (MiniClimaX):

1.Input Data (X):

- Model takes **6 consecutive months** of 4 climate forcing variables → **CO_2, CH_4, SO_2, BC**
- Each month = a **96×144 global grid** (~250 km resolution).
- Input shape → (6, 4, 96, 144) for training.
- (B, 6, 4, 48, 72) for prediction.

2.Temporal Aggregation:

- The 6-month sequence is **averaged** to represent one "context window".
- This gives a single map showing the short-term climate forcing state.

3. **Patch Embedding (Local Features):**

- Each grid cell (lat, lon) is passed through a **1×1 convolution** → turns 4 forcings into a **192-dimensional vector** (feature embedding).
- These vectors form a sequence of **tokens** — one per spatial location.

4. **Positional Encoding (Spatial Awareness):**

- Adds information about **where** each token is on the globe (latitude/longitude).

5. **Transformer Encoder (Global Attention):**

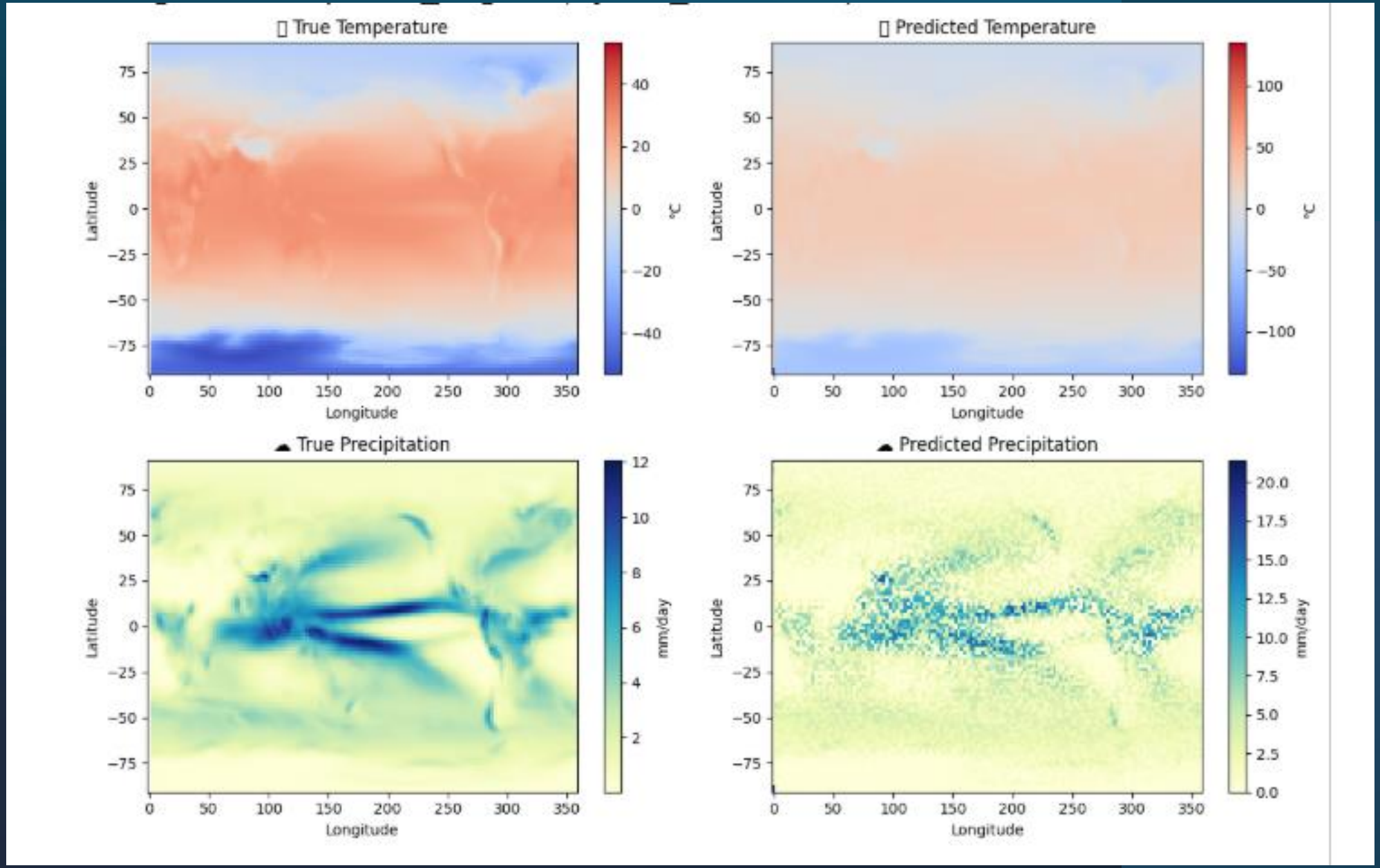
- Each token (grid cell) looks at **every other cell** using **self-attention**.
- The model learns **global dependencies** —
  e.g., CO_2 increase in one region affects temperature elsewhere.
- 5 encoder layers, 6 attention heads per layer.

6. **Output Head (Regression):**

- Final layer converts the token features → 2 output channels:
  - **Temperature (TAS)**
  - **Precipitation (PR)**
- Output reshaped back to grid → `(2, 96, 144)` map.

My Predictions vs Actual model

Metric	ClimaX (Paper)	MiniClimaX (Mine)	Comment
Temp RMSE	~3.8 °C	**9.4 °C**	Higher error due to smaller model & dataset
Temp Corr	~0.95	**0.905**	Captures trends well
Prec RMSE	~1.5 mm/day	**2.76 mm/day**	Rainfall harder to generalize
Prec Corr	~0.80	**0.508**	Lower due to single-model bias


True Temperature
Predicted Temperature
True Precipitation
Predicted Precipitation
Latitude
Longitude
°C
mm/day

- Inferences:

- My **MiniClimaX** model shows **strong temperature correlation (0.905)** — meaning it learns overall spatial–temporal patterns well.
- However, **higher RMSE** indicates that the magnitude of temperature anomalies isn’t perfectly captured.
- **Precipitation** remains more challenging because it’s **noisier and localized**, even in CMIP6 benchmarks.

Future Improvements:

No.	Proposed Improvement	Expected Benefit
1 **Multi-Model Training**	Integrate data from multiple CMIP6 models (e.g., CESM2, MIROC6, IPSL-CM6A)	Better generalization and robustness across different climate simulators.
2 **Multi-Scenario Inputs**	Include future pathways (SSP126, SSP585) during training	Enables cross-scenario emulation and long-term climate forecasting.
3 **Higher Spatial Resolution**	Upgrade from 2.5° (~250 km) to 1° (~100 km) grid	Improves regional pattern accuracy, especially for temperature and precipitation.
4 **More Input Variables**	Add surface temperature, humidity, sea-level pressure, etc.	Provides richer physical context and improves learning of global relationships.
5 **Longer Training / Fine-Tuning**	Run more epochs on GPU with early stopping and learning-rate scheduling	Reduces RMSE, stabilizes convergence, and enhances prediction accuracy.

Climate Change: A Global Crisis

Our planet is at a critical juncture. Rising temperatures, fueled by human activity, are causing unprecedented extreme weather, melting polar ice, and threatening ecosystems. Urgent, collective action is essential to mitigate these impacts and secure a sustainable future.