* Research
* API
* ChatGPT
* [Safety](https://openai.com/safety)
* Company

Search

**Navigation quick links**

* [Log in](https://platform.openai.com/login?launch)
* [Try ChatGPT](https://chat.openai.com/)

[Blog](https://openai.com/blog)

**GPT-3.5 Turbo fine-tuning and API updates**

Developers can now bring their own data to customize GPT-3.5 Turbo for their use cases.

Illustration: Ruby Chen

August 22, 2023

**Authors**

* [Andrew Peng](https://openai.com/blog/gpt-3-5-turbo-fine-tuning-and-api-updates#AndrewPeng)
* [Michael Wu](https://openai.com/blog/gpt-3-5-turbo-fine-tuning-and-api-updates#MichaelWu)
* [John Allard](https://openai.com/blog/gpt-3-5-turbo-fine-tuning-and-api-updates#JohnAllard)
* [Logan Kilpatrick](https://openai.com/blog/gpt-3-5-turbo-fine-tuning-and-api-updates#LoganKilpatrick)
* [Steven Heidel](https://openai.com/blog/gpt-3-5-turbo-fine-tuning-and-api-updates#StevenHeidel)

Fine-tuning for GPT-3.5 Turbo is now available, with fine-tuning for GPT-4 coming this fall. This update gives developers the ability to customize models that perform better for their use cases and run these custom models at scale. Early tests have shown a fine-tuned version of GPT-3.5 Turbo can match, or even outperform, base GPT-4-level capabilities on certain narrow tasks. As with all our APIs, data sent in and out of the fine-tuning API is owned by the customer and is [not used by OpenAI](https://openai.com/api-data-privacy), or any other organization, to train other models.

**Fine-tuning use cases**

Since the release of GPT-3.5 Turbo, developers and businesses have asked for the ability to customize the model to create unique and differentiated experiences for their users. With this launch, developers can now run supervised fine-tuning to make this model perform better for their use cases.

In our private beta, fine-tuning customers have been able to meaningfully improve model performance across common use cases, such as:

* **Improved steerability**: Fine-tuning allows businesses to make the model follow instructions better, such as making outputs terse or always responding in a given language. For instance, developers can use fine-tuning to ensure that the model always responds in German when prompted to use that language.
* **Reliable output formatting:** Fine-tuning improves the model's ability to consistently format responses—a crucial aspect for applications demanding a specific response format, such as code completion or composing API calls. A developer can use fine-tuning to more reliably convert user prompts into high-quality JSON snippets that can be used with their own systems.
* **Custom tone:** Fine-tuning is a great way to hone the qualitative feel of the model output, such as its tone, so it better fits the voice of businesses’ brands. A business with a recognizable brand voice can use fine-tuning for the model to be more consistent with their tone.

In addition to increased performance, fine-tuning also enables businesses to **shorten their prompts**while ensuring similar performance.  Fine-tuning with GPT-3.5-Turbo can also handle 4k tokens—double our previous fine-tuned models. Early testers have reduced prompt size by up to 90% by fine-tuning instructions into the model itself, speeding up each API call and cutting costs.

Fine-tuning is most powerful when combined with [other techniques](https://platform.openai.com/docs/guides/gpt-best-practices) such as prompt engineering, information retrieval, and function calling. Check out our [fine-tuning guide](https://platform.openai.com/docs/guides/fine-tuning) to learn more. Support for fine-tuning with function calling and gpt-3.5-turbo-16k will be coming later this fall.

**Fine-tuning steps**

Step 1

**Prepare your data**

{

"messages": [

{ "role": "system", "content": "You are an assistant that occasionally misspells words" },

{ "role": "user", "content": "Tell me a story." },

{ "role": "assistant", "content": "One day a student went to schoool." }

]

}

Step 2

**Upload files**

curl https://api.openai.com/v1/files \

-H "Authorization: Bearer $OPENAI\_API\_KEY" \

-F "purpose=fine-tune" \

-F "file=@path\_to\_your\_file"

Step 3

**Create a fine-tuning job**

curl https://api.openai.com/v1/fine\_tuning/jobs \

-H "Content-Type: application/json" \

-H "Authorization: Bearer $OPENAI\_API\_KEY" \

-d '{

"training\_file": "TRAINING\_FILE\_ID",

"model": "gpt-3.5-turbo-0613"

}'

Once a model finishes the fine-tuning process, it is available to be used in production right away and has the same shared [rate limits](https://platform.openai.com/docs/guides/rate-limits/what-are-the-rate-limits-for-our-api) as the underlying model.

Step 4

**Use a fine-tuned model**

curl https://api.openai.com/v1/chat/completions \

-H "Content-Type: application/json" \

-H "Authorization: Bearer $OPENAI\_API\_KEY" \

-d '{

"model": "ft:gpt-3.5-turbo:org\_id",

"messages": [

{

"role": "system",

"content": "You are an assistant that occasionally misspells words"

},

{

"role": "user",

"content": "Hello! What is fine-tuning?"

}

]

}'

We will also be debuting a fine-tuning UI in the near future, which will give developers easier access to information about ongoing fine-tuning jobs, completed model snapshots, and more.

**Safety**

It is very important to us that the deployment of fine-tuning is safe. To preserve the default model's safety features through the  fine-tuning process, fine-tuning training data is passed through our Moderation API and a GPT-4 powered moderation system to detect unsafe training data that conflict with our safety standards.

**Pricing**

Fine-tuning costs are broken down into two buckets: the initial training cost and usage cost:

* Training: $0.008 / 1K Tokens
* Usage input: $0.012 / 1K Tokens
* Usage output: $0.016 / 1K Tokens

For example, a gpt-3.5-turbo fine-tuning job with a training file of 100,000 tokens that is trained for 3 epochs would have an expected cost of $2.40.

**Updated GPT-3 models**

In July, [we announced](https://openai.com/blog/gpt-4-api-general-availability) that the original GPT-3 base models (ada, babbage, curie, and davinci) would be turned off on January 4th, 2024. Today, we are making babbage-002 and davinci-002 available as replacements for these models, either as base or fine-tuned models. Customers can access those models by querying the [Completions API](https://platform.openai.com/docs/api-reference/completions).

These models can be fine-tuned with our new API endpoint /v1/fine\_tuning/jobs. This new endpoint offers pagination and more extensibility to support the future evolution of the fine-tuning API. Transitioning from /v1/fine-tunes to the updated endpoint is straightforward and more details can be found in our new [fine-tuning guide](https://platform.openai.com/docs/guides/fine-tuning). This deprecates the old /v1/fine-tunes endpoint, which will be turned off on January 4th, 2024.

Pricing for base and fine-tuned GPT-3 models is as follows:

**Base models**

**Fine-tuned models**

**Model**

**Input tokens**

**Output tokens**

**Training**

**Input tokens**

**Output tokens**

babbage-002

$0.0004 / 1K tokens

$0.0004 / 1K tokens

$0.0004 / 1K tokens

$0.0016 / 1K tokens

$0.0016 / 1K tokens

davinci-002

$0.002 / 1K tokens

$0.002 / 1K tokens

$0.006 / 1K tokens

$0.012 / 1K tokens

$0.012 / 1K tokens

**Authors**

* **Andrew Peng**

[View all articles](https://openai.com/blog?authors=andrew-peng)

* **Michael Wu**

[View all articles](https://openai.com/blog?authors=michael-wu)

* **John Allard**

[View all articles](https://openai.com/blog?authors=john-allard)

* **Logan Kilpatrick**

[View all articles](https://openai.com/blog?authors=logan-kilpatrick)

* **Steven Heidel**

[View all articles](https://openai.com/blog?authors=steven-heidel)