AutoClimDS: Climate Data Science Agentic AI — A Knowledge Graph is All You Need*

Ahmed Jaber^{†1}, Wangshu Zhu¹, Karthick Jayavelu², Justin Downes², Sameer Mohamed², Candace Agonafir¹, Linnia Hawkins¹, and Tian Zheng^{‡1}

¹NSF STC Learning the Earth with AI and Physics (LEAP), Columbia University, New York, NY, USA

²AWS Generative AI Innovation Center, Seattle, WA, USA

Abstract

Climate data science faces persistent barriers stemming from the fragmented nature of data sources, heterogeneous formats, and the steep technical expertise required to identify, acquire, and process datasets. These challenges limit participation, slow discovery, and reduce the reproducibility of scientific workflows. In this paper, we present a proof of concept for addressing these barriers through the integration of a curated knowledge graph (KG) with AI agents designed for cloud-native scientific workflows. The KG provides a unifying layer that organizes datasets, tools, and workflows, while AI agents—powered by generative AI services—enable natural language interaction, automated data access, and streamlined analysis. Together, these components drastically lower the technical threshold for engaging in climate data science, enabling non-specialist users to identify and analyze relevant datasets. By leveraging existing cloud-ready API data portals, we demonstrate that "a knowledge graph is all you need" to unlock scalable and agentic workflows for scientific inquiry. The open-source design of our system further supports community contributions, ensuring that the KG and associated tools can evolve as a shared commons. Our results illustrate a pathway toward democratizing access to climate data and establishing a reproducible, extensible framework for human-AI collaboration in scientific research.

Keywords: Knowledge Graphs; AI Agents; Climate Data Science; Generative AI; Cloud-Native Data Access; Human–AI Collaboration.

^{*}We acknowledge funding from NSF through the Learning the Earth with Artificial Intelligence and Physics (LEAP) Science and Technology Center (STC) (Award #2019625).

[†]Corresponding author. Email: amj2234@columbia.edu

[‡]Corresponding author. Email: tian.zheng@columbia.edu

1 Introduction

1.1 Current Bottlenecks in Climate Data Science

Climate research is broadly understood as the scientific investigation of Earth's climate systems, their variability, and the effects of climate change on both natural and human environments [Eyring et al., 2024]. This work often draws from a growing body of observational datasets, simulation outputs, and analytical tools that span multiple scientific domains [Gettelman et al., 2022]. The central task remains consistent: to extract meaningful, reproducible insights about how climate behaves, and how those changes interact with societal and ecological systems. This endeavor is complicated by the immense volume of climate data, which requires sophisticated methods to bridge theory with observation. The successful application of these techniques hinges on thoughtful workflow design that carefully integrates physical knowledge with data-driven approaches [Karniadakis et al., 2021, Selz and Craig, 2023]. Despite the proliferation of data sources, the research process itself remains deeply fragmented. Observational datasets and Earth System Model outputs are stored in heterogeneous formats, described through inconsistent metadata, and distributed across institutions without standardized mechanisms for access or integration. Researchers are often forced to spend significant time reconciling structural differences before scientific questions are even posed [Ceccato et al., 2012]. This not only slows progress but also limits the extent to which new computational methods are reliably integrated into climate science workflows. Perhaps more fundamentally, existing systems are designed for retrieval rather than reasoning. Climate data repositories, like NASA CMR, rely on keyword search for data retrieval [Shum et al., 2017]. These systems assume a user already knows what they are looking for. These assumptions become problematic as research questions grow more interdisciplinary and data-driven tools play a larger role in hypothesis generation and evaluation. In such settings, the inability to express research intent in ways that systems interpret and act on becomes a bottleneck.

1.2 Opportunities with Agentic AI

Agentic AI systems [Acharya et al., 2025], defined as artificial intelligence models capable of autonomous reasoning, planning, and tool use, offer a promising avenue for overcoming these limitations. Recent evidence underscores the transformative potential of human—AI collaboration, particularly in domains that demand both scale and precision. A large workplace study [Moreno, 2023] found that access to a generative AI assistant boosted worker productivity by 14% on average, with the greatest improvements observed among less-experienced users who were able to match or even surpass the performance of more seasoned colleagues. This finding is striking because it suggests that AI not only accelerate routine workflows but also actively narrows expertise gaps, allowing broader participation in complex, data-driven tasks.

In the context of climate data science, where research bottlenecks often stem from technical barriers, these results imply that Agentic AI could play a similar equalizing role—elevating entry-level researchers while simultaneously streamlining the work of experts. Beyond efficiency, the productivity gains translate into more time for higher-order reasoning and hypothesis generation, positioning AI systems as genuine collaborators rather than passive tools. Such evidence strengthens the case that knowledge-graph-enabled Agentic AI could fundamentally shift how climate research is conducted, making discovery more inclusive, reproducible, and rapid. As GenAI

continues to improve with time, the reliance on Agentic AI tools will grow too.

1.3 Related Work

Several initiatives have sought to improve climate data access and interoperability. Foundational projects like NASA Earthdata [nas], the Coupled Model Intercomparison Project (CMIP) archives [Meehl et al., 2000], and the Earth System Grid Federation (ESGF) [Williams et al., 2011] provide access to a wealth of observational and model datasets. Computational tools such as Pangeo [Odaka et al., 2019] and ESMValTool [Righi et al., 2020] have further facilitated cloud-based analysis and model evaluation, while ontologies like Semantic Web for Earth and Environmental Terminology (SWEET) [Raskin and Pan, 2005] and frameworks like GeoLink [Zhou et al., 2020] offer structured vocabularies for Earth science concepts. While invaluable, these platforms were designed as passive repositories and toolkits for human researchers. They excel at data provision but lack the integrated, machine-readable structure required to support autonomous Climate Data Science AI Agents. Such agents require a relational understanding of how to select, combine, and process that data to achieve a specific scientific goal. More critically, existing systems do not capture researcher intent or support automated, problem-aware workflow composition.

The development of knowledge graphs for climate science represents a significant step toward solving this challenge. Recent efforts such as LinkClimate [Wu et al., 2022] have demonstrated the potential of knowledge graph infrastructures to support interoperable access to observational climate datasets. However, its underlying design reflects a static view of data usage, where discovery is driven by keyword matching. The system treats datasets as endpoints for retrieval. It does not represent the procedural knowledge or scientific reasoning—such as task dependencies or model-data compatibility—that would allow an AI system to construct or adapt a workflow. As a result, while LinkClimate enhances data discovery for a human user, it does not provide the necessary framework to support agentic behavior effectively. This reveals a critical gap in existing infrastructure: the lack of a system designed explicitly to serve as the reasoning backbone for an Agentic AI. The transition from data retrieval to automated scientific discovery requires a new paradigm where the infrastructure itself encodes the knowledge needed for an agent to plan, execute, and adapt complex research pipelines. Our work addresses this gap by developing a knowledge graph tailored to empower a new class of Climate Data Science AI Agents, demonstrating that "a knowledge graph is all you need".

1.4 Contributions of This Paper

In this work, we introduce a semantic infrastructure that encodes climate data entities into a unified, queryable knowledge graph (KG). Our contribution is twofold. First, we present an ontology-driven methodology for integrating heterogeneous NASA Common Metadata Repository (CMR) records and institutional catalogs into a semantically consistent graph using OpenCypher [Green et al., 2018]. To align raw observational metadata with standardized Earth System Model variables, we develop a fine-tuned transformer classifier built on ClimateBERT [Webersinke et al., 2021], achieving 99.17% semantic accuracy in linking text to CESM variable labels [Kay et al., 2015]. Second, we demonstrate how this KG becomes the reasoning substrate for our Agentic AI system, *AutoClimDS*, capable of reproducing scientific workflows end-to-end. Given only a research objective expressed in natural language, the agent leverages the KG to autonomously

identify relevant data sources, reconcile their metadata, and execute preprocessing steps before generating analytical outputs such as figures and graphs. In doing so, the system replicates results from published climate studies, thereby illustrating a new paradigm of AI-driven scientific reproducibility. Through this design, our approach addresses three key goals of climate data science: enabling intent-aware data discovery, streamlining data acquisition, and supporting reproducible climate modeling.

2 Method

2.1 Graph Ontology

To enable semantic reasoning over heterogeneous climate datasets, we define a domain-specific ontology that captures the conceptual schema of climate research workflows. Metadata from the NASA Common Metadata Repository (CMR) was ingested via their API and structured into a JSON format, where each entry corresponds to a dataset and its associated metadata fields. Based on this representationn, we constructed the structural backbone of the knowledge graph using a combination of entity-centric (e.g., datasets, variables, model components) and process-centric nodes (e.g., workflows, analytical tasks). This hybrid design allows the graph to encode both the content of the scientific data and also their procedural context, enabling reasoning over both what data exist and how it can be operationalized within research pipelines.

2.1.1 Dual-Format Metadata Integration

The foundation of our knowledge graph construction rests on a data fusion that harmonizes NASA's Common Metadata Repository (CMR) dual-format architecture. Let $\mathcal{D}_{\mathtt{JSON}}$ and $\mathcal{D}_{\mathtt{UMM}}$ represent the sets of metadata records retrieved from the collections. json and collections.umm_json endpoints, respectively. For each dataset with concept identifier c, we define the merge operation as:

$$\mathcal{M}(c) = \text{merge}(\mathcal{D}_{\text{JSON}}[c], \mathcal{D}_{\text{UMM}}[c]) \tag{1}$$

where the merge function implements a field-wise preference hierarchy:

$$\text{field}(c) = \begin{cases} \mathcal{D}_{\text{UMM}}[c].\text{field} & \text{if } \mathcal{D}_{\text{UMM}}[c].\text{field} \neq \emptyset \\ \mathcal{D}_{\text{JSON}}[c].\text{field} & \text{otherwise.} \end{cases}$$
 (2)

This strategy ensures maximal metadata completeness while preserving the structured semantics of the Unified Metadata Model format. The resulting harmonized dataset $\mathcal{H} = \{\mathcal{M}(c) : c \in \mathcal{C}\}$ contains approximately $|\mathcal{F}| \approx 60$ standardized attributes per record, where \mathcal{C} represents the set of all concept identifiers and \mathcal{F} the attribute schema.

2.1.2 Geospatial Processing

To enable consistent spatial reasoning across heterogeneous datasets, we developed a geospatial processing pipeline that standardizes spatial representations in CMR and resolves their geographic context via boundary-based classification. The pipeline consists of three stages: (1) geometry standardization, (2) boundary-based classification, and (3) spatial scope determination. Each dataset

record $d \in \mathcal{H}$ is transformed into a standardized polygonal representation. The pipeline supports three types of CMR spatial formats: bounding boxes, complex polygons, and point coordinates.

Bounding boxes are converted into polygons as follows:

$$\mathcal{B}(b_i) = \text{Polygon}\{(w, s), (e, s), (e, n), (w, n), (w, s)\}, \quad b_i = [s, w, n, e]. \tag{3}$$

Complex polygons are parsed from alternating latitude—longitude pairs:

$$\mathcal{P}(p_j) = \text{Polygon}\{(\lambda_k, \phi_k) : \phi_k = c_{2k}, \lambda_k = c_{2k+1}\}$$

$$\tag{4}$$

where ring closure is enforced when necessary.

All shapes are unified into a single geometry:

$$\mathcal{U}(d) = \begin{cases} \text{shapes}[0] & |\text{shapes}| = 1, \\ \text{unary_union(shapes)} & |\text{shapes}| > 1, \\ \text{None} & |\text{shapes}| = 0. \end{cases}$$
 (5)

Only valid polygonal geometries are retained:

$$\mathcal{G}(d) = \begin{cases} \mathcal{U}(d) & \mathcal{U}(d) \in \{\text{Polygon}, \text{MultiPolygon}, \} \\ \text{unary_union}(\{g \in \mathcal{U}(d).\text{geoms}\}), & \mathcal{U}(d) = \text{GeometryCollection} \\ \text{None} & \text{otherwise.} \end{cases}$$

Boundary-Based Geographic Classification. To associate geometries with geographic entities, we employ boundary-based classification using offline boundary datasets and an R-tree spatial index [Zhang et al., 2007]. For a dataset geometry $g = \mathcal{G}(d)$, candidate boundaries are efficiently retrieved via bounding-box intersection:

$$\mathcal{I}_{\texttt{candidates}}(g) = \{i : \texttt{bbox}(\mathcal{B}_{\texttt{world}}[i]) \cap \texttt{bbox}(g) \neq \emptyset\}. \tag{7}$$

Candidate geometries are then filtered using GeoPandas [Jordahl et al., 2021] overlay operations:

$$\mathcal{C}_{\text{intersect}}(g) = \text{overlay}(\text{GeoDataFrame}([g]), \text{GeoDataFrame}(\mathcal{C}_{\text{potential}}(g)), (8)$$

how =' intersection').

Geographic labels are derived as:

$$\mathcal{N}_{\text{countries}}(g) = \{ \text{name}(c) : c \in \mathcal{C}_{\text{intersect}}(g) \},$$

$$\mathcal{N}_{\text{continents}}(g) = \{ \text{continent_map}[n] : n \in \mathcal{N}_{\text{countries}}(g) \}.$$
(10)

Spatial Scope Determination. Spatial scope is classified hierarchically:

$$\text{scope}(g) = \begin{cases} \text{ocean} & \mathcal{C}_{\text{intersect}}(g) = \emptyset, \\ \text{global} & |\mathcal{N}_{\text{continents}}(g)| > 1, \\ \text{continental} & |\mathcal{N}_{\text{continents}}(g)| = 1 \land |\mathcal{N}_{\text{countries}}(g)| > 1, \\ \text{country} & |\mathcal{N}_{\text{countries}}(g)| = 1, \\ \text{multinational} & |\mathcal{N}_{\text{countries}}(g)| > 3, \\ \text{regional} & 1 < |\mathcal{N}_{\text{countries}}(g)| \leq 3. \end{cases}$$

$$(11)$$

Implementation Details. The pipeline runs in offline-first mode (use_geocoding_api = False) to ensure reproducibility; records without boundary data are returned with scope value being "unclassified". The R-tree spatial index (BOUNDARIES_SINDEX) reduces candidate searches from O(n) to $O(\log n)$ across 258 global boundaries. GeoPandas overlay operations then refine results with minimal computational overhead, enabling efficient large-scale geospatial classification across heterogeneous climate datasets. This allows for high scalability and faster dataset processing.

2.1.3 Resolution Extraction

Resolution metadata is extracted using predefined keyword lists and regex pattern matching. This is necessary since the NASA CMR API does not directly come with resolution fields. The system searches for spatial resolution using 13 predefined attribute names (e.g., spatial_resolution, grid_spacing, dx) and temporal resolution using 9 terms (e.g., time_resolution, dt, frequency). When structured attributes are unavailable, regex patterns match resolution information in text fields: spatial patterns capture formats like "\d+\s*km" and "\d+\s*degree", while temporal patterns match terms like "daily" and "weekly". Rather than extracting just numeric values, the system returns full sentences containing resolution context, preserving semantic meaning for downstream processing.

2.1.4 Transformer-Based Semantic Variable Mapping.

A critical component of the Knowledge Graph involves establishing relationships between NASA CMR datasets and standardized Earth System Model variables. To bridge this gap, we developed a transformer-based classification system that learns to predict CESM variable names from CESM variable descriptions, which can then be applied to match similar textual content in CMR dataset metadata. We formulate this as a multi-class classification problem over the CESM variable space $\mathcal{V}_{\text{CESM}}$ with $|\mathcal{V}_{\text{CESM}}| = 2,308$ distinct variables.

2.1.5 Model Architecture and Training Data

Our variable prediction system builds on ClimateBERT (distilroberta-base-climate-f) Webersinke et al. [2021], a domain-adapted language model pre-trained on climate science literature. The classifier $f_{\theta}: \mathcal{X} \to \mathcal{V}_{\text{CESM}}$ employs the following formulation:

$$\mathbf{h} = \text{DistilRoBERTa}_{\text{climate}}(\mathbf{x}),$$
 (12)

$$\mathbf{p} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \mathbf{h}_t \odot \mathbf{m}_t, \tag{13}$$

$$\mathbf{z} = \text{Linear}(\text{Dropout}(\mathbf{p}, p = 0.3)),$$
 (14)

$$\hat{y} = \arg\max_{i} \operatorname{softmax}(\mathbf{z})_{i}, \tag{15}$$

where $\mathbf{h} \in \mathbb{R}^{L \times d}$ denotes contextualized representations with hidden dimension d = 768, \mathcal{T} represents valid token positions, \mathbf{m} is the attention mask, and \odot indicates attention-masked mean pool-

ing over real tokens. Training minimizes cross-entropy loss with Adam optimization ($\alpha = 1 \times 10^{-5}$, batch size 16, 50 epochs).

Training data consists of 2,308 CESM variables extracted from model documentation, representing atmospheric, oceanic, land, and ice component processes. Analysis reveals systematic redundancy where of the 2,289 variables tested, 1,981 represent unique variable names with 1,830 distinct descriptions, indicating substantial description overlap where multiple variables share identical descriptions but differ in technical specifications such as aerosol species, grid levels, or temporal averaging.

2.1.6 Similarity-Based Semantic Clustering

To address the semantic redundancy inherent in the CESM variable space, we implement an automated clustering algorithm based on string similarity analysis. Let $S: \mathcal{V}_{\text{CESM}} \times \mathcal{V}_{\text{CESM}} \to [0,1]$ denote the SequenceMatcher similarity function. We define the similarity relation:

$$similar(v_i, v_j) \Leftrightarrow S(normalize(desc(v_i)), normalize(desc(v_j))) \ge \tau_d \lor S(v_i, v_j) \ge \tau_n,$$
(16)

where $\tau_d=0.7$ and $\tau_n=0.8$ represent the description and name similarity thresholds, respectively. The normalization function removes component prefixes and standardizes text formatting.

The clustering algorithm constructs an undirected graph $G = (\mathcal{V}_{\text{CESM}}, E)$ where $(v_i, v_j) \in E$ if $\text{similar}(v_i, v_j)$. Connected components of G form semantic clusters $\mathcal{C} = \{C_1, C_2, \dots, C_k\}$ where each $C_i \subseteq \mathcal{V}_{\text{CESM}}$ represents variables describing identical physical processes.

2.1.7 Evaluation

Our classification system achieves strong performance on the CESM self-validation task. Let $\mathcal{A}_{\text{exact}}$ and $\mathcal{A}_{\text{group}}$ denote exact match and similarity-group accuracies, respectively. The baseline exact match accuracy is:

$$\mathcal{A}_{\text{exact}} = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{d \in \mathcal{D}_{\text{test}}} \mathbb{I}[f_{\theta}(d) = y_d] = 93.45\%, \tag{17}$$

where $\mathbb{I}[\cdot]$ represents the indicator function and y_d the ground truth label for dataset d. With similarity grouping, the group accuracy improves significantly:

$$\mathcal{A}_{\text{group}} = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{d \in \mathcal{D}_{\text{test}}} \mathbb{I}[\text{cluster}(f_{\theta}(d)) = \text{cluster}(y_d)] = 99.87\%. \tag{18}$$

This represents a $\frac{\mathcal{A}_{\text{group}} - \mathcal{A}_{\text{exact}}}{1 - \mathcal{A}_{\text{exact}}} = 98\%$ reduction in error rate, with only $|\mathcal{E}| = 3$ variables (0.13%) remaining unmatched.

2.1.8 Text Inference

Metadata from CMR datasets—including titles, summaries, abstracts, DataCategory summaries, and Variable fields—is combined with platform and instrument information. The text is segmented into 2–9 word n-grams, filtered to retain up to 20 meaningful climate-related tokens per dataset. These tokens are then input into a transformer-based classifier to generate hasCESMVariable relationships for each corresponding dataset

2.2 Graph Construction

The final, pre-processed metadata is stored as a unified JSON document, which is then transformed into OpenCypher-compatible [Green et al., 2018] CSVs for ingestion by Amazon Neptune Analytics. The graph construction process consists of two main phases: node synthesis and relationship generation.

First, the diverse set of entities are materialized as nodes in the graph. This includes the primary data-centric concepts (directly retrieved from the json) outlined in **Table 1** (e.g., Dataset, Platform, Variable) and the procedural workflow schemas from **Table 2**. Each node is assigned a globally unique ID using deterministic content-hashing to prevent duplication. To enable semantic querying, text embeddings are generated for vertices marked for embedding in the tables and stored as Neptune vectors. Following node synthesis, a comprehensive set of relationships is generated to interconnect the graph, as detailed in **Table 3**. These edges include structural links, such as hasPlatform and hasVariable, which connect datasets to their constituent parts. ML-assisted links are also forged; for instance, the crucial hasCESMVariable edge is created by the ClimateBERT classifier to bridge observational data with the standardized CESM vocabulary. Finally, a procedural layer is integrated by connecting the workflow nodes from **Table 2** to relevant data entities using the specialized workflow relationships, giving the graph a dynamic, operational dimension ready to support agentic reasoning.

2.3 AutoClimDS Agentic AI Architectures

The knowledge graph provides grounding for the use of Agentic AI in climate data science applications. In this paper, we propose a Climate Data Science Agentic AI with three core objectives: *Data Discovery, Data Acquisition*, and *Climate Modeling and Analytics*. These objectives are achieved by using a combination of Agentic Tools and Resources. The system is implemented using LangChain [LangChain] and ReAct-style reasoning [Yao et al., 2023], with Bedrock Claude Sonnet 4 [Amazon Web Services and Anthropic, 2025] serving as the core model.

2.3.1 Data Discovery Agent

The agent implements semantic dataset discovery by encoding research queries into 384-dimensional vectors through the sentence-transformers model. The embedding process applies multi-layer transformer attention mechanisms that capture contextual relationships between climate terminology, mapping input query text q to vector $\mathbf{v}_q \in \mathbb{R}^{384}$ through attention-weighted token aggregation and mean pooling operations. The core vector search functionality operates through Neptune Analytics' native topkByEmbedding() procedure [Amazon Web Services, 2025], which implements hierarchical navigable small world graphs for efficient approximate nearest neighbor retrieval. The algorithm executes through OpenCypher query construction. This approach leverages Neptune's internal cosine similarity computation without requiring custom similarity implementations, ensuring scalable vector search across large knowledge graphs containing thousands of climate datasets. The mathematical foundation relies on Neptune's optimized similarity scoring that maintains semantic consistency by ensuring climatically related concepts cluster appropriately in the 384-dimensional embedding space. The agent implements intelligent search routing through embedding availability detection, where node types are categorized

into vector-enabled and text-only categories. Vector-enabled types include <code>DataCategory</code>, <code>Variable</code>, <code>CESMVariable</code>, <code>ScienceKeyword</code>, <code>Location</code>, <code>TemporalResolution</code>, <code>SpatialResolution</code>, and various workflow node types, while remaining node types utilize text-based Neptune query matching. The routing decision operates through conditional branching where vector search executes for embedding-enabled nodes and text search provides coverage for non-embedding nodes, ensuring comprehensive search capability across the entire knowledge graph schema.

The agent extends single-criterion search through composite multi-criteria functionality that combines vector search results with relationship-based filtering. This addresses the real-world complexity of climate dataset selection where researchers must consider temporal coverage, spatial resolution, variable availability, and institutional provenance simultaneously. The multi-criteria algorithm constructs complex OpenCypher queries that incorporate vector search results as node constraints while applying additional filtering through temporal overlap detection, spatial relationship traversal, and organizational affiliation matching. The temporal filtering mechanism implements mathematical interval intersection testing where dataset temporal bounds $[t_{\text{start}}, t_{\text{end}}]$ undergo comparison with query temporal constraints. For temporal extent queries, the algorithm applies date comparison logic through OpenCypher date functions, supporting after, before, and between temporal specifications. The spatial constraint processing leverages vector search on Location nodes to identify relevant geographic regions, then applies relationship traversal to discover datasets associated with those locations through hasLocation relationships. Unlike static query interfaces, the agent operates with a degree of autonomy in dataset discovery. It can dynamically reformulate queries, adjust search strategies (vector vs. text-based), and traverse the graph flexibly to maximize relevance. Once suitable datasets are identified, their metadata and relationship information are stored in a local SQLite database. This persistent memory allows the system to recall previously retrieved datasets, accelerate future queries, and support iterative research workflows by maintaining continuity across sessions.

2.3.2 Data Acquisition and Processing Pipeline Agent

Once the knowledge graph returns a set of relevant datasets, the Agentic AI system transitions from discovery to acquisition. Each dataset entry in the research database is inspected, and the agent retrieves the corresponding links from the hasLink relationship of the dataset node. These links may point to NASA Earthdata, NOAA archives, or AWS Open Data S3 buckets. Based on the research query q, the Agent determines the next action $a \in \mathcal{A}$, where

$$\mathcal{A} = \{\texttt{retrieve}, \texttt{preprocess}, \texttt{analyze}\}.$$

If $a=\mathtt{retrieve}$, the agent invokes the appropriate API (with authentication handled via preconfigured tokens) to fetch the dataset. The raw data, denoted $D=\{d_1,d_2,\ldots,d_n\}$, may arrive in heterogeneous formats such as CSV, NetCDF, HDF, or JSON. An automated transformation function $T:D\mapsto \hat{D}$ standardizes the collection into tabular or array-based structures, enabling interoperability. Quality validation steps are expressed as a constraint-checking function $V(\hat{D}) \in \{0,1\}$, which enforces link validity, accessibility, and structural consistency. Only datasets satisfying $V(\hat{D})=1$ are retained for downstream workflows. These steps are executed through the CodeExecution tool, which grants the Data Acquisition Agent controlled access to a secure runtime environment for code execution and data manipulation. The adaptive nature

of the pipeline allows for domain-specific preprocessing, including spatial subsetting, temporal aggregation, or variable-level transformations, as well as analytics such as automated graphing.

By integrating discovery, retrieval, validation, and preprocessing into a single agent-driven workflow. This makes data acquisition not only reproducible and cloud-resilient but also adaptive to evolving research needs. In particular, the linkages between the graph and cloud-hosted data repositories ensure persistence, while the transformation T and validation V guarantee that datasets are continuously usable for climate modeling or machine learning applications.

2.3.3 Climate Modeling and Analytics Agent

The CESM LENS Climate Ensemble modeling and analytics agent provides direct access to the Community Earth System Model Large Ensemble (CESM-LENS) dataset, enabling sophisticated climate modeling workflows through automated ensemble analysis and uncertainty quantification. The agent operates on the 40-member CESM-LENS ensemble spanning 1920-2100, with historical simulations (1920-2005) and RCP8.5 future projections (2006-2100) stored in AWS S3 at approximately 1-degree global resolution across atmospheric, oceanic, land, and ice components.

Ensemble Data Access and Optimization. The climate modeling pipeline implements efficient data loading which leverages the official Intake-ESM catalog for standardized access to CESM-LENS datasets. The system automatically handles CESM's non-standard calendar format through cftime [cftime] decoding, converting temporal coordinates to pandas-compatible datetime objects while preserving calendar-specific attributes. For large datasets exceeding memory constraints, the agent applies intelligent subsampling strategies where spatial resolution is reduced by factor s and temporal resolution by factor t, maintaining statistical representativeness while ensuring computational feasibility.

The data transformation pipeline converts multidimensional xarray [Hoyer and Hamman, 2017] datasets to efficient tabular format through function, which implements chunked processing to handle datasets exceeding available memory. The algorithm processes ensemble data in chunks of size

$$C = \min(10^6, \frac{M_{available}}{5 \cdot \text{sizeof(float32)}})$$

where $M_{available}$ represents available system memory, ensuring stable processing of large climate datasets. The resulting Polars DataFrame [Polars Developers, 2025] structure enables vectorized operations for subsequent ensemble analysis while maintaining memory efficiency through columnar storage.

Observational-Model Integration. The climate modeling workflow leverages the knowledge graph's semantic structure to bridge observational datasets with CESM-LENS simulations through automated discovery and variable mapping. The system initiates observational dataset retrieval by querying the knowledge graph for datasets D_{obs} that contain hasCESMVariable relationships, identifying observational records already linked to CESM variable nomenclature. When researchers specify observational dataset identifiers, the agent traverses knowledge graph relationships to extract connected CESM variables, their metadata, and temporal constraints $[t_{start}^{obs}, t_{end}^{obs}]$ from the observational record. This relationship-driven approach enables the system to automatically determine which CESM-LENS variables correspond to the observational measurements,

then execute targeted queries against the CESM ensemble using the CESMLENSDataTool with matching temporal bounds and spatial domains. The knowledge graph thus serves as the semantic bridge that transforms observational dataset specifications into executable CESM-LENS queries, ensuring variable compatibility and temporal alignment between observed and modeled climate data.

Spatial and Temporal Subsetting Pipeline. Spatial subsetting capabilities enable regional climate analysis through coordinate-based filtering, where geographical constraints $(lat_{min}, lat_{max}) \times (lon_{min}, lon_{max})$ are applied during data loading to reduce computational overhead. The agent handles longitude convention differences automatically, converting between [-180°, 180°] and [0°, 360°] coordinate systems as needed for seamless integration with different climate datasets. Temporal subsetting operates through year-based filtering that respects CESM experiment boundaries, automatically selecting appropriate experiments (20C for historical periods, RCP85 for future scenarios) based on requested time ranges, ensuring temporal continuity while maximizing data availability across the full CESM-LENS temporal domain.

The climate modeling and analytics agent architecture enables researchers to transition seamlessly from dataset discovery through the knowledge graph to ensemble analysis of CESM-LENS simulations, providing a comprehensive framework for climate research that spans observational data integration, model analysis, and uncertainty quantification within a unified Agentic AI system.

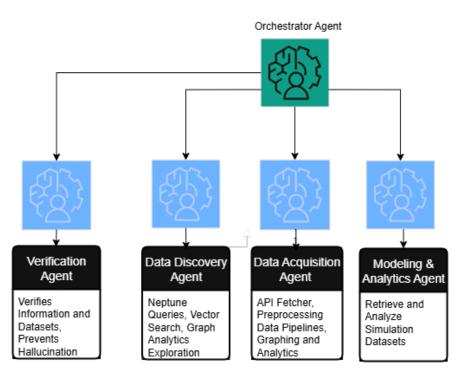


Figure 1: An overview of the multi-agent system architecture.

2.3.4 AutoClimDS: Multi-Agentic AI System

The multi-agentic AI system (Figure 1 is designed to modularize climate research tasks through a central Orchestrator Agent, which interprets user objectives, maintains research state, and delegates sub-tasks to specialized agents. The Data Discovery Agent queries the *Knowledge Graph* using semantic search to identify relevant resources, while the Data Acquisition Agent retrieves and preprocesses datasets from sources such as AWS OpenData S3. The Climate modeling and analytics agent then integrates these datasets with climate model ensembles to produce harmonized, comparable simulations. Supporting this pipeline, a *Verification Agent* serves as an automated peer reviewer, validating data quality, logical consistency, and adherence to physical constraints. Together, these agents form a coordinated workflow that mirrors the collaborative dynamics of a human research team.

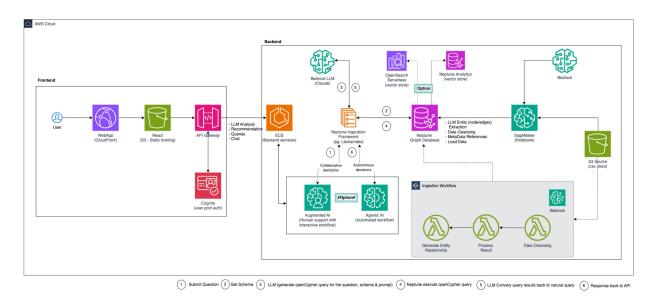


Figure 2: End-to-end architecture for our deployed, scalable AutoClimDS Agentic AI knowledge graph pipeline on AWS. The frontend (CloudFront, React, API Gateway, and Cognito) enables user interaction, while the backend integrates Bedrock LLMs, Neptune Graph Database, and SageMaker for ingestion, querying, and reasoning. Data flows include schema extraction, entity-relationship ingestion, natural language to OpenCypher translation, and Agentic AI workflows for collaborative or autonomous decision-making.

2.4 Cloud Deployment and Workflow

To ensure scalability, adaptability, and continuous integration of new climate datasets, the proposed AutoClimDS system is deployed within an AWS-based architecture (Figure 2). The workflow follows a modular design: data are periodically ingested from external sources into Amazon S3, transformed into a standardized CSV-based format, and harmonized through automated routines executed via AWS Lambda. This standardized representation enables consistent integration of heterogeneous datasets and allows the knowledge graph to be continuously updated as new data

becomes available. Each dataset is enriched according to the ontology and stored in Amazon Neptune, which supports both symbolic querying and vector-based similarity search through Neptune Analytics.

Future Scalability and Computational Considerations: The architecture is designed to accommodate future growth in both data volume and analytical complexity. The ingestion pipeline can be extended to handle streaming or real-time data, while graph and machine learning components can scale through Amazon ECS and SageMaker for more advanced analytics. Additional components, such as Neptune Analytics for high-throughput GraphRAG search or Bedrock for LLM-powered query generation, can be seamlessly integrated without major architectural changes. While leveraging cloud services provides elasticity and ensures accessibility across domains, it also introduces computational costs tied to storage, data transfer, and on-demand processing. These costs scale with dataset size, update frequency, and analytical intensity, highlighting the trade-off between maintaining an up-to-date, continuously enriched knowledge graph and the resources required to sustain such infrastructure. AWS offers a *Pricing Calculator* that allows you to estimate the cost of various AWS services based on your specific usage requirements.

3 Case Study: Sea Level Trends

To demonstrate the effectiveness of our AutoClimDS system, we replicate selected figures and graphs from the *New York City Climate Risk Information 2022 (NPCC4)* report [Braneon et al., 2024]. These replications validate the ability of the Knowledge Graph and Agentic AI Pipeline to reproduce key climate risk indicators using openly available data and automated workflows. Data were found using natural language queries to the Data Discovery Agent, while data was loaded and plotted using natural language instructions to the Data Acquisition Agent.

Figure 3 and Figure 5a show sea level trends produced using the AutoClimDS system through only natural language queries. AutoClimDS successfully replicated the original NPCC4 figures (Figure 4 and Figure 5b) from Braneon et al. [2024]. Following natural language instructions, the Agentic AI system ran the entire workflow, from data discovery to data acquisition. The natural language instructions followed a three-part structure:

- 1. **Objective** the broad analytical task (e.g., "analyze historical sea level change for New York City").
- 2. **Context/Constraints** scope of analysis such as time span, geographic region, or variable of interest.
- 3. **Desired Output** specification of both the *form of the result* (e.g., figure, time series plot, table) and the *measures to be extracted* (e.g., long-term trend, annual anomalies, regression slope).

Crucially, no underlying datasets, numerical values, or regression coefficients were provided by the user. The agents autonomously located appropriate datasets, carried out preprocessing, calculated the requested measures, and generated the figures. By using AutoClimDS, the user receives not only the figures themselves but also the underlying datasets locally, along with a file containing

information about the data sources and interactive agents that can acquire other similar datasets, transform the data in different ways, and analyze the data further. This prompt structure highlights how AutoClimDS transforms high-level natural language requests into complete analytical workflows. The Modeling and Analytics agent reproduced the linear sea level rise trend and overall graph structure with high fidelity. In particular, the replicated graph aligns exactly with the original in terms of numerical trends, and, in some aspects, the agent-generated version demonstrates improved precision. All logs, data files, and figures are available on the GitHub repository.

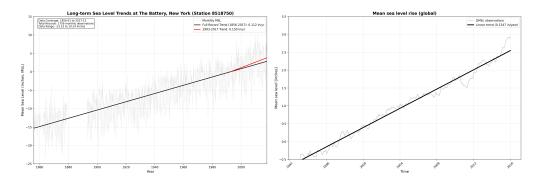


Figure 3: AutoClimDS replicated NPCC4 observed sea level trends.

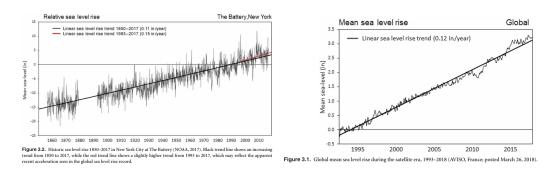


Figure 4: Original figures of sea level trends from Braneon et al. [2024], under the terms of the CC BY-NC license.

In this case study, AutoClimDS demonstrated a nuanced understanding of the analytical objective, distinguishing between observed and corrected sea level rise, correctly interpreted specialized climate terminology, such as vertical land motion (VLM) and handled higher-level statistical considerations inherent in the data.

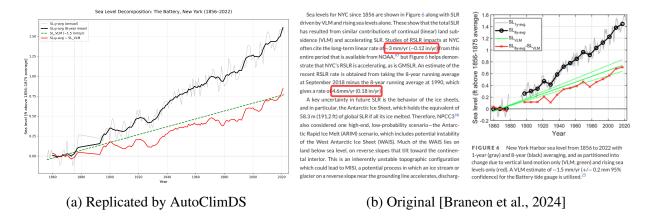


Figure 5: Sea level trends with SLR driven by VLM and rising sea levels alone.

4 Open Science and Community Contributions

A central goal of this project is also to advance open science by making our proof of concept fully accessible and extensible by the community. Reproducibility, transparency, and accessibility are guiding principles: by sharing both the knowledge graph (KG) and the AI agent workflows, we aim to lower barriers not only for end-users but also for researchers and practitioners who wish to build upon this work.

All resources associated with this proof of concept are openly available through a GitHub repository. The repository includes the KG schema and seed entries, example AI agent workflows, scripts for data access and processing, as well as documentation and tutorials. These materials allow users to reproduce our experiments and adapt the workflows for their own scientific inquiries.

The repository also serves as a hub for community contributions. We welcome contributions through GitHub pull requests in several forms:

- Submission of new KG entries, such as datasets, workflows, or domain-specific ontologies;
- addition of scripts and tools for data acquisition or processing;
- improvements to documentation, tutorials, or examples; and
- reporting issues or suggesting enhancements through the GitHub Issues interface.

A list of open issues can be found in our GitHub repository cited in the paper.

To maintain quality and consistency, contributions are reviewed before incorporation. The KG and agent framework are designed for modularity, so that new resources can be integrated without disrupting existing workflows. This ensures that the system can evolve as a shared commons, enriched by a diversity of perspectives and expertise.

Looking ahead, we envision the repository as a foundation for collaborative growth. By lowering the technical threshold for participation, the project invites contributions not only from climate scientists and data scientists but also from educators, students, and citizen science communities. Future activities such as community challenges can further expand the resource base and strengthen the socio-technical infrastructure of open, reproducible science.

¹https://github.com/Ajaberr/AutoClimDS

5 Conclusion

In this paper, we built a *proof of concept* for utilizing a well-curated knowledge graph (KG) to develop highly capable AI agents for climate data science workflows. We demonstrated that these agents can substantially lower barriers to data identification, acquisition, processing, and analysis for non-technical users. The system we built integrates cloud-ready API data portals with generative AI tools from AWS, illustrating that "a knowledge graph is all you need" can be a reachable vision for Agentic AI workflows in scientific inquiry. By leveraging the KG as both an extensible memory and a unifying reasoning layer across tools and datasets, our approach aligns with existing cloud data science solutions while creating space for community contributions in the form of KG entries, data access utilities, and workflow scripts.

Beyond the technical proof of concept, this work highlights the broader potential of KGs and AI agents to democratize climate data science. Lowering technical barriers opens opportunities for participation from policy, education, and citizen science communities, while the modular and cloud-native design ensures scalability and interoperability. Crucially, the KG provides a foundation for an evolving, community-driven commons that can grow with new datasets, tools, and domain knowledge, positioning it as both a technical and socio-technical infrastructure for collaborative science. Looking forward, this approach provides a pathway toward integrating more advanced reasoning capabilities, fostering reproducibility, and ultimately accelerating discovery through human–AI partnerships in climate and beyond.

Table 1: Node classes defined in the Knowledge Graph

Node label	Description	Emb.
Dataset	One CMR collection: identifiers, title, DOI, temporal coverage, spatial footprint, datacentre flags.	No
DataCategory	Text summary of a dataset.	Yes
DataFormat	Physical file/packaging format (e.g. NetCDF-4, HDF-EOS2, GeoTIFF).	No
CoordinateSystem	CRS / projection definition (name, datum, units).	No
Location	Bounding boxes, polygons or points plus derived place-names.	Yes
Station	Ground station or deployment site linked to a dataset.	No
Organization	Data provider, processing centre, or programme office.	No
Platform	Spacecraft, aircraft, float, buoy, or other carrier of instruments.	No
Consortium	Multilateral project or data-sharing alliance.	No
TemporalExtent	Start and end timestamps and last-update time.	No
Variable	Native CMR variable extracted per dataset (name, units, description).	Yes
CESMVariable	Canonical CESM variable (domain, component, long-name, units).	No
Component	CESM Model Components (ATM, OCN, LND, ICE, ROF, GLC, WAV).	No
Contact	Person or group: name, roles, email/phone, affiliation.	No
Project	NASA/NOAA mission or research campaign.	No
Link	Ancillary link (documentation, OPeNDAP, visualiser, etc.).	No
SpatialResolution	Parsed horizontal grid size with units.	Yes
TemporalResolution	Parsed reporting or output frequency (e.g. 3-hourly).	Yes
ScienceKeyword	Controlled GCMD / SWEET keyword hierarchy levels 0–3.	Yes
ProcessingLevel	NASA processing level 0–4 descriptor.	No

Table 2: Workflow node classes defined in the $Knowledge\ Graph$

Workflow Node Label	Description	
SurrogateModelingWorkflow	Learned surrogate model trained to emulate physical simulations.	Yes
HybridMLPhysicsWorkflow	Hybrid system combining ML components with physics-based simulation.	Yes
EquationDiscoveryWorkflow	Process that extracts governing equations from data.	Yes
ParameterizationBenchmark	Evaluation setup for comparing parameterization methods.	Yes
UncertaintyQuantification	Process for estimating predictive uncertainty.	Yes
ParameterInferenceWorkflow	Inference system for estimating physical parameters from data.	Yes
SubseasonalForecastingWorkflow	Forecasting models for 2–6 week prediction horizons.	Yes
TransferLearningWorkflow	Transfer pipeline from synthetic to observational datasets.	Yes

Table 3: Relationship (edge) types in the Knowledge Graph

Edge Label	$\textbf{From} \rightarrow \textbf{To}$	Meaning / semantics
hasDataCategory	Dataset → DataCategory	Links a dataset to its summary.
hasDataFormat	$Dataset \rightarrow DataFormat$	Declares the physical file format.
usesCoordinateSystem	$Dataset \rightarrow CoordinateSystem$	Specifies the projection of the dataset.
hasLocation	$Dataset \rightarrow Location$	Attaches the spatial footprint (boxes / polygons / points).
hasStation	$Dataset \rightarrow Station$	Associates ground station or platform deployment sites.
hasOrganization	Dataset \rightarrow Organization	Producing or archiving centre.
hasPlatform	$Dataset \rightarrow Platform$	Carrier of the measuring instrument(s).
hasConsortium	$Dataset \rightarrow Consortium$	Higher-level project or alliance.
hasTemporalExtent	$Dataset \rightarrow TemporalExtent$	Time coverage start / end.
hasVariable	$Dataset \rightarrow Variable$	Native CMR variables parsed directly from metadata.
hasCESMVariable	Dataset \rightarrow CESMVariable	ML-predicted CESM counterparts.
hasSpatialResolution	$Dataset \rightarrow SpatialResolution$	Parsed horizontal grid spacing.
hasTemporalResolution	$Dataset \rightarrow Temporal Resolution$	Parsed reporting or output frequency.
hasProcessingLevel	$Dataset \rightarrow ProcessingLevel$	NASA processing level (0–4).
hasLink	$Dataset \rightarrow Link$	Ancillary documentation / access links.
hasProject	$Dataset \rightarrow Project$	NASA/NOAA mission or research campaign.
hasScienceKeyword	$Dataset \rightarrow ScienceKeyword$	Controlled GCMD / SWEET tags.
hasContact	$Dataset \rightarrow Contact$	Person or group responsible for dataset.
belongsToComponent	$CESMVariable \rightarrow Component$	Maps a CESM variable to its parent model component.
describesVariable	$Science Keyword \rightarrow CESM Variable$	Semantic bridge from keyword hierarchy to CESM variable.
operatesAtLocation	Platform \rightarrow Location	Deployment area of the platform.
worksForOrganization	$Contact \rightarrow Organization$	Staff affiliation / stewardship.
belongsToConsortium	Organization \rightarrow Consortium	Organization's membership in a larger alliance.
similarCESMVariables	$CESMVariable \rightarrow CESMVariable$	Groups similar CESM variables together based on string similarity.

References

- Veronika Eyring, William D Collins, Pierre Gentine, Elizabeth A Barnes, Marcelo Barreiro, Tom Beucler, Marc Bocquet, Christopher S Bretherton, Hannah M Christensen, Katherine Dagon, et al. Pushing the frontiers in climate modelling and analysis with machine learning. *Nature Climate Change*, pages 1–13, 2024.
- Andrew Gettelman, Alan J Geer, Richard M Forbes, Greg R Carmichael, Graham Feingold, Derek J Posselt, Graeme L Stephens, Susan C Van den Heever, Adam C Varble, and Paquita Zuidema. The future of earth system prediction: Advances in model-data fusion. *Science Advances*, 8(14):eabn3488, 2022.
- George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.
- Tobias Selz and George C Craig. Can artificial intelligence-based weather prediction models simulate the butterfly effect? *Geophysical Research Letters*, 50(20):e2023GL105747, 2023.
- P Ceccato, S Maxwell, RG Rommel, GM Jacquez, KK Benedict, SA Morain, P Yang, Q Huang, ML Golden, RS Chen, et al. Data discovery, access and retrieval. *Environmental Tracking for Public Health Surveillance*, page 229, 2012.
- Dana Shum, Chris Durbin, James Norton, and Andrew Mitchell. Harvesting nasa's common metadata repository (cmr). In *American Geophysical Union (AGU) 2017 Fall Meeting*, number IN51A-0003, 2017.
- Deepak Bhaskar Acharya, Karthigeyan Kuppan, and B. Divya. Agentic AI: Autonomous Intelligence for Complex Goals—A Comprehensive Survey. *IEEE Access*, 13:18912–18936, 2025. ISSN 2169-3536.
- Johan Moreno. In a real-world study, ai boosts worker productivity by 14%, 2023. URL https://www.forbes.com/sites/johanmoreno/2023/04/25/in-a-real-world-study-ai-boosts-worker-productivity-by-14/. Forbes, April 25, 2023.
- Open Science | NASA Earthdata. URL https://www.earthdata.nasa.gov/about/open-science.
- Gerald A Meehl, George J Boer, Curt Covey, Mojib Latif, and Ronald J Stouffer. The coupled model intercomparison project (cmip). *Bulletin of the American Meteorological Society*, 81(2): 313–318, 2000.
- Dean N Williams, Karl E Taylor, Luca Cinquini, Ben Evans, Michio Kawamiya, Michael Lautenschlager, Bryan Lawrence, Don Middleton, and Contributors ESGF. The earth system grid federation: Software framework supporting cmip5 data analysis and dissemination. *ClIVAR Exchanges*, 56(2):40–42, 2011.

- Tina Erica Odaka, Anderson Banihirwe, Guillaume Eynard-Bontemps, Aurelien Ponte, Guillaume Maze, Kevin Paul, Jared Baker, and Ryan Abernathey. The pangeo ecosystem: interactive computing tools for the geosciences: benchmarking on hpc. In *Annual Workshop on HPC User Support Tools*, pages 190–204. Springer, 2019.
- Mattia Righi, Bouwe Andela, Veronika Eyring, Axel Lauer, Valeriu Predoi, Manuel Schlund, Javier Vegas-Regidor, Lisa Bock, Björn Brötz, Lee de Mora, et al. Earth system model evaluation tool (esmvaltool) v2. 0–technical overview. *Geoscientific Model Development*, 13(3): 1179–1199, 2020.
- Robert G Raskin and Michael J Pan. Knowledge representation in the semantic web for earth and environmental terminology (sweet). *Computers & geosciences*, 31(9):1119–1125, 2005.
- Lu Zhou, Michelle Cheatham, Adila Krisnadhi, and Pascal Hitzler. Geolink data set: A complex alignment benchmark from real-world ontology. *Data Intelligence*, 2(3):353–378, 2020.
- Jiantao Wu, Fabrizio Orlandi, Declan O'Sullivan, and Soumyabrata Dev. Linkclimate: An interoperable knowledge graph platform for climate data. *Computers & Geosciences*, 169:105215, 2022.
- Alastair Green, Martin Junghanns, Max Kießling, Tobias Lindaaker, Stefan Plantikow, and Petra Selmer. opencypher: New directions in property graph querying. In *EDBT*, pages 520–523, 2018.
- Nicolas Webersinke, Mathias Kraus, Julia Anna Bingler, and Markus Leippold. Climatebert: A pretrained language model for climate-related text. *arXiv preprint arXiv:2110.12010*, 2021.
- Jennifer E Kay, Clara Deser, A Phillips, A Mai, Cecile Hannay, Gary Strand, Julie Michelle Arblaster, SC Bates, Gokhan Danabasoglu, James Edwards, et al. The community earth system model (cesm) large ensemble project: A community resource for studying climate change in the presence of internal climate variability. *Bulletin of the American Meteorological Society*, 96(8): 1333–1349, 2015.
- Ze-Bao Zhang, Jian-Pei Zhang, Jing Yang, and Yue Yang. A new approach to creating spatial index with r-tree. In 2007 International Conference on Machine Learning and Cybernetics, volume 5, pages 2645–2648. IEEE, 2007.
- Kelsey Jordahl, Joris Van den Bossche, Jacob Wasserman, James McBride, Martin Fleischmann, Jeffrey Gerard, Jeff Tratner, Matthew Perry, Carson Farmer, Geir Arne Hjelle, et al. geopandas/geopandas: v0. 7.0. *Zenodo*, 2021.
- LangChain. Langgraph, built by langchain inc. https://langchain-ai.github.io/langgraph/.
- Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

- Amazon Web Services and Anthropic. Anthropic's claude in amazon bedrock. https://aws.amazon.com/bedrock/anthropic/, 2025. Accessed: 24 September 2025.
- Amazon Web Services. Amazon Neptune Analytics User Guide, 2025. URL https://docs.aws.amazon.com/neptune-analytics/latest/userguide/what-is-neptune-analytics.html.
- cftime. cftime: Python library for decoding time units and variable values in netcdf files. https://unidata.github.io/cftime/.
- Stephan Hoyer and Joe Hamman. xarray: Nd labeled arrays and datasets in python. *Journal of open research software*, 5(1):10–10, 2017.
- Polars Developers. Polars: A lightning-fast DataFrame library. https://pola.rs, 2025. Version; insert version number here.
- Christian Braneon, Luis Ortiz, Daniel Bader, Naresh Devineni, Philip Orton, Bernice Rosenzweig, Timon McPhearson, Lauren Smalls-Mantey, Vivien Gornitz, Talea Mayo, Sanketa Kadam, Hadia Sheerazi, Equisha Glenn, Liv Yoon, Amel Derras-Chouk, Joel Towers, Robin Leichenko, Deborah Balk, Peter Marcotullio, and Radley Horton. NPCC4: New York City climate risk information 2022—observations and projections. *Annals of the New York Academy of Sciences*, 1539(1):13–48, 2024. ISSN 1749-6632. doi: 10.1111/nyas.15116.