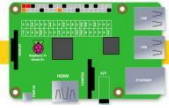
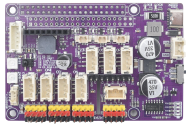



Lesson 28 Examples of AI Access

28.1 Overview

In this class, we will demonstrate a specific example of integrating AI. In this example, we will select DeepSeek as the AI tool, utilize pyttsx3 to implement the text-to-speech function, and rely on sherpa-ncnn to build the speech recognition framework. By completing this AI example, we aim to help beginners gain a basic and intuitive understanding of AI integration, thus opening the door for them to explore the world of AI.

28.2 Required Components

Components	Quantity	Picture
Raspberry Pi	1	
Adeept Robot HAT V3.2	1	
USB Audio - Video Module	1	

28.3 Principle Introduction

In this AI example, we'll implement intelligent interactions from the user's voice input to the voice output. The specific process is as follows:

First, the user starts talking, giving voice commands.

The sherpa-ncnn speech recognition system then converts the captured speech information into text content.

The resulting text content is then used to invoke Deepseek. Deepseek is a powerful AI tool that enables deep thinking and analysis based on input text information to generate relevant responses.

When Deepseek gives a response, use Pytts (pyttsx3) to convert Deepseek's text response to speech.

Finally, the converted voice is played, and the user can hear the answer given by the AI, so as to realize the interactive flow of the dialogue with the AI.

Of course, this is just an example. Once you are familiar with the process, you can also use other options to access AI, such as ChatGPT and so on.

28.4 Demonstration

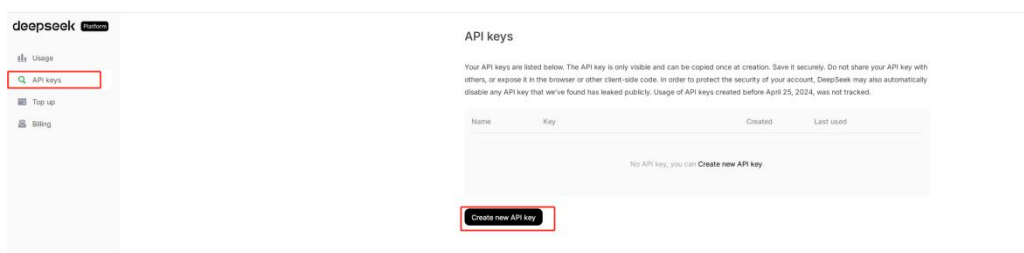
1. Insert the USB recording device into any USB port of the Raspberry Pi. Then power on the Raspberry Pi and log in to the Raspberry Pi terminal remotely. (You also need to insert an audio playback device to play the AI's responses. Alternatively, you can directly check the information that has not been processed by the Text-to-Speech (TTS) system in the console.)

2. Enter the following commands to install the required libraries.

```
sudo apt install espeak-ng libespeak1 --break-system-packages
sudo pip3 install pyttsx3 --break-system-packages
sudo pip3 install openai --break-system-packages
```

3. Apply for the API key of DeepSeek so that you can use the DeepSeek service. For details, please visit this link to apply: <https://platform.deepseek.com/>

Enter the DeepSeek Platform and create an API key



API keys

Your API keys are listed below. The API key is only visible and can be copied once at creation. Save it securely. Do not share your API key with others, or expose it in the browser or other client-side code. In order to protect the security of your account, DeepSeek may also automatically disable any API key that we've found has leaked publicly. Usage of API keys created before April 25, 2024, was not tracked.

Name	Key	Created	Last used
No API key, you can Create new API key .			

Create new API key

Create new API key

Name

test key

Cancel

Create API key

API keys

✓ API key created.

Your API keys are listed below. The API key is only visible and can be copied once at creation. Save it securely. Do not share your API key with others, or expose it in the browser or other client-side code. In order to protect the security of your account, DeepSeek may also automatically disable any API key that we've found has leaked publicly. Usage of API keys created before April 25, 2024, was not tracked.

Name	Key	Created	Last used
test key	sk-57549*****86a8	2025-03-17	-

Create new API key

Create new API key

Please save this API key somewhere safe and accessible. For security reasons, you won't be able to view it again through your account. If you lose this API key, you'll need to generate a new one.

sk-5754957ac4364f64a0d03643e17586a8

Done

Copy

deepseek Platform

Usage
API keys
Top up
Billing

Top up

Please complete real-name verification to enjoy more services [Verify](#)

Online recharge Bank transfer

Amount

¥10 ¥20 ¥50 ¥100 ¥300 ¥500 Custom Pricing

Off-Peak Discounts: DeepSeek-V3 with 50% off and DeepSeek-R1 with 75% off at off-peak hours (16:30-00:30 UTC daily). Optimize your workflow while enjoying these exclusive savings. [View Details](#)

Payment method

支付宝 Alipay
微信支付 WeChat Pay

Next step

Note

1. The recharge amount is solely for API service usage. Conversations on the web version and app are free and do not require any recharge.
2. After recharging, you can visit the billing page to request an invoice.

You can use the API key you just created after topping up.

4. Enter the following directory and modify the TalkToAI.py file.

```
cd Adeept_PiCar-Pro/Examples/14_Example_Of_AI
```

```
pi@raspberrypi:~ $ cd Adeept_PiCar-Pro/Examples/14_Example_Of_AI
pi@raspberrypi:~/Adeept_PiCar-Pro/Examples/14_Example_Of_AI $
```

Check the playback device and remember the number of the playback device's sound card, e.g. Card 2

```
aplay -l
```

```
pi@raspberrypi:~/Adeept_PiCar-Pro/Examples/14_Example_Of_AI $ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: vc4hdmi0 [vc4-hdmi-0], device 0: MAI PCM i2s-hifi-0 [MAI PCM i2s-hifi-0]
  Subdevices: 1/1
    Subdevice #0: subdevice #0
card 1: vc4hdmi1 [vc4-hdmi-1], device 0: MAI PCM i2s-hifi-0 [MAI PCM i2s-hifi-0]
  Subdevices: 1/1
    Subdevice #0: subdevice #0
card 2: Device [USB2.0 Device], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
    Subdevice #0: subdevice #0
```

```
sudo nano TalkToAI.py
```

Replace the value of DEEPSEEK_API_KEY with the API key you just applied for.

```

audio_file = "./response_audio.wav"

# Assume you've completed local speech recognition and obtained the recognized text
local_recognition_text = "hello"

# DeepSeek API settings, replace with your own information
DEEPSEEK_API_URL = "https://api.deepseek.com"
DEEPSEEK_API_KEY = "sk-cb3625243e214fea872c815d41624201"

```

Replace the sound card number with the number you just remembered. For example, if it is card 2, after modification, it should be `plughw:2,0`.

```

# Function to play the audio file
def play_audio(audio_file):
    """
    Play the audio file using the system's default media player.
    :param audio_file: The path to the audio file.
    """
    try:
        os.system('aplay -D "plughw:2,0" ' + audio_file)
    except Exception as e:
        print(f"Error occurred when playing the audio file: {e}")

```

Press **Ctrl + X**, **Y**, and then **Enter** to save the file and exit.

6. Check the available recording device numbers. Open **Speech.py** and then replace the recording device number.

```

arecord -l

pi@raspberrypi:~/Adeept_PiCar-Pro/Examples/14_Example_Of_AI $ arecord -l
**** List of CAPTURE Hardware Devices ****
card 2: Device [USB2.0 Device], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 3: Camera [FHD C3 Camera], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0

```

```

sudo nano Speech.py

```

```
def main():
# cmd = "sudo ~/home/pi/sherpa-ncnn/build/bin/sherpa-ncnn-microphone \
# /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/tokens.txt \
# /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/encoder_jit_trace-pnnx.ncnn.param \
# /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/encoder_jit_trace-pnnx.ncnn.bin \
# /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/decoder_jit_trace-pnnx.ncnn.param \
# /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/decoder_jit_trace-pnnx.ncnn.bin \
# /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/joiner_jit_trace-pnnx.ncnn.param \
# /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/joiner_jit_trace-pnnx.ncnn.bin"

# You can also use the 'sherpa-ncnn-alsa' command for speech recognition. Note that you need to replace the 'plughw:3,0' parameter with
cmd = "sudo /home/pi/sherpa-ncnn/build/bin/sherpa-ncnn-alsa \
/home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/tokens.txt \
/home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/encoder_jit_trace-pnnx.ncnn.param \
/home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/encoder_jit_trace-pnnx.ncnn.bin \
/home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/decoder_jit_trace-pnnx.ncnn.param \
/home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/decoder_jit_trace-pnnx.ncnn.bin \
/home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/joiner_jit_trace-pnnx.ncnn.param \
/home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/joiner_jit_trace-pnnx.ncnn.bin \
plughw:3,0 \
# \
greedy_search"
os.system(f"{cmd} > output.txt 2>&1") #Run a command-line program and save the output results to a file named 'output. txt'
```

Press **Ctrl + X**, **Y**, and then **Enter** to save the file and exit.

7.Run the **ExampleOfAI.py** program and start a conversation with the AI.

```
sudo python3 ExampleOfAI.py
```

```
pi@raspberrypi:~/Adeept_PiCar-Pro/Examples/14_Example_Of_AI $ sudo python3 ExampleOfAI.py
I:  hello
    deepseek: Hello! How can I assist you today? □
                                     Playing WAVE './response_audio.wav' : Signed 16 b
it Little Endian, Rate 22050 Hz, Mono
```

28.5 Code

ExampleOfAI.py

```
01  #!/usr/bin/env/python
02  # File name   : ExampleOfAI.py
03  # Website    : www.Adeept.com
04  # Author     : Adeept
05  # Date      : 2025/03/13
06  import subprocess
07  import time
08  # Define the paths to two Python programs to run
09  program1_path = "./Speech.py"
10  program2_path = "./TalkToAI.py"
11
12  # Create two sub processes and run two Python programs separately
13  process1 = subprocess.Popen(["python3", program1_path])
14  time.sleep(3)          # Waiting for speech recognition to start
15  process2 = subprocess.Popen(["python3", program2_path])
16
17  # Waiting for two child processes to complete
18  process1.wait()
19  process2.wait()
```

Speech.py

```

01  #!/usr/bin/env/python
02  # File name   : Speech.py
03  # Website    : www.Adeept.com
04  # Author     : Adeept
05  # Date      : 2025/03/13
06  import os
07
08  def main():
09      # cmd = "sudo ~/home/pi/sherpa-ncnn/build/bin/sherpa-ncnn-microphone \
10      #      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/tokens.txt
11      \
12      #      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
13      13/encoder_jit_trace-pnnx.ncnn.param \
14      #      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
15      13/encoder_jit_trace-pnnx.ncnn.bin \
16      #      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
17      13/decoder_jit_trace-pnnx.ncnn.param \
18      #      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
19      13/decoder_jit_trace-pnnx.ncnn.bin \
20      #      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
21      13/joiner_jit_trace-pnnx.ncnn.param \
22      #      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
23      13/joiner_jit_trace-pnnx.ncnn.bin"
24
25      # You can also use the `sherpa-ncnn-alsa` command for speech recognition. Note that you need to
26      replace the `plughw:3,0` parameter with the serial number of your own sound card.
27      cmd = "sudo /home/pi/sherpa-ncnn/build/bin/sherpa-ncnn-alsa \
28      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-13/tokens.txt
29      \
30      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
31      13/encoder_jit_trace-pnnx.ncnn.param \
32      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
33      13/encoder_jit_trace-pnnx.ncnn.bin \
34      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
35      13/decoder_jit_trace-pnnx.ncnn.param \
36      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
37      13/decoder_jit_trace-pnnx.ncnn.bin \
38      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
39      13/joiner_jit_trace-pnnx.ncnn.param \
40      /home/pi/sherpa-ncnn/sherpa-ncnn-streaming-zipformer-bilingual-zh-en-2023-02-
41      13/joiner_jit_trace-pnnx.ncnn.bin \
42      plughw:3,0 \
43      4 \
44      greedy_search"
45
46      os.system(f"{cmd} > output.txt 2>&1") #Run a command-line program and save the output results to a
47      file named 'output. txt'
48
49  if __name__ == "__main__":
50      main()

```


TalkToAI.py

```
01  #!/usr/bin/env/python
02  # File name   : TalkToAI.py
03  # Website    : www.Aadept.com
04  # Author     : Aadept
05  # Date      : 2025/03/13
06
07  import pyttsx3
08  import os
09  from openai import OpenAI
10  import time
11
12  audio_file = "./response_audio.wav"
13
14  # Assume you've completed local speech recognition and obtained the recognized text
15  local_recognition_text = "hello"
16
17  # DeepSeek API settings, replace with your own information
18  DEEPSEEK_API_URL = "https://api.deepseek.com"
19  DEEPSEEK_API_KEY = "sk-cb3625243e214fea872c815d41624201"
20
21  # Function to send a request to the DeepSeek cloud service and get the response
22  def get_deepseek_response(text):
23      """
24      Send a request to the DeepSeek cloud service and return the response.
25      :param text: The input text for the DeepSeek service.
26      :return: The output text from the DeepSeek service, or an empty string if an error occurs.
27      """
28      try:
29          client = OpenAI(api_key=DEEPSEEK_API_KEY, base_url=DEEPSEEK_API_URL)
30          response = client.chat.completions.create(
31              model="deepseek-chat",
32              messages=[
33                  {"role": "system", "content": "You are a helpful assistant"},
34                  {"role": "user", "content": text},
35              ],
36              stream=False
37          )
38          # print("deepseek response: " +str(response))
39          return response.choices[0].message.content
40      except Exception as e:
41          print(f"Error occurred when requesting the DeepSeek service: {e}")
42          return ""
43
44
45  def text_to_speech(text, output_file):
46      engine = pyttsx3.init()
47      engine.setProperty('rate', 150)    # Speed adjustment (50-200)
48      engine.setProperty('volume', 0.8)  # Volume control (0.0-1.0)
49
50      voices = engine.getProperty('voices')
51      engine.setProperty('voice', voices[29].id)    #29: English (America, New York City) (['en-us-nyc'])
52      # If you want to change the language, you can run the following code to get the list of languages,
53      and then select the appropriate language.
```



```
54     # voices = engine.getProperty('voices')
55     # for i, voice in enumerate(voices):
56     #     print(f"{i}: {voice.name} ({voice.languages})")
57
58     engine.save_to_file(text, output_file)
59     engine.runAndWait()
60
61
62 # Function to play the audio file
63 def play_audio(audio_file):
64     """
65     Play the audio file using the system's default media player.
66     :param audio_file: The path to the audio file.
67     """
68     try:
69         os.system('aplay -D "plughw:2,0" ' + audio_file) #In the example, the playback device is
70         card 2, so "plughw:2,0" is used. You can use the command "aplay -l" to check your own playback device.
71     except Exception as e:
72         print(f"Error occurred when playing the audio file: {e}")
73
74 file_position = 0
75 while True:
76     with open("output.txt", "r") as file: # Read the file named "output.txt"
77         file.seek(file_position)
78         new_lines = file.readlines() # Read all lines from the current file pointer position to the end
79 of the file
80         if new_lines:
81             for line in new_lines:
82                 if "Started" in line:
83                     local_recognition_text = line.split("Started")[-1].strip()
84                     elif file_position > 0: # Ensure we print lines after the first "Started"
85                         local_recognition_text = line.strip()
86
87             file_position = file.tell()
88             print("I: " + local_recognition_text)
89
90             if local_recognition_text:
91                 deepseek_response = get_deepseek_response(local_recognition_text)
92                 if deepseek_response:
93                     print("deepseek: " + deepseek_response)
94                     text_to_speech(deepseek_response, audio_file)
95                     play_audio(audio_file)
96                     if os.path.exists(audio_file):
97                         os.remove(audio_file)
98
99             time.sleep(5) # Read every 5 second
```