LLM-Assisted Modeling and Simulations for Public Sector Decision-Making: Bridging Climate Data and Policy Insights

Charles Cao[1], **Jie Zhuang**[1], **Qiang He**[1]

[1]University of Tennessee, Knoxville, TN, 37996
Email: cao@utk.edu, jzhuang@utk.edu, qianghe@utk.edu

Abstract

This paper presents a transformative framework aimed at enhancing decision-making in the public sector, especially in the context of environmental policy and climate change. Traditional approaches to handling vast and complex climate data often create barriers due to the need for specialized expertise. Our framework, based on the GPTs platform from OpenAI, overcomes this challenge by enabling users without technical knowledge to easily access and interpret climate datasets and simulations. Through natural language interaction, stakeholders can engage with data and explore various policy scenarios effectively. This approach not only simplifies the decision-making process but also opens doors for a wider range of stakeholders to contribute to policy development and strategic planning in the public sector. The paper discusses the framework's design, its application in real-world scenarios, and its potential to facilitate informed, data-driven decisions in addressing environmental challenges.

Introduction

In public sector decision-making, especially concerning environmental policy and climate change, the volume and complexity of data presents a formidable challenge (Green, Armstrong, and Soon 2009; Nocke et al. 2008). Effective decision-making in this context requires not only access to comprehensive data but also the ability to interpret and utilize this information in a meaningful way. Traditional approaches often require expert knowledge for data processing and modeling, leading to the gap between technical data and actionable insights.

On the other hand, another important development in decision-making is the emergence of digital twin technology (Jones et al. 2020; Tao et al. 2022). By creating virtual replicas of physical systems, digital twins integrate real-time data with advanced simulation models, providing a dynamic and interactive platform for scenario analysis to answer "what-if" questions. However, just like domain-specific datasets, this technology also poses a barrier to non-technical users, particularly those in the public sector responsible for policy decisions.

In this paper, we address this common technical barrier of expert knowledge for interpreting datasets and carrying out digital twin based simulations by introducing an LLM-agent based data modeling and simulation framework designed to democratize access to complex climate datasets and digital twin-based simulations, enabling users without specialized technical knowledge to engage effectively with data repositories which may come in a wide range of formats, as well as domain-specific simulation tools that require expert knowledge to carry out experiments. Central to our methodology is a novel, GPTs-based AI agent, which is based on OpenAI's recently announced GPTs Platform (OpenAI 2023). The AI agent interacts with users in natural language, and performs customized functions on data analysis and remote simulations without human intervention. Hence, users do not need to upload their data or manually submit experiment results. Instead, this AI agent interacts with dataset servers through API calls directly. Hence, it not only provides universal access to various data sources of different formats, but also implements actions with digital twin simulators for in-depth validation of what-if decisions. In our prototype implementation, the AI agent utilizes RESTful APIs for efficient data retrieval, and we also implement a Linux-based sandbox environment for digital twin simulations using domain-specific tools. Hence, the specific actions accessible to this agent can be easily extended with up-to-date and relevant environment datasets, and the well-known AI hallucinations are minimized due to that all analysis results are based on up-to-date data instead of pre-trained weights. This paper demonstrates the preliminary results on how this framework can be applied to a wide variety of areas for decision-making purposes. The envisioned applications of this framework is shown in Figure 1, which spans multiple areas of interest.

There are several advantages of this approach. First, the natural language processing capabilities of GPT allow users to interact with the system using simple queries and commands, thus lowering the barrier of entry for engaging with complex datasets. Second, we take advantage of the GPTs platform to build customized environment to manipulate data modeling tasks. Doing so allows the AI agent to transform raw data into visually compelling and easily understandable graphs. This capability is crucial for stakeholders, enabling them to understand the implications of environmental data and the potential impacts of various policy decisions. Finally, decisions made by the policy makers can be simulated using domain-specific simulators. As we pro-

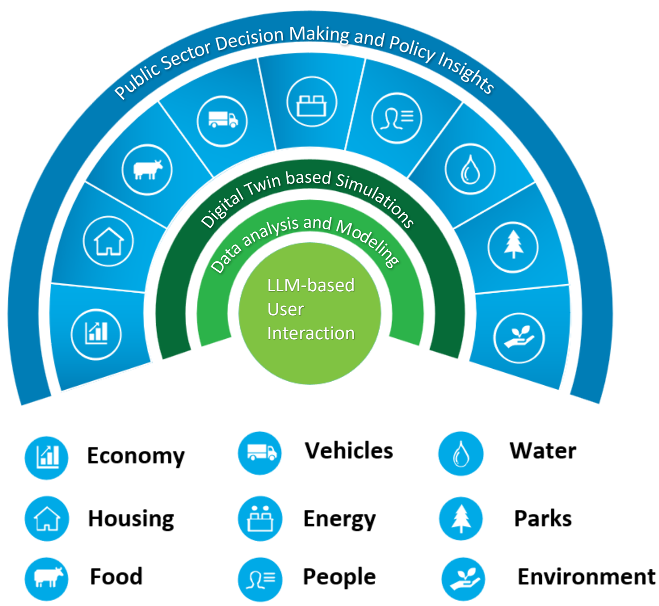


Figure 1: Application Areas of the Proposed AI Agent

vide interfaces to the generic Linux platform as a sandbox, digital twin simulations can be easily experimented with. In doing so, the AI agent can control the terminal of the Linux sandbox environment directly without human intervention, such as experiment with different parameter combinations in an autonomous manner. This allows the AI agent to carry out detailed scenario analysis, offering a practical tool for policy experimentation and foresight. For stakeholders, they can manipulate variables and observe the outcomes of different policy decisions in real-time, providing a powerful tool for predictive analysis and strategic planning.

In the rest of this paper, we first discuss the design of the proposed framework, including the architecture of the AI agent and the Linux sandbox environment. We then describe the implementation of the framework, including the data sources and simulators used. We then present three concrete case studies of using this framework in real-world analysis and simulation tasks. Finally, we discuss the related work and conclude with a summary of our contributions.

Design

In this section, we describe the design of the proposed framework, including the architecture of the AI agent and the Linux sandbox environment. Figure 2 shows the overview of the design of the building blocks. As shown in this figure, the design features a GPTs frontend, a Node.js based server for serving datasets from a database, an adapter for Linux terminal sandbox (using Docker) to provide RESTful APIs, and the digital twin simulator running inside the Linux sandbox. This platform not only simplifies the interaction with complex datasets but also significantly enhances the decision-making process, particularly in the areas of environmental policy and climate change. We use sample datasets, such as the global average temperature anomaly datasets and the polar ice extent datasets as examples to show how LLM can understand and model long-term trends of climate. For the digital twin simulator, we integrate the well-known MAGICC 7 (Magicc Development Team 2023), which stands for "Model for the Assessment of Greenhouse Gas Induced Climate Change" for complex climate modeling purposes. MAGICC has been one of the widely used climate models in various IPCC assessment reports. We demonstrate that, as MAGICC runs under Linux, the LLM frontend can greatly simplify user interactions by reading the documentation manuals of MAGICC autonomously, and carry out simulations under various parameter settings all without user intervention. This allows stakeholders to experiment with different scenario analysis on reducing greenhouse gas emissions, which serves to validate what-if hypotheses. Hence, Through this work, we contribute to the evolving landscape of technology-enabled policy-making, offering a path forward for informed, data-driven decisions in the public sector.

GPTs Frontend

The users of this system interfaces with the GPTs based AI agent. This agent is tailored to meet the specific needs of stakeholders by interacting with them in natural language, effectively bridging the gap between users and the complex datasets. Its integration enables users, irrespective of their technical expertise, to engage meaningfully with the datasets served by the backend and simulation toolkits. This significantly lowers the barrier to entry, making the system accessible and user-friendly for policymakers who might not possess in-depth technical knowledge.

For the purpose of data retrieval and integration, the frontend employs RESTful APIs to access a diverse array of data sources, which are stored in server side databases. This capability ensures that the system is continuously fed with up-to-date and relevant environmental datasets, crucial for accurate analysis and modeling. It also reduces the possibility of hallucinations of AI models, which is a common problem in GPT-based models. The integration process is designed to handle various data formats and sources, thereby maintaining the versatility and comprehensiveness of the data repository.

Once the agent receives data from the backend Node.js based server, it performs visualization and modeling tasks, which are powered by the combined strengths of Python and the GPTs platform. This integration facilitates the transformation of raw data into formats that are not only visually appealing but also easily interpretable by users. The agent also provides support for statistical predictions, such as regression analysis.

Node.js Server Backend

In order to serve the data to the frontend, we developed a Node.js based server backend. This server is designed to handle various datasets stored in databases. The server is designed to be modular, allowing for easy integration of new data sources and formats, which can also be directly provided from external sources. The Node.js server also provides capabilities to be bridged to digital twin simulators,

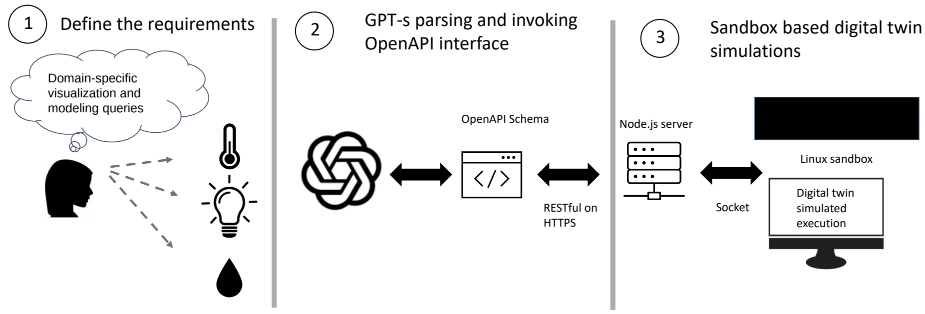


Figure 2: Overview of Framework Design

which run in a Linux sandbox environment. To this end, we developed a Python-based adapter that exposes the Linux terminal through RESTful APIs. To prevent errors and file misoperations, all commands are executed in a Dockerized environment. Various simulation tools can therefore be supported without modifications. This development is crucial in enabling the AI agent to interact with simulation environment, such as running a simulator with various parameter settings. This also allows the AI agent to use Linux commands remotely to further customize the simulation environment as needed. For example, we demonstrate it is feasible for the LLM agent to edit the contents of the configuration files of the MAGICC simulator directly. Such flexibility allows for a wide range of scenario analyses and decision validation processes for exploring ”what-if” scenarios, essential for policy planning and evaluations.

Integration and Workflows

Based on these foundational elements, our vision extends to enabling stakeholders to engage interactively with datasets, conduct modeling and predictions, and observe the outcomes of various policy scenarios through advanced simulations. This interactive capability is not just an add-on but a critical component for iterative policy development and optimization. It allows for the exploration of diverse scenarios and their potential impacts in a controlled, risk-free environment. Consequently, the framework serves not only as a theoretical model but as a practical tool for predictive analysis and strategic planning.

In summary, our framework's design is a confluence of technical expertise in data processing, visualization techniques, and realistic simulations, all encapsulated behind a user-friendly interface. This interface masks the underlying complexity, making the tool accessible to a broader range of users, including those without deep technical expertise. The integration of GPT-based AI agents, renowned for their unprecedented natural language processing capabilities, positions this tool as a transformative solution in the public sector. This is particularly pertinent in the realm of environmental policy and climate change strategies, where the need for rapid, informed decision-making is critical.

The AI agent's role in this framework is multifaceted. It not only facilitates the interpretation and analysis of complex datasets but also assists in simulating various policy outcomes. Its ability to understand and generate human-like text allows for a more intuitive interaction between the stakeholders and the tool, making the process of policy formulation and analysis more natural and efficient.

Furthermore, the framework's potential in environmental policy and climate change strategies cannot be overstated. With the increasing urgency to address environmental challenges, our tool offers a way to model the impact of various policies on environmental outcomes. This could range from carbon footprint reduction strategies to resource allocation in renewable energy initiatives (Franchetti and Apul 2012; Hertwich and Peters 2009). By providing a platform for testing and visualizing the effects of these policies before implementation, the framework aids in identifying the most effective strategies for combating climate change and promoting sustainable practices.

Through this design, we aim to contribute significantly to the evolving landscape of technology-enabled policy-making. Our goal is to pave the way for more informed, data-driven decisions in the public sector, particularly in areas where the stakes are high and the need for strategic planning is critical. By harnessing the power of AI and data analytics, we believe this framework can play a pivotal role in shaping policies that are not only effective but also sustainable and forward-thinking.

Implementation

GPTs Customization

Our approach to customizing the GPTs platform involved the development of a detailed OpenAPI schema, designed to structure the AI agent's interaction with backend datasets and user queries. The schema also supports dataset discoveries so that users can easily find out which datasets are currently available. Specifically, this schema served as a blueprint, defining API endpoints, parameter data types, query formats, and response structures.

Besides the schema, we also leveraged GPTs's advanced prompt engineering capabilities, by providing customized requirements so that the AI agent will interpret user inputs accurately and generate responses that are coherent and directly relevant to the users' inquiries. Specifically, we provided a list of keywords and phrases that the agent should be able to recognize and respond to. This list was designed to cover a wide range of possible user queries, ensuring that the agent was able to handle a diverse array of requests. The agent was also trained to recognize and respond to various types of questions, including those that require data retrieval, visualization, and modeling. This capability was crucial for the agent to perform its functions effectively, ensuring that it was able to respond to user queries in a meaningful and relevant manner.

Node.js Server Backend

In the backend, we developed a Node.js based server to serve the data to the frontend. This server is designed to handle various datasets stored in databases. We are using MySQL database in our prototype, but other types of databases can also be used. The server provides a RESTful interface that allows queries from the frontend to be processed. All queries and responses are implemented using the JSON format, which provides a lightweight and flexible format for data exchange. We choose JSON for its ease of use and its ability to structure data in a way that is both human-readable and machine-parsable.

Adapter for Linux Sandbox

To support various types of digital twin simulators, such as MAGICC, we develop an adapter for the Linux terminal sandbox based on the use of Docker images. By creating a sandbox environment, we ensured that the AI agent's interactions with the backend simulators are always limited to a controlled and secure space. This precaution was essential to prevent potential security vulnerabilities that could arise from direct file system-level access. The adapter allowed the AI agent to execute Linux commands within the sandbox without compromising the underlying system's integrity. The Node.js server communicates with the dockerized sandbox using socket programming, and we implement a Python-based controller for intercepting and monitoring all communication messages, ensuring that all interactions were governed by strict access controls and security protocols.

In conclusion, the implementation of the GPTs customization, RESTful interface, and the Linux sandbox adapter was carried out with a focus on functionality, security, and scalability.

Case Studies

In this section, we present three detailed case studies to demonstrate the utility and effectiveness of the proposed LLM-assisted data analysis framework in bridging complex climate data with actionable policy insights.

Global Average Temperature Anomaly Analysis

This case study focuses on the analysis of global temperature anomaly data spanning from 1981 to 2022. We use the HadCRUT5 dataset (Met Office Hadley Centre 2022), and employs the GPTs AI agent to interpret and model this data effectively.

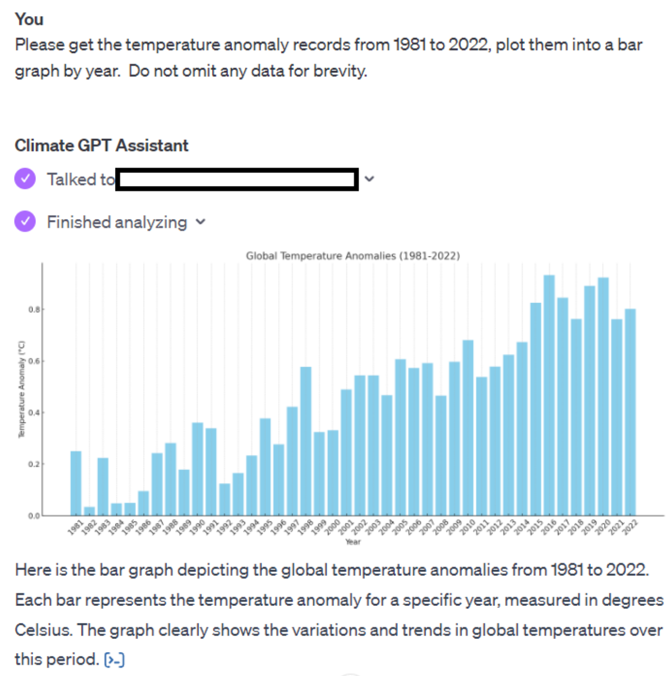


Figure 3: Global Temperature Anomalies from 1981 to 2022. The backend server's address is hidden.

When queried, the data is served by the Node.js server using JSON format. The GPTs then analyzes the data and generate graphs. The results are shown in Figure 3, which is completely generated autonomously. This demonstrates that the GPTs frontend is able to plot the data effectively for understanding the trends and variations in global temperatures.

Next, we apply a linear regression model to this data to project future trends. The regression model, detailed in Figure 4, is generated as Python code by the frontend. To enhance interpretability, the AI agent produces a visual representation of this model alongside the actual data points, as depicted in Figure 5. The graph illustrates the consistency of the model with observed data and underscores the increasing trend of temperature anomalies.

Finally, we also ask the AI agent by predicting the year in which the temperature anomaly is likely to reach the critical threshold of 2 degrees Celsius. Based on the linear regression model, this threshold is projected to be crossed approximately in the year 2080, as demonstrated in Figure 6.

This case study exemplifies the AI agent's capability to not only process and visualize large datasets but also to perform predictive modeling that is crucial for policy-making. By translating complex data into comprehensible forecasts, the agent effectively aids policymakers in understanding the potential future impacts of climate change, thus facilitating informed decision-making. Since the users of this frontend does not need to learn any technical details of the data, the AI agent can greatly reduce the barrier of entry for non-technical users to engage with complex datasets.

```
import numpy as np

# Converting the years to a numpy array for regression analysis
years_array = np.array(years).reshape(-1, 1)

# Linear Regression
from sklearn.linear_model import LinearRegression

# Creating a linear regression model
model = LinearRegression()
model.fit(years_array, temperature_anomalies)

# Predicting temperature anomalies using the model
predicted_anomalies = model.predict(years_array)

# Plotting the data and the trendline
plt.figure(figsize=(15, 7))
plt.bar(years, temperature_anomalies, color='skyblue', label='Actual An
plt.plot(years, predicted_anomalies, color='red', linewidth=2, label='T

plt.xlabel('Year')
plt.ylabel('Temperature Anomaly (°C)')
plt.title('Global Temperature Anomalies with Linear Trendline (1981-202
plt.xticks(years, rotation=45)
plt.legend()
plt.grid(axis='y')
plt.tight_layout()

plt.show()

# Coefficients of the linear regression model
slope = model.coef_[0]
intercept = model.intercept_

slope, intercept
```

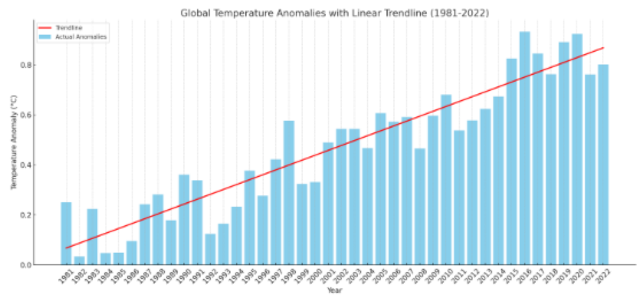
Figure 4: Python Model for Linear Regression Analysis

You

Add a trendline based on linear trend, and develop the model for regression analysis.

Climate GPT Assistant

Finished analyzing



The bar graph now includes a red linear trendline, which shows the overall trend in global temperature anomalies from 1981 to 2022.

The linear regression model developed for this trendline is described by the equation:

$$\text{Temperature Anomaly} = 0.0195 \times \text{Year} - 38.588$$

Where:

The slope of the trendline is approximately 0.0195. This indicates that on average, the temperature anomaly increases by about 0.0195°C per year.

The intercept is approximately -38.588. This value represents the theoretical temperature anomaly at year 0 in this model, which helps in understanding the trend over time but isn't practically meaningful by itself.

Figure 5: Linear Trend Analysis of Temperature Anomalies

Polar Ice Extent Trend Analysis

The second case study aims to assess the long-term trend of polar ice extent, another important issue in climate change that usually requires expertise to interpret datasets. Utilizing the proposed platform, the case study analyzes monthly polar ice extent data from selected years to observe changes and predict future trends. Figure 7 provides a visual representation of the ice extent over the years, with each year indicated by a unique color. Note that this graph requires that the AI agent to repeatedly query the backend server for each year to collect the dataset. This demonstrates the agent's ability to interact with the backend server and retrieve data in a structured manner.

Next, we ask the AI agent to use a regression model to predict the future of polar ice extents, as shown in Figure 8. The model was based on the historical lowest ice extent data points, allowing us to extrapolate the trend into the future and estimate when the ice might disappear entirely during the summer months.

According to the linear regression model, the polar ice is predicted to disappear completely during the summer by the year 2090. This is a critical insight for environmental policy-making, as it provides a tangible timeline for the potential disappearance of polar ice. Actual outcomes may vary due to the complexity of climate systems and the influence of various factors, including potential changes in climate policies, natural variability, and other unforeseen factors that might influence future ice extent trends. Again, this case study underscores the capabilities of the proposed framework in facilitating in-depth analyses and predictions that are essential for informed decision-making.

Remote Simulations in the Linux Sandbox

In the third case study, we demonstrate the capabilities of the GPTs AI agent in executing commands on a remote Linux sandbox and customizing software for specific applications. The focus lies on the AI agent's ability to autonomously navigate through software installation packages, interpret documentation, execute simulations, and interpret the results in an autonomous and structured manner.

As illustrated in Figure 9, the AI agent successfully reads and interprets the contents of the software installation directory, identifying key files such as README, LICENSE, and VERSION. This demonstrates the agent's capacity to understand and summarize complex technical documentation, making it accessible to users who may lack the technical expertise to comprehend such details.

Figure 10 showcases the actual commands run by the AI

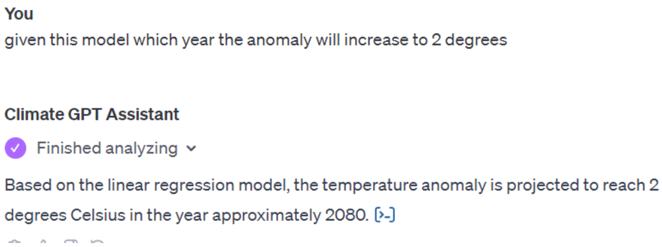


Figure 6: Prediction of Temperature Anomaly Threshold Crossing

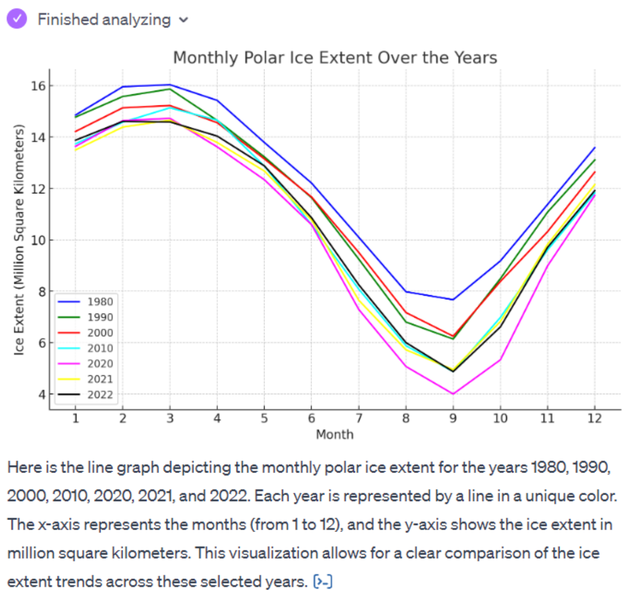


Figure 7: Monthly Polar Ice Extent Over the Years

agent within the Linux sandbox environment when it discovers the contents of the simulator installation. As shown, multiple Linux commands are chosen to be executed. The agent's ability to interact with the terminal and execute a series of commands underlines its potential as a powerful interface for software interaction and management in remote environments.

Finally, Figure 11 confirms the successful execution of the climate simulation software by the AI agent, where the simulator.py is a wrapper for the actual MAGICC simulator. The output indicates that the MAGICC simulation was run without manual intervention, signifying a significant step towards automating complex simulation tasks. This underscores the utility of GPTs in streamlining the operation of specialized simulation software, and changing how simulations are conducted in various scientific and policy-oriented scenarios with the assistance of LLMs.

Overall, this case study validates the GPTs AI agent as an effective tool for managing and executing tasks in a remote Linux sandbox, enhancing the accessibility and user-friendliness of complex simulation software. This capability is crucial for empowering policymakers and stakeholders,

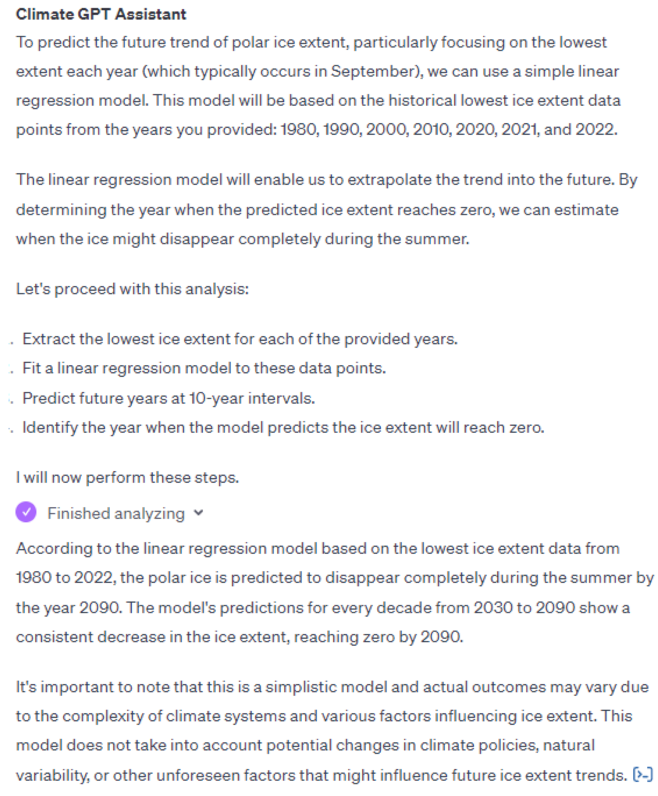


Figure 8: Predictive Model for Polar Ice Extent

allowing them to conduct advanced simulations and analyses without the need for in-depth technical knowledge.

Related Work

The Evolving Role of Large Language Models (LLMs) in Data Analytics: Large Language Models (LLMs) (Brown et al. 2020; Liu et al. 2023a) like GPT have redefined data analysis, especially in handling and simplifying complex datasets. However, existing ChatGPT website still requires manual uploading of datasets for analysis, which can be difficult for non-technical users. In the public sector, LLMs can be instrumental in breaking down barriers to accessing and interpreting extensive climate data, thus facilitating broader stakeholder engagement. With the wide deployment of specialized GPTs, this shift will significantly ease the requirement for specialized expertise, enabling diverse groups of users to gain insights from intricate datasets, which is vital for informed policy-making in environmental contexts.

Advancements in Digital Twins for Public Sector Decision-Making: Digital twins (Tao et al. 2022; Jones et al. 2020), virtual representations of physical systems, have extended their impact beyond manufacturing to significantly influence public sector decision-making, particularly in environmental policy. These advanced systems enable real-time data integration, detailed predictive modeling, and intricate scenario simulations, which are crucial for policymakers dealing with complex and dynamic environmental issues.

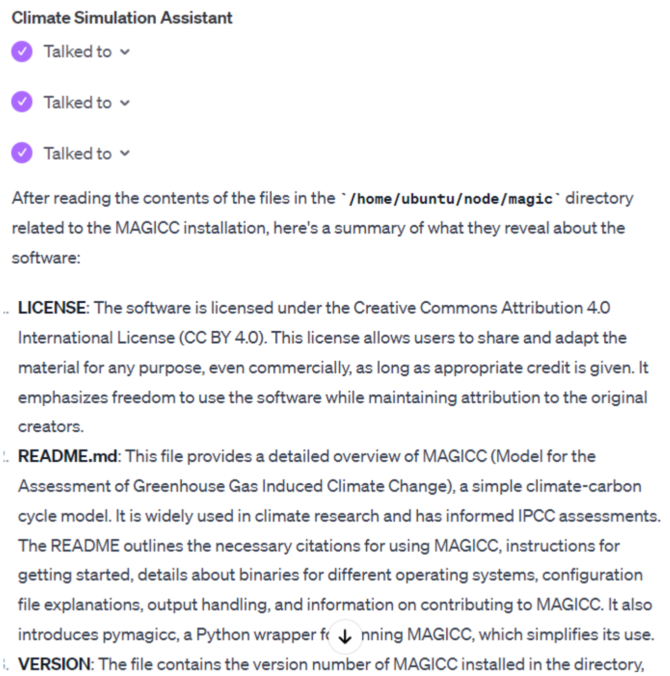


Figure 9: AI Agent's Discovery of Software Installation Contents

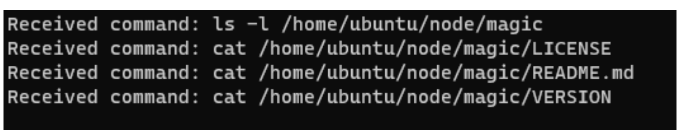


Figure 10: Execution of Commands in the Linux Sandbox. Commands are chosen and issued by the AI Agent autonomously.

Recent developments in this domain have seen an increased emphasis on interactive and user-friendly digital twins that can be leveraged for effective climate change management and policy formulation (VanDerHorn and Mahadevan 2021; Liu et al. 2023b).

Bridging Research Gaps and Setting New Directions: We find that there remains a notable research gap in making advanced technologies like digital twins and LLMs more accessible to non-technical users in the public sector. Therefore, we hope that our work can stimulate future research towards simplifying user interfaces, bolstering data security measures, and widening the scope of these technologies across various facets of public sector decision-making. One possibility is to develop APIs for various public databases to directly interact with LLMs (Overpeck et al. 2011). This future direction is essential for equipping policymakers with the tools needed for tackling the multifaceted challenges posed by environmental issues in an increasingly data-driven world, where the reduced technical barriers are key to enabling effective, data-oriented strategies in addressing global environmental challenges.

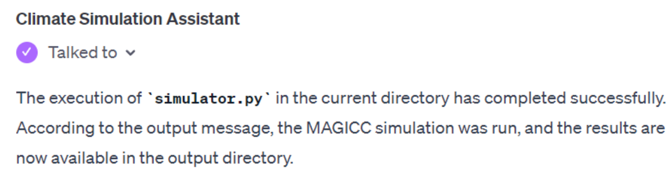


Figure 11: Successful Execution of the Climate Simulator by the AI Agent

Conclusions

Our research introduces a transformative framework that enables non-technical stakeholders in the public sector to engage effectively with complex climate data and simulations. This LLM-Assisted modeling and simulation framework for public sector decision-making integrates a GPTs based AI agent with a secure and scalable Node.js based backend server system to simplify the interpretation and modeling of environmental data, making it accessible for non-technical users. We further demonstrate that the LLM-based agent can interact with a Linux-based sandbox environment to carry out digital twin simulations, thus enabling stakeholders to explore various policy scenarios and their potential impacts

Through three case studies, we demonstrate the framework's ability to turn complex datasets into understandable visualizations and predictive models. In essence, this framework democratizes data-driven insights in environmental policy, paving the way for more informed, effective decision-making in the public sector, particularly in addressing climate change. Its potential for broader applications underscores its value as a versatile tool in technology-enabled policy-making.

Acknowledgments

This work was supported in part by the National Institute of Food and Agriculture/USDA under Grant 2023-67021-40613. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the authors and do not necessarily reflect the view of the U.S. Department of Agriculture.

References

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Franchetti, M. J.; and Apul, D. 2012. *Carbon footprint analysis: concepts, methods, implementation, and case studies*. CRC press.

Green, K. C.; Armstrong, J. S.; and Soon, W. 2009. Validity of climate change forecasting for public policy decision making. *International Journal of Forecasting*, 25(4): 826–832.

Hertwich, E. G.; and Peters, G. P. 2009. Carbon footprint of nations: a global, trade-linked analysis. *Environmental science & technology*, 43(16): 6414–6420.

Jones, D.; Snider, C.; Nassehi, A.; Yon, J.; and Hicks, B. 2020. Characterising the Digital Twin: A systematic literature review. *CIRP journal of manufacturing science and technology*, 29: 36–52.

Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; and Neubig, G. 2023a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9): 1–35.

Liu, X.; Jiang, D.; Tao, B.; Xiang, F.; Jiang, G.; Sun, Y.; Kong, J.; and Li, G. 2023b. A systematic review of digital twin about physical entities, virtual models, twin data, and applications. *Advanced Engineering Informatics*, 55: 101876.

Magicc Development Team. 2023. Magicc simulator. https://magicc.org/.

Met Office Hadley Centre. 2022. HadCRUT5 Global Temperature Dataset. https://www.metoffice.gov.uk/hadobs/hadcrut5/. Accessed: [Insert date of access].

Nocke, T.; Sterzel, T.; Böttinger, M.; Wrobel, M.; et al. 2008. Visualization of climate and climate change data: An overview. *Digital earth summit on geoinformatics*, 226–232.

OpenAI. 2023. Introducing GPTs. https://openai.com/blog/introducing-gpts/.

Overpeck, J. T.; Meehl, G. A.; Bony, S.; and Easterling, D. R. 2011. Climate data challenges in the 21st century. *science*, 331(6018): 700–702.

Tao, F.; Xiao, B.; Qi, Q.; Cheng, J.; and Ji, P. 2022. Digital twin modeling. *Journal of Manufacturing Systems*, 64: 372–389.

VanDerHorn, E.; and Mahadevan, S. 2021. Digital Twin: Generalization, characterization and implementation. *Decision support systems*, 145: 113524.