モデル管理APIチュートリアル
はじめに

本書は、ファインチューニング済みモデルのアップロードとデプロイするチュートリアルです。

本書の対象読者は以下を想定しています。

- チューニング済モデル利用のオプションメニューを契約し、ファインチューニング済みのモデルを本サービスに登録したいSI開発者およびシステム管理者

チュートリアルの流れ

チュートリアルは以下の流れで進みます。

1. モデルリポジトリへのアップロード
2. モデルリポジトリの一覧取得
3. モデルのデプロイ
4. デプロイ済みモデルの一覧取得
5. モデルのアンデプロイ
6. モデルリポジトリの削除

前提条件

ファインチューニング済みのモデルをデプロイするには、オプション契約が必要です。デプロイできるモデルの数はオプション契約数に依存します。

事前準備

ファインチューニング済みのモデルをお手元にご用意ください。

また、モデルのファイルのアップロードにAzCopyを使用します。

> LoRAモデルを利用する場合は、ベースとなるモデルを格納した場所に ”lora-model” ディレクトリを作成し、LoRAモデルのファイルを ”lora-model” ディレクトリに格納してください。
>
> また、モデルの設定やパラメータを定義する config.json を、モデルを格納した場所 と “lora-model “ ディレクトリの両方に格納してください。

モデルリポジトリへのアップロード

アップロード用URLとトークンの取得

ファインチューニング済みのモデルをモデルのリポジトリにアップロードします。

まず、アップロード用のURLとトークンを取得します。`{モデル名}`には、モデルリポジトリで使うためのモデルの名称を指定してください。

```
curl -X POST https://api.genai-api.nec-cloud.com/genai-modelrepos-api/modelrepos/{モデル名}/generate_upload_url ¥
-H ”Authorization: Bearer {アクセストークン}” -H ”Content-Length: 0”
```

以下のように、URLとトークンが返ってきます。

```
{
  ”upload_url”: ”https://...”,
  ”token”: ”........”
}
```

トークンの有効期限は1時間です。有効期限が切れた場合は、モデルリポジトリを削除した後、アップロード用URLとトークンを取得しなおしてください。

ファイルのアップロード

AzCopyを使用する場合

続いてAzCopyを使用して、モデルのファイルをアップロードします。`{model_dir}`にはモデルのファイルがあるディレクトリを指定してください。`{upload_url}`と`{token}`には、上記で返ってきたURLとアクセストークンを入れてください。

```
azcopy copy '{model_dir}/*' '{upload_url}?{token}' --recursive
```

curlを使用する場合

curlでアップロードする場合は、各ファイルをPUTでアップロードします。`{model_dir}`にはモデルのファイルがあるディレクトリを、`{ファイル名}`にはアップロードするファイルの名前を指定してください。`{upload_url}`と`{token}`には、上記で返ってきたURLとアクセストークンを入れてください。また、リクエストヘッダで`x-ms-blob-type: BlockBlob`を指定する必要があります。

```
curl -X PUT -H "x-ms-blob-type: BlockBlob" --upload-file {model_dir}/{ファイル名} {upload_url}/{ファイル名}?{token}
```

ファイルサイズが5,000MiBを超える場合は、ファイルを4,000MiB以下のブロックに分割してアップロードする必要があります。

大きなファイル`file.bin`を、`file.bin.001`, `file.bin.002`, `file.bin.003`に分割してアップロードする場合で説明します。

まず、以下のように分割したファイルをブロックとしてアップロードします。クエリパラメータで`comp=block`を指定します。`blockid`はブロックを識別する一意なIDを付けて、Base64文字列にエンコードします。Base64文字列はURLエンコードする必要があります。

```
curl -X PUT --upload-file {model_dir}/file.bin.001 {upload_url}/file.bin?comp=block&blockid=AAAAAA%3D%3D&{token}
curl -X PUT --upload-file {model_dir}/file.bin.002 {upload_url}/file.bin?comp=block&blockid=AAAAAB%3D%3D&{token}
curl -X PUT --upload-file {model_dir}/file.bin.003 {upload_url}/file.bin?comp=block&blockid=AAAAAC%3D%3D&{token}
```

次に、以下のようにブロックIDを並べたブロックリストを作成します。

```
<?xml version="1.0" encoding="utf-8"?>
<BlockList>
  <Uncommitted>AAAAAA==</Uncommitted>
  <Uncommitted>AAAAAB==</Uncommitted>
  <Uncommitted>AAAAAC==</Uncommitted>
</BlockList>
```

作成したブロックリストをPUTします。

```
curl -X PUT --upload-file {model_dir}/blocklist.xml {upload_url}/file.bin?comp=blocklist&{token}
```

ファイルのアップロードについての詳細は、Microsoft AzureのBLOBサービスのドキュメントを参照してください。

Put Blob (REST API) - Azure Storage | Microsoft Learn

Put Block (REST API) - Azure Storage | Microsoft Learn

Put Block List (REST API) - Azure Storage | Microsoft Learn

モデルリポジトリの一覧取得

モデルリポジトリにあるモデルの一覧を取得します。アップロード用のURLを取得した時点で、モデルリポジトリに登録されます。

```
curl https://api.genai-api.nec-cloud.com/genai-modelrepos-api/modelrepos ¥
-H "Authorization: Bearer {アクセストークン}"
```

以下のようにモデルリポジトリに登録されたモデルの情報が取得できます。

```
[
  {
    "model_name": "{モデル名}",
    "deployed": false,
    "aliases": []
  },
    ...
]
```

`deployed`はモデルがデプロイ済みかどうかを示します。モデルがデプロイ済みの場合、デプロイ時に指定したエイリアスが`aliases`に入ります。また、デプロイ済みのモデルは削除できません。削除する前にアンデプロイしてください。

モデルのデプロイ

モデルをデプロイします。以下は、 `curl` コマンドを使用してJSONデータをAPIにPOSTする基本的な例です。

```
curl -X PUT https://api.genai-api.nec-cloud.com/genai-models-api/models/{エイリアス} ¥
     -H "Authorization: Bearer {アクセストークン}" ¥
     -d "{¥"model_name¥": ¥"{モデル名}¥", ¥"use_lora¥": true, ¥"llm_type¥": ¥"{プリセット名}¥"}"
```

URLの `{エイリアス}` には、モデル名のエイリアスを指定してください。推論リクエスト時にこの名前を指定して、推論に使用するモデルを特定します。

リクエストボディの `{モデル名}` にはモデルリポジトリに登録したモデルの名前を指定してください。

LoRAモデルをデプロイする場合は、 `use_lora` パラメータに `true` を指定してください。 LoRAモデルを使用しない場合は、 `use_lora` パラメータに `false` を指定するか、 `use_lora` パラメータを削除します。

`llm_type` パラメータには デプロイするモデルのサイズに応じて、次のプリセット名を指定します。このパラメータを省略した場合は `g2.small` が採用されます。

プリセット名一覧：

プリセット名	モデルのサイズ
`g2.small`	cotomi-fast-v2.0 ベース
`g2.large`	cotomi-pro-v2.0 ベース
`g3.xlarge`	cotomi-v3.0 ベース

デプロイ済みモデルの一覧取得

デプロイされたモデルを取得します。

```
curl https://api.genai-api.nec-cloud.com/genai-models-api/models -H "Authorization: Bearer {アクセストークン}"
```

返される一覧には、デプロイしたファインチューニング済みのモデルだけでなく、全テナント共通で提供されるNEC cotomiのモデルも含まれます。

```
{
  "object": "list",
  "data": [
    {
      "id": "cotomi-fast-v2.0",
      "object": "model",
      "created": 1234567890,
      "owned_by": "nec",
      "ready": true
    },
    {
      "id": "cotomi-pro-v2.0",
      "object": "model",
      "created": 1234567890,
      "owned_by": "nec",
      "ready": true
    },
    {
      "id": "cotomi-v3.0",
      "object": "model",
      "created": 1234567890,
      "owned_by": "nec",
      "ready": true
    },
    {
      "id": "{エイリアス}",
      "object": "model",
      "created": 1234567890,
      "owned_by": "{サブスクリプション名}",
      "model_name": "{モデルリポジトリのモデル名}"
      "ready": true,
      "llm_type": "g2.small"
    },
      ...
  ]
}
```

`ready` が `true` になれば、モデルにリクエストできるようになります。

モデルのアンデプロイ

使用しないモデルをアンデプロイします。 `{エイリアス}` にはアンデプロイするモデルのエイリアスを指定してください。

```
curl -X DELETE https://api.genai-api.nec-cloud.com/genai-models-api/models/{エイリアス} ¥
-H "Authorization: Bearer {アクセストークン}"
```

モデルリポジトリの削除

モデルのファイルをモデルリポジトリから削除します。`{モデル名}`には、削除するモデルの名前を指定してください。デプロイ済みのモデルは削除できません。削除する前にアンデプロイしてください。

```
curl -X DELETE https://api.genai-api.nec-cloud.com/genai-modelrepos-api/modelrepos/{モデル名} ¥
-H "Authorization: Bearer {アクセストークン}"
```