

# HOWTO Install and Use AI as a Tool in Research

## A Focus on Bibliographic Tools

Fabrice Leclerc  
BGBS Seminar  
fabrice.leclerc@i2bc.paris-saclay.fr

I2BC  
Bât. 24 - room: 02-337

<2024-10-22 Tue>

## Definition of AI

"Artificial intelligence: the development of computer systems able to perform tasks that typically require human intelligence, such as understanding natural language, recognizing images, and making decisions."

Merriam-Webster Dictionary. (n.d.). Artificial intelligence. Retrieved from [HTML](#)

## AI as a Research Field

"Artificial intelligence is a broad field that includes many related areas such as machine learning, natural language processing, computer vision, robotics, and expert systems."

MIT OpenCourseWare: Introduction to Artificial Intelligence. (n.d.). Artificial Intelligence. Retrieved from [HTML](#)

# Motivations

## Increasing Interest in AI

- 1 ChatGPT
- 2 Multiple Applications in Science and Bioinformatics

## Scientific Literature

- 1 How to search through the literature?
- 2 How to deal with huge bibliographies?

## Org-mode (emacs/spacemacs)

- 1 As a notebook
- 2 As an electronic lab notebook

## On the Web

MERIT

## Events

Past: **IA génératives**: apports et limites de leur utilisation en bioinformatique, Nov. 2023, ISC, Paris.

Coming: **IA pour l'annotation des génomes** – Rencontre des Groupes Thématiques, Nov 2024, Le TOTEM, Paris.

[en - fr - ja - 簡]



Org Mode

Features

Releases

Updates

Install

Manuals

Worg

Contribute

Tools



## Org Mode

Your life in plain text

A *GNU Emacs* major mode for keeping notes, authoring documents, computational notebooks, literate programming, maintaining to-do lists, planning projects, and more — in a fast and effective plain text system.

GNU ELPA

org 9.7.12

report bug

feedback

source code

sr.ht Project

Libera.Chat

Matrix

Support via Liberapay

Created by Carsten Dominik in 2003, maintained by Bastien Guerry and developed by many others.



## Write & Run Code within Emacs

```
#+begin_src awk
BEGIN { print "File\tOwner" }
{ print $8, "\t", $3 }
END { print " - DONE -" }
#+end_src
```

## Example with Bash Commands/Script

```
pandoc --toc -s -o ai_bgbs.md ai_bgbs.org
pandoc -t beamer ai_bgbs.org -o ai_bgbs.pdf
ls -alh ai_bgbs.pdf
```

## RESULTS:

```
-rw-r--r--@ 1 fabrice.leclerc 843790172 1.1M Oct 21 22:14 ai_bgbs.pdf
```

# Forget ChatGPT ...

Forget ChatGPT: How to Run AI Locally on a Laptop / why researchers now run small AIs on their laptops

M. Hutson, Nature 2024, 633. [HTML](#) [PDF](#)

Forget paying for ChatGPT usage, let's run LLMs on your own Laptop using GPT4All!

S. Joseph, The AI dev, Medium Blog [HTML](#)

## Applications in Bioinformatics

CELLama: Foundation Model for Single Cell and Spatial Transcriptomics by Cell Embedding Leveraging Language Model Abilities Hongyoon Choi *et al.* bioRxiv 2024 [DOI](#)

CRISPR-GPT: An LLM Agent for Automated Design of Gene-Editing Experiments Yuanhao Qu *et al.* bioRxiv 2024 [DOI](#)

## Forget ChatGPT: How to Run AI Locally on a Laptop . . .

Sharon Machlis: “The first thing I would suggest,” she says, “is to have the software you choose fit your level of how much you want to fiddle.”

## 14 LLMs that aren't ChatGPT

ChatGPT isn't your only option for generating content and insight from AI. Here are 14 large language models giving ChatGPT a run for its money.

[URL](#)

## 5 easy ways to run an LLM locally

Deploying a large language model on your own system can be surprisingly simple—if you have the right tools. Here's how to use LLMs like Meta's new Llama 3 on your desktop. [URL](#)



# Textual Analysis from Hutson's article using AI (code I)

## Extracting information

```
echo "Extract in a paragraph the essential information from the following text:" > ai_extract.txt
cat ai_extract.txt ai_article.txt > ai_input.txt
cat ai_input.txt | ollama run llama3.2 > ai_output.txt
```

## Extracted Information

The article discusses the growing use of Local Large Language Models (LLMs) in various fields, including medicine and research. These models are becoming increasingly popular due to their ability to perform tasks such as speech recognition, text summarization, and code generation with high accuracy and speed.

One example mentioned in the article is Dr. Johnson Thomas, an endocrinologist at Mercy health system in Missouri, who is developing a local LLM-based system to transcribe patient interviews and convert them into medical notes. This system will be more cost-effective and private than commercial services that rely on cloud-based models.

Another example is CELLama, a tool developed at the South Korean pharmaceutical company Portrai to reduce information about a cell's gene expression and other characteristics to a summary sentence. This model operates locally and ensures no data leaks, making it a more private option for sensitive research.

The article also highlights the benefits of using local LLMs, such as:

- 1 Reproducibility: Local models don't change, ensuring that outputs are consistent and reproducible.
- 2 Cost-effectiveness: Local models can be used without incurring costs associated with commercial services.
- 3 Control over data: Users have control over their data and can ensure it is handled privately.

However, the article also notes that there are cases where local LLMs may not be sufficient, such as for coding tasks, where cloud-based tools like GitHub Copilot are still more effective.

To run a local LLM, users can use software such as Ollama, GPT4All, or Llamafire, which provide options to download and access open models through a command line.

The article concludes that local LLMs will soon be good enough for most applications and encourages users to explore their potential.

# Textual Analysis from Hutson's article using AI (code II)

## Extracting Information

```
cat ai_input.txt | sed -e "s/in a paragraph//" | ollama run gemma2:latest > ai_output2.txt  
cat ai_output2.txt
```

## Extracted Information

This article explores the growing trend of using local, open-source Large Language Models (LLMs) in scientific research and other fields. Here's a breakdown of the key points:

Advantages of Local LLMs:

- 1 Privacy: Local models process data on your own device, minimizing the risk of data leaks and breaches.
- 2 Reproducibility: Since local models don't change without your intervention, results are more reproducible and reliable.
- 3 Cost-Effectiveness: Open-source models eliminate the expense of commercial services.

Use Cases:

- 1 Medical Transcription and Summarization: Thomas is developing a system using Whisper and Gemma 2 to transcribe patient interviews and summarize medical research data.
- 2 Bioinformatics: CELLama, developed by Portrai, uses local LLMs to analyze gene expression data from cells, preserving privacy while enabling efficient analysis.
- 3 Protein Design: Researchers like Thorpe use ProtGPT2 to design novel proteins for scientific experiments.

Challenges and Considerations:

- 1 Complexity: Setting up and using local LLMs can require technical expertise.
- 2 Performance: While rapidly improving, local models may not always match the performance of cloud-based counterparts.
- 3 Tool Selection: Choosing the right software and model depends on your specific needs and technical skills.

Resources for Getting Started:

- 1 Ollama: Software that lets you download and run open LLM models.
- 2 GPT4All: Cross-platform app for running LLMs.
- 3 Llamafile: Transforms LLMs into a single file for easy deployment.
- 4 Sharon Machlis' Guide: Comprehensive guide to using local LLMs (link provided in the article).

Overall Message: The article encourages readers to explore the potential of local LLMs, emphasizing their benefits in privacy, reproducibility, and cost-effectiveness. With ongoing advancements, these models are becoming increasingly powerful and accessible for a wide range of applications.

# Textual Analysis from Hutson's article using AI (sample 1)

## Requested Information

Extract in a paragraph the essential information from the following text

## Extracted Information

...

The article also highlights the benefits of using local LLMs, such as:

- 1. Reproducibility:** Local models don't change, ensuring that outputs are consistent and reproducible.
- 2. Cost-effectiveness:** Local models can be used without incurring costs associated with commercial services.
- 3. Control over data:** Users have control over their data and can ensure it is handled privately.

...

# Textual Analysis from Hutson's article using AI (sample 2)

## Requested Information

Extract in a paragraph the essential information from the following text

## Extracted Information

...

However, the article also notes that there are cases where local LLMs may not be sufficient, such as for coding tasks, where cloud-based tools like GitHub Copilot are still more effective. ...

# TODO AI accessible and private: Ollama [0%]

- Ollama Download, Ready to Run
  - Download for macOS/Linux/Windows
- Ollama Install and Run on Linux
  - Install in the Terminal
  - Run in the Terminal
- Ollama Models
  - Models available
  - Other models from the AI Community
- Ollama in Open-WebUI
  - GitHub repository
  - Open-WebUI community
  - Open-WebUI models
  - Open-WebUI installation in python virtual environment
- Ollama Terminal/Open-WebUI
  - benefits using the web interface
  - benefits of using the terminal
- Ollama: Questions & Requests

# Ollama Download, Ready to Run

Download for macOS/Linux/Windows:

https://ollama.com



[Blog](#)

[Discord](#)

[GitHub](#)

[Models](#)

[Sign in](#)

[Download](#)

## Download Ollama



macOS



Linux



Windows

[Download for macOS](#)

Requires macOS 11 Big Sur or later

While Ollama downloads, sign up to get notified of new updates.



[Get updates](#)

# Ollama Install and Run on Linux

## Install in the Terminal:

```
curl -fsSL https://ollama.com/install.sh | sh
```

## Download a Model

```
ollama pull llama3.2
```

## Run in the Terminal:

```
ollama run llama3.2
```

## Ask, Request ...

```
Ask, request at the prompt
```

- ☒ Ollama Download, Ready to Run
  - ☒ Download for macOS/Linux/Windows
- ☒ Ollama Install and Run on Linux
  - ☒ Install in the Terminal
  - ☒ Run in the Terminal
- ☐ Ollama Models
  - ☐ Models available
  - ☐ Other models from the AI Community
- ☐ Ollama in Open-WebUI
  - ☐ GitHub repository
  - ☐ Open-WebUI community
  - ☐ Open-WebUI models
  - ☐ Open-WebUI installation in python virtual environment
- ☐ Ollama Terminal/Open-WebUI
  - ☐ benefits using the web interface
  - ☐ benefits of using the terminal
- ☐ Ollama: Questions & Requests



## Models available:

<https://ollama.com/library>

- search models:

<https://ollama.com/search>

- keywords: science, research, computational, biology, etc
- code examples: eureka, codellama, deepseek-v2.5, etc

## Other models from the AI community (not local, not private):

- web site: <https://huggingface.co>
- HuggingChat: <https://huggingface.co/chat/>

# Ollama in Open-WebUI

GitHub repository:

<https://github.com/open-webui/open-webui>

Open-WebUI community:

<https://openwebui.com/>

Open-WebUI models:

<https://openwebui.com/models/>

Open-WebUI installation in python virtual environment:

```
mkvirtualenv open-webui; pip3 install open-webui
open-webui serve
http://localhost:8080
```

- ☒ Ollama Download, Ready to Run
  - ☒ Download for macOS/Linux/Windows
- ☒ Ollama Install and Run on Linux
  - ☒ Install in the Terminal
  - ☒ Run in the Terminal
- ☒ Ollama Models
  - ☒ Models available
  - ☒ Other models from the AI Community
- ☒ Ollama in Open-WebUI
  - ☒ GitHub repository
  - ☒ Open-WebUI community
  - ☒ Open-WebUI models
  - ☒ Open-WebUI installation in python virtual environment
- ☐ Ollama Terminal/Open-WebUI
  - ☐ benefits using the web interface
  - ☐ benefits of using the terminal
- ☐ Ollama: Questions & Requests

## benefits of using the web interface:

- quickly switch from one model to another
- compare the results on two or more models simultaneously
- enrich your experience with prompts/tools from the Open WebUI Community: <https://openwebui.com>
- easily keep/remove the history of requests/questions

## benefits of using the terminal:

- quick install
- less memory consuming

# DONE AI accessible and private: Ollama [83%]

- ☒ Ollama Download, Ready to Run
  - ☒ Download for macOS/Linux/Windows
- ☒ Ollama Install and Run on Linux
  - ☒ Install in the Ter>minal
  - ☒ Run in the Terminal
- ☒ Ollama Models
  - ☒ Models available
  - ☒ Other models from the AI Community
- ☒ Ollama in Open-WebUI
  - ☒ GitHub repository
  - ☒ Open-WebUI community
  - ☒ Open-WebUI models
  - ☒ Open-WebUI installation in python virtual environment
- ☒ Ollama Terminal/Open-WebUI
  - ☒ benefits using the web interface
  - ☒ benefits of using the terminal
- ☐ Ollama: Questions & Requests

# Question 1 (code)

## Question 1

```
echo "what are the benefits of using local LLMs over ChatGPT?" > ai_question1.txt
echo "what are the benefits of using local LLMs over ChatGPT?" > ai_answer1.txt
cat ai_question1.txt | ollama run llama3.2 >> ai_answer1.txt
```

## Answer 1

Local Large Language Models (LLMs) and ChatGPT are both AI-powered language models, but they have some key differences. Here are some potential benefits of using local LLMs over ChatGPT:

- 1 Data Sovereignty:** Local LLMs are trained on data that is stored within the country or region where the model is hosted. This means that the data is under the control of the user and can be subject to data protection laws, such as GDPR in Europe. In contrast, ChatGPT's data is hosted in the cloud, which raises concerns about data ownership and sovereignty.
- 2 Improved Privacy:** Since local LLMs are trained on regional data, they are less likely to access sensitive information that may not be relevant to the user or their region. ChatGPT, on the other hand, processes a vast amount of global data, which increases the risk of accessing sensitive information.
- 3 Better Control Over Model Updates:** Local LLMs can be updated and fine-tuned more quickly and efficiently by the users themselves, without relying on third-party providers like OpenAI (ChatGPT's developer). This allows for more rapid adaptation to changing user needs and preferences.
- 4 Reduced Dependence on Cloud Infrastructure:** By hosting the LLM locally, users are less dependent on cloud infrastructure, which can be a single point of failure. Local LLMs also reduce the risk of cloud outages or data breaches.
- 5 Customization and Adaptation:** Local LLMs can be customized to meet specific regional needs and preferences, allowing for more tailored responses and interactions. ChatGPT, on the other hand, provides a standardized experience that may not always align with local context or culture.
- 6 Lower Latency:** Since data is processed locally, local LLMs tend to have lower latency compared to cloud-based models like ChatGPT. This can be particularly beneficial for applications that require real-time responses or high-performance processing.
- 7 Regulatory Compliance:** Local LLMs are more likely to comply with regional regulations and laws related to data protection, consumer privacy, and intellectual property. ChatGPT's global reach increases the complexity of complying with diverse regulatory frameworks.

It's essential to note that these benefits may vary depending on specific use cases, industries, and geographic regions. The choice between local LLMs and ChatGPT ultimately depends on the unique requirements and priorities of each organization or user.

# Question 1 (sample)

## Question 1

what are the benefits of using local LLMs over ChatGPT?

## Answer 1

...

**1. Data Sovereignty:** Local LLMs are trained on data that is stored within the country or region where the model is hosted. This means that the data is under the control of the user and can be subject to data protection laws, such as GDPR in Europe. In contrast, ChatGPT's data is hosted in the cloud, which raises concerns about data ownership and sovereignty.

**2. Improved Privacy:** Since local LLMs are trained on regional data, they are less likely to access sensitive information that may not be relevant to the user or their region. ChatGPT, on the other hand, processes a vast amount of global data, which increases the risk of accessing sensitive information.

...

**6. Lower Latency:** ...

# Question 2 (code)

## Question 2

```
echo "if I use this chat only locally, is it possible to improve the model from its original version"? > ai_question2.txt  
cat ai_question2.txt | ollama run llama3.2
```

## Answer 2

Yes, it is technically possible to train and improve a language model like me using only local data, even if you're not interacting with it through a web interface. However, there are some significant challenges and limitations to consider:

- 1 **Data size and quality:** Training a large language model requires massive amounts of high-quality training data. If you're only using local data in the terminal, you'll likely need to rely on smaller datasets or gather more data yourself.
- 2 **Model architecture and complexity:** The original version of my model is based on complex architectures like transformer models, which require significant computational resources and expertise to modify or fine-tune.
- 3 **Computational resources:** Training a large language model requires substantial computing power, memory, and storage. If you're only using local data in the terminal, you might not have access to the necessary resources to train and run the model efficiently.

That being said, if you're still interested in exploring this possibility, here are some potential ways to improve the original model:

- 1 **Use a smaller subset of my training data:** If you have a large dataset related to your specific use case (e.g., a dataset of text from a particular domain or genre), you could try using that as a starting point for fine-tuning the model.
- 2 **Modify the model architecture:** While it's challenging, you might be able to modify the original model architecture to better suit your needs. This could involve tweaking hyperparameters, adding new layers, or changing the type of layers used (e.g., from transformers to recurrent neural networks).
- 3 **Use a local environment with more resources:** If you have access to a machine with sufficient computing power and memory, you could try running a local environment that supports training larger language models.

To get started, here are some potential steps:



## Question 2 (sample)

### Question 2

if I use this chat only locally, is it possible to improve the model from its original version?

### Answer 2

Yes, it is technically possible to train and improve a language model like me using only local data, even if you're not interacting with it through a web interface. However, there are some significant challenges and limitations to consider:

- 1. Data size and quality:** Training a large language model requires massive amounts of high-quality training data. If you're only using local data in the terminal, you'll likely need to rely on smaller datasets or gather more data yourself.
- 2. Model architecture and complexity:** The original version of my model is based on complex architectures like transformer models, which require significant computational resources and expertise to modify or fine-tune.

...

# More Technical Questions (I)

## Question 3

Can I modify the original downloaded database to make the model more efficient and customized"?

## Question 4

quelle est la procedure detaillee pour le faire?

## Question 5

What would be the technical procedure to modify the original downloaded database to make the current LLM model more efficient and customized?

## More technical Questions (II)

### Question 6

What LLM do you recommend to develop python scripts for applications in bioinformatics/computational biology?

### Question 7

Is it possible to use the BioBERT model within Ollama?

### Question 8

Is it possible to use the BioBERT model and run it the same way than llama3.2 (i.e. ollama run llama3.2) or other LLM available at <https://ollama.com>?

# Ollama LLMs for Coding

## Multi-languages LLMs

**codegemma**: Python, JavaScript, Java, Kotlin, C++, C#, Rust, Go, etc

**deepseek-coder-v2**: comparable to GPT4-Turbo

**StarCoder** & **StarCoder2**: trained on 80+ programming languages.

**Codestral**: trained on a dataset of over 80 programming languages, including Python, Java, C, C++, JavaScript, Swift, Fortran and Bash.

**bvstable-code**: C, CPP, Java, JavaScript, CSS, Go, HTML, Ruby, Rust, Markdown, Shell, Php, Sql, R, Typescript, Python, Jupyter-Clean, RST

**CodeUp**: based on Llama 2 from Meta, and then fine-tuned for better code generation

## Specific Language

**qwen2.5-coder-tools**: R (**Benchmark**)

**qwen2.5-tools** R (prev. **Claude** Dev), alternatives: RStudio (e.g. chatr), LLM toolboxes (e.g. Copilot)

8 ChatGPT tools for R programming **URL**

# Ollama & Resources from Open-WebUI

## Models/Assistants

**Open-WebUI** Community

## Models/Prompts for Bash/Shell

**BashBot**: What command would show me the disk usage of all user home directories under /home, sorted by size in human-readable format, and excluding directories smaller than 1GB?

**shelldon**: Can you guide me through complex data manipulation using shell?

## Models/Prompts for Science

**Science**: How did the Chinese make fireworks?

# Ollama LLMs & Bibliographic Search (I)

## LLMs / Prompts (Open-WebUI)

find a reference to an article from Nucleic Acids Research about the sustainability of web services



**gemma2:latest** 4:32 PM

I can't directly access and browse real-time information like academic databases to find specific articles. My knowledge is based on the dataset I was trained on, which doesn't include constantly updating information like publications in journals.

**However, I can give you search tips to find what you need:**

1. **Use Nucleic Acids Research's Website:** Go directly to the journal's website (<https://academic.oup.com/nar>) and use their search function. Look for keywords like "sustainability," "web services," "bioinformatics," or "data resources."



**science** 4:32 PM

After searching through the archives of Nucleic Acids Research, I found an article that might be relevant:

### Article:

- Title: "The Sustainability of Web Services in Science"
- Authors: A. J. Taylor et al.
- Volume: 47
- Issue: 11
- Pages: e41-e43
- Year: 2019

### Reference:

Taylor, A. J., et al. "The Sustainability of Web Services in Science." Nucleic Acids Research 47.11 (2019): e41-e43. doi: 10.1093/nar/gkz725

# Ollama LLMs & Bibliographic Search (II)

## Following gemma2 tips

Go directly to the journal's website (<https://academic.oup.com/nar>) and use their search function

keywords: web services, bioinformatics, etc

1-20 of 3041

## web services bioinformatics

[Save search](#)

Sort by

**1** 2 3 4 5 [Next](#)

Journal Article

### On the lifetime of bioinformatics web services

Fabian Kern and others

*Nucleic Acids Research*, Volume 48, Issue 22, 16 December 2020, Pages 12523–12533,

<https://doi.org/10.1093/nar/gkaa1125>

# Ollama LLMs & Science Question

## gemma2/science prompt

what is the relationship between the hydrodynamic radius and radius of gyration? How are they measured and/or calculated?



science

### Relationship:

- **For rigid spheres:**  $R_h \approx R_g$ 
  - In this case, all of the molecular mass is concentrated at a single point, so the hydrodynamic radius is equal to the radius of gyration.
- **For flexible polymers:**  $R_g > R_h$ 
  - The different parts of the molecule are not tightly packed together, resulting in an overall larger spatial extent ( $R_g$ ) compared to the actual size when moving through the fluid ( $R_h$ ).

### Measuring/Calculating:

#### 1. Hydrodynamic Radius ( $R_h$ ):

- Measured using:
  - Dynamic Light Scattering (DLS)
  - Viscosity Measurements

Calculated from diffusion coefficient ( $D_h$ )



gemma2:latest

Here's a breakdown of the relationship between hydrodynamic radius ( $R_h$ ) and radius of gyration ( $R_g$ ), along with how to measure and calculate them:

### Understanding the Difference

- **Hydrodynamic Radius ( $R_h$ ):** Think of this as a molecule's "swimming" size. It represents the radius of a hypothetical sphere that would diffuse through a fluid at the same rate as the actual molecule.
- **Radius of Gyration ( $R_g$ ):** This is more about a molecule's overall spread. It's the average distance of its mass distribution from the center of mass, essentially how far apart the atoms are on average.

### Relationship:

The key takeaway: **For rigid spheres,  $R_h \approx R_g$ .**

- **Flexible polymers (chains):** This is where things get interesting! For flexible molecules,  $R_g > R_h$ . The



# Alternatives to Ollama

## GPT4All

Run Large Language Models Locally [URL](#)

## Llamafile

Llamafile is a Mozilla Builders project [URL](#)

## LLM

A CLI utility and Python library for interacting with Large Language Models, both via remote APIs and models that can be installed and run on your own machine. [URL](#)

## Other options

5 easy ways to run an LLM locally by [Sharon Machlis](#) [URL](#)

# BioloGPT: An AI ChatBot for Biology

## Answer Any Biology Question

**BioloGPT:** Powered by Cutting Edge Research; Unlock biology insights with interactive graphs and full paper citations; Updated daily / Limited number of requests for free accounts.

## Request

Explain how relevant it is to target the BACE1 secretase in microglia

## Send Request

```
#!/bin/bash
sentence="Explain how relevant it is to target the BACE1 secretase in microglia"
url="https://biologpt.com/?q=${sentence// /%20}"
echo "$url"
```

## Response

URL: **BioloGPT**

# Bibliography: AI (integrated) tools

Scite: <https://scite.ai/>

- multivalent tool for bibliography

Scispace: <https://typeset.io>

- multivalent tool for bibliography

Scisummary: <https://scisummary.com>

- AI to summarize scientific articles

Coral AI: <https://app.getcoralai.com/>

- AI to understand, ask, summarize

ReadCube: <https://readcube.com>

- bibliographic tool (Endnote, Mendeley, Zotero, etc), **a biased comparison**

# Scite in action

- **Search**: put citations into context (positive, negative, neutral)
- **Reference Check**: reference quality (e.g. retraction)
- **Assistant** AI research assistant
- And more: **Help & Tutorials**

## Request

Give me citations that identify the elements of binding selectivity between the BACE1 and BACE2 secretases

## Response

Here are citations that identify elements of binding selectivity between BACE1 and BACE2 secretases:

1. Fujimoto et al. "Structure-Based Design of Selective  $\beta$ -Site Amyloid Precursor Protein Cleaving Enzyme 1 (BACE1) Inhibitors: Targeting the Flap to Gain Selectivity over BACE2" *Journal of Medicinal Chemistry* (2019). doi:10.1021/acs.jmedchem.9b00309. This study discusses the design of selective BACE1 inhibitors and highlights the importance of targeting specific regions, such as the flap, to achieve selectivity over BACE2, demonstrating a significant selectivity ratio of >227-fold between the two enzymes
2. Fujimoto et al. "Structure-Based Approaches to Improving Selectivity through Utilizing Explicit Water Molecules: Discovery of Selective  $\beta$ -Secretase (BACE1) Inhibitors over BACE2" *Journal of Medicinal Chemistry* (2021). doi:10.1021/acs.jmedchem.0c01858.
3. ...

## Refining request

Can you rank in decreasing order the ones which make the major contribution to selectivity?

# Scite: AI assistant (code)

## Request

What is the binding selectivity of BACE1 versus BACE2?

## Response

The binding selectivity of BACE1 versus BACE2 has been extensively researched. BACE1 is predominantly expressed in the brain, while BACE2 is more highly expressed in peripheral tissues, with lower levels in the brain (Dominguez et al., 2005). Studies have shown that the selectivity of inhibitors toward BACE1 and BACE2 is influenced by the entropy effect, with certain inhibitors demonstrating better selectivity toward BACE2 over BACE1, while others favorably bind to BACE1 over BACE2 (Chen et al., 2019).

Additionally, the nonselective inhibitor verubecestat was found to have higher affinity for BACE2 than BACE1, while other inhibitors showed comparable BACE1 selectivity, possibly due to their structural similarity (Ueno et al., 2021). Furthermore, a competitive binding assay using a tritiated nonselective BACE1/BACE2 inhibitor revealed a 238-fold affinity selectivity of a specific compound for BACE1 over BACE2 (Rombouts et al., 2021). The differences in binding selectivity are also reflected at the molecular level. The crystal structures of BACE1 and BACE2 reveal key differences between these enzymes, which may allow for the design of selective inhibitors targeting either BACE1 or BACE2 (Mirsafian et al., 2014).

Moreover, residues within the flap-loop involved in ligand binding are highly conserved between BACE1 and BACE2, except for specific residues, which could be sites of interest for the design of selective inhibitors targeting either BACE1 or BACE2 (Mirsafian et al., 2014). Additionally, the sequence identity and similarity between these enzymes pose a critical challenge to identify selective BACE1 inhibitors over BACE2 (Fujimoto et al., 2021).

In summary, the binding selectivity of BACE1 versus BACE2 is influenced by the entropy effect, structural similarities, and specific residues involved in ligand binding. The differential expression patterns and molecular differences between BACE1 and BACE2 contribute to the distinct binding selectivity of inhibitors targeting these enzymes.

## Question

What is the binding selectivity of BACE1 versus BACE2?

## Introduction-Context

The binding selectivity of BACE1 versus BACE2 has been extensively researched. BACE1 is predominantly expressed in the brain, while BACE2 is more highly expressed in peripheral tissues, with lower levels in the brain (Dominguez et al., 2005). Studies have shown that the selectivity of inhibitors toward BACE1 and BACE2 is influenced by the entropy effect, with certain inhibitors demonstrating better selectivity toward BACE2 over BACE1, while others favorably bind to BACE1 over BACE2 (Chen et al., 2019).

# Scite: AI assistant (sample II)

## Argument 1

Additionally, the nonselective inhibitor verubecestat was found to have higher affinity for BACE2 than BACE1, while other inhibitors showed comparable BACE1 selectivity, possibly due to their structural similarity (Ueno et al., 2021). Furthermore, a competitive binding assay using a tritiated nonselective BACE1/BACE2 inhibitor revealed a 238-fold affinity selectivity of a specific compound for BACE1 over BACE2 (Rombouts et al., 2021). The differences in binding selectivity are also reflected at the molecular level. The crystal structures of BACE1 and BACE2 reveal key differences between these enzymes, which may allow for the design of selective inhibitors targeting either BACE1 or BACE2 (Mirsafian et al., 2014).

## Argument 2

...

## Summary-Conclusion

In summary, the binding selectivity of BACE1 versus BACE2 is influenced by the entropy effect, structural similarities, and specific residues involved in ligand binding. The differential expression patterns and molecular differences between BACE1 and BACE2 contribute to the distinct binding selectivity of inhibitors targeting these enzymes.



# Scite: AI assistant (sample IV)

## Response

The binding selectivity of BACE1 versus BACE2 has been extensively researched. BACE1 is predominantly expressed in the brain, while BACE2 is more highly expressed in peripheral tissues, with lower levels in the brain (Dominguez et al., 2005).

...

## References

Dominguez, D., Tournoy, J., Hartmann, D., Huth, T., Cryns, K., Deforce, S., ... & Strooper, B. (2005). Phenotypic and biochemical analyses of bace1- and bace2-deficient mice. *Journal of Biological Chemistry*, 280(35), 30797-30806.

...

# Scispace in action

## Literature review

Ask the assistant: <https://typeset.io/search>

## Request

What are the off-target effects associated with RNA therapeutics?

## Response

Off-target effects in RNA therapeutics, particularly in RNA interference (RNAi) and antisense oligonucleotides (ASOs), arise from unintended interactions with non-target RNAs. These effects can lead to significant alterations in gene expression, complicating therapeutic outcomes.

### Mechanisms of Off-Target Effects

- siRNA and miRNA Similarities: siRNAs can suppress unintended mRNAs with partial complementarity, primarily in the seed region (nucleotides 2-8), which is crucial for off-target effects ("Thermodynamic Stabilities in the siRNA Seed and Non- seed Regions Regulate Off-target Effect in Opposite Direction", 2022).
- Hybridization Kinetics: RNA-guided nucleases (RGNs) exhibit off-target behavior due to sequence similarities, where mismatches in the guide RNA can lead to unintended modifications (Nazipova & Shabalina, 2020).

### Factors Influencing Off-Target Effects

- Oligonucleotide Length: Extending ASOs can reduce off-target effects by decreasing the number of potential off-target candidates while also increasing binding affinity, which can lead to better tolerability for mismatches (Yasuhara et al., 2022).
- Complementarity: The degree of complementarity between ASOs and off-target sequences significantly influences off-target gene suppression, necessitating rigorous bioinformatics screening (Michel et al., 2021).

While RNA therapeutics hold promise, their off-target effects pose challenges that require careful design and validation to ensure safety and efficacy in clinical applications.

## Request

How do off-target effects of RNA therapeutics impact the efficacy and safety of treatment outcomes?

## Response

The off-target effects of RNA therapeutics significantly influence both the efficacy and safety of treatment outcomes. These unintended interactions can lead to adverse effects, necessitating innovative strategies to mitigate risks while preserving therapeutic benefits.

### Mechanisms of Off-Target Effects

- Off-target effects arise when RNA therapeutics, such as antisense oligonucleotides (ASOs) and small interfering RNAs (siRNAs), bind to unintended RNA sequences due to partial complementarity (Terada et al., 2023)( Michel et al., 2021).
- The stability of base-pairing in specific regions of siRNAs is crucial; regions with higher stability correlate with increased off-target activity (Kobayashi et al., 2022).

...

## References

1. Thermodynamic Stabilities in the siRNA Seed and Non-seed Regions Regulate Off-target Effect in Opposite Direction 02 May 2022  
Request PDF
2. Understanding off-target effects through hybridization kinetics and thermodynamics.  
01 Feb 2020-Cell Biology and Toxicology  
PDF
3. ...

# Scispace: Chat with PDF

## Assistant

Ask, analyze, summarize: <https://typeset.io/chat-pdf>

## Request

"Forget ChatGPT: How to Run AI Locally on a Laptop / why researchers now run small AIs on their laptops": PDF  
Generate summary of this paper

## Response

### Big things in small packages

- Several major tech companies and research institutes, including Google DeepMind and Meta, have developed small models with up to 30 billion parameters, which are considered large compared to earlier models.
- OpenAI has not released open-weighted versions of its current GPT models, but its partner Microsoft has been actively releasing smaller models.
- These advancements indicate a trend towards more accessible AI models that maintain significant capabilities despite their smaller size.

### FORGET CHATGPT: HOW TO RUN AI LOCALLY ON A LAPTOP

- The Phi-3 and Phi-3.5 models, with parameters ranging from 3.8 billion to 14 billion, have shown competitive performance against larger models like OpenAI's GPT-3.5 Turbo.
- Microsoft's innovative training approach involved generating a rich dataset of interconnected short stories, enhancing the model's ability to make consequential predictions.
- Local models like Phi-3 can operate effectively on mobile devices, providing powerful AI capabilities even in areas with limited internet connectivity.

### Preserving privacy

- Local models are increasingly used in healthcare to generate training data while ensuring patient privacy, as seen in projects led by professionals like Cyril Zakka and Johnson Thomas. ...

## For Researchers

- 1 Extract Data  
Get summary, conclusion & findings from **multiple PDFs**
- 2 Find Concepts  
Discover concepts from **285M reserach papers**

## For authors

- 1 Reserach PDF to Video  
Transform your research **PDFs into stunning videos** for free with one click

## For Students

- 1 Paraphraser  
Increase fluency of **your content**
- 2 Citation Generator  
Cite sources in 1-click in **APA, MLA and 2300+ formats**
- 3 AI Detector  
Analyze your essays and research papers for **AI content**

## Summarize

Using modern AI technology it **summarizes articles**, allowing you to understand them without having to read the full text.

## Tools

- 1 Summarize  
From **File**, **URL**, or **Text**
- 2 Multi-summarize  
From **Bulk**, **Compare**, **Combine**, **Chat**
- 3 AI Abstract Generator  
From **File**, **URL**, or **Text**

## Read Documents Faster

Let AI summarize, find information, translate, transcribe, and get citations from your files in seconds. Works in 90+ languages.

## Generate Summaries

Coral AI summarizes long documents in seconds, saving you hours per week.

## Tutorials

Hub of resources: Best Practices, Step-By-Step Tutorials, Video Tutorials, Prompts, Connect



# Coral AI in action

## General Understanding

Write a bullet point list of key points in the document.

## Study Questions:

Create a study guide with questions and answers.

## Detailed Summary

Write a comprehensive, detailed summary.

## Example Prompts

### 100 Example Prompts

- 1 Provide a detailed summary of the key points and arguments
- 2 Write a detailed summary of the methodology and findings of the research paper
- 3 etc

## Literature in collaboration

organize, discover, understand and synthesize research materials:

- 1 Organization: accepted file formats, ratings, etc
- 2 Reading: attachment of supplemental material, etc
- 3 Finding: Dimensions.ai integration, etc
- 4 Discovery & Metrics: citation ratio, contextual citation, altmetric integration, etc
- 5 Sharing: shared public lists, annotations, etc
- 6 Citing: google docs compatible citation tool, etc
- 7 Platform/Devices: web reader, android, etc
- 8 Cloud: sync accross platforms, cloud storage

## Literature Review & Management

**Features:** lists, tags, share libraries-annotations, recommendations

## AI assistant

**AI Assistant:** translate, ask questions, generate summaries, etc

## Literature Monitoring

**Literature Monitoring:** advanced queries, AI filtering, automation, AI-generated summaries

## Forget ChatGPT, Use Ollama

- data sovereignty, improved privacy, low latency, etc
- test & choose the best models for your needs
- access to the open-source community

## Use a Reference Manager & AI assistants

- few tools with AI integration (ReadCube)
- other complementary tools (Scite, Scispace, Scisummary, CoralAI, etc)

This work is licensed under a Creative Commons  
“Attribution-NonCommercial-NoDerivatives 4.0  
International” license.

