

9. Voice recognition

National Chiao Tung University

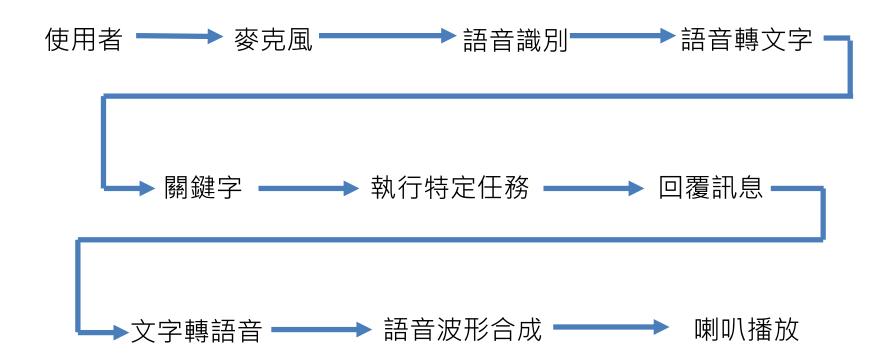


Outline

- □語音助理
 - □ 語音識別 (Speech recognition)
 - □ 自動語音辨識 (Automatic Speech Recognition, ASR)
 - □電腦語音識別 (Computer Speech Recognition)
 - □語音轉文字識別 (Speech To Text, STT)
 - □ 自然語言處理 (Natural Language Processing, NLP)
 - ■讓電腦擁有理解人類語言的能力



語音助理流程



https://zh.wikipedia.org/wiki/%E6%A2%85%E7%88%BE%E5%80%92%E9%A0%BB%E8%AD%9Chttps://en.wikipedia.org/wiki/Mel-frequency_cepstrum

Mel-Frequency Cepstral Coefficients

- MFCCs are commonly used as features in speech recognition systems, such as the systems which can automatically recognize numbers spoken into a telephone.
- □ MFCC(梅爾倒頻譜係數)
 - 1. Take the Fourier transform of a signal (with sliding window)
 - 2. Map the powers of the spectrum obtained above onto the mel scale, using triangular overlapping windows.
 - 3. Take the logs of the powers at each of the mel frequencies.
 - Take the discrete cosine transform of the list of mel log powers, as if it were a signal.
 - The MFCCs are the amplitudes of the resulting spectrum.
- Application: music information retrieval
 - audio similarity measures



Google assistant

1:06





這是什麼歌



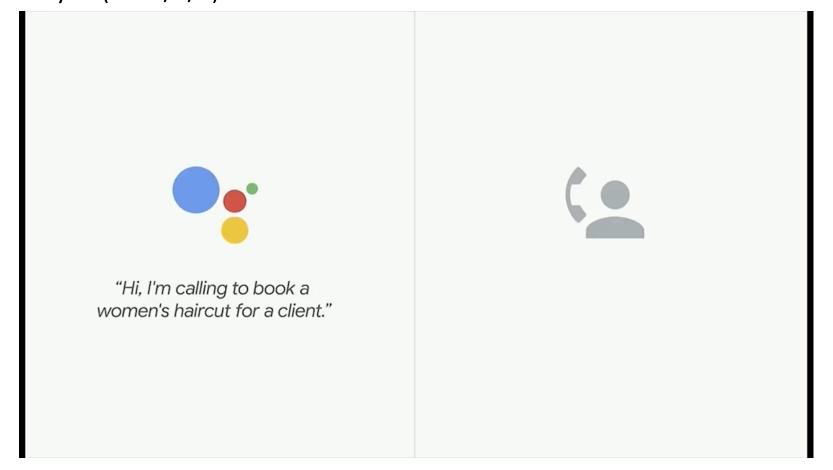
這是 Kate Ryan 的《Voyage voyage》

下午1:06



Google assistant

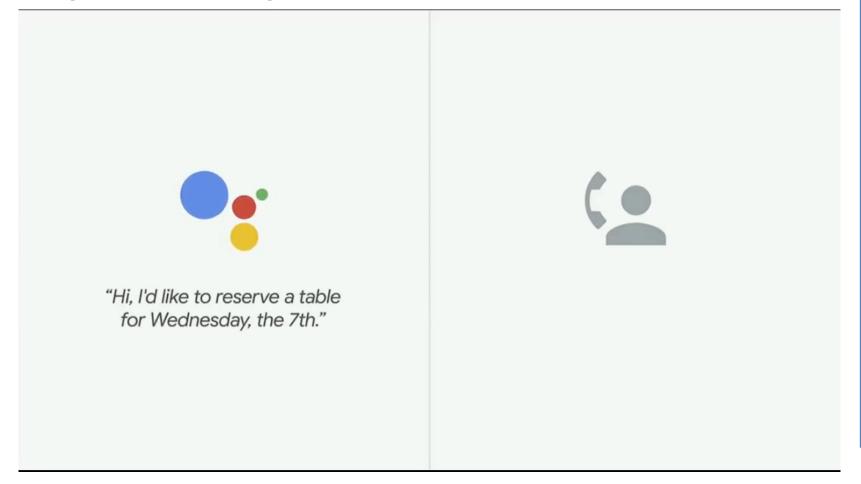
 Google Assistant will soon be able to call restaurants and make a reservation for you (2018/5/9)





Google assistant

Google Assistant calling a restaurant for a reservation





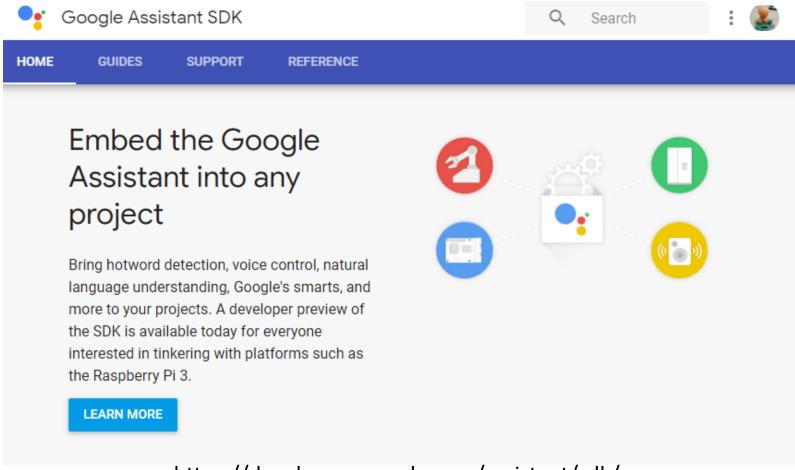


Google IO 2019 Next Gen Google Assistant (2019/5/7)





Google assistant SDK

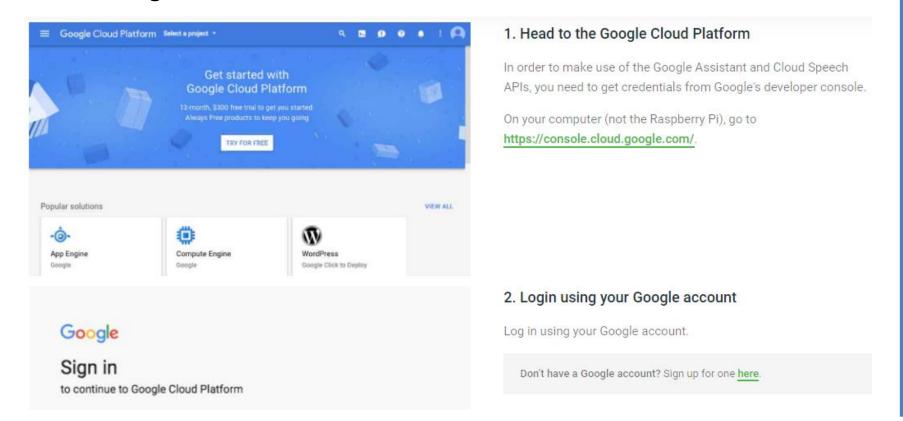


https://developers.google.com/assistant/sdk/





 Do-it-yourself intelligent speaker. Experiment with voice recognition and the Google Assistant.



https://aiyprojects.withgoogle.com/voice/

Azure



語音轉換文字 - 將語音轉換成文字以取得直覺式互動

輕鬆將即時語音轉換文字的功能新增到您的應用程式之中,以應用在語音命令、即時轉譯、自動會議記錄,或是話務中心 的記錄分析等等。





深入了解

文字轉換語音 - 為您的應用程式提供自然語音

建置智慧型應用程式和服務,使用文字轉換語音服務自然地與使用者交談。近乎即時地將文字轉換成音訊,並根據說話速度、音調、音量等變化進行調整。

使用自訂語音模型,為您的應用程式提供獨特且可辨識的品牌語音。只要錄製並上傳定型資料,服務就會建立專為您的錄音調整的獨特語音效果。



深入了解

語音翻譯

為您的應用程式提供任何支援語言的即時語音翻譯功能,並接收文字或語音翻譯。語音翻譯模型是以尖端語音辨識和神經機器翻譯系統 (NMT) 技術為基礎。這些模型已經過最佳化,能夠理解人們在真實生活中的說話方式,並產生絕佳品質的翻譯。



深入了解



SpeechRecognition

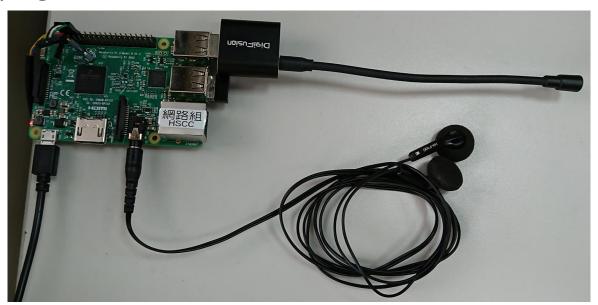
- Library for performing speech recognition, with support for several engines and APIs, online and offline.
- Speech recognition engine/API support:
 - CMU Sphinx (works offline) (卡内基大學)
 - **□** Google Speech Recognition
 - Google Cloud Speech API
 - Wit.ai (Facebook, Messenger ChatBot)
 - Microsoft Bing Voice Recognition
 - Houndify API (SoundHound,音樂識別平台)
 - IBM Speech to Text
 - Snowboy Hotword Detection (works offline)

https://pypi.org/project/SpeechRecognition/



Dependecy

- sudo pip install SpeechRecognition
- sudo pip install gTTS
- □ sudo apt-get install libasound2-dev
- sudo apt-get install python-pyaudio
- sudo apt-get install flac



Test and play microphone

- In terminal
 - Check device
 - aplay -l
 - arecord -l
 - Record you voice
 - arecord -D plughw:1 -f cd Filename.mp3
 - use "ctrl + c" to stop recording
 - Play audio
 - omxplayer -o local -p Filename.mp3



Check your device

aplay -l

```
(COM8) [80x24]
                                                                               ×
                                                                         連線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H)
pi@raspberrypi:~$ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: ALSA [bcm2835 ALSA], device 0: bcm2835 ALSA [bcm2835 ALSA]
 Subdevices: 7/7
 Subdevice #0: subdevice #0
 Subdevice #1: subdevice #1
 Subdevice #2: subdevice #2
 Subdevice #3: subdevice #3
 Subdevice #4: subdevice #4
 Subdevice #5: subdevice #5
 Subdevice #6: subdevice #6
card 0: ALSA [bcm2835 ALSA], device 1: bcm2835 ALSA [bcm2835 IEC958/HDMI]
 Subdevices: 1/1
 Subdevice #0: subdevice #0
card 1: Device [USB Audio Device], device 0: USB Audio [USB Audio]
 Subdevices: 1/1
 Subdevice #0: subdevice #0
pi@raspberrypi:~$
```

arecord -l

```
正 (COM8) [80x24] — □ × 注線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H) pi@raspberrypi:~$ arecord -1 **** List of CAPTURE Hardware Devices **** card 1: Device [USB Audio Device], device 0: USB Audio [USB Audio] Subdevices: 1/1 Subdevice #0: subdevice #0 pi@raspberrypi:~$
```

Test and play microphone

- Record you voice
 - arecord -D plughw:1 -f cd Filename.mp3
 - use "ctrl + c" to stop recording

```
(COM8) [80x24] — □ X 連線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H)

pi@raspberrypi:~$ arecord -D plughw:l -f cd Filename.mp3

Recording WAVE 'Filename.mp3': Signed 16 bit Little Endian, Rate 44100 Hz, Stereo

^CAborted by signal Interrupt...

pi@raspberrypi:~$
```

- Play audio
 - omxplayer -o local -p Filename.mp3

```
正 (COM8) [80x24] — □ ★ 連線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H) pi@raspberrypi:~$ omxplayer -o local -p Filename.mp3 Audio codec pcm_sl6le channels 2 samplerate 44100 bitspersample 16 Subtitle count: 0, state: off, index: 1, delay: 0 have a nice day;) pi@raspberrypi:~$
```

Speech to text (microphone

```
import speech recognition as sr
#obtain audio from the microphone
r=sr.Recognizer()
with sr.Microphone() as source:
  print("Please wait. Calibrating microphone...")
  #listen for 1 seconds and create the ambient noise energy level
  r.adjust for ambient noise(source, duration=1)
  print("Say something!")
  audio=r.listen(source)
# recognize speech using Google Speech Recognition
try:
  print("Google Speech Recognition thinks you said:")
  print(r.recognize google(audio))
except sr.UnknownValueError:
  print("Google Speech Recognition could not understand audio")
except sr.RequestError as e:
  print("No response from Google Speech Recognition service: {0}".format(e))
```

Speech to text (audio file)

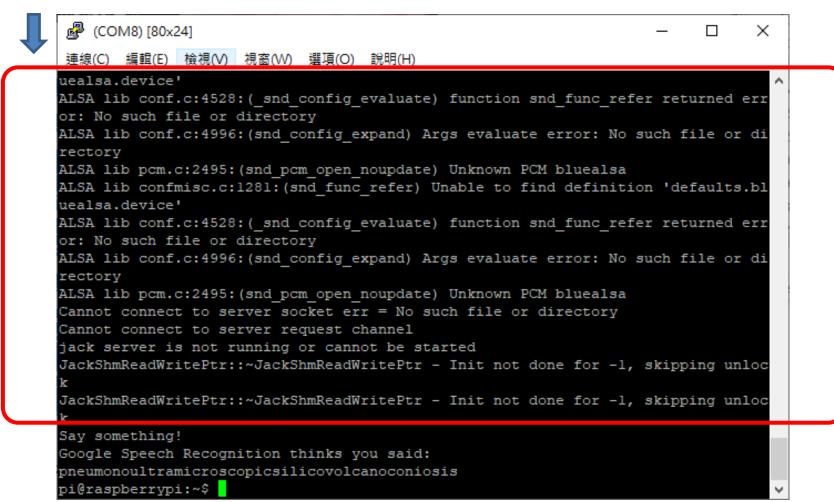
```
import speech recognition as sr
#obtain audio from the microphone
r=sr.Recognizer()
myvoice = sr.AudioFile('hello.flac')
with myvoice as source:
  print("Use audio file as input!")
  audio = r.record(source)
# recognize speech using Google Speech Recognition
try:
  print("Google Speech Recognition thinks you said:")
  print(r.recognize google(audio))
except sr.UnknownValueError:
  print("Google Speech Recognition could not understand audio")
except sr.RequestError as e:
  print("No response from Google Speech Recognition service: {0}".format(e))
```

Input format: PCM WAV, AIFF/AIFF-C, or Native FLAC



Speech to text (result)

You can ignore the ALSA warning messages





SpeechRecognition

Speech recognition engine/API support:

- CMU Sphinx (works offline)
- Google Speech Recognition
- Google Cloud Speech API
- Wit.ai
- Microsoft Bing Voice Recognition
- Houndify API
- IBM Speech to Text

- r.recognize_sphinx(audio)
- r.recognize google(audio)
- r.recognize_google_cloud(audio, credentials_json=GOOGLE_CLOUD_SPEECH_CREDENTIALS)
- r.recognize_wit(audio, key=WIT_AI_KEY)
- r.recognize_azure(audio, key=AZURE_SPEECH_KEY)
- r.recognize_bing(audio, key=BING_KEY)
- r.recognize_houndify(audio, client_id=HOUNDIFY_CLIENT_ID, client_key=HOUNDIFY_CLIENT_KEY)
- r.recognize_ibm(audio, username=IBM_USERNAME, password=IBM_PASSWORD)

Speech Recognition Library Reference

https://github.com/Uberi/speech_recognition/blob/master/reference/library-reference.rst



Text to speech

```
from gtts import gTTS import os

tts = gTTS(text='hello', lang='en') 
tts.save('hello.mp3')

os.system('omxplayer -o local -p hello.mp3 > /dev/null 2>&1')
```

The output format is mp3!

Parameters:

- text (string) The text to be read.
- lang (string, optional) The language (IETF language tag) to read the text in.
 Defaults to 'en'.
- slow (bool, optional) Reads text more slowly. Defaults to False .
- lang_check (bool, optional) Strictly enforce an existing lang, to catch a language error early. If set to True, a ValueError is raised if lang doesn't exist. Default is True.

gTTS (Google Text-to-Speech)

An interface to Google Translate's Text-to-Speech API.

gTTS (gtts.gtts)

Parameters:

- text (string) The text to be read.
- lang (string, optional) The language (IETF language tag) to read the text in.
 Defaults to 'en'.
- slow (bool, optional) Reads text more slowly. Defaults to False .
- lang_check (bool, optional) Strictly enforce an existing lang, to catch a language error early. If set to True, a ValueError is raised if lang doesn't exist. Default is True.



Discussion 1

- When we speak other languages, ex: Mandarin, german, français... etc. How to make r.recognize_google(audio) understand our language?
 - Hint: read the library code
 - https://github.com/Uberi/speech_recognition/blob/master/speech_recognition/__init__.py

Origin: r.recognize_google(audio)





Discussion 2

Similar to discussion 1, how to make gTTS speak other language?

```
gTTS (gtts.gtts)
```

class gtts.tts.gTTS(text, lang='en', slow=False, lang_check=True, pre_processor_funcs=[<function tone_marks>, <function end_of_line>, <function abbreviations>, <function word_sub>], tokenizer_func=<bound method Tokenizer.run of re.compile('(?<=\?).|(?<=\!).|(?<=\!).|(?<=\!).|(?<!\.[a-z])\. |(?<!\.[a-z])\, |(?<!\d)\:|\;|\-|\ \ |\...|\]|\ \circ ', re.IGNORECASE) from: [<function tone_marks>, <function period_comma>, <function other_punctuation>]>) [source]



Quiz 1

- Say a specific command to Raspberry PI, it will start to measure the temperature and humidity.
 - Input could be microphone or audio file or gTTS

```
Say something!

Google Speech Recognition thinks you said:

pneumonoultramicroscopicsilicovolcanoconiosis

pi@raspberrypi:~$
```



```
pi@raspberrypi ~ $ cd Adafruit_Python_DHT/examples/
pi@raspberrypi ~/Adafruit_Python_DHT/examples $ sudo ./AdafruitDHT.py 11 4
Temp=26.0* Humidity=37.0%
```



Quiz 1

- Input
 - **Microphone**: talk to microphone directly
 - Audio file: recode your voice on PC, then send it to Raspberry PI. The file should be PCM WAV, AIFF/AIFF-C, or Native FLAC.
 - gTTS: generate the audio file from text
 - The default output is mp3. Use the following command to convert
 - ffmpeg -i input.mp3 output.flac



Quiz 2

- After measuring temperature, use gTTS (Google Textto-Speech) to speak out the result.
 - Ex: the temperate is 26 degree

```
pi@raspberrypi ~ $ cd Adafruit_Python_DHT/examples/
pi@raspberrypi ~/Adafruit_Python_DHT/examples $ sudo ./AdafruitDHT.py 11 4
Temp=26.0* Humidity=37.0%
```

1896

Summary

- Practice Lab (STT and TTS)
- Write down the answer for discussion
 - Discussion
 - 1. how to make recognize_google understand other language?
 - 2. how to make gTTS speak other language?
- Demonstrate Quiz 1 and Quiz 2 to TAs
 - Quiz1: Say command to execute task
 - Quiz2: Speak out the task result
 - You can combine quiz1 and quiz2 together.