Accessing and Running Large Language Models (LLMs): A Comprehensive Guide

Shubin Yu[1]

[1]Generative AI for Research Initiative, https://gaiforresearch.com

May 15, 2025

Abstract

This guide provides comprehensive instructions for accessing and utilizing Large Language Models (LLMs) through various methods including commercial APIs, local execution tools, and cloud deployment options. Researchers, developers, and general users will find practical information on obtaining API keys, understanding pricing models, and implementing alternative approaches for running models locally or through cloud services.

Contents

1 Commercial LLM API Access

API keys enable your applications to interact with large language models hosted by various providers. This section guides you through obtaining API keys from major LLM providers.

1.1 Google Gemini API

Google Gemini API Access

1. Visit Google AI Studio: `https://aistudio.google.com/`
2. Sign in with your Google account
3. Navigate to ”API” section (typically in left sidebar)
4. Follow prompts to create a Google Cloud project if needed
5. Generate and copy your API key
6. Store the key securely as it may only be displayed once

1.2 OpenAI API

OpenAI API Access

1. Visit OpenAI Platform: `https://platform.openai.com/`
2. Create an account or log in
3. Navigate to ”API keys” section
4. Click ”Create new secret key”
5. Optionally name your key for organization
6. Copy and securely store the key immediately (it won’t be shown again)

1.3 Anthropic Claude API

Anthropic Claude API Access

1. Go to Anthropic Console: `https://console.anthropic.com/`
2. Create an account or sign in
3. Navigate to the API Keys section
4. Generate a new API key
5. Copy and securely store your key

1.4 DeepSeek API

DeepSeek API Access

1. Visit DeepSeek's developer portal
2. Register for an account or sign in
3. Navigate to API management section
4. Generate a new API key
5. Save your API key in a secure location

1.5 Mistral AI API

Mistral AI API Access

1. Visit Mistral AI's platform: `https://console.mistral.ai/`
2. Create an account or sign in
3. Access the API Keys section
4. Generate a new API key
5. Securely store your API key

2 Understanding LLM Pricing

Most LLM providers charge based on the number of tokens processed. A token is roughly 4 characters or 3/4 of a word in English.

2.1 Pricing Comparison Tool

The LLM API Pricing Calculator at `https://yourgpt.ai/tools/openai-and-other-llm-api-pricing-calculator` provides up-to-date comparisons of pricing across providers.

2.2 Sample Pricing Highlights (per 1,000 tokens)

Model	**Input Cost**	**Output Cost**	**Context Length**
OpenAI GPT-4o	$0.005	$0.015	128K
Claude 3.5 Haiku	$0.0008	$0.004	200K
Gemini 1.5 Flash	$0.000075	$0.0003	1M
Mistral Large	$0.008	$0.024	32K
DeepSeek-R1	$0.00055	$0.00219	64K

Cost Saving Tips

- Use smaller, efficient models for simpler tasks
- Optimize prompts to reduce token usage
- Leverage free tiers and credits for experimentation
- Consider running open-source models locally for high-volume applications

3 Running Models Locally

For researchers who prefer to run models locally for privacy, reduced costs, or specialized research applications, several tools make this possible without extensive technical expertise.

3.1 LM Studio

LM Studio Overview

LM Studio is a desktop application for downloading and running open-source LLMs with a user-friendly interface.

Key Features:

- Graphical user interface (GUI) for easy model management
- Built-in model discovery and download functionality
- Support for GGUF and MLX model formats
- Chat interface for direct interaction with models
- Local server compatible with OpenAI API format
- Available for Windows, macOS, and Linux

Getting Started:

1. Download from `https://lmstudio.ai/`
2. Install and launch the application
3. Browse the model catalog
4. Download a model that fits your hardware capabilities
5. Start chatting or set up the server for API access

3.2 Ollama

Ollama Overview

Ollama is a lightweight command-line tool for running LLMs locally with minimal setup, making it ideal for research applications.
Key Features:

- Simple command-line interface
- Easy model management
- Built-in API server
- Low resource overhead
- Support for popular open-source models
- Available for macOS, Linux, and Windows

Installation: For Linux:

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

For macOS and Windows, download the installer from `https://ollama.com/`.
Basic Commands:

```
# Download a model
ollama pull llama3

# Run a model in chat mode
ollama run llama3

# List installed models
ollama list

# Start the API server
ollama serve
```

4 Using Ollama in Google Colab

Google Colab provides free access to GPUs, making it an excellent platform for research involving LLMs even without specialized hardware.

4.1 Setting Up the Environment

Configure Colab Runtime

1. Open Google Colab: `https://colab.research.google.com/`
2. Create a new notebook
3. Go to Runtime → Change runtime type
4. Select ”T4 GPU” as the Hardware accelerator
5. Click ”Save”

4.2 Installing Terminal Extension

Colab doesn’t have a built-in terminal, so we’ll install a terminal extension:

```
!pip install colab_xterm
%load_ext colab_xterm
%xterm
```

4.3 Installing Ollama

Once the terminal is available, install Ollama:

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

4.4 Setting Up and Running a Model

Start the Ollama server and download a model:

```
# Start the Ollama server in the background
ollama serve &

# Download a model (smaller models work better in Colab)
ollama pull deepseek-r1:1.5b
```

4.5 Installing Python Library and Testing

Install the Python library to interact with Ollama:

```
!pip install ollama
```

Test your model with a simple Python script:

```
import ollama
import asyncio

async def run_inference():
  try:
    response = await ollama.AsyncClient().chat(
        model=’deepseek-r1:1.5b’,
        messages=[
            {’role’: ’user’, ’content’: ’Explain how LLMs can be used in academic research.’
    }
        ]
    )
    print(response['message']['content'])
  except Exception as e:
    print(f"An error occurred: {e}")
    print("Ensure the Ollama server is running and the model is downloaded.")

# Run the async function
asyncio.run(run_inference())
```

Important Note

You need to reinstall Ollama, download models, and install the Python library each time you start a new Colab session or reconnect after inactivity since Colab environments are temporary.

5 Research Applications of LLMs

Large Language Models are transforming research methodologies across disciplines. The Generative AI for Research Initiative (`https://gaiforresearch.com`) provides resources for researchers looking to incorporate AI into their workflow.

Research Applications

- **Literature Review:** Summarizing papers, identifying research gaps
- **Data Analysis:** Text mining, content analysis of qualitative data
- **Writing Assistance:** Drafting sections, improving clarity
- **Interview Research:** Tools like MimiTalk for AI-mediated interviews
- **Brainstorming:** Generating research questions and hypotheses
- **Code Generation:** Creating analysis scripts for quantitative research

Ethical Considerations

When using LLMs in research, consider:

- Check your institution's AI policy
- Review journal policies on AI-assisted writing
- Maintain transparency about AI use
- Verify all AI-generated content for accuracy
- Consider data privacy when using API-based services

6 Conclusion

This guide has provided multiple approaches to working with Large Language Models for research purposes:

- **Commercial APIs:** For production use, access to cutting-edge models, and when simplicity is key
- **Local Tools:** For privacy, reduced costs, and specialized applications
- **Cloud Deployment:** For leveraging free GPU resources without local hardware investment

The approach you choose should depend on your specific research needs, technical capabilities, and resources. Many researchers combine these approaches—using commercial APIs for critical applications while experimenting with local models for exploratory research.

For more specialized guidance on integrating generative AI into your research methodology, visit the Generative AI for Research Initiative at `https://gaiforresearch.com`.

References

[1] Generative AI for Research Initiative (2025). *Home — Generative AI for Research*. Retrieved from `https://www.gaiforresearch.com/`

[2] YourGPT (2025). *LLM Cost Calculator: Compare OpenAI, Claude2, PaLM, Cohere & More*. Retrieved from `https://yourgpt.ai/tools/openai-and-other-llm-api-pricing-calculator`

[3] Ollama (2025). *Ollama - Get up and running with large language models*. Retrieved from `https://ollama.com/`

[4] LM Studio (2025). *LM Studio - Discover, download, and run local LLMs*. Retrieved from `https://lmstudio.ai/`