Nanalogue-chat or how I developed an agent around my package using LLMs

Talk to your genomics data in plain English!

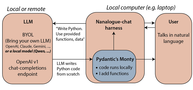
Sathish Thiyagarajan

Nieduszynski group at the Earlham Institute

LLMs impact on analysis

- Lowers skill barrier to writing code
- 'Unprepared' users manually approve code
- Privacy, safety, security compromised due to this

Chat workflow



Monty: an innovative Python interpreter

- https://github.com/pydantic/monty
- Can be accessed as a library
- Written for running untrusted LLM-made code
  - No imports, no filesystem access, no env vars
  - User supplies external functions
- Avoids problems like the following

```
>>> import dog
>>> dog.woof() # Malicious, rm -rf /*
```

What is Nanalogue-gui?

Sequencing data processing with AI chat features.

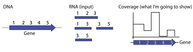
- Inspect system prompts and tools in plain English
- Run analyses on text and binary BAM data
- Generate figures quickly

/dump_llm_instructions

Plain-English prompt → published RNA-seq result

Just describe the analysis in plain English.

Generates code from scratch that counts spanning reads, computes coverage, and generates a plot.



/dump_llm_instructions

Thoughts for future

For tool builders:

- Build custom functions/tools for LLMs
  - So they access custom data or algorithms
  - This also constrains inexperienced users
- Then, just make a chat interface to the LLM
- Allow bring your own LLM (privacy, interop)
- Automate code generation and safe execution

For tool users:

- More complex stuff doable with no coding!
- Driven to do more and more complex stuff.

Thank you

- Angela Man
- Daria Cuthbert
- Charlotte Cannon
- Lukasz Sitko
- Eduardo de-La-Vega
- Anna Rogers
- Isabel Diez Santos
- Nerea Murugarren-Garrido
- Conrad Nieduszynski
- NBI Research computing

Visit nanalogue.com for code or further info and more tools!

Find out more

- Website: nanalogue.com
- Docs and examples
- Rust CLI, Node.js, Python, Docker