

# wota-translater v5.6.2

— 快速剪輯 御宅藝 用副歌檔案

(完整使用手冊 · Markdown 版, 已含目錄, 可直接另存為 PDF)

---

## 目 錄

1. 系統概要
  2. 環境需求
  3. 資料與檔案結構
  4. Excel 格式
  5. 安裝
  6. 執行
  7. 輸出內容
  8. 錯誤紀錄與防呆
  9. 剪輯邏輯與可調參數
  10. 打包 EXE
  11. 常見問題
  12. 版本紀錄
-

## 1. 系統概要

項目	說明
目的	依 <b>tracklist.xlsx</b> 將 <b>songs/</b> 內音檔批次裁剪成御宅藝用副歌
支援格式	MP3・M4A・WAV・FLAC
輸出	<b>output/</b> 320 kbps MP3 (自動清洗副標／非法字元)
特色	標準化音量、模糊比對、重複檢查、時間驗證、缺配對偵測、彩色進度條

---

## 2. 環境需求

Python  $\geq$  3.9

Python套件需要: colorama tqdm pydub pandas openpyxl

需安裝 FFmpeg 並加入 PATH (打包時可進階打包打包時可進階打包)

---

## 3. 資料與檔案結構

專案資料夾 /

```
|— songs/                # 原始音檔 (mp3, wav, m4a, flac)

|— output/              # 剪輯完成後的輸出

|— tracklist.xlsx       # Excel 曲目清單

|— wota-translater-v5.6.2.py

|— duplicate_titles.txt
```

|— duplicate\_matches.txt

|— invalid\_time.txt

|— processing\_errors.txt

|— unmatched\_titles.txt

|— unmatched\_audio.txt

(除songs/、tracklist.xlsx外 其餘檔案/資料夾執行後自動產生)

---

## 4. Excel 格式(tracklist.xlsx)

**A**(歌名)   **B** (開頭進入)   **C** (開頭退出)   **D** (第一段進入)   **E** (第一段退出)   **F**(第x段進入)   .....

歌名	mm:ss	mm:ss	mm:ss	mm:ss	mm:ss
----	-------	-------	-------	-------	-------

- 必填：
  - 歌名(A)
- 可選：
  - 開頭進入(B, 選填)、開頭退出(C)
  - 第一段進入(D)、第一段退出(E)
  - 第X段進入、第X段退出 (GHIJK.....可無限擴充, [X]支援中文或阿拉伯數字)

---

## 5. 安裝

1. 安裝 Python 與所需套件。

```
pip install colorama tqdm pydub pandas openpyxl
```

2. 安裝 ffmpeg 並加入 PATH。(在命令列輸入 `ffmpeg -version` 應有回應)
3. 建立 `songs/` 與 `tracklist.xlsx`。
4. 將 `songs/` 填充需要剪輯的音檔, 並將 `tracklist.xlsx` 編輯好格式

---

## 6. 執行

```
python wota-translater.py
```

- 綠字: 版本與功能簡介
- 黃色 `tqdm`: 整體進度條
- 黃字 `[n/N]` 歌名 `處理中...`: 當前歌曲
- 紅字: 即時錯誤
- 結尾: 綠(全成功)或紅(部分失敗)統計 + 一次性彈窗列出所有錯誤

---

## 7. 輸出內容

檔案 / 目錄

說明

`output/*.mp3`

副歌成品

`duplicate_titles.txt` 重複歌名

`duplicate_matches` 音檔被多歌名配對  
`.txt`

`invalid_time.txt` 時間格式 / 邏輯錯誤

`processing_errors` 音檔讀取 / 輸出異常  
`.txt`

`unmatched_titles.` 歌名找不到音檔  
`txt`

`unmatched_audio.t` songs/ 中未被使用之檔案  
`xt`

- 剪輯後的音檔會存放在 `output/`，格式為 **320 kbps MP3**。
  - 剪輯邏輯：
    - 只保留 [進入 ~ 退出] 區段。
    - 每段自動套用淡入/淡出。
    - **0:00** 例外：若首段 進入=0 → 不做淡入。
  - 最後進行 **Peak** 正規化 **-1 dBFS**。
-

## 8. 錯誤紀錄與防呆

- 模糊比對:若歌名與檔名不完全相符, 會自動嘗試比對。
  - 錯誤分類檔案:
    - `duplicate_titles.txt` → Excel 內歌名重複
    - `duplicate_matches.txt` → 同一音檔被多首歌使用
    - `invalid_time.txt` → 時間格式錯誤 / 區間重疊 / 退出早於進入
    - `processing_errors.txt` → 處理失敗(超過長度、匯出錯誤等)
    - `unmatched_titles.txt` → Excel 歌名未找到音檔
    - `unmatched_audio.txt` → 音檔未被配對
  - 輸出:
    - 錯誤即時顯示於終端機(紅字)。
    - 完成後一次性彈窗, 列出總結與錯誤清單。
-

## 9. 剪輯邏輯與可調參數

程式片段：

```
# ① 開頭片段 (intro)
```

```
intro_seg = a[:intro_len_ms].fade_out(2000)      # ← 這個 2000 =  
Intro 淡出長度
```

```
# ② 副歌段
```

```
seg = a[start_ms:end_ms]                        # 切出副歌
```

```
seg = seg.fade_in(min(2000, end_ms-start_ms))    # ← 2000 = 副歌淡入  
長度
```

```
# ③ 結尾淡出
```

```
if len(seg) > 1000:                             # 距離退出點 1 秒
```

```
    seg = seg[:-1000].fade_out(min(2000, len(seg))) # ← 2000 = 結尾  
淡出長度
```

```
# ④ Peak 正規化到 -1 dBFS (v5.5.0 新增)
```

```
gain = -1.0 - result.max_dBFS    # result 為 intro_seg + seg
```

```
result = result.apply_gain(gain)
```

# ⑤ 自訂輸入資料夾名稱

```
INPUT_FOLDER = "songs"
```

# ⑥ 自訂輸出資料夾名稱

```
OUTPUT_FOLDER = "output"
```

# ⑦ 自訂歌曲資料表格名稱

```
TRACKLIST = "tracklist.xlsx"
```

# ⑧ 搜尋歌名時的模糊比對嚴格程度(若希望更模糊則減少, 最嚴格為1)

```
FUZZY_TH = 0.4
```

參數	位置 / 說明	如何修改
單段淡入秒數	<code>fade_in(2000)</code>	2000 → 1500 (1.5 秒)...
單段淡出秒數	<code>fade_out(2000)</code>	同上
結尾前緩衝	<code>seg[: -1000]</code>	1000 ms → 自訂



TARGET_PEAK	TARGET_PEAK = -1.0 dBFS	-1.0 → -0.3...
自訂輸入資料夾名稱	INPUT_FOLDER = "songs"	songs → def
自訂輸出資料夾名稱	OUTPUT_FOLDER = "output"	output → def
自訂歌曲資料表格名稱	TRACKLIST = "tracklist.xlsx"	tracklist.xlsx → def.xlsx
模糊比對嚴格程度	FUZZY_TH = 0.4	0.4 → 0.2

---

## 10. 打包 EXE

若需打包為無命令列視窗的 EXE, 可使用 **pyinstaller**:

```
pyinstaller --noconsole --onefile --icon=icon.ico
wota-translater-v5.6.2.py
```

# 保留主控台則把 "--noconsole" 刪除

其中 **icon.ico** 可換成任意 **.ico** 圖示。

進階打包(打包成單一檔案, 不必額外安裝套件和 **FFmpeg**, 單exe啟動):

從 <https://www.gyan.dev/ffmpeg/builds/> 下載 full-release

從中抽取出 /bin/ffmpeg.exe 以及 /bin/ffprobe.exe 放在主py程式資料夾

接著py腳本內在 `from pydub import AudioSegment` 下方加入以下程式碼

```
if getattr(sys, 'frozen', False):  
    # EXE 模式下  
    base_dir = sys._MEIPASS  
else:  
    # Python 腳本模式  
    base_dir = os.path.dirname(os.path.abspath(__file__))  
  
AudioSegment.converter = os.path.join(base_dir, "ffmpeg.exe")  
AudioSegment.ffprobe = os.path.join(base_dir, "ffprobe.exe")  
  
pyinstaller的命令改成 pyinstaller --onefile --add-binary "ffmpeg.exe;."  
--add-binary "ffprobe.exe;." --icon=wota.ico  
wota-translater-v5.6.2.py
```

---

## 11. 常見問題

Q	A
找不到 <b>ffmpeg</b>	下載 FFmpeg 並加入 PATH
彩色亂碼	Windows cmd: <code>chcp 65001</code> ; 或改用 PowerShell / Windows Terminal
日文檔名被轉底線	Windows 禁符號; 可修改 <code>sanitize()</code> 正則

---

## 12. 版本紀錄

版本

重點

5.6.2 靜默化python套件安裝

5.6.1 正式化功能, 使其具有更加通用的剪輯功能

5.5.7 修正報錯提示、簡介輸出

5.5.0 新增 Peak 正規化 -1 dBFS

5.4.3 0:00 例外 & 單數字秒數

5.4.2 開頭時間防呆完善

5.3.x 彩色進度條、錯誤統整

≤5.2.x 基礎裁剪、重複檢查

---

若有進階需求(例如自訂淡入淡出長度、加入尾奏、歸一化音量), 歡迎隨時提出, 程式碼已預留易改動欄位

祝剪輯愉快!

遇到任何問題, 先查看結尾彈窗與對應 TXT, 仍無法解決再回報即可。

聯繫方式:Discord

NAME:endercup