EGU25-16905, updated on 16 Jan 2026
https://doi.org/10.5194/egusphere-egu25-16905
EGU General Assembly 2025
© Author(s) 2026. This work is distributed under
the Creative Commons Attribution 4.0 License.

An AI-Copilot for JupyterLab for climate data analyses using FrevaGPT

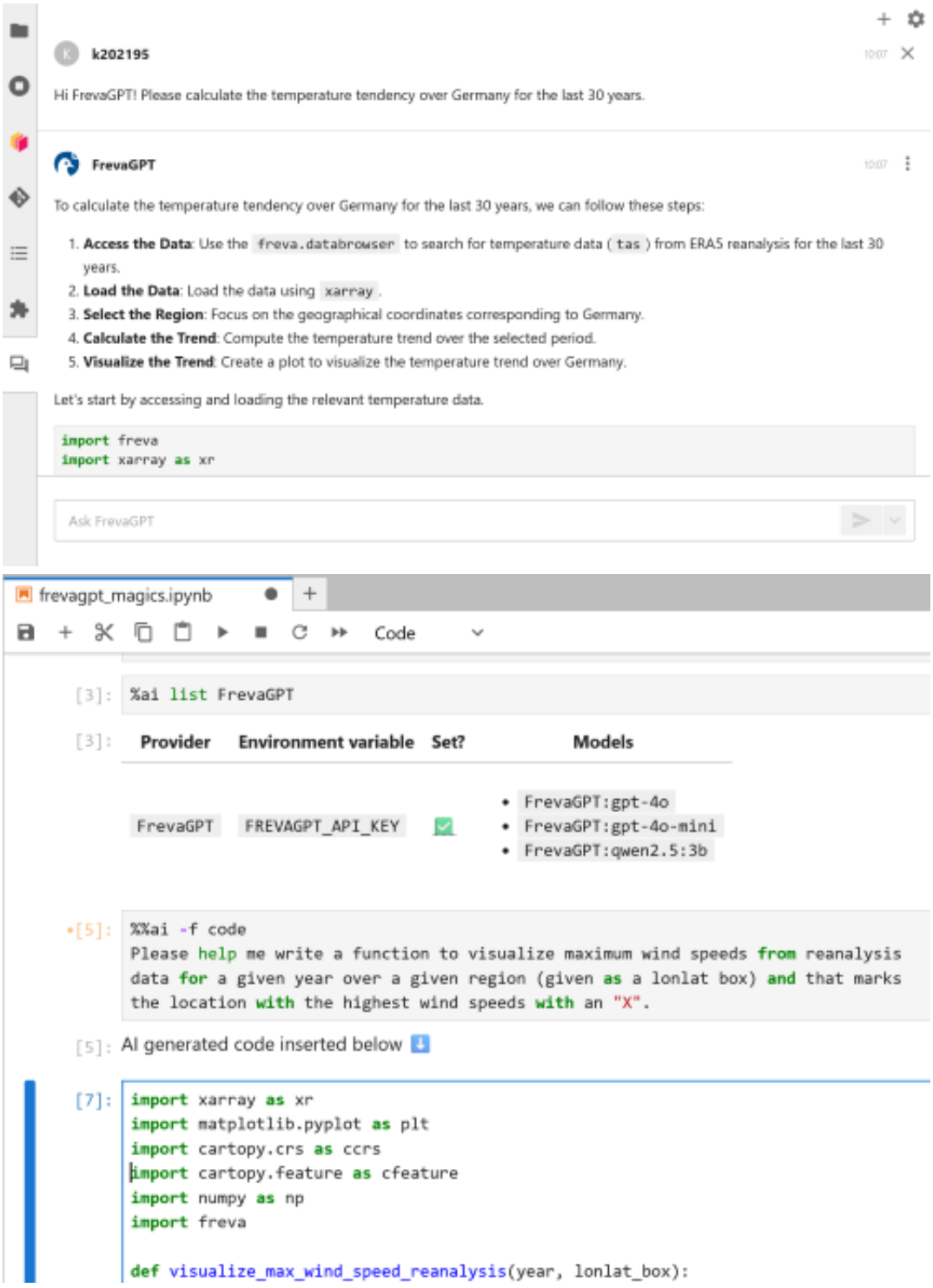
Felix Oertel, Etor Lucio Eceiza, Sebastian Willmann, Bianca Wentzel, Martin Bergemann, and Christopher Kadow
Deutsches Klimarechenzentrum GmbH (DKRZ), Data Analysis, Germany (oertel@dkrz.de)

JupyterLab is a web-based interactive development platform that is widely used in the Earth science community. Using Jupyter Notebooks, it is possible to perform data analysis tasks, annotate and visualize results in a way that is easy to reproduce, present and share with others. JupyterLab allows the use of "extensions", which add functionality to the platform. One of these is Jupyter-AI [1], which allows the use of Large Language Models (LLMs), such as ChatGPT, Claude Sonnet and Ollama, within the JupyterLab environment, through a chat interface or directly within notebooks. By integrating LLMs into JupyterLab, it is possible to leverage their code generation capabilities to assist a user to translate their analysis tasks from an idea to actual executable code in an efficient manner. One drawback of using these LLMs in tasks involving spatio–temporal data is that the models typically do not have access to the data necessary for the analysis task and will often resort to generating fictional data or using placeholders in the code that they create. This requires the user to adapt the provided code to their data, which removes some of the utility provided by the LLM.

In this context we make use of FrevaGPT, an approach for using LLMs in climate data analysis that allows for quick, complex and reproducible analyses of data sets, such as decadal climate model forecasts. Leveraging LLM's capability to write code and using few-shot prompting (in-context learning) allows the LLM to utilize Freva [2,3] (Free Evaluation Framework), a data search and analysis platform, which provides a standardised interface to spatio-temporal datasets hosted on an HPC cluster [4].

FrevaGPT integrates seamlessly into Jupyter-AI and, by making use of the Freva library, combines the code-generating capabilities of LLMs with contextual understanding of how to access relevant datasets on the HPC cluster. This in addition with FrevaGPT's ability to execute generated code in an isolated environment on an HPC node, annotating and explaining any intermediate results, as well as automatically correcting errors encountered along the way, could serve as a starting ramp for researchers to efficiently produce new analysis products based on spatio-temporal climate data.

This PICO will include examples of using FrevaGPT within JupyterLab to analyse spatio-temporal datasets from the climate of the past, as well seasonal to decadal climate predictions.



References:

[1] Jupyter-AI GitHub Repository: https://github.com/jupyterlab/jupyter-ai
[2] Kadow, Christopher, Sebastian Illing, Etor E. Lucio-Eceiza, Martin Bergemann, Mahesh

Ramadoss, Philipp S. Sommer, Oliver Kunst, et al.. 2021. “Introduction to Freva – A Free Evaluation System Framework for Earth System Modeling”. *Journal of Open Research Software* 9 (1): 13. https://doi.org/10.5334/jors.253.
[3] Freva GitHub Repository: https://github.com/FREVA-CLINT/freva
[4] Public Freva Instance: https://www.freva.dkrz.de/