

[Open in app](#)[Sign up](#)[Sign In](#)

WhisperをFineTuningして専門用語を認識可能にする

Kazuki Kyakuno · [Follow](#)

Published in axinc

28 min read · May 1

 [Share](#)

Whisperを少量のデータセットでFine Tuningして専門用語を認識可能にする方法を解説します。Tacotron2の合成音声でデータセットを作成することで、専門用語を認識可能なWhisperモデルを作成します。

Whisperについて

WhisperはOpenAIの開発した音声認識モデルです。日本語を含む多言語に対応しており、高精度な音声認識が可能です。ただし、学習時に使用していない専門用語は認識できないという問題があります。

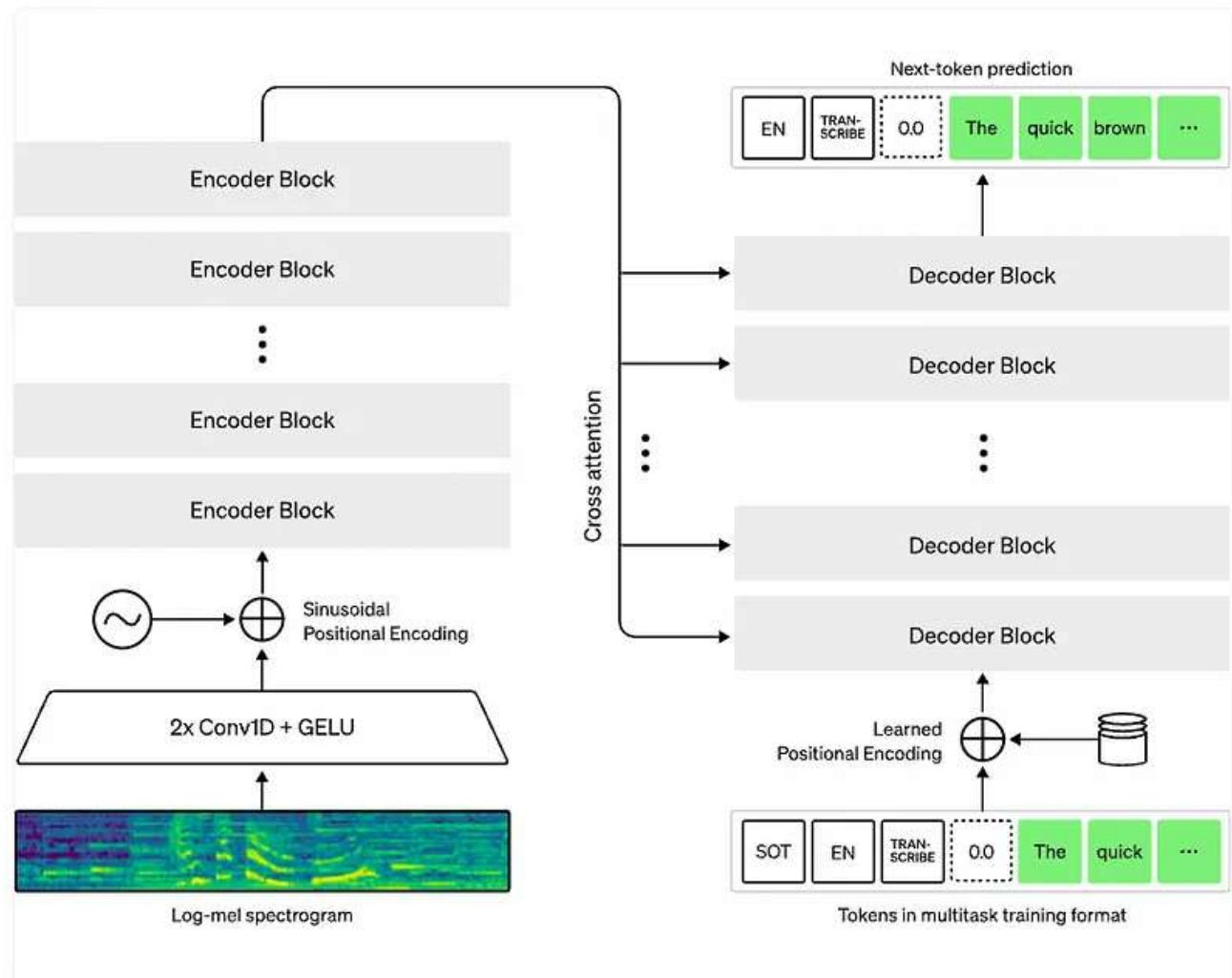


Figure 1: Whisper model. The architecture follows the standard Transformer-based encoder-decoder model. A log-Mel spectrogram is input to the encoder. The last encoder hidden states are input to the decoder via cross-attention mechanisms. The decoder autoregressively predicts text tokens, jointly conditional on the encoder hidden states and previously predicted tokens. Figure source: [OpenAI Whisper Blog](#).

Whisperのアーキテクチャ（出典：<https://huggingface.co/blog/fine-tune-whisper>）

Whisperにおける専門用語の扱いについて

Whisperで専門用語を取り扱う場合、`initial_prompt`に専門用語を埋め込むという方法があります。しかし、`initial_prompt`にはコンテキストサイズの半分の224トークンまでしか使用できないという制約があり、専門用語の数が多い場合には使用できません。

WhisperにおけるPrompt Engineering

音声認識を行うAIモデルであるWhisperも言語モデルを使用しているためPrompt Engineeringを適用することが可能ですが、本記事で...

medium.com/axinc/whisperをfine-tuningして専門用語を認識可能にする-3744e2779c71

WhisperにおけるFineTuning

PytorchのTransformersを開発しているHuggingFaceが、TransformersにWhisperのFine Tuning機能を追加しました。

Fine Tuningの方法は下記のBLOGに記載されています。音声とテキストのペアを与えることで、Whisperに新しい単語を覚えさせることができます。

WhisperはUTF8のバイトコードを直接、予測するため、新しい単語を追加しても、vocabの更新は不要です。

Fine-Tune Whisper For Multilingual ASR with 😊 Transformers

In this blog, we present a step-by-step guide on fine-tuning Whisper for any multilingual ASR dataset using Hugging...

huggingface.co

Whisperにおける日本語のFineTuning

日本語でFine Tuningする場合、日本語はスペース区切りされていないので、compute_metrics関数が正しく単語誤り率（WER）を計算できません。そこで、ginzaによる形態素解析を使用してcompute_metrics関数を改良し、単語で区切る必要があります。この方法は、以下のBLOGで解説されています。

Hugging FaceでOpenAIの音声認識"Whisper"をFine Tuningする方法が公開されました | DevelopersIO

こんちには。データアナリティクス事業本部 機械学習チームの中村です。OpenAIがリリースしたWhisperについて、先日Hugging...

dev.classmethod.jp

データセットの構築

独自のデータセットを使用するには、TransformersのDatasetDictとDatasetを使用します。fileListに音声ファイルのパス、sentenceListに学習するテキストを用意すると、学習用のデータセットを構築可能です。

```
from datasets import DatasetDict, Dataset
common_voice = DatasetDict()
fileList = ["test/tsukuyomi1.wav", "test/tsukuyomi2.wav"]
```

```
sentenceList = ["こんにちは。今日は新しいAIエンジンであるailia SDKを紹介します。ailia は日本語を理解することができる音声認識エンジンです。"]  
common_voice["train"] = Dataset.from_dict({"audio": fileList, "sentence": sentenceList})
```

学習

学習はHuggingFaceのBLOGに沿って行い、Seq2SeqTrainingArgumentsとSeq2SeqTrainerを使用して、データセットを設定し、trainer.train()を実行します。

```
trainer.train()
```

音声ファイルが5つで、40 Epochs、Whisper Small、M1 MacのCPU実行だと、20分程度で学習が完了します。学習が完了すると、whisper-small-jp/checkpoint-40に結果が出力されます。

推論

学習したモデルを使用して推論するには下記のように、WhisperForConditionalGeneration.from_pretrainedでモデルを読み込み、推論します。

```
import torch  
from transformers import AutoProcessor, WhisperForConditionalGeneration  
from datasets import load_dataset, DatasetDict, Dataset  
from datasets import Audio  
  
processor = AutoProcessor.from_pretrained("openai/whisper-small")  
  
# original  
#model = WhisperForConditionalGeneration.from_pretrained("openai/whisper-small")  
  
# fine tuned  
model = WhisperForConditionalGeneration.from_pretrained("whisper-small-jp/checkpoint-40")  
  
model.config.forced_decoder_ids \  
    = processor.get_decoder_prompt_ids(language = "ja", task = "transcribe")  
model.config.suppress_tokens = []  
  
common_voice = DatasetDict()  
fileList = ["test/ailia.wav"]  
common_voice["train"] = Dataset.from_dict({"audio": fileList}).cast_column("audio", Audio)
```

```

for i in range(len(common_voice["train"])):
    inputs = processor(common_voice["train"][i]["audio"]["array"], return_tensors="pt")
    input_features = inputs.input_features

    generated_ids = model.generate(inputs=input_features)

    transcription = processor.batch_decode(generated_ids, skip_special_tokens=True)

    print(transcription)

```

推論コードの例は、HuggingFaceの公式ドキュメントにあります。

Whisper

The Whisper model was proposed in Robust Speech Recognition via Large-Scale Weak Supervision by Alec Radford, Jong Wook...

huggingface.co

公式のWhisperのWeightへの変換

convert_openai_whisper_to_tfmsで公式のWhisperから、HuggingFaceのWhisperへの変換を行なっているので、この逆操作を行います。この処理で、HuggingFaceで学習したモデルを、通常のWhisperで推論が可能になります。

transformers/convert_openai_to_hf.py at main ·

[huggingface/transformers](https://github.com/huggingface/transformers)

🤗 Transformers: State-of-the-art Machine Learning for Pytorch, TensorFlow, and JAX. ...

[github.com](https://github.com/huggingface/transformers)

```

from transformers import AutoProcessor, WhisperForConditionalGeneration
finetune_model = WhisperForConditionalGeneration.from_pretrained("whisper-small")

WHISPER_MAPPING = {
    "encoder.ln_post.weight": "encoder.layer_norm.weight", # added by ax
    "encoder.ln_post.bias": "encoder.layer_norm.bias", # added by ax
    "blocks": "layers",
    "mlp.0": "fc1",
    "mlp.2": "fc2",
}

```

```

"mlp_ln": "final_layer_norm",
".attn.query": ".self_attn.q_proj",
".attn.key": ".self_attn.k_proj",
".attn.value": ".self_attn.v_proj",
".attn_ln": ".self_attn_layer_norm",
".attn.out": ".self_attn.out_proj",
".cross_attn.query": ".encoder_attn.q_proj",
".cross_attn.key": ".encoder_attn.k_proj",
".cross_attn.value": ".encoder_attn.v_proj",
".cross_attn_ln": ".encoder_attn_layer_norm",
".cross_attn.out": ".encoder_attn.out_proj",
"decoder.ln.": "decoder.layer_norm.",
"encoder.ln.": "encoder.layer_norm.",
"token_embedding": "embed_tokens",
"encoder.positional_embedding": "encoder.embed_positions.weight",
"decoder.positional_embedding": "decoder.embed_positions.weight",
#"ln_post": "layer_norm", # disabled by ax
}

def rename_keys(s_dict):
    keys = list(s_dict.keys())
    for key in keys:
        new_key = key
        for v, k in WHISPER_MAPPING.items():
            if k in key:
                new_key = new_key.replace(k, v)

        print(f"{key} -> {new_key}")

    s_dict[new_key] = s_dict.pop(key)
    return s_dict

state_dict = finetune_model.model.state_dict()
rename_keys(state_dict)

import whisper
model = whisper.load_model("small")

missing, unexpected = model.load_state_dict(state_dict, strict = False)

if len(missing):
    print("Weight name not found", missing)
    raise

result = model.transcribe("test/axell_130.wav", language="ja", verbose=False)

for s in result["segments"]:
    start = s['start']
    end = s['end']
    text = s['text']
    print(str(start) + "\t" + str(end) + "\t" + text)

```

合成音声での学習

学習前の認識結果は下記となります。「ハードウェア」を「アードメア」、「ソフトウェア」を「ソフトメア」、「ailia SDK」を「Irea SDK」、「コア技術」を「小和技術」と誤認識しています。

0.0 6.0 自動運転の未来を加速させるアクセルの取り組み
6.0 12.0 AIをより早く、より使いやすく、すぐに始められる
12.0 17.0 豊かな未来を想像する企業の良き理解者になる
17.0 23.0 AI実装に必要な学習から推論までトータルソリューションを提供できる
23.0 29.0 アクセルはテクノロジーで未来を加速させる
29.0 36.0 アクセルでは、アードメア、ソフトメア、要素技術の3つの開発力を基づいて、
36.0 40.0 5つの事業領域に展開をしています
40.0 46.0 現在、目覚ましい発展を遂げているAIや日々重要性が増している
46.0 49.0 情報セキュリティなど、様々な分野で
49.0 54.0 アードメアからソフトウェアまで一貫したソリューションを提供しています
54.0 61.0 特にAI及び自動運転領域では、Irea SDKを小和技術として
61.0 66.0 モデルやアプリケーションなどの快適な開発環境の構築と
66.0 71.0 それを動かす実行基盤としてのアードメアの開発を進めることで
71.0 77.0 理想的なAI開発環境の構築を目指しています
77.0 83.0 AIやSDKはあらゆるアードメアで動き

そこで、下記の辞書を使用して再学習しました。辞書においては左に音声合成用のテキスト、右に音声認識用のテキストを配置しています。音声合成用と音声認識用のテキストが同じ場合は、一つのテキストになります。

アイリアエスディーケー , ailia SDK
ハードウェア , ハードウェア
ソフトウェア , ソフトウェア
目覚ましい発展
コア技術

FineTuningにおいては、辞書からTacotron2を使用して合成した音声を使用して学習を行いました。音声合成にはax株式会社でつくよみちゃんコーパスで学習した日本語モデルを使用しています。

学習に使用した環境は下記です。

```
macOS 12.6 (M1 Pro Max)
Python 3.9
transformers == 4.29.0.dev0
whisper == 1.0
torch == 2.0.0
```

学習後の認識結果は下記となります。Epoch10から、Epoch40に増加するについて、新しい単語が認識できるようになっていきます。

Epochs10

0.0 7.0 自動運転の未来を加速させる アクセルの取り組み
7.0 12.0 AIをより早くより使いやすく すぐに始められる
12.0 17.0 豊かな未来を想像する企業の よき理解者になる
17.0 21.0 AI実装に必要な学習から 推論まで
21.0 24.0 トータルソリューションを提供できる
24.0 29.0 アクセルはテクノロジーで 未来を加速させる
29.0 36.0 アクセルではアードウェア、ソフトウェア 要素技術の3つの開発力を基づ
36.0 40.0 いちつの事業領域に展開をしています
40.0 44.0 現在 メザマシー発展を遂げているAIや
44.0 47.0 日々重要性が増している情報セキュリティなど
47.0 51.0 様々な分野でアードウェアからソフトウェアまで
51.0 54.0 一貫したソリューションを提供しています
54.0 59.0 特にAI及び自動運転領域では
59.0 62.0 Airiya SDKをコア技術として
62.0 66.0 モデルやアプリケーションなどの 快適な開発環境の構築と
66.0 71.0 それを動かす実行基盤としての アードウェアの開発を進めることで
71.0 77.0 理想的なAI開発環境の構築を 目指しています
77.0 84.0 AIリア SDKはあらゆるアードウェアで動き

Epochs40

0.0 7.0 自動運転の未来を加速させる アクセルの取り組み
7.0 12.0 AIをより早くより使いやすく すぐに始められる
12.0 17.0 豊かな未来を想像する企業の よきり会社になる
17.0 23.0 AI実装に必要な学習から 推論までトータルソリューションを提供できる
23.0 29.0 アクセルはテクノロジーで未来を加速させる
29.0 36.0 アクセルではアードウェア ソフトウェア 要素技術の3つの開発力を基づ
36.0 40.0 いちつの事業領域に展開をしています
40.0 44.0 現在 レザマシー発展を遂げているAIや
44.0 47.0 日々重要性がましている情報セキュリティなど

47.0 51.0 様々な部屋でアードウェアからソフトウェアまで
51.0 54.0 一貫したソリューションを提供しています
54.0 58.0 特にAI及び自動運転領域では
58.0 61.0 ailia SDKをコア技術として
61.0 66.0 モデルやアプリケーションなどの 快適な開発環境の構築と
66.0 71.0 それを動かす実行基盤としての アードウェアの開発を進めることで
71.0 78.0 理想的なAI開発環境の構築を目指しています
78.0 82.0 ailia SDKはあらゆるハードウェアで動き

Epochs40では「ailia SDK」と「コア技術」という新しい単語を認識できるようになっています。また、タイムスタンプの情報も問題なく取得できています。

しかし、「5つの事業領域」を「いつつの事業領域」、「目覚ましい発展」を「レザマシー発展」、「開発力に基づいて」を「開発力に基づい」と、元々、正しく認識できていた箇所にエラーが見られます。

新しい知識のみでFine Tuningしているため、過去の知識を一部、忘却しているようです。

エンコーダのFreeze

全てのWeightを再学習するのではなく、エンコーダをFreezeし、デコーダのみを再学習します。Fine Tuningで使用できるAPIは下記のコードに記載されています。

**transformers/src/transformers/models/whisper at main ·
huggingface/transformers**

You can't perform that action at this time. You signed in with another tab or window. You signed out in another tab or...

[github.com](https://github.com/huggingface/transformers)

今回は、WhisperForConditionalGenerationで定義されている、freeze_encoder APIを呼び出しました。

```
model.freeze_encoder()
```

Epochs10

0.0 6.0 自動運転の未来を加速させるアクセルの取り組み
6.0 12.0 AIをより早く、より使いやすく、すぐに始められる
12.0 17.0 豊かな未来を想像する企業の良き理解者になる
17.0 23.0 AI実装に必要な学習から推論までトータルソリューションを提供できる
23.0 29.0 アクセルはテクノロジーで未来を加速させる
29.0 36.0 アクセルでは、ハードウェア、ソフトウェア、要素技術の3つの開発力を基づいて
36.0 40.0 5つの事業領域に展開をしています
40.0 44.0 現在、目覚ましい発展を遂げているAIや
44.0 47.0 日々重要性が増している情報セキュリティなど
47.0 51.0 様々な分野で、ハードウェアからソフトウェアまで
51.0 54.0 一貫したソリューションを提供しています
54.0 59.0 特にAI及び自動運転領域では
59.0 62.0 Airiya SDKをコア技術として
62.0 67.0 モデルやアプリケーションなどの快適な開発環境の構築と
67.0 72.0 それを動かす実行基盤としてのハードウェアの開発を進めることで
72.0 77.0 理想的なAI開発環境の構築を目指しています
77.0 83.0 Airiya SDKはあらゆるハードウェアで動き

Epochs40

0.0 7.0 自動運転の未来を加速させる アクセルの取り組み
7.0 12.0 AIをより早くより使いやすく すぐに始められる
12.0 17.0 豊かな未来を想像する企業の 良き理解者になる
17.0 24.0 AI実装に必要な学習から 推論までトータルソリューションを提供できる
24.0 32.0 アクセルはテクノロジーで未来を加速させる アクセルではハードウェア
32.0 40.0 ソフトウェア 要素技術の3つの開発力を基づい 5つの事業領域に展開をしています
40.0 44.0 現在 レザマシー発展を遂げているAIや
44.0 49.0 日々重要性がましている情報セキュリティなど 様々な分野で
49.0 54.0 ハードウェアからソフトウェアまで 一貫したソリューションを提供しています
54.0 58.0 特にAI及び自動運転領域では
58.0 63.0 ailia SDKをコア技術として モデルやアプリケーションなどの
63.0 66.0 快適な開発環境の構築と
66.0 71.0 それを動かす実行基盤としての ハードウェアの開発を進めることで
71.0 78.0 理想的なAI開発環境の構築を目指しています
78.0 82.0 ailia SDKはあらゆるハードウェアで動き

かなり良くなりました。新しい単語は全て学習できました。ただ、「目覚ましい発展」->「レザマシー発展」や、「開発力に基づいて」->「開発力を基づい」など、一部、精度低下があります。

これについては、元々、Smallモデルでは間違える単語の確信度が微妙で、元々の推論の精度が不足しているのではないかと考えています。Tinyモデルで同じ音声フ

イルを推論した場合に、「レザマシー発展」や「開発力を求め」など、同様の箇所で間違いをするため、Whisperにとって難易度の高い箇所なのではないかと考えています。そのため、Fine Tuningによる若干の精度低下が、出力エラーに繋がっているのではないかと考えています。

Tinyモデルの出力

0.0 7.0 自動運転の未来を重くさせる アクセルの取り組み
7.0 12.0 AIをより早くより使いやすく 常に始められる
12.0 17.0 ユタカな未来を想像する気によるのを よき理解さになる
17.0 24.0 AI実装に必要な学習からスイロンマリ ドーバルソリューションの提供できる
24.0 29.0 アクセルはテクノロジーで未来を重くさせる
29.0 36.0 アクセルではアドメーや ソップトメーや 両速実の3つの開発力を求め
36.0 40.0 実の事業量意気に展開をしています
40.0 47.0 現在、レザマシー発展を止めているAIや ギリ重要性がなしている情報セキュリティなど
47.0 54.0 様々な分野でアドメーからソフトエアまで 一感した措利を進んで 影響しています
54.0 62.0 特にAI及び自動運転量意気ではAIエスディー系を 割実として
62.0 67.0 モデルやプリケーションなどの 入的な開発環境の構築拡させ
67.0 71.0 それを動かす実行期監としてのハードビアの開発を進めることで
71.0 78.0 実装的なAI開発環境の構築を目指しています
78.0 85.0 AIでなエスディー系は アラユルハードビアでいもち

この問題の解決には、より精度の良いモデルで学習を行なった方が良さそうです。精度の高いモデルでは、他の単語との確信度の距離が十分に取れていることが期待されるため、Fine Tuningによる若干の精度低下の影響があったとしても、他の単語に転ぶケースが減少するものと考えています。

Mediumモデルでの学習

今までではSmallモデルで学習を行なっていましたが、Mediumモデルで再学習を行います。

Fine Tuning前では、「ailia SDK」を「AiRia SDK」と誤認識しています。

Fine Tuning前

0.0 4.0 自動運転の未来を加速させる
4.0 6.0 アクセルの取り組み
7.0 12.0 AIをより早く、より使いやすく、すぐに始められる
12.0 17.0 豊かな未来を創造する企業の良き理解者になる
17.0 21.0 AI実装に必要な学習から推論まで

21.0 24.0 トータルソリューションを提供できる
24.0 28.0 アクセルはテクノロジーで未来を加速させる
28.0 32.0 アクセルではハードウェア、ソフトウェア、
32.0 36.0 要素技術の3つの開発力をもとに
36.0 39.0 5つの事業領域に展開をしています
39.0 43.0 現在目覚ましい発展を遂げているAIや
43.0 47.0 日々重要性が増している情報セキュリティなど
47.0 51.0 様々な分野でハードウェアからソフトウェアまで
51.0 54.0 一貫したソリューションを提供しています
54.0 58.0 特にAI及び自動運転領域では
58.0 61.0 Aria SDKをコア技術として
61.0 63.0 モデルやアプリケーションなどの
63.0 66.0 快適な開発環境の構築と
66.0 70.0 それを動かす実行基盤としてのハードウェアの開発を
70.0 74.0 進めることで理想的なAI開発環境の
74.0 76.0 構築を目指しています
76.0 82.0 Aria SDKはあらゆるハードウェアで動き

Fine Tuning後では、「ailia SDK」も正しく認識できました。

FIne Tuning後 (Epochs 10)

0.0 4.0 自動運転の未来を加速させる
4.0 6.0 アクセルの取り組み
7.0 12.0 AIをより早く、より使いやすく、すぐに始められる
12.0 17.0 豊かな未来を創造する企業の良き理解者になる
17.0 21.0 AI実装に必要な学習から推論まで
21.0 24.0 トータルソリューションを提供できる
24.0 28.0 アクセルはテクノロジーで未来を加速させる
28.0 32.0 アクセルではハードウェア、ソフトウェア、
32.0 36.0 要素技術の3つの開発力をもとに
36.0 39.0 5つの事業領域に展開をしています
39.0 43.0 現在目覚ましい発展を遂げているAIや
43.0 47.0 日々重要性が増している情報セキュリティなど
47.0 51.0 様々な分野でハードウェアからソフトウェアまで
51.0 54.0 一貫したソリューションを提供しています
54.0 58.0 特にAI及び自動運転領域では
58.0 61.0 ailia SDKをコア技術として
61.0 63.0 モデルやアプリケーションなどの
63.0 66.0 快適な開発環境の構築と
66.0 70.0 それを動かす実行基盤としてのハードウェアの開発を
70.0 74.0 進めることで理想的なAI開発環境の
74.0 76.0 構築を目指しています
76.0 82.0 ailia SDKはあらゆるハードウェアで動き

医療用語への応用

次に、大規模な辞書への適用可能性を評価するため、10123種類の医療用語でMediumモデルを追加学習させました。通常のMediumモデルでは、「痙攣」->「経連」、「動悸」->「動悸」、「心電図」->「心レンズ」という誤認識があります。

Fine Tuning前

0.0 5.0 こんにちは、先生。最近手足の経連があり、動機も感じます。
5.0 9.0 こんにちは。手足の経連と動機がある場合、心臓の問題が考えられます。
9.0 13.0 経連や動機の発作がいつ頻繁に起こるか教えていただけますか?
13.0 16.0 経連はほとんど毎日、動機はたまにです。
16.0 22.0 経連と動機は日常的に起こる場合は、心臓や血液循環経営などの問題がある可能性があります。
22.0 25.0 一度、心臓の検査を受ける必要があります。
25.0 27.0 心臓の検査を受ける必要があるのですか?
27.0 29.0 どのような検査が必要ですか?
29.0 34.0 はい、検査が必要です。まずは、心臓の電気活動を測定するために心レンズを撮影します。
34.0 39.0 また、動脈効果の状態を調べるために、エコー検査やMRIを行うことがあります。
39.0 42.0 そうですか、わかりました。心臓の検査を予約しましょうか?
42.0 49.0 はい、お予約ください。また、心臓病が疑われる場合は、内科医や心臓専門医に紹介することになります。
49.0 59.0 ありがとうございます、先生。どういたしまして、心配なことがあれば、いつでも連絡してください。

合成音声でFine Tuningすることで、「痙攣」と「心電図」が正しく認識可能になりました。「動悸」は読みが一般的なものであり、単語だけでの学習は不十分な可能性があり、文脈を加味したデータセットを加えた方がよさそうです。

Fine Tuning後（Epochs20）

0.0 5.0 こんにちは、先生、最近手足の痙攣があり、動氣も感じます。
5.0 9.0 こんにちは。手足の痙攣と動氣がある場合、心臓の問題が考えられます。
9.0 13.0 痙攣や動氣の発作がいつ頻繁に起こるか教えていただけますか?
13.0 16.0 痙攣はほとんど毎日、動氣はたまにです。
16.0 22.0 痙攣と動氣が日常的に起こる場合は、心臓や血液循環痙攣などの問題がある可能性があります。
22.0 25.0 一度、心臓の検査を受ける必要があります。
25.0 27.0 心臓の検査を受ける必要があるのですか?
27.0 29.0 どのような検査が必要ですか?
29.0 30.0 はい。検査が必要です。
30.0 34.0 まずは、心臓の電気活動を測定するために心電図を撮影します。
34.0 39.0 また、動脈硬化の状態を調べるために、エコー検査やMRIを行うことがあります。
39.0 42.0 そうですか?分かりました。心臓の検査を予約しましょうか?

2023/05/23 12:01

WhisperをFineTuningして専門用語を認識可能にする。Whisperを少量のデータセットでFine... | by Kazuki Kyakuno | axinc | ...

- | | | |
|------|------|---------------------------------------|
| 42.0 | 49.0 | はい。お予約ください。また、心臓病が疑われる場合は、内科医や心臓専門医に紹 |
| 49.0 | 54.0 | ありがとうございます。先生。どういたしまして、心配なことがあれば、いつでも |

「肺臍腫」->「肺腑子」という誤認識も、Fine Tuningすることで解消します。

Fine Tuning前

0.0 7.0 こんにちは。それは心配ですね。その症状は肺腑子の可能性があります。あなたは

◀ ▶

Fine Tuning後 (Epochs20)

0.0 7.0 こんにちは。それは心配ですね。その症状は肺臍腫の可能性があります。あなたは

◀ ▶

まとめ

本BLOGでは、Tacotron2の合成音声を使用して、Hugging FaceのTransformersを使用したWhisperの再学習を行いました。その結果、Whisperが新しい単語を獲得できることを確認しました。

Whisperは、UTF8のバイトコードをバイグラムにして直接、推論するアーキテクチャとなっています。そのため、知らない単語は、バイトコードの組み合わせを知らないために出力できないものと考えられます。そこで、Fine Tuningして、バイトコードの組み合わせさえ教えられれば、音素までは内部構造で獲得できていることが期待されるので、比較的、少数のサンプルでも専門用語を獲得できるようになっていると考えられます。

ただし、Smallモデルの場合、Fine Tuningで、従来検出できていた単語が検出できなくなるケースがありました。これは、確信度が低い状態で正解していた単語がFine Tuningの誤差により誤検出へと変化したものと考えています。そのため、Fine TuningではMediumモデルを使用した方が良いと考えています。

また、10123種類の医療用語でFine Tuningすることで、大規模な辞書に対してもFine Tuningは成立し、一定のスケーラビリティがあることを確認しました。

ailia AI Speechの紹介

ax株式会社では、エッジ向けAI推論エンジンのailia SDKを使用して、Whisperをオフラインで高速に実行可能にするAI音声認識ライブラリ、[ailia AI Speech](#)を開発しています。今後、WhisperのFine Tuningによって、業界ごとに特化した辞書を持つ音声認識モデルを提供していくことを計画しています。

ax株式会社はAIを実用化する会社として、クロスプラットフォームでGPUを使用した高速な推論を行うことができるailia SDKを開発しています。ax株式会社ではコンサルティングからモデル作成、SDKの提供、AIを利用したアプリ・システム開発、サポートまで、AIに関するトータルソリューションを提供していますのでお気軽に[お問い合わせください](#)。

Ailia Technology



Written by Kazuki Kyakuno

78 Followers · Editor for axinc

CTO — ax Inc., Managing Director — AXELL

More from Kazuki Kyakuno and axinc

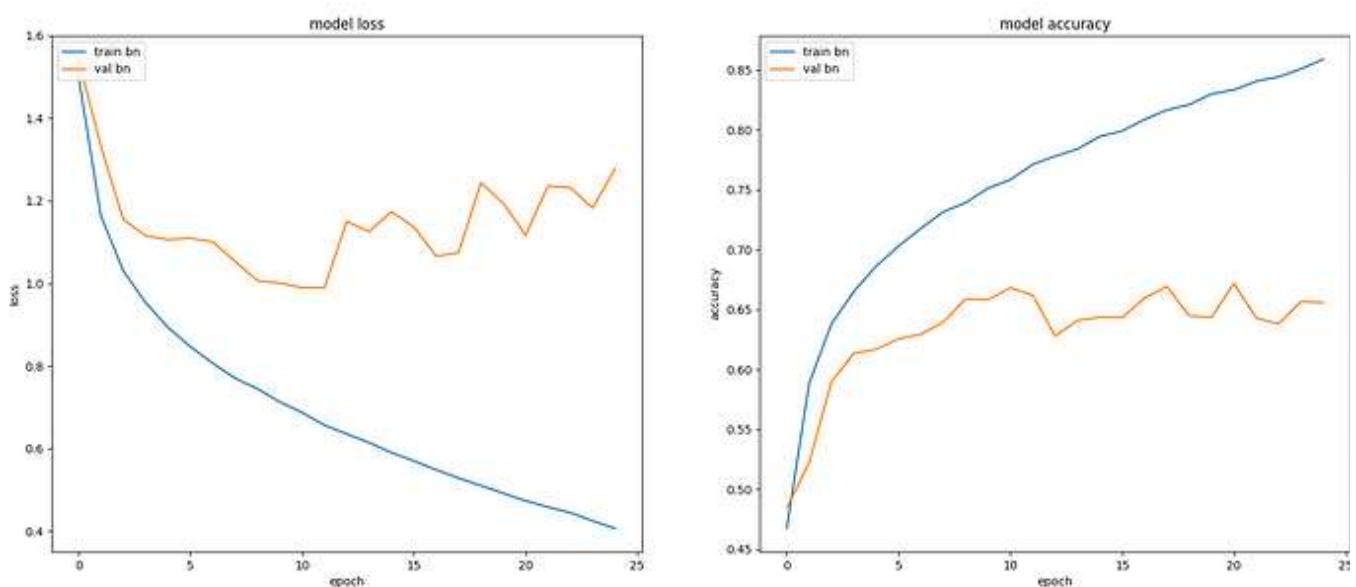


Kazuki Kyakuno in axinc

StableDiffusionWebUIとKohya's GUIで生成したLoRAを使って特定のキャラクターの画像を生成する

PC上で簡単にイラストを生成できるStableDiffusionWebUIと、特定のキャラクターに特化したLoRAをKohya'sGUIで生成して、特定のキャラクターの画像を生成します。

19 min read · Apr 19



Kazuki Kyakuno in axinc

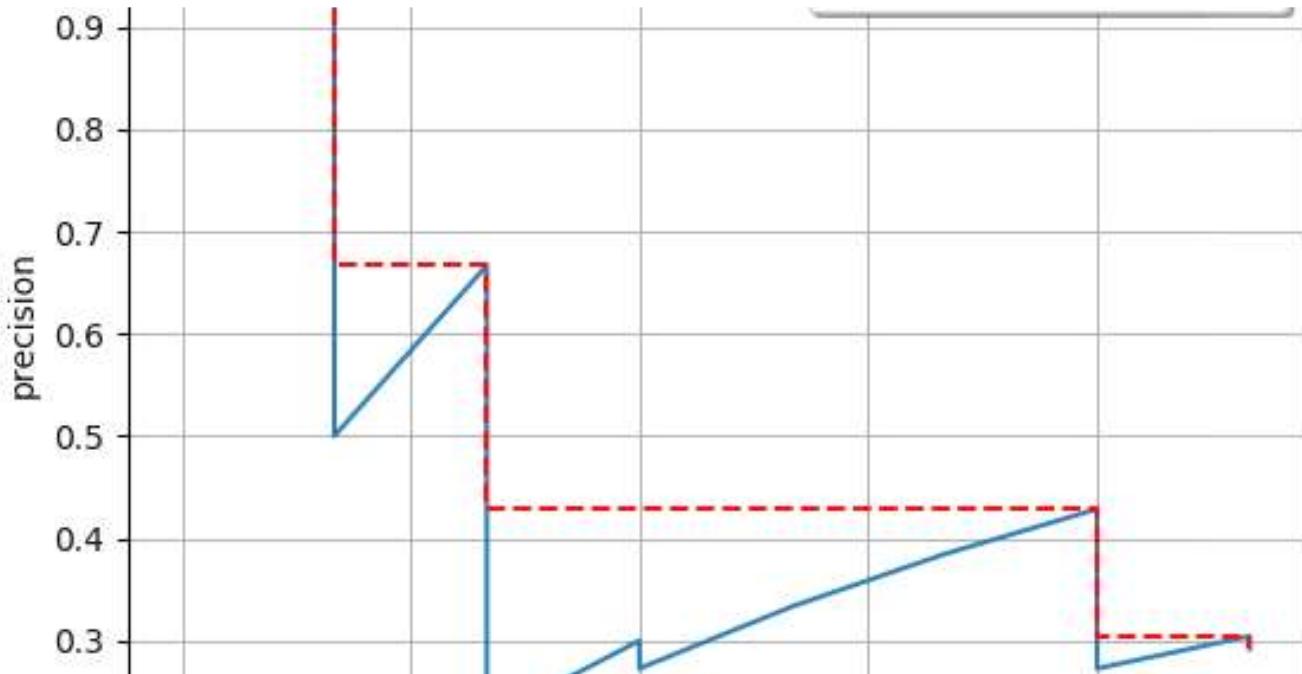
Dropoutによる過学習の抑制

Kerasを使用して機械学習を行い、実アプリケーションに適用する場合、いかに過学習を抑制するかが重要になります。本記事では、Dropoutによる過学習の抑制の効果を検討します。

10 min read · May 13, 2020



53



Kazuki Kyakuno in axinc

mAP: 物体検出モデルの評価指標

物体検出モデルの評価指標であるmAPを解説します。

11 min read · Jun 23, 2021



1



		BoolQ	PIQA	SIQA	HellaSwag	WinoGrande	ARC-e	ARC-c	OBQA
GPT-3	175B	60.5	81.0	-	78.9	70.2	68.8	51.4	57.6
Gopher	280B	79.3	81.8	50.6	79.2	70.1	-	-	-
Chinchilla	70B	83.7	81.8	51.3	80.8	74.9	-	-	-
PaLM	62B	84.8	80.5	-	79.7	77.0	75.2	52.5	50.4
PaLM-cont	62B	83.9	81.4	-	80.6	77.0	-	-	-
PaLM	540B	88.0	82.3	-	83.4	81.1	76.6	53.0	53.4
LLaMA	7B	76.5	79.8	48.9	76.1	70.1	72.8	47.6	57.2
	13B	78.1	80.1	50.4	79.2	73.0	74.8	52.7	56.4
	33B	83.1	82.3	50.4	82.8	76.0	80.0	57.8	58.6
	65B	85.3	82.8	52.3	84.2	77.0	78.9	56.0	60.2

Table 3: Zero-shot performance on Common Sense Reasoning tasks.

ax Kazuki Kyakuno in axinc

Llama.cppとLoRAを使用してPC上で日本語LLMモデルを実行する

PC上でLLMモデルを実行できるllama.cppと、LLMモデルをFineTuningするLoRAを使って、日本語でのLLM推論を行う方法を解説します。

7 min read · Apr 11



See all from Kazuki Kyakuno

See all from axinc

Recommended from Medium



 Salvatore Raieli in Towards Data Science

Emergent Abilities in AI: Are We Chasing a Myth?

Changing Perspective on Large Language Models emerging properties

◆ · 11 min read · 6 days ago

 620  9





 Rui Alves in The Generator

Snapchat Influencer Uses GPT-4 to Become a High-End AI Girlfriend

This is how she made \$71,610 in a single month

◆ · 4 min read · May 13

2.6K

30

Lists



Staff Picks

307 stories · 75 saves



Stories to Help You Level-Up at Work

19 stories · 33 saves



Self-Improvement 101

20 stories · 76 saves



Productivity 101

20 stories · 67 saves



Daniel Rizea in Entrepreneur's Handbook

What I've Learned After Holding One Thousand Interviews

How hiring works in startups and big companies

◆ · 9 min read · 5 days ago

1K

6



Martin Heinz in Better Programming

Real Multithreading is Coming to Python—Learn How You Can Use It Now

True multi-core concurrency is coming to Python in 3.12 release and here's how you can use it right now using sub-interpreter API

• 6 min read • May 14

654

8



Patrick Metzger in Crow's Feet

What I've Learned About Job Hunting Over 50

Relationships are everything



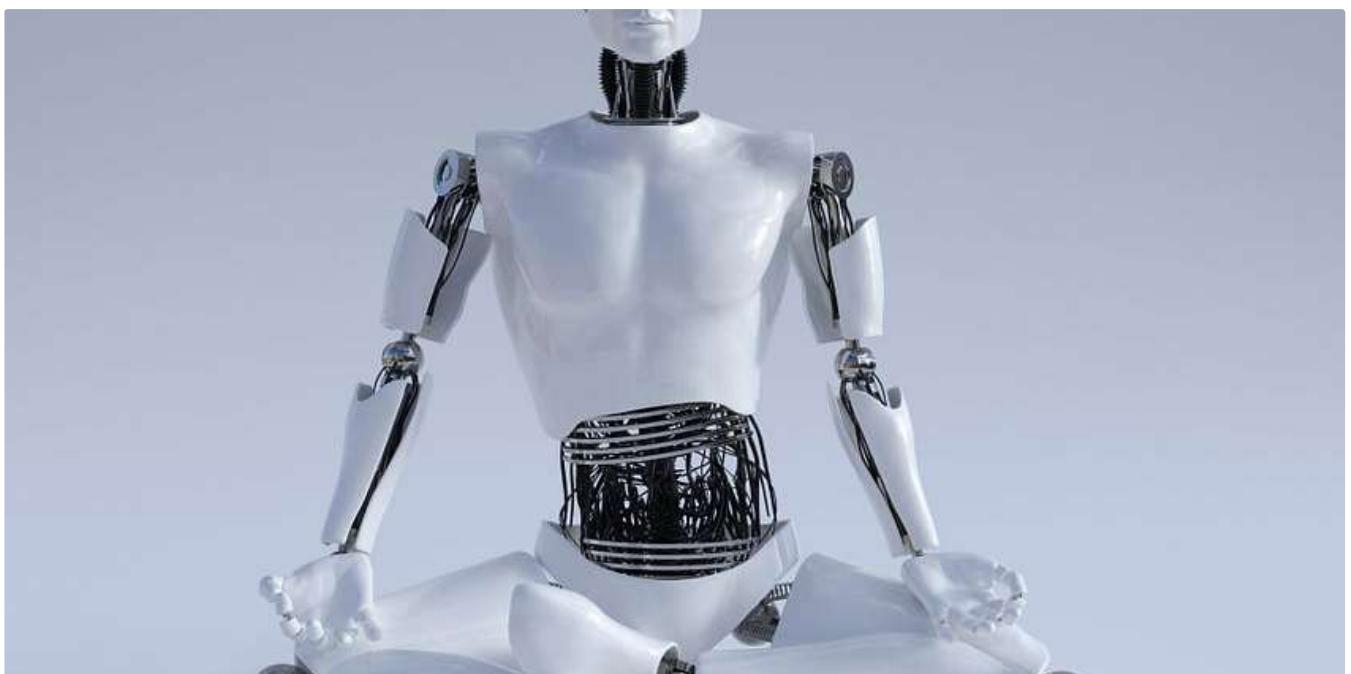
· 6 min read · 4 days ago



2.9K



56



The PyCoach in Artificial Corner

You're Using ChatGPT Wrong! Here's How to Be Ahead of 99% of ChatGPT Users

Master ChatGPT by learning prompt engineering.

◆ · 7 min read · Mar 18

👏 19.97K

🗨 352



See more recommendations