

Off line 語音轉換器規劃及進度

Student : Sian-Yi Chen

Advisor : Tay-Jyi Lin and Chingwei Yeh

Outline

1. 介紹傳統個人語音轉換裝置

- 既有技術說明
- 架構圖

2. 新產品：離線語音轉換器

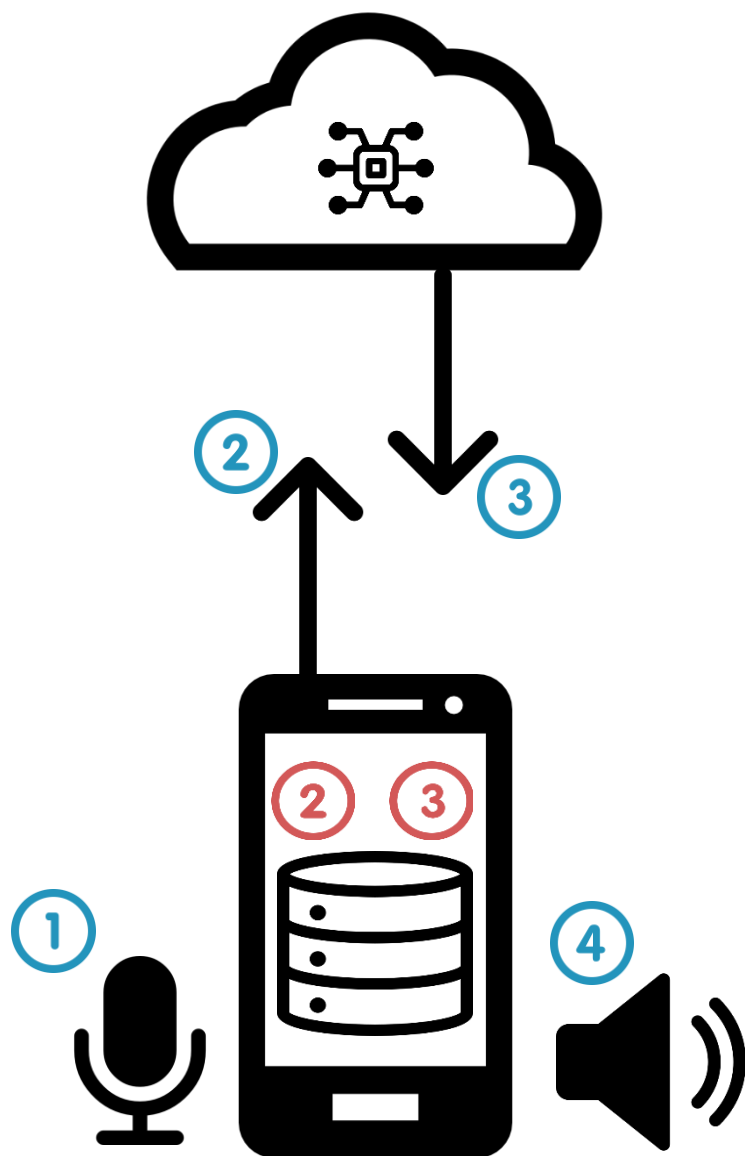
- 傳統裝置缺點改善
- 架構圖

3. 實作架構

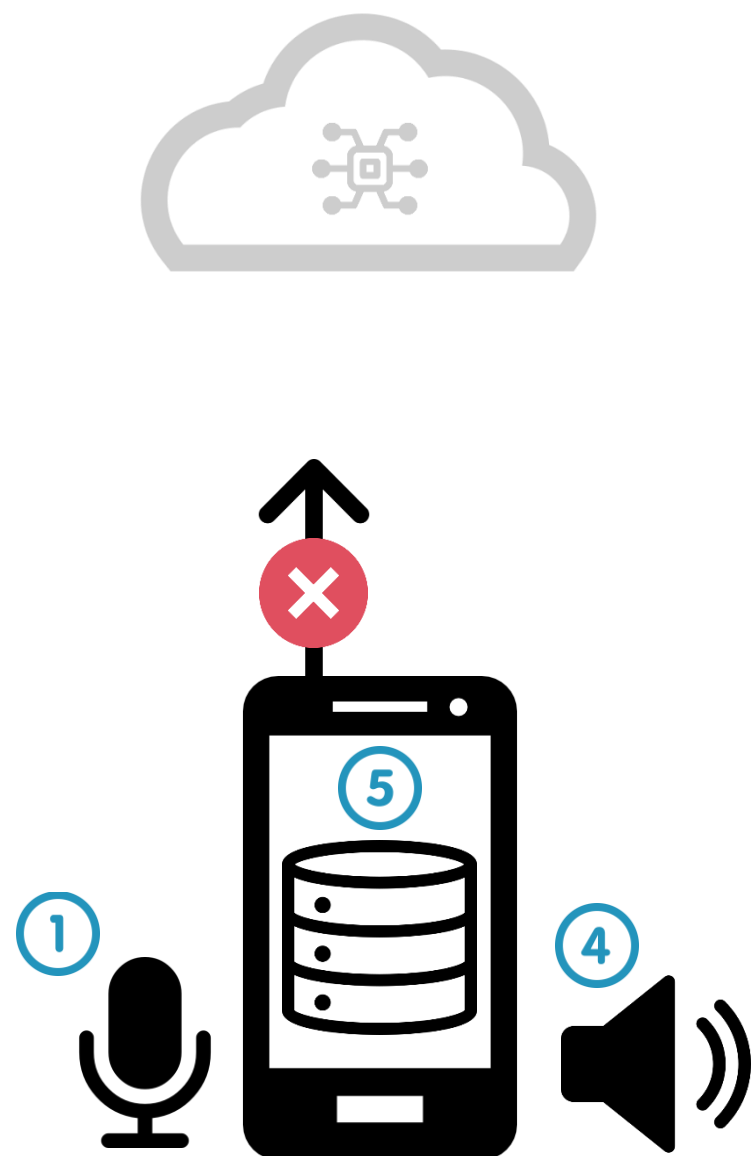
- 程式架構
- 使用語言選擇
 - Python
 - 資源多，有套件可使用 (預估3個禮拜完成在連網狀態下使用ASR與TTS)
 - C 語言
 - Google API 沒有提供 C 的 libraries，實作上感覺有困難

傳統裝置

- ① 輸入音訊
- ② 上傳至雲端運算
- ② 同時儲存(歷史)語句
- ③ 下載轉換後聲音
- ③ 同時儲存(歷史)轉換後語句
- ④ 輸出音訊
- ⑤ 查表



雲端運算並支援離線轉換

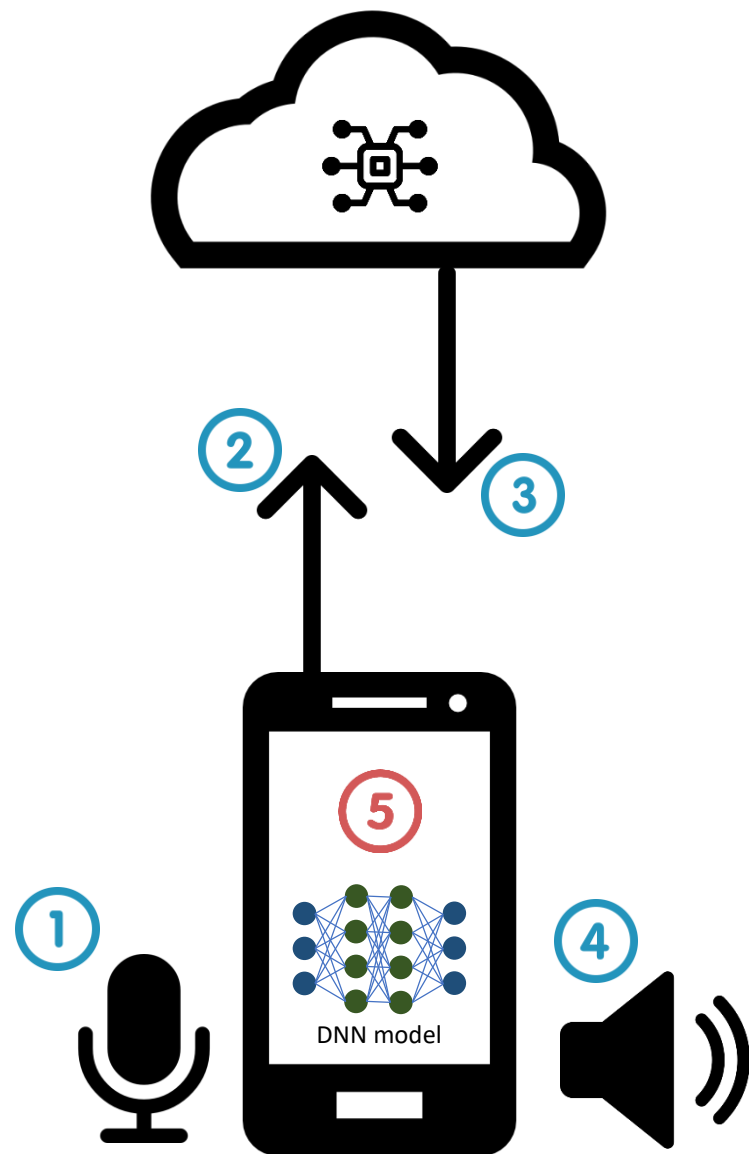


離線轉換

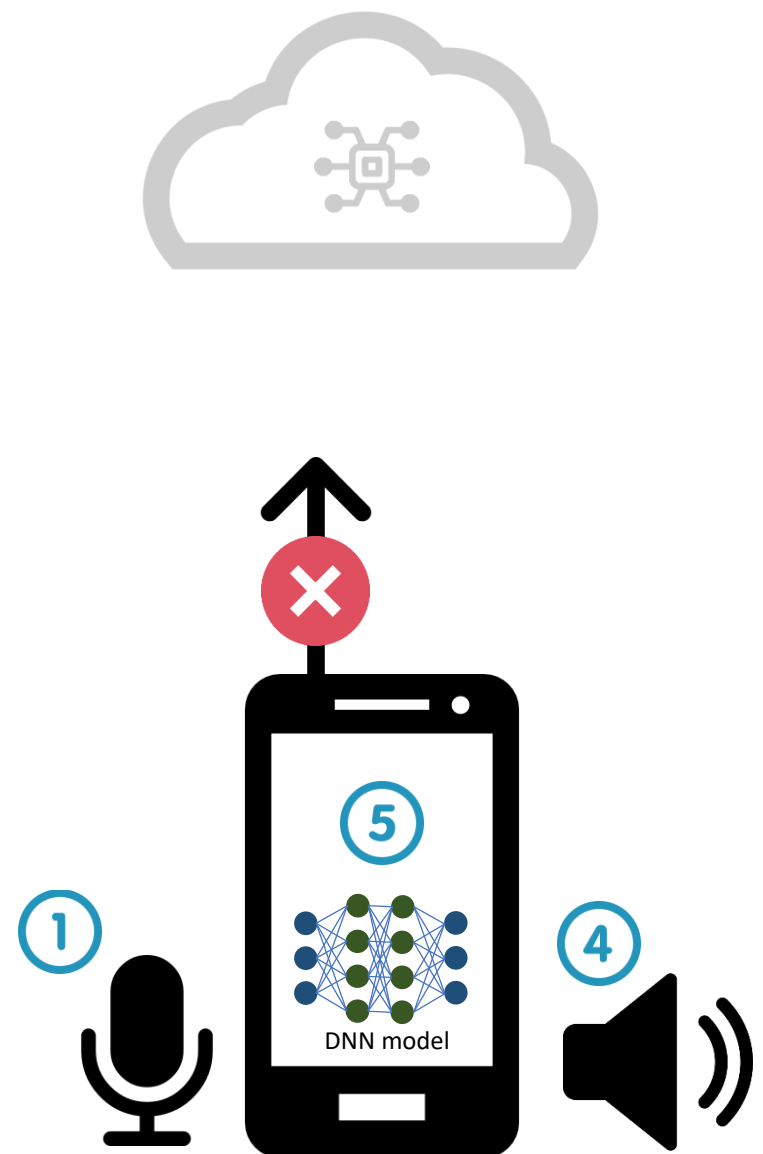
■ 新產品：

離線語音轉換器

- ① 輸入音訊
- ② 上傳至雲端運算
- ③ 下載轉換後聲音
- ④ 輸出音訊
- ⑤ 下載雲端模型並儲存
- ⑤ 利用DNN模型轉換



雲端運算並支援離線轉換



離線轉換

■ 程式架構

麥克風輸入音檔

是否連上網際網路

```
if ( 能連網 ) {
```

```
    Google_API_ASR  // automatic speech recognition 語音變文字
```

```
    Google_API_TTS  // text to speed 文字轉換成語音
```

```
    儲存音檔
```

```
} else {
```

```
    使用儲存的 DNN 模型執行儲存的參數 // 實驗室的 DVC
```

```
}
```

播放音檔

未來規劃：

1. 在連網狀態下，輸入我的聲音，轉換成 google 小姐的聲音 (ASR+TTS)
2. 如何確認是否連網
3. 與實驗室已經做好的 DVC 做整合