

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

**Abstract**—Climate data science remains constrained by fragmented data sources, heterogeneous formats, and steep technical expertise requirements. These barriers slow discovery, limit participation, and undermine reproducibility. We present *AutoClimDS*, a proof-of-concept Agentic AI system that addresses these challenges by integrating a curated climate knowledge graph (KG) with a set of Agentic AI workflows designed for cloud-native scientific analysis. The KG unifies datasets, metadata, tools, and workflows into a machine-interpretable structure, while AI agents—powered by generative models—enable natural-language query interpretation, automated data discovery, programmatic data acquisition, and end-to-end climate analysis. A key result is that AutoClimDS can reproduce published scientific figures and analyses from natural-language instructions alone, completing the entire workflow from dataset selection to preprocessing to modeling. When given the same tasks, state-of-the-art general-purpose LLMs (e.g., ChatGPT GPT-5.1) cannot independently identify authoritative datasets or construct valid retrieval workflows using standard web access. This highlights the necessity of structured scientific memory for agentic scientific reasoning. By encoding procedural workflow knowledge into a KG and integrating it with existing technologies (cloud APIs, LLMs, sandboxed execution), AutoClimDS demonstrates that the KG serves as the essential enabling component—the irreplaceable structural foundation—for autonomous climate data science. This approach provides a pathway toward democratizing climate research through human–AI collaboration.

Knowledge Graphs, AI Agents, Climate Data Science, Generative AI, Cloud-Native Data Access, Human–AI Collaboration

## I. INTRODUCTION

Climate research investigates Earth’s climate systems, variability, and change effects on natural and human environments [1], drawing from observational datasets, simulation outputs, and analytical tools across domains [2]. Despite data proliferation, research remains fragmented: datasets in heterogeneous formats with inconsistent metadata lack standardized access [3]. Existing retrieval systems rely on keyword search requiring users to know dataset names [4], while general-purpose LLMs lack the structured scientific memory needed for autonomous data acquisition and workflow construction.

**Contributions:** We introduce a semantic infrastructure encoding climate data entities into a unified, queryable knowledge graph (KG) serving as structured scientific memory for agentic workflows. First, we present an ontology-driven methodology integrating NASA CMR, NOAA OneStop [5], ERA5 [6], CMIP6 [7] records into a semantically consistent graph using OpenCypher [8]. Unlike prior KGs providing conceptual vocabularies or keyword-driven

retrieval, this approach encodes procedural workflow knowledge: executable access links with authentication protocols, variable-level semantic mappings via fine-tuned ClimateBERT [9] (99.17% accuracy), geospatial relationships, and preprocessing operation metadata. Second, we demonstrate this KG as reasoning substrate for *AutoClimDS*, reproducing scientific workflows end-to-end where existing approaches fail. The KG provides essential structural constraints—typed relationships like `hasLink`, `hasCESMVariable`, spatial containment—irreplaceable by pure vector search or LLM reasoning, evidenced by successful replication of published climate studies (Section III).

### A. Agentic AI and Human–AI Collaboration

Agentic AI systems [10] capable of autonomous reasoning, planning, and tool use offer promising solutions. Evidence shows generative AI assistants boost productivity 14% on average, with greatest gains for less-experienced users [11], narrowing expertise gaps in complex tasks. In climate data science, where bottlenecks stem from technical barriers, knowledge-graph-enabled Agentic AI could elevate entry-level researchers while streamlining expert workflows, making discovery more inclusive and reproducible.

### B. Related Work

Several initiatives have sought to improve climate data access and interoperability. Foundational projects like NASA Earthdata [12], CMIP [13], and ESGF [14] provide observational and model datasets. Computational tools such as Pangeo [15] and ESMValTool [16] facilitate cloud-based analysis, while ontologies like SWEET [17] and GeoLink [18] offer structured vocabularies for Earth science concepts. However, these were designed for human researchers: SWEET provides conceptual hierarchies (e.g., "Precipitation"  $\subseteq$  "AtmosphericPhenomenon") but lacks executable data access paths; Pangeo requires manual dataset specification; ESMValTool demands expert-configured recipes. None capture the procedural reasoning needed for autonomous agentic workflows—how to authenticate with NASA Earthdata, which variable to extract from multi-dimensional NetCDF, or which preprocessing operations to apply for specific analyses.

Recent efforts such as LinkClimate [19] demonstrated knowledge graph infrastructures for climate datasets. However, its design reflects a static view where discovery is keyword-driven, treating datasets as retrieval endpoints without representing procedural knowledge or scientific reasoning

needed for workflow construction. When tasked with reproducing the NPCC4 sea level analyses (Section III), LinkClimate’s keyword search returned dataset titles but relied on exact string matching over relatively shallow metadata fields. Similarly, GPT-5.1 with web search but without KG guidance failed to autonomously locate authoritative datasets, instead hallucinating dataset names or selecting inappropriate sources with mismatched temporal/spatial coverage. This reveals a critical gap: the lack of infrastructure serving as reasoning backbone for Agentic AI. This work addresses this gap by developing a knowledge graph encoding not just data locations but procedural reasoning paths, demonstrating that structured scientific memory enables reliable agentic workflows where pure retrieval or LLM reasoning alone fails.

## II. METHOD

### A. Knowledge Graph Ontology and Construction

The knowledge graph integrates metadata from NASA CMR, NOAA OneStop [5], ERA5 [6], and CMIP6 [7] via respective APIs. The graph contains  $|\mathcal{V}| \approx 1,300,000$  nodes across 42 types and  $|\mathcal{E}| \approx 3,600,000$  edges across 45 relationship types, encompassing  $\sim 208,000$  climate datasets (106,000 observational, 102,000 simulation outputs). The schema partitions into two branches: observational data (NASA CMR/NOAA OneStop capturing measurements, satellite retrievals, reanalysis) and simulation data (CMIP6/ERA5 encoding experiments, ensembles, model outputs). This hybrid design encodes both data content and procedural context.

**Data ingestion pipeline:** NASA CMR collections/granules retrieved via Search API (`cmr.earthdata.nasa.gov`) with UMM-JSON parsing for temporal/geospatial metadata. NOAA OneStop (`data.noaa.gov/onestop`) integrates NCEI, ERDDAP, CO-OPS repositories with 5 req/s rate limiting. ERA5 datasets discovered via Copernicus CDS web crawling (`cds.climate.copernicus.eu`) with HTML parsing and JSON normalization. CMIP6 metadata resolves via ESGF distributed index API with 10-field DRS tuple filtering (`mip_era`, `activity_id`, `institution_id`, `source_id`, `experiment_id`, `variant_label`, `table_id`, `variable_id`, `grid_label`, `version`) and multi-node failover.

**Geospatial processing:** Polygon coordinates from dataset metadata undergo Shapely geometric processing for bounding box extraction. Mapbox Geocoding API classifies spatial scope hierarchically: ocean, global, continental, country, multinational, or regional across 258 location boundaries. This enables spatial relationship traversal via `hasLocation` edges during multi-criteria search.

**Link scoring and validation:** URLs undergo automated downloadability assessment. Domain analysis identifies API types (OpenDAP, ERDDAP, WMS, WFS, REST) and authentication requirements (public vs. restricted access). Downloadability weights reflect empirical retrieval success: direct protocols (HTTP/HTTPS/FTP,  $w = 1.0$ ), scientific services (OPeNDAP/THREDDS,  $w = 0.8$ ), web service APIs

( $w = 0.6$ ), documentation ( $w = 0.4$ ). Endpoint verification tests validate data accessibility before graph ingestion.

**Semantic variable mapping:** We fine-tuned ClimateBERT [9]<sup>1</sup> for multi-class classification over 2,308 CESM variables. Architecture employs attention-masked mean pooling with dropout ( $p = 0.3$ ), trained via cross-entropy loss with Adam optimizer ( $\alpha = 1 \times 10^{-5}$ , batch 16, 50 epochs) on curated CESM training set. Similarity-based clustering addresses variable redundancy via  $S(\text{desc}(v_i), \text{desc}(v_j)) \geq 0.7$ , achieving  $\mathcal{A}_{\text{exact}} = 93.45\%$  and  $\mathcal{A}_{\text{group}} = 99.87\%$ . The trained model maps observational dataset variables to standardized CESM nomenclature, enabling cross-dataset variable discovery.

**OpenCypher graph construction:** JSON metadata converts to Neptune-compatible OpenCypher CSV format via schema-driven transformation. Node types partition into core entities and climate-simulation specific types. Relationships encode typed edges with source-target node constraints. Embeddings for vector-enabled nodes are generated via sentence-transformers. The complete schema with node/edge definitions is available in the supplementary materials [20].

### B. AutoClimDS Agentic AI Architecture

The system implements three core objectives: *Data Discovery*, *Data Acquisition*, and *Climate Modeling/Analytics*. Implementation uses LangChain [21] with ReAct reasoning [22] and Bedrock Claude Sonnet 4 [23].

1) *Data Discovery Agent:* The agent implements semantic dataset discovery by encoding research queries into 384-dimensional vectors through sentence-transformers. Vector search operates through Neptune Analytics’ `topKByEmbedding()` procedure implementing hierarchical navigable small world graphs for efficient approximate nearest neighbor retrieval [24].

The agent implements intelligent search routing through embedding availability detection, categorizing node types into vector-enabled and text-only categories. Vector-enabled types include `DataCategory`, `Variable`, `CESMVariable`, `ScienceKeyword`, `Location`, `TemporalResolution`, and `SpatialResolution`, while remaining nodes utilize text-based Neptune query matching.

Multi-criteria search extends single-criterion functionality by combining vector results with relationship-based filtering addressing real-world complexity where researchers must consider temporal coverage, spatial resolution, variable availability, and institutional provenance simultaneously. The algorithm constructs complex OpenCypher queries incorporating vector results as node constraints while applying additional filtering through temporal overlap detection  $[t_{\text{start}}, t_{\text{end}}]$ , spatial relationship traversal via `hasLocation` relationships, and other edges. This hybrid approach enables queries like ”precipitation datasets over Pacific Northwest from 1980-2020 with daily resolution”

<sup>1</sup>`climatebert/distilroberta-base-climate-f`

to leverage vector search for "precipitation" while applying structured filters for location, temporal range, and resolution metadata.

Retrieved datasets undergo link reranking for automated prioritization: for dataset  $D$  with links  $L_D = \{l_1, \dots, l_n\}$ , each link  $l_i$  possesses preprocessed weight  $w_i \in [0, 1]$  assigned during graph construction. The reranking function orders by descending weight via permutation  $\sigma: l_{\sigma(1)}, \dots, l_{\sigma(n)}$  where  $w_{\sigma(1)} \geq \dots \geq w_{\sigma(n)}$ . This prioritization ensures direct download protocols (HTTP/HTTPS/FTP,  $w = 1.0$ ) are attempted before specialized scientific data services (OPeN-DAP/THREDDS,  $w = 0.8$ ), web service APIs ( $w = 0.6$ ), or documentation links ( $w = 0.4$ ). The reranking operation executes as post-processing following dataset retrieval, transforming unordered link collections into priority-ordered access sequences maximizing probability of successful automated data acquisition. Results persist to local SQLite database enabling session continuity, recall of previously retrieved datasets, and support for iterative research workflows.

2) *Data Acquisition Agent*: Once the knowledge graph returns relevant datasets with reranked access links, the Agentic AI system transitions from discovery to acquisition. Each dataset entry in the research database is inspected, and the agent retrieves corresponding links from `hasLink` relationships of dataset nodes, pre-ordered by downloadability weight. These links may point to NASA Earthdata, NOAA archives, AWS Open Data S3 buckets, or specialized climate data services. Based on research query  $q$ , the agent determines next action  $a \in \mathcal{A}$  where  $\mathcal{A} = \{\text{retrieve, preprocess, analyze}\}$ .

If  $a = \text{retrieve}$ , the agent first attempts data acquisition through highest-weighted links. When links provide explicit API endpoints or direct download URLs, the agent invokes appropriate retrieval protocol with authentication handled via preconfigured tokens (e.g., NASA Earthdata credentials, NOAA CDO API keys). However, when links lack clear programmatic access methods or when initial retrieval attempts fail, the agent engages in dynamic access discovery. This process leverages web search and website fetching tools to query documentation, data portals, and technical specifications, enabling the agent to discover access protocols autonomously rather than relying on hardcoded patterns. The agent can read API documentation, identify authentication requirements, locate endpoint specifications, and generate custom retrieval code adapted to the specific data source. For NOAA datasets requiring location-based queries, the agent utilizes a location code resolution API with token validation to translate geographic descriptors into standardized location identifiers before constructing data requests.

Raw data, denoted  $D = \{d_1, d_2, \dots, d_n\}$ , 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 constraint-checking function  $V(\hat{D}) \in \{0, 1\}$ , which enforces link validity, accessibility, and struc-

tural consistency. Only datasets satisfying  $V(\hat{D}) = 1$  are retained for downstream workflows. These steps execute through `CodeExecutionTool` operating within sandboxed Docker containers with restricted filesystem access, network egress limited to whitelisted repositories (NASA Earthdata, NOAA, AWS S3), resource quotas, and output sanitization. The adaptive pipeline enables protocol discovery for previously unseen data sources.

By integrating discovery, retrieval, validation, and preprocessing into a single agent-driven workflow, the system achieves reproducible and cloud-resilient data acquisition. Graph linkages to cloud repositories ensure persistence, while transformation  $T$  and validation  $V$  guarantee dataset usability. The agent's dynamic protocol discovery ensures robustness across heterogeneous climate data infrastructure where access methods vary significantly.

3) *Climate Simulation Agents*: Two specialized agents handle ERA5/CMIP6 datasets. *Discovery agent* executes OpenCypher queries against Neptune for `SimDataset` nodes, leveraging relationships. For CMIP6, DRS filtering uses `institution/scenario/ensemble` relationships. Link validation confirms CDS API (ERA5) or ESGF HTTP (CMIP6) accessibility. *Acquisition agent* invokes `cdsapi` for ERA5 asynchronous retrieval with authentication, queuing, polling. For location queries (e.g., "NYC temperature"), geocoding resolves descriptors to bounding boxes for spatial subsetting—a query for "New York City SSP2-4.5 temperature projections" automatically translates to coordinates [40.7°N, -74.0°W] with appropriate buffer for spatial subsetting of global model grids. CMIP6 uses authenticated HTTP from ESGF nodes prioritized by link weights. Downloaded NetCDF/GRIB files undergo `xarray`-based loading with specialized tools (`ProcessERA5DataTool`, `ProcessCMIP6DataTool`, `TimeSeriesAnalysisTool`, `SpatialAnalysisTool`, `CompareERA5CMIP6Tool`) handling calendar conventions, coordinate transforms, unit conversions.

4) *State Management and Error Recovery*: The system implements persistent state management via SQLite for dataset discovery results and `LangChain's ConversationBufferWindowMemory` ( $k=15$  exchanges for Orchestrator,  $k=5$  for specialized agents) maintaining conversation context. When data acquisition fails, cascading fallback executes: (1) iterate through reranked links  $l_{\sigma(1)}, \dots, l_{\sigma(n)}$  from Section 2.2.1, (2) invoke dynamic discovery via web search/documentation fetching when all pre-indexed links fail, (3) request semantically similar datasets from Discovery Agent using vector similarity  $\cos(\mathbf{v}_D, \mathbf{v}_{D'}) > 0.85$  if dataset remains inaccessible. `CodeExecutionTool` persists state through mounted volume file writes and maintains rolling window of last 5 tool outputs, enabling multi-step preprocessing without re-execution.

5) *Agentic Loop Mechanics and Guardrails*: The system implements ReAct (Reasoning + Acting) [22] through iterative cycles. For query  $q$ , the loop exe-

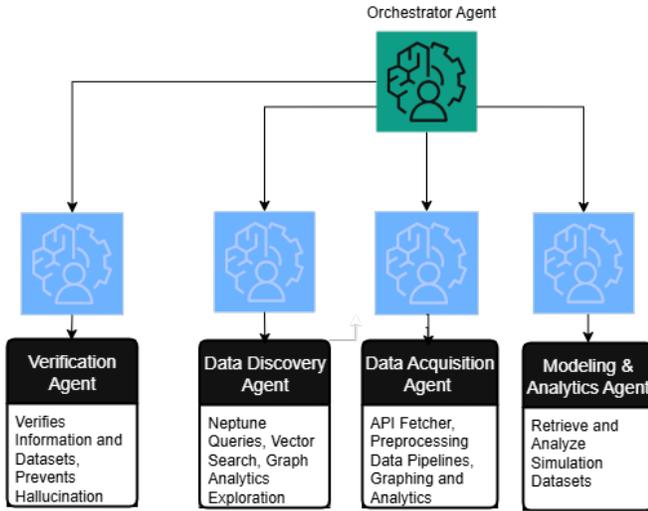


Fig. 1. Multi-agent system architecture.

cutes:  $\text{Thought}_t = \text{LLM}(q, \mathcal{H}_{t-1}, \mathcal{O}_{t-1})$ ,  $\text{Action}_t = \text{parse}(\text{Thought}_t) \in \mathcal{A}$ ,  $\mathcal{O}_t = \text{Tool}(\text{Action}_t)$ ,  $\mathcal{H}_t = \mathcal{H}_{t-1} \cup \{(\text{Thought}_t, \text{Action}_t, \mathcal{O}_t)\}$ , where  $\mathcal{H}_t$  is conversation history,  $\mathcal{O}_t$  is tool observation, and  $\mathcal{A}$  is the action space.

To prevent infinite loops and token exhaustion, agents enforce: (1) `max_iterations=15` for discovery/acquisition agents, 100 for Orchestrator handling complex workflows, (2) `handle_parsing_errors=True` for malformed LLM outputs, (3) `max_execution_time=300s` timeout, (4) semantic cycle detection terminating when consecutive thought embeddings exceed cosine similarity 0.95 for three steps, (5) circuit breaker for identical tool calls within 5-step window. Multi-agent workflows use LangChain’s `StateGraph` defining directed acyclic graphs with explicit routing logic: Orchestrator  $\rightarrow$  Discovery Agent  $\rightarrow$  Acquisition Agent, with conditional fallback edges routing back to Discovery upon acquisition failure rather than allowing indefinite Acquisition loops.

6) *Multi-Agent System*: The architecture (Fig. 1) features a central Orchestrator Agent interpreting objectives, maintaining session state via persistent storage (Section 2.2.5), and delegating to specialized agents through `StateGraph`-based workflow execution (Section 2.2.6). Data Discovery Agent queries the KG, Data Acquisition Agent retrieves from cloud sources with automated link fallback and error recovery, and Climate Modeling Agent integrates model ensembles. When validation fails ( $V(\hat{D}) = 0$ ), the system logs failure reason and routes back to Discovery for alternative datasets, ensuring transparency and preventing silent failures. When Acquisition Agent exhausts all links, workflow routes to Discovery for alternatives rather than halting, ensuring robustness through automated fault tolerance.

### C. Cloud Deployment

The reference implementation deploys on AWS Neptune for graph storage and Bedrock for LLM inference (Fig. 2). However, the KG schema and agent workflows are cloud-

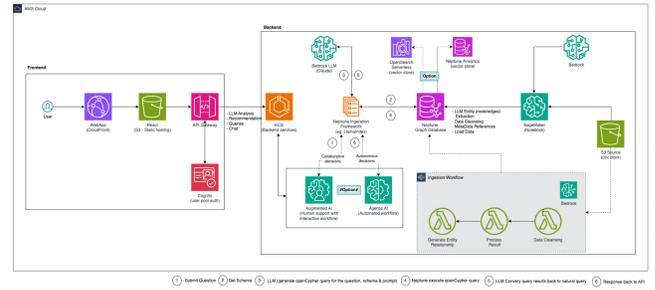


Fig. 2. End-to-end AWS architecture with frontend (CloudFront, React, API Gateway, Cognito) and backend (Bedrock, Neptune, SageMaker) integration.

agnostic: the 42-node-type, 45-relationship-type schema exports to OpenCypher CSV format compatible with Neo4j, ArangoDB, or any OpenCypher graph database; agents can use open-source models (Llama, Mistral) via HuggingFace; vector search can use FAISS or Qdrant. The scientific contribution lies in the KG’s encoding of procedural climate data science workflows, not the cloud infrastructure. Implementation details and computational costs are documented in the supplementary materials [20].

## III. CASE STUDIES

### A. Observational Data: Sea Level Trends

We replicate figures from *NYC Climate Risk Information 2022 (NPCC4)* [25] to validate AutoClimDS’ capability for reproducing climate risk indicators using natural language queries. Data discovery and acquisition proceeded via natural language instructions only: no datasets, numerical values, or coefficients are provided. The instructions followed a three-part structure: (1) *Objective*, (2) *Context/Constraints*, (3) *Desired Output*. Agents autonomously located datasets, preprocessed, calculated measures, generated figures.

**Baseline Comparisons:** As established in Section 1.2, existing approaches (LinkClimate, GPT-5.1, NASA CMR) failed to complete these workflows autonomously. In contrast, AutoClimDS successfully replicated all NPCC4 figures (Figs. 3–5). Statistical validation confirms high fidelity across all indicators. For Battery Park sea level, AutoClimDS calculates a long-term trend of 0.112 in/yr, exceeding the precision of the reported 0.11 in/yr, and exactly reproduces the recent acceleration of 0.150 in/yr (1993–2017). Similarly, the system accurately recovers the Vertical Land Motion (VLM) contribution of  $-1.5$  mm/yr and the Global Mean Sea Level (GMSL) trend of 0.12 in/yr. Additionally, the Jensen-Shannon Divergence (JSD) is 0, confirming that the graph pairs are exactly the same. These metrics demonstrate AutoClimDS reproduces published analyses with statistical equivalence. All logs, prompts, data, and figures available in supplementary materials [20].

### B. Climate Simulation: Temperature Projections

To demonstrate simulation data handling and generalization beyond observational datasets, we tasked AutoClimDS with analyzing future temperature projections for New York

City using CMIP6 and ERA5 datasets. Given only a natural language prompt, the system autonomously: (1) queried the KG for relevant CMIP6/ERA5 experiments, (2) retrieved multi-model ensemble data via authenticated ESGF and CDS API access, (3) performed spatial subsetting to NYC coordinates via geocoding, (4) computed ensemble means and uncertainty ranges, (5) generated comparison visualizations (Fig. 6). This workflow required distinct capabilities from observational sea level analysis—specifically multi-model ensemble handling—demonstrating the KG’s generalizability across observational and simulation data modalities. GPT-5.1 without KG guidance failed on this task.

#### IV. OPEN SCIENCE

Code, data, and documentation are available in supplementary materials [20]. Resources include KG schema/seed entries, agent workflows, data access scripts, documentation, and tutorials. Users can reproduce experiments and adapt workflows. Importantly, all agent-generated Python codes (via CodeExecutionTool) were logged and preserved, allowing users to inspect preprocessing operations, learn data science workflows, and understand how analyses were constructed—supporting educational use cases beyond pure automation. The modular design enables extensibility through new KG entries (datasets, workflows, ontologies), tools, and documentation. We envision this work as foundation for collaborative growth by climate scientists, data scientists, educators, students, and citizen science communities.

#### V. CONCLUSION

We built a proof of concept demonstrating that a well-curated knowledge graph enables highly capable AI agents for climate data science workflows, substantially lowering barriers for non-technical users. The claim that “*a knowledge graph is all you need*” reflects a fundamental insight: while agentic systems require LLMs, tools, and infrastructure, the KG provides the irreplaceable structural foundation that makes autonomous scientific reasoning possible. Without procedural knowledge of KG: knowledge—executable access paths, authentication protocols, variable mappings, preprocessing workflows—frontier LLMs hallucinate datasets, select inappropriate sources, and fail to construct workflows, as the GPT-5.1 baseline demonstrates. The KG transforms general-purpose AI into domain-competent scientific agents by encoding the structured memory and reasoning constraints that pure retrieval or LLM reasoning alone cannot provide. In this sense, the KG is the essential enabling component: integrate it with existing technologies (cloud APIs, LLMs, sandboxed execution), and autonomous climate data science becomes achievable. The KG serves as extensible memory and unifying reasoning layer across tools/datasets, aligning with cloud data science solutions while creating space for community contributions.

Beyond technical capabilities, this work highlights broader potential of KGs and AI agents to democratize climate data science. Lowering technical barriers opens opportunities for policy, education, and citizen science participation,

while modular cloud-native design ensures scalability and interoperability. The KG provides foundation for evolving, community-driven commons growing with new datasets, tools, and domain knowledge—positioning it as technical and socio-technical infrastructure for collaborative science. This approach provides pathways toward integrating advanced reasoning capabilities, fostering reproducibility, and accelerating discovery through human–AI partnerships in climate and beyond.

#### REFERENCES

- [1] V. Eyring, W. D. Collins, P. Gentine, E. A. Barnes, M. Barreiro, T. Beucler, M. Bocquet, C. S. Bretherton, H. M. Christensen, K. Dagon *et al.*, “Pushing the frontiers in climate modelling and analysis with machine learning,” *Nature Climate Change*, pp. 1–13, 2024.
- [2] A. Gettelman, A. J. Geer, R. M. Forbes, G. R. Carmichael, G. Feingold, D. J. Posselt, G. L. Stephens, S. C. Van den Heever, A. C. Varble, and P. Zuidema, “The future of earth system prediction: Advances in model-data fusion,” *Science Advances*, vol. 8, no. 14, p. eabn3488, 2022.
- [3] P. Ceccato, S. Maxwell, R. Rommel, G. Jacquez, K. Benedict, S. Morain, P. Yang, Q. Huang, M. Golden, R. Chen *et al.*, “Data discovery, access and retrieval,” *Environmental Tracking for Public Health Surveillance*, p. 229, 2012.
- [4] D. Shum, C. Durbin, J. Norton, and A. Mitchell, “Harvesting nasa’s common metadata repository (cmr),” in *American Geophysical Union (AGU) 2017 Fall Meeting*, no. IN51A-0003, 2017.
- [5] NOAA, “OneStop: A geospatial search platform for environmental data discovery,” <https://www.ncei.noaa.gov/products/onestop>, 2023, national Centers for Environmental Information.
- [6] H. Hersbach, B. Bell, P. Berrisford, S. Hirahara, A. Horányi, J. Muñoz-Sabater, J. Nicolas, C. Peubey, R. Radu, D. Schepers *et al.*, “The era5 global reanalysis,” *Quarterly Journal of the Royal Meteorological Society*, vol. 146, no. 730, pp. 1999–2049, 2020.
- [7] V. Eyring, S. Bony, G. A. Meehl, C. A. Senior, B. Stevens, R. J. Stouffer, and K. E. Taylor, “Overview of the coupled model intercomparison project phase 6 (cmip6) experimental design and organization,” *Geoscientific Model Development*, vol. 9, no. 5, pp. 1937–1958, 2016.
- [8] A. Green, M. Junghanns, M. Kiebling, T. Lindaaker, S. Plantikow, and P. Selmer, “opencypher: New directions in property graph querying,” in *EDBT*, 2018, pp. 520–523.
- [9] N. Webersinke, M. Kraus, J. A. Bingler, and M. Leippold, “Climatebert: A pretrained language model for climate-related text,” *arXiv preprint arXiv:2110.12010*, 2021.
- [10] D. B. Acharya, K. Kuppan, and B. Divya, “Agentic AI: Autonomous Intelligence for Complex Goals—A Comprehensive Survey,” *IEEE Access*, vol. 13, pp. 18 912–18 936, 2025.
- [11] J. Moreno. (2023) In a real-world study, ai boosts worker productivity by 14%. Forbes, April 25, 2023. [Online]. Available: <https://www.forbes.com/sites/johanmoreno/2023/04/25/in-a-real-world-study-ai-boosts-worker-productivity-by-14/>
- [12] “Open Science | NASA Earthdata.” [Online]. Available: <https://www.earthdata.nasa.gov/about/open-science>
- [13] G. A. Meehl, G. J. Boer, C. Covey, M. Latif, and R. J. Stouffer, “The coupled model intercomparison project (cmip),” *Bulletin of the American Meteorological Society*, vol. 81, no. 2, pp. 313–318, 2000.
- [14] D. N. Williams, K. E. Taylor, L. Cinquini, B. Evans, M. Kawamiya, M. Lautenschlager, B. Lawrence, D. Middleton, and C. ESGF, “The earth system grid federation: Software framework supporting cmip5 data analysis and dissemination,” *CIIVAR Exchanges*, vol. 56, no. 2, pp. 40–42, 2011.
- [15] T. E. Odaka, A. Banihirwe, G. Eynard-Bontemps, A. Ponte, G. Maze, K. Paul, J. Baker, and R. Abernathey, “The pangeo ecosystem: interactive computing tools for the geosciences: benchmarking on hpc,” in *Annual Workshop on HPC User Support Tools*. Springer, 2019, pp. 190–204.
- [16] M. Righi, B. Andela, V. Eyring, A. Lauer, V. Predoi, M. Schlund, J. Vegas-Regidor, L. Bock, B. Brötz, L. de Mora *et al.*, “Earth system model evaluation tool (esmvaltool) v2. 0—technical overview,” *Geoscientific Model Development*, vol. 13, no. 3, pp. 1179–1199, 2020.

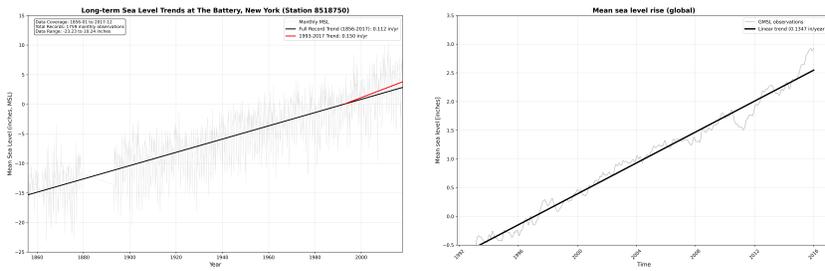


Fig. 3. AutoClimDS replicated NPCC4 sea level trends.

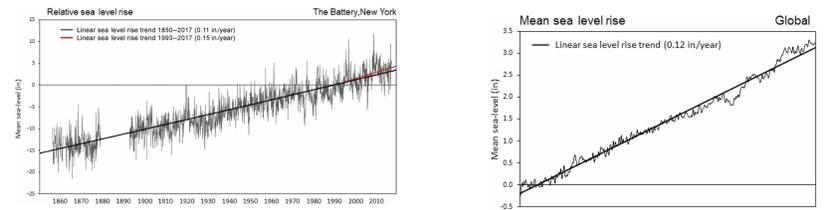


Figure 3.2. Historic sea level rise 1850–2017 in New York City at The Battery (NOAA, 2017). Black trend line shows an increasing trend from 1850 to 2017, while the red trend line shows a slightly higher trend from 1993 to 2017, which may reflect the apparent recent acceleration seen in the global sea level rise record.

Figure 3.1. Global mean sea level rise during the satellite era, 1993–2018 (AVISO, France; posted March 26, 2018).

Fig. 4. Original figures from [25] (CC BY-NC license).

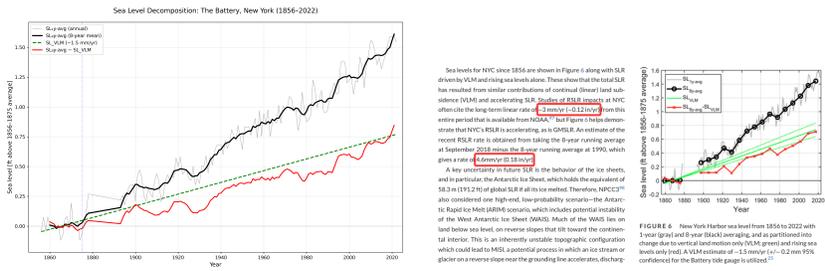


Fig. 5. Sea level trends with VLM-driven SLR: AutoClimDS (left) vs. Original [25] (right).

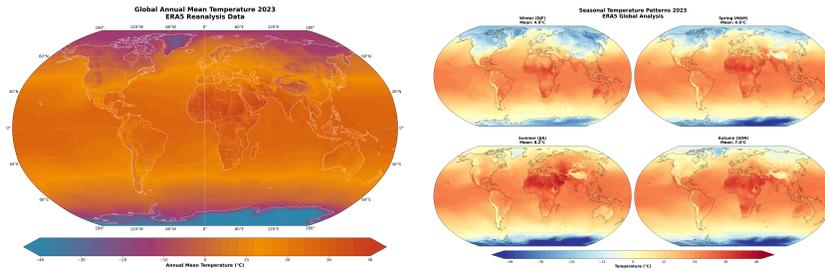


Fig. 6. CMIP6/ERA5 temperature analysis for NYC: historical reanalysis and multi-model SSP2-4.5 projections with ensemble uncertainty.

[17] R. G. Raskin and M. J. Pan, “Knowledge representation in the semantic web for earth and environmental terminology (sweet),” *Computers & geosciences*, vol. 31, no. 9, pp. 1119–1125, 2005.

[18] L. Zhou, M. Cheatham, A. Krishnathi, and P. Hitzler, “Geolink data set: A complex alignment benchmark from real-world ontology,” *Data Intelligence*, vol. 2, no. 3, pp. 353–378, 2020.

[19] J. Wu, F. Orlandi, D. O’Sullivan, and S. Dev, “Linkclimate: An interoperable knowledge graph platform for climate data,” *Computers & Geosciences*, vol. 169, p. 105215, 2022.

[20] Anonymous, “Autoclimds: Climate data science agentic ai - supplementary materials,” <https://anonymous.4open.science/r/AutoClimDS-E041/README.md>, 2025, anonymous repository for double-blind review.

[21] LangChain, “Langgraph, built by langchain inc,” <https://langchain-ai.github.io/langgraph/>.

[22] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, “React: Synergizing reasoning and acting in language models,” in *International Conference on Learning Representations (ICLR)*, 2023.

[23] Amazon Web Services and Anthropic, “Anthropic’s claude in amazon bedrock,” <https://aws.amazon.com/bedrock/anthropic/>, 2025, accessed: 24 September 2025.

[24] Amazon Web Services. (2025) Amazon Neptune Analytics User Guide. [Online]. Available: <https://docs.aws.amazon.com/neptune-analytics/latest/userguide/what-is-neptune.html>

[25] T. Braneon, L. Ortiz, D. Bader, N. Devineni, P. Orton, B. Rosenzweig, C. McPhearson, L. Smalls-Mantey, V. Gornitz, T. Mayo, S. Kadam, H. Sheerazi, E. Glenn, L. Yoon, A. Derras-Chouk, J. Towers, R. Leichenko, D. Balk, P. Marcotullio, and R. Horton, “NPCC4: New York City climate risk information 2022—observations and projections,” *Annals of the New York Academy of Sciences*, vol. 1539, no. 1, pp. 13–48, 2024.