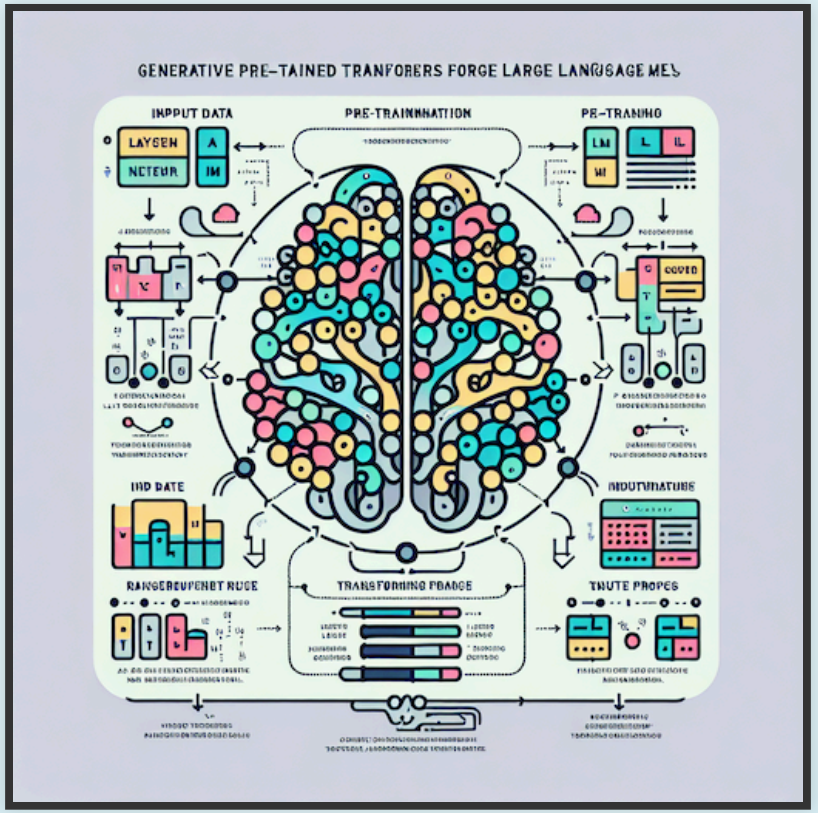
LARGE LANGUAGE MODELS

LARGE LANGUAGE MODELS

LARGE LANGUAGE MODELS

- Example LLMs: ChatGPT, Gemini, Llamma, Claude, capable of understanding and generating human-like language, images, music.
- Data scope: Trained on massive datasets, including PetaBytes of internet text, Wikipedia and Pubmed.
- Applications: LLMs are utilized in chatbots, text generation, reasoning and problem solving, creative output.

ARTIFICIAL NEURAL NETWORKS



ARTIFICIAL NEURAL NETWORKS

- Scalability: LLMs are based on Generative Pre-trained Transformers (GPTs), can be "prompt-engineered" for complex tasks.
- Definition: Artificial Neural Networks (ANNs), are fundamentally complex non-linear function estimators - pattern classifiers.
- Innovation: GPTs are ANNs that implement "multi-head attention", enabling capture of long-range patterns in training data, emergent "intelligence".

GENERATION AND INFERENCE

GENERATION AND INFERENCE

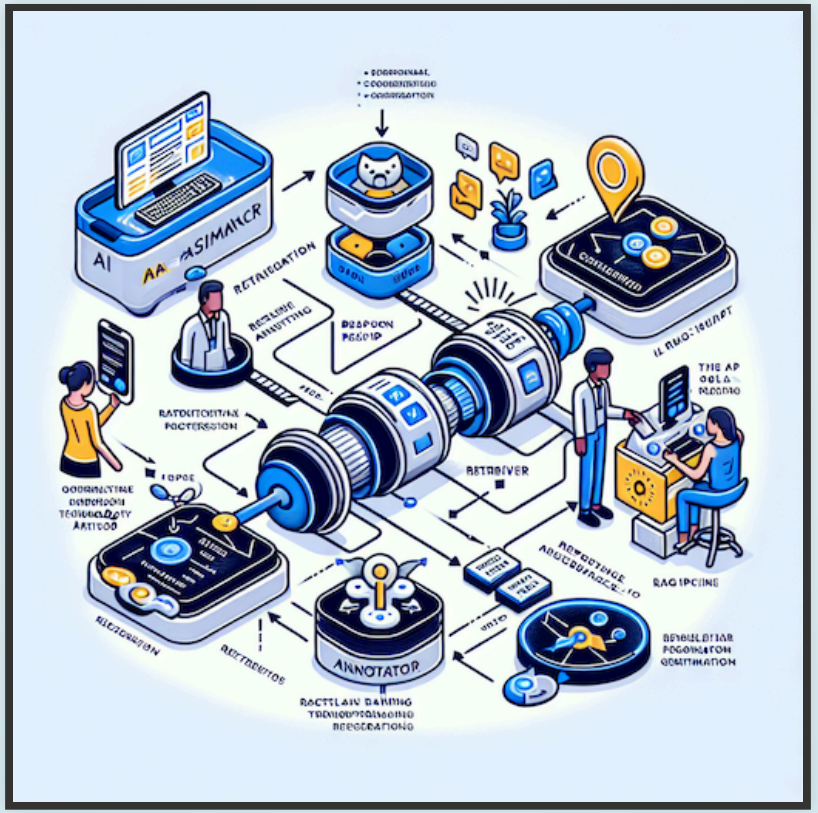
- Text Generation: LLMs create responses by predicting likely sequences of words, based on billions of probabilities.
- Inference Techniques: The models use sophisticated algorithms to generate text that aligns with context and user input.
- Diversity: Can produce a wide range of responses, from factual information to hallucinations and "deepfakes".

FINE-TUNING

FINE-TUNING

- Fine-Tuning: LLMs can be fine-tuned with data from specific domains, enhancing their relevance and performance.
- Task-Specific: Fine-tuning produces tailored AI models for specialized applications, i.e. bioinformatics / biomedical research.
- Alternative: Fine-tuning costs computing time, instead similar achievements via carefully designed prompt-engineering.

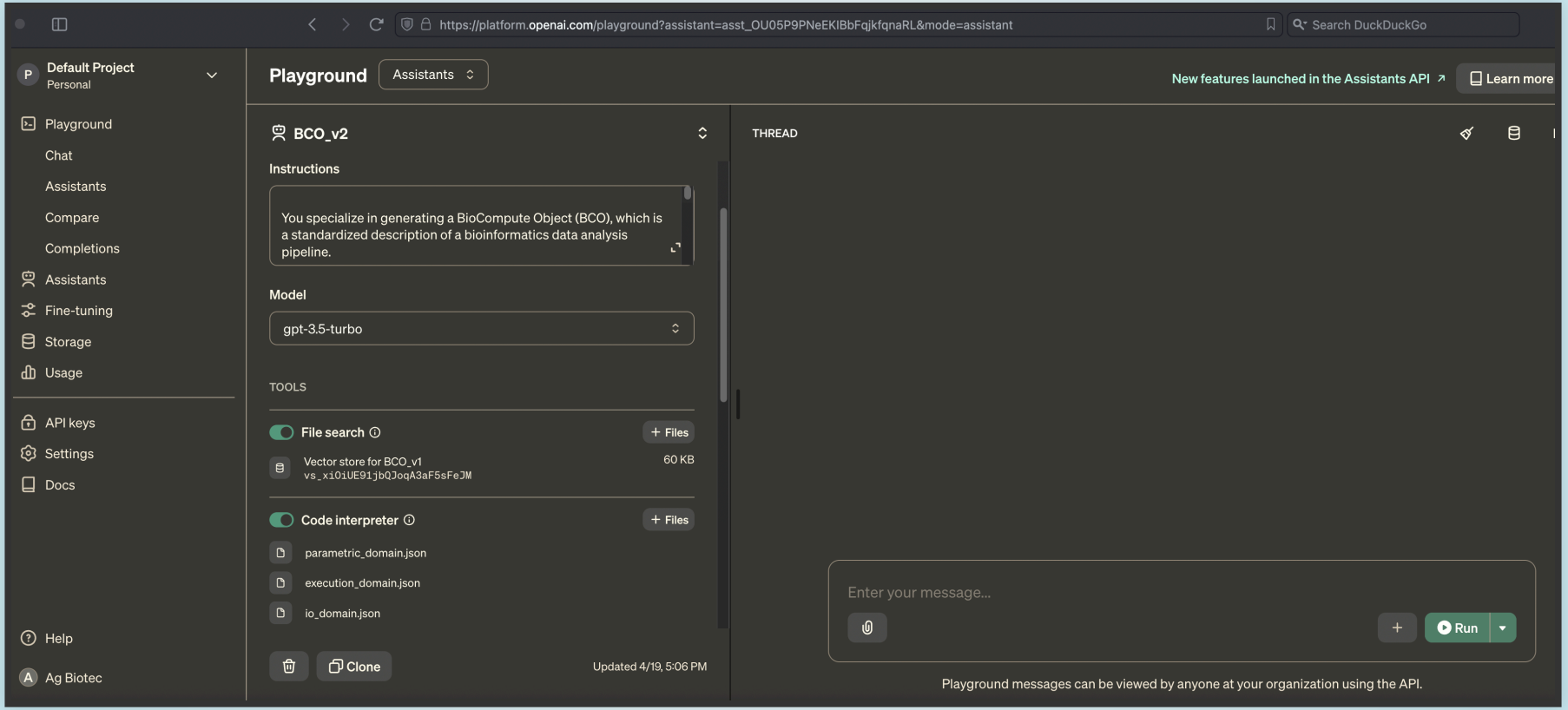
OPENAI ASSISTANTS



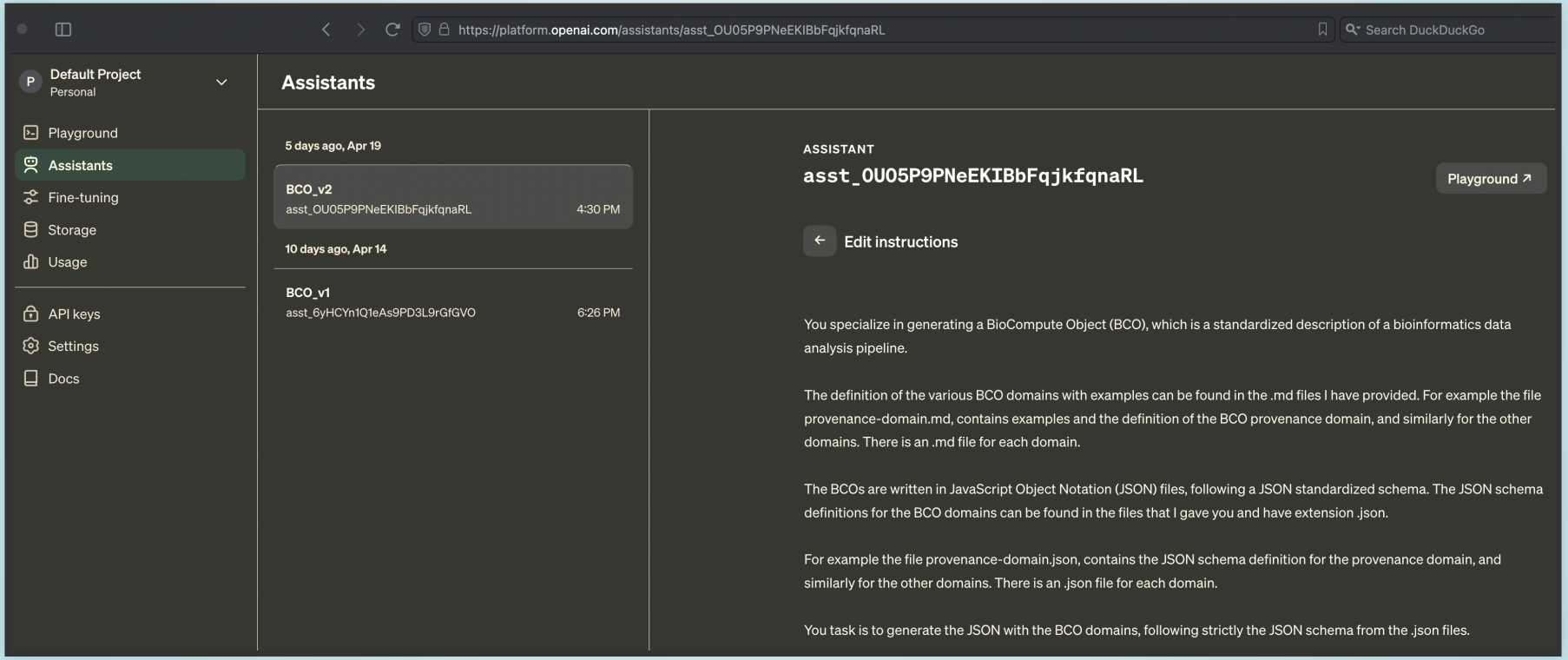
OPENAI ASSISTANTS

- Clarification: OpenAI (the company behind ChatGPT) offers rich functionality through their API
- Assistants: user file search and code Interpreter, external API function calling by the AI
- Functionality: build custom AI applications around user’s data

GPT FOR BCO


https://platform.openai.com/playground?assistant=asst_OU05P9PNeEKIBbFqjkfqnaRL&mode=assistant
Search DuckDuckGo
Default Project
Personal
Playground
Chat
Assistants
Compare
Completions
Assistants
Fine-tuning
Storage
Usage
API keys
Settings
Docs
Help
Ag Biotec
Playground
Assistants
New features launched in the Assistants API
Learn more
BCO_v2
Instructions
You specialize in generating a BioCompute Object (BCO), which is a standardized description of a bioinformatics data analysis pipeline.
Model
gpt-3.5-turbo
TOOLS
File search
+ Files
Vector store for BCO_v1
vs_xiOiUE91jbQJoqA3aF5sFeJM
60 KB
Code interpreter
+ Files
parametric_domain.json
execution_domain.json
io_domain.json
Clone
Updated 4/19, 5:06 PM
THREAD
Enter your message...
Run
Playground messages can be viewed by anyone at your organization using the API.

PROMT ENGINEERING



EXAMPLE PUBLICATION

Astling *et al. BMC Genomics* (2017) 18:614
DOI 10.1186/s12864-017-3976-z

BMC Genomics

METHODOLOGY ARTICLE Open Access

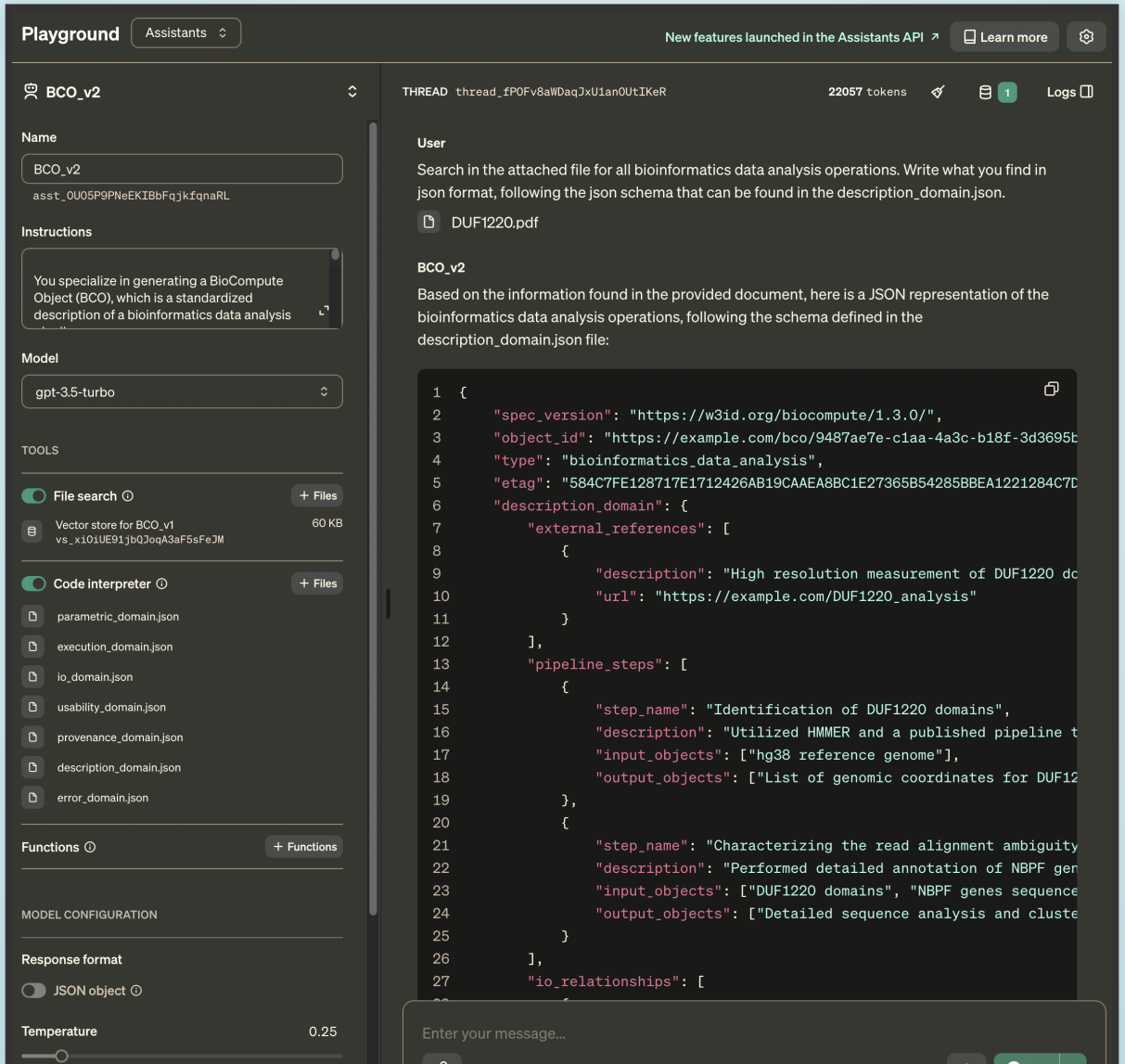
High resolution measurement of DUF1220 domain copy number from whole genome sequence data

David P. Astling[1], Ilea E. Heft[1], Kenneth L. Jones[2] and James M. Sikela[1*]

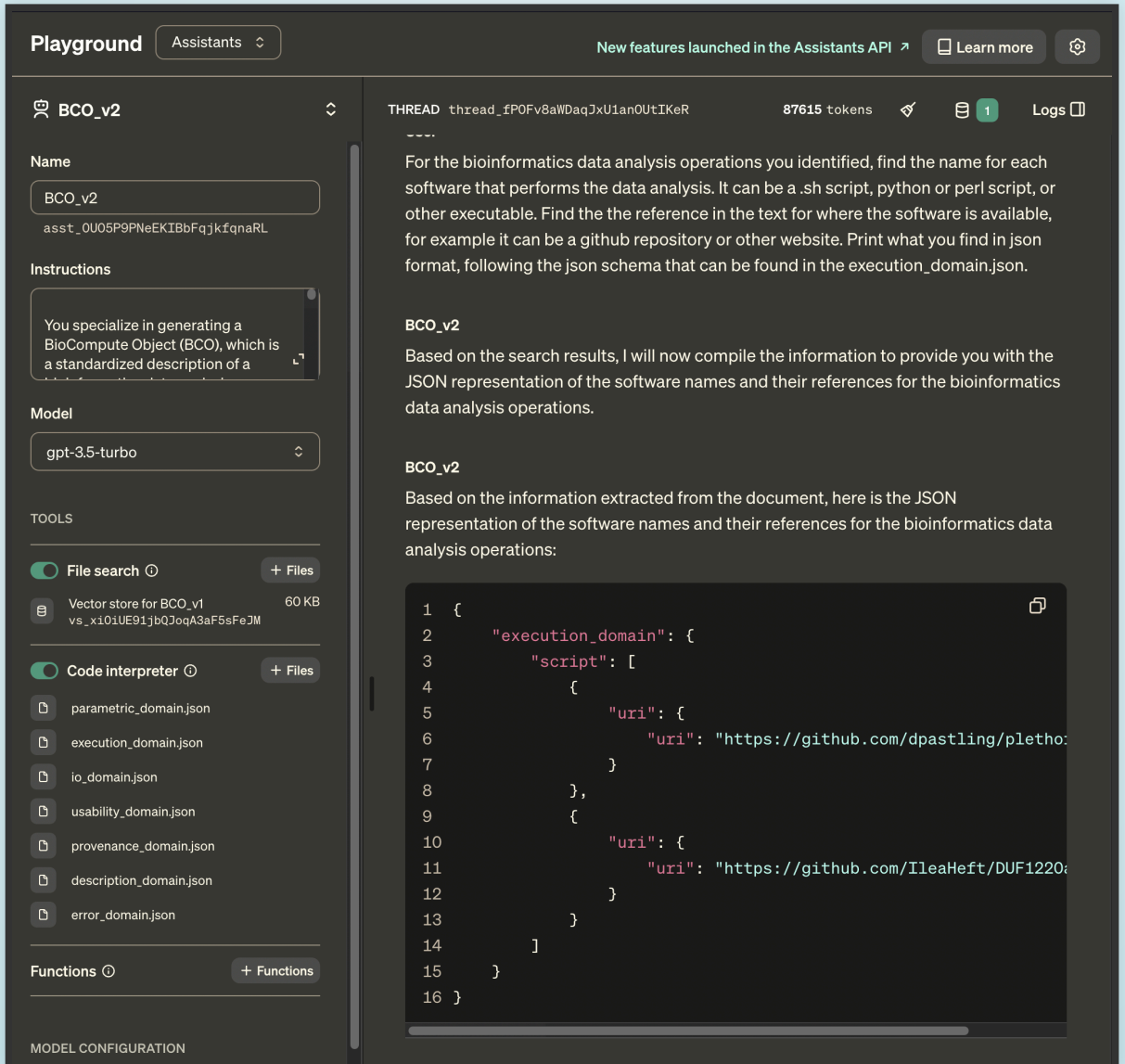
Abstract

Background: DUF1220 protein domains found primarily in Neuroblastoma BreakPoint Family (*NBPF*) genes show the greatest human lineage-specific increase in copy number of any coding region in the genome. There are 302 haploid copies of DUF1220 in hg38 (~160 of which are human-specific) and the majority of these can be divided into 6 different subtypes (referred to as clades). Copy number changes of specific DUF1220 clades have been associated in a dose-dependent manner with brain size variation (both evolutionarily and within the human population), cognitive aptitude, autism severity, and schizophrenia severity. However, no published methods can directly measure copies of DUF1220 with high accuracy and no method can distinguish between domains within a clade.

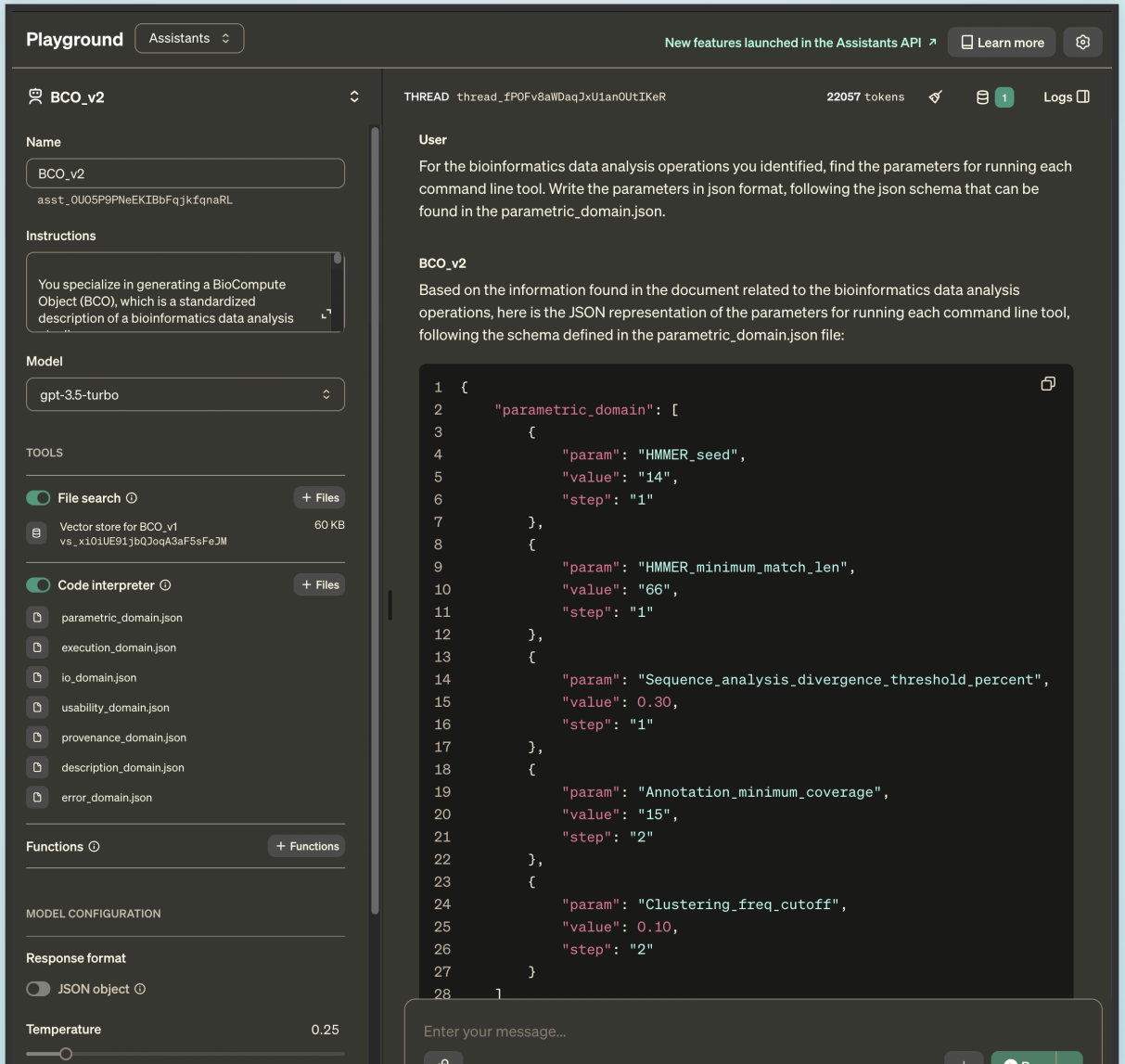
BCOS VIA GPT


Playground
Assistants
New features launched in the Assistants API
Learn more
BCO_v2
Name
BCO_v2
asst_0U05P9PNeEKIBbFqjkfqnaRL
Instructions
You specialize in generating a BioCompute Object (BCO), which is a standardized description of a bioinformatics data analysis
Model
gpt-3.5-turbo
TOOLS
File search
+ Files
Vector store for BCO_v1
vs_xiOiUE91jbQJoqA3aF5sFeJM
60 KB
Code interpreter
+ Files
parametric_domain.json
execution_domain.json
io_domain.json
usability_domain.json
provenance_domain.json
description_domain.json
error_domain.json
Functions
+ Functions
MODEL CONFIGURATION
Response format
JSON object
Temperature
0.25
THREAD thread_fPOFv8aWDaqJxU1anOUtIKeR
22057 tokens
Logs
User
Search in the attached file for all bioinformatics data analysis operations. Write what you find in json format, following the json schema that can be found in the description_domain.json.
DUF1220.pdf
BCO_v2
Based on the information found in the provided document, here is a JSON representation of the bioinformatics data analysis operations, following the schema defined in the description_domain.json file:
{
"spec_version": "https://w3id.org/biocompute/1.3.0/",
"object_id": "https://example.com/bco/9487ae7e-c1aa-4a3c-b18f-3d3695b
"type": "bioinformatics_data_analysis",
"etag": "584C7FE128717E1712426AB19CAAEA8BC1E27365B54285BBEA1221284C7D
"description_domain": {
"external_references": [
{
"description": "High resolution measurement of DUF1220 do
"url": "https://example.com/DUF1220_analysis"
}
],
"pipeline_steps": [
{
"step_name": "Identification of DUF1220 domains",
"description": "Utilized HMMER and a published pipeline t
"input_objects": ["hg38 reference genome"],
"output_objects": ["List of genomic coordinates for DUF12
},
{
"step_name": "Characterizing the read alignment ambiguity
"description": "Performed detailed annotation of NBPF gen
"input_objects": ["DUF1220 domains", "NBPF genes sequence
"output_objects": ["Detailed sequence analysis and cluste
}
],
"io_relationships": [
Enter your message...

BCOS VIA GPT


Playground
Assistants
New features launched in the Assistants API
Learn more
BCO_v2
Name
BCO_v2
asst_OUO5P9PNeEKIBbFqjkfqnaRL
Instructions
You specialize in generating a BioCompute Object (BCO), which is a standardized description of a
Model
gpt-3.5-turbo
TOOLS
File search
+ Files
Vector store for BCO_v1
vs_xiOiUE91jbQJoqA3aF5sFeJM
60 KB
Code interpreter
+ Files
parametric_domain.json
execution_domain.json
io_domain.json
usability_domain.json
provenance_domain.json
description_domain.json
error_domain.json
Functions
+ Functions
MODEL CONFIGURATION
THREAD thread_fPOFv8aWDaqJxU1anOUtIKeR
87615 tokens
Logs
For the bioinformatics data analysis operations you identified, find the name for each software that performs the data analysis. It can be a .sh script, python or perl script, or other executable. Find the the reference in the text for where the software is available, for example it can be a github repository or other website. Print what you find in json format, following the json schema that can be found in the execution_domain.json.
BCO_v2
Based on the search results, I will now compile the information to provide you with the JSON representation of the software names and their references for the bioinformatics data analysis operations.
BCO_v2
Based on the information extracted from the document, here is the JSON representation of the software names and their references for the bioinformatics data analysis operations:
{
"execution_domain": {
"script": [
{
"uri": {
"uri": "https://github.com/dpastling/pletho
}
},
{
"uri": {
"uri": "https://github.com/IleaHeft/DUF1220
}
}
]
}
}

BCOS VIA GPT


Playground
Assistants
New features launched in the Assistants API
Learn more
BCO_v2
Name
BCO_v2
asst_OUO5P9PNeEKIBbFqjkfqnaRL
Instructions
You specialize in generating a BioCompute Object (BCO), which is a standardized description of a bioinformatics data analysis
Model
gpt-3.5-turbo
TOOLS
File search
+ Files
Vector store for BCO_v1
vs_xiOiUE91jbQJoqA3aF5sFeJM
60 KB
Code interpreter
+ Files
parametric_domain.json
execution_domain.json
io_domain.json
usability_domain.json
provenance_domain.json
description_domain.json
error_domain.json
Functions
+ Functions
MODEL CONFIGURATION
Response format
JSON object
Temperature
0.25
THREAD thread_fPOFv8aWDaqJxU1anOUtIKeR
22057 tokens
Logs
User
For the bioinformatics data analysis operations you identified, find the parameters for running each command line tool. Write the parameters in json format, following the json schema that can be found in the parametric_domain.json.
BCO_v2
Based on the information found in the document related to the bioinformatics data analysis operations, here is the JSON representation of the parameters for running each command line tool, following the schema defined in the parametric_domain.json file:
{
"parametric_domain": [
{
"param": "HMMER_seed",
"value": "14",
"step": "1"
},
{
"param": "HMMER_minimum_match_len",
"value": "66",
"step": "1"
},
{
"param": "Sequence_analysis_divergence_threshold_percent",
"value": 0.30,
"step": "1"
},
{
"param": "Annotation_minimum_coverage",
"value": "15",
"step": "2"
},
{
"param": "Clustering_freq_cutoff",
"value": 0.10,
"step": "2"
}
]
Enter your message...

SUMMARY & CONCLUSIONS

- Strong NLP capabilities of GPTs, good results.
- Iterative training, prompting with canonical BCO.
- Fine tuning with BCO json - text chunks dataset.

THANK YOU !