User Guide

Public documentation on using the inference platform

Overview

The **Vector Inference Platform** is a service provided by the AI Engineering team at Vector Institute. It hosts large, state-of-the-art open-source language models that anyone in the Vector community can use freely and easily.

Unlike previous efforts to provide inference services on Vector's compute environment, this platform is a **production-grade, always-available service**. Users do not need to spin up their own models via Slurm jobs or worry about time limits — models remain persistently online.

The source code and technical documentation for this project are available on the GitHub repository.

For the current list of available models and their specifications, visit inference.vectorinstitute.ai.

The AI Engineering team will continue to add new models as the service matures. Feedback and model requests are welcome — contact the AI Engineering team.

Getting Started

The Vector Inference Platform is available to all Vector Institute community members. For an up-to-date list of available models and their specifications, visit inference.vectorinstitute.ai.

Getting an API Key

Access is managed via invite links. To get started:

1. Contact the AI Engineering team to request access. They will create your account and send you an invite email.
2. Click the link in the invite email. It is valid for **48 hours** and takes you to the Vector Proxy dashboard.
3. Your API key is generated and displayed **once** — copy and save it immediately. The key will not be shown again.

API keys have the format `vp_xxxxxxxx.yyyyyy...`.

Prerequisites

Install the OpenAI Python client:

```
pip install openai
```

Usage

The platform exposes an OpenAI-compatible API at **https://proxy.vectorinstitute.ai/v1**. Use it as a drop-in replacement for any OpenAI client by changing the `base_url` and `model` parameters.

```
from openai import OpenAI

client = OpenAI(
    base_url="https://proxy.vectorinstitute.ai/v1",
    api_key="vp_xxxxxxxx.yyyyyy..."
)

stream = client.chat.completions.create(
    model="<model-id>",  # see inference.vectorinstitute.ai for available models
    messages=[{"role": "user", "content": "Explain attention mechanisms in transformers."}],
    stream=True,
)
```

```
for chunk in stream:
    if chunk.choices:
        print(chunk.choices[0].delta.content or "", end="", flush=True)
```

You can also use curl:

```
curl https://proxy.vectorinstitute.ai/v1/chat/completions \
  -H "Authorization: Bearer vp_xxxxxxxx.yyyyyy..." \
  -H "Content-Type: application/json" \
  -d '{
    "model": "<model-id>",
    "messages": [{"role": "user", "content": "Hello!"}]
  }'
```

Listing Available Models

Retrieve the current list of enabled models programmatically:

```
curl https://proxy.vectorinstitute.ai/v1/models \
  -H "Authorization: Bearer vp_xxxxxxxx.yyyyyy..."
```

Or visit inference.vectorinstitute.ai for a visual overview.