

Prompt

与传统的 ASR 模型不同，Whisper 在某些情况下可以借助 Prompt 提升音频转文字的准确度。

虽然同属于 OpenAI，但 Whisper 和 chatGPT 的 Prompt 差别非常大。

如果 Whisper 的 Prompt 是 "请尽力保持每句话的完整 并给每句话都添加标点符号"，你会发现这句 Prompt 没有任何作用，Whisper 的转录结果里可能还是会出现大量中断的句子，或者转录结果无标点。

这是因为 Whisper 是在尝试学习 Prompt 里的「风格(style)」，而非具体的指令。

那什么是 Prompt 的「风格」？

根据 Whisper 的 [cookbook](#)、[Whisper prompt 的 issues](#) 和我的实际体验，Prompt 的风格可以表现在下面几个方面：

风格 1：Prompt 里是否包含标点符号。

Whisper 的转录结果有时会缺乏标点符号，这个时候你可以使用带标点符号的 Prompt 的来引导 Whisper 添加标点。

1. 「生于忧患，死于安乐。岂不快哉？」
2. 「请为转写的文本添加标点符号」

Prompt 1 是有意义的，Prompt 2 是无意义的。

不过 Whisper 也不是能每一次都学会 Prompt 的风格添加标点符号。实测在 Whisper-large v2 模型下，通常英文内容使用 prompt 可以为结果添加标点，中文内容则要看运气。

风格2：一些容易拼写错误的人名、产品名、公司名

这里比较好理解，不同的词汇可能会有同一个发音。你可以在 Prompt 里指定你希望的拼写格式。

比如一段对话里提到了张三和李四，而「张三」的音同「章三」、「樟叁」...

那 Prompt 里就可以包含「张三，李四。」

风格3：是否包含一些填充词（filler words）

这里直接引用官方的说法来解释，翻译成中文会有点怪异。

The model may also leave out common filler words in the audio. If you want to keep the filler words in your transcript, you can use a prompt that contains them: "Umm, let me think like, hmm... Okay, here's what I'm, like, thinking."

该模型还可能遗漏音频中常见的填充词。如果您想在成绩单中保留填充词，您可以使用包含它们的提示：“嗯，让我想想，嗯.....好吧，这就是我的想法。”

风格4：简体中文和繁体中文

因为中文有两种，有的时候 Whisper 会把音频以繁体中文的形式转录。此时用简体中文来陈述 Prompt 大概率可以解决此问题。

一般建议 Whisper 的 Prompt 语言和转录的音频语言一致。即英文音频用英语 Prompt，中文音频用简中/繁中 Prompt。

风格5：通过对话风格的 Prompt 变相实现转录结果区分说话人

这个技巧来自于 [prompt vs prefix in DecodingOptions](#)，有的时候会有效果。

单纯的 Whisper 模型并不能在转录结果里区分出说话人，想要区分说话人，一般会配合 Pyannote 来做。但在某些场景下，我们可以通过 Prompt 来变相实现区分说话人——**每一句的转录结果都属于单个说话人**，不会出现某一句的转录结果属于多个说话人的情况。

方法就是用一个对话式的 Prompt：

- 「- How are you? - I'm fine, thank you.」
- 「- 吃了没? - 吃了。」

播客这种场景可以试试。

Whisper Prompt 的其他 Tips

- Whisper 的 Prompt 应该尽可能长一些，太短的 prompt 的风格难以学习。
- Whisper 的 Prompt 只会使用 224 个 token，如果超过 224 个 token 则会默认使用最后 224 个 token。token 的概念可以参考[这篇文章](#)，但 Memo 的 Prompt 直接限制了字符串长度，这里就不用在乎 token 长度了，只需要尽量让自己的 prompt 完整输入到 Memo 输入框里就行。
- Whisper 的 Prompt 最好可以和音频的源语言保持一致。

Note

摘自 [少数派](#)

hotwords

可以使用 hotwords 来给定一些特定的指示内容，比如专业术语、新兴词汇等。

比如：**comfyUI** 是一个新词，它是功能最强大，模块化稳定的 diffusion GUI 和后端。

测试视频：<https://www.youtube.com/watch?v=Ybu6qTbEsew>

没有热词的情况下，默认参数调用 faster-whisper:

```
segments, info = model.transcribe(input_file, beam_size=5, language="en", vad_filter=False, vad_parameters = dict(min_silence_duration_ms=1000))
```

转写结果:

```
[261.76s -> 263.12s] The first thing you need to do is,
[263.12s -> 265.36s] of course, to copy the web address
[265.36s -> 266.12s] up here.
[266.12s -> 267.84s] Then you go into your Conf UI
[267.84s -> 270.04s] folder, again in the Conf UI
[270.04s -> 272.08s] folder, in there in the custom
[272.08s -> 274.08s] nodes folder and then up here
[274.08s -> 276.28s] in the address bar type CMD,
[276.28s -> 277.40s] hit enter.
[277.40s -> 279.40s] This opens up your command
[279.40s -> 281.24s] window. In here you type
[281.24s -> 283.36s] git clone and then
[283.36s -> 285.32s] put the web address and hit
[285.32s -> 287.36s] enter to clone the git
[287.36s -> 289.68s] project into your custom
[289.68s -> 290.56s] nodes folder.
[290.56s -> 291.60s] After you've done this, you're going
[291.60s -> 293.32s] to find in here the Conf UI
```

该词汇被错误识别为: Conf UI

添加热词:

```
segments, info = model.transcribe(input_file, hotwords="the video is about comfyUI", beam_size=5, language="en", vad_filter=False, vad_parameters = dict(min_silence_duration_ms=1000))
```

所添加的 hotwords 为:

```
1 hotwords = "the video is about comfyUI"
```

结果:

[261.76s -> 263.12s] The first thing you need to do is,
[263.12s -> 264.84s] of course, to copy the web
[264.84s -> 266.68s] address up here, then you go
[266.68s -> 268.48s] into your comfyUI folder,
[268.48s -> 270.80s] again in the comfyUI folder,
[270.80s -> 272.48s] in there in the custom nodes
[272.48s -> 274.28s] folder, and then up here in the
[274.28s -> 276.28s] address bar type cmd,
[276.28s -> 277.40s] hit enter.
[277.40s -> 279.40s] This opens up your command
[279.40s -> 281.20s] window. In here you type
[281.20s -> 283.08s] git clone and
[283.08s -> 285.00s] then put the web address and
[285.00s -> 286.92s] hit enter to clone
[286.92s -> 288.88s] the git project into
[288.88s -> 290.56s] your custom nodes folder.
[290.56s -> 291.48s] After you've done this, you're
[291.48s -> 293.32s] going to find in here the comfyUI

comfyUI 这个词被正确识别出来了。

Note

摘自: [github](#)