DaVinci – The ChatGPT Virtual Assistant

DaVinci - The ChatGPT virtual assistant is a voice-controlled and voice-response assistant that uses OpenAI's artificial intelligence language model to assist with a wide range of tasks, such as answering questions, providing information, giving suggestions, telling jokes, writing stories and much more. In addition to providing responses verbally, DaVinci also provide written responses if your device is connected to a display.

Davinci - The ChatGPT virtual assistant is a python program that utilizes Picovoice Porcupine (and (PyAudio) to detect a wake word; Picovoice Cobra voice activity detection to determine when the user begins and finishes speaking their query; Picovoice Leopard to convert the spoken query to text; OpenAI ChatGPT to respond to the query; Amazon Polly text to speech to convert the response into a natural-sounding human voice; and Pygame to play the audio.

DaVinci uses the most current ChatGPT API that was released on March 1, 2023. DaVinci will be updated to run on GPT4 as soon as that API is widely available to developers.

The instructions below are for running the DaVinci – The ChatGPT Virtual Assistant on a Raspberry Pi 4, but it can also be run on other systems with minimal modifications. The process for obtaining access keys for ChatGPT, Picovoice solutions and Amazon Polly is the same regardless of the system used.

You can watch a brief video demo of DaVinci – The ChatGPT Virtual Assistant here: https://youtu.be/r2R4Yoy8XzU

How to Run Davinci on a Raspberry Pi 4 Mini-Computer

The following steps are required to run Davinci - The ChatGPT Virtual Assistant on your Raspberry Pi 4:

- Obtain the necessary hardware
- Create an OpenAI account and obtain your personal secret API key
- Create a Picovoice account and obtain your personal secret access key
- Create an AWS account and obtain your personal access key and secret access key
- Follow the other steps below for preparing your Raspberry Pi and downloading DaVinci - The ChatGPT Virtual Assistant

Note that although account registrations are required to obtain the keys to use the outside services and that production-use of those resources requires the payment of fees, these services should all be free to a user who only utilizes them for the hobbyist requirements of DaVinci - The ChatGPT Virtual Assistant for private use.

Hardware Requirements

Raspberry Pi 4 – This needs to be a Raspberry Pi 4, so that you can run the 64-bit operating system. Earlier versions of Raspberry Pis are likely to throw memory errors while running this program. The 2 GB RAM model is sufficient: https://www.adafruit.com/product/4292
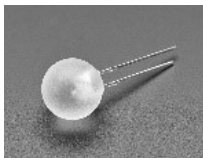
Power Supply – 5v power supply. I recommend the official Raspberry Pi power supply: https://www.adafruit.com/product/4298
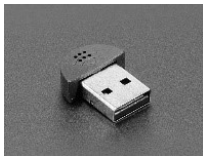
USB Microphone – for talking to the assistant. Any USB mic should work, and this inexpensive one is sufficient: https://www.adafruit.com/product/3367

USB speaker – for the assistant to talk back to you. I used this one: https://www.adafruit.com/product/3369

LEDs (optional) – The program optionally lights LEDs when it detects the wake word and pulses them when it is responding to your queries. These spherical LEDs were used in the video demo: https://www.adafruit.com/product/5430

Resistors (220 ohm) (optional) – Only needed if you decide to use the optional LEDs. I used these: https://www.adafruit.com/product/2780

Heatsinks (optional) – The heatsinks are optional but are highly recommended if you will put your Raspberry Pi in a case or other confined container such as the one used in the video demo. I used these: https://www.amazon.com/dp/B07ZLZRDXZ?psc=1&smid=A2PL6UUXJC5K5G&ref_=chk_typ_imgToDp

STLs for the DaVinci 3D-printed enclosure shown in the video demo are also available on this GitHub repository if you have access to a 3D printer and would like to print one.

CAUTION: Raspberry Pi 4s can run hot. If you use an enclosure, make certain that it is properly ventilated and that you use heatsinks or a fan.

Optional LED Wiring

DaVinci – The ChatGPT Virtual Assistant includes the ability to light up LEDs that help show the status of the program. If connected, the LEDs will light up when the program hears the wake word and will pulse after the user finishes their query and while the assistant provides its response. The proper wiring of the LEDs and resistors to the Raspberry Pi is shown on this diagram. The LEDs are controlled by GPIO pins 18 and 23. The use of the LEDs is optional and will not otherwise impact the functionality of DaVinci.
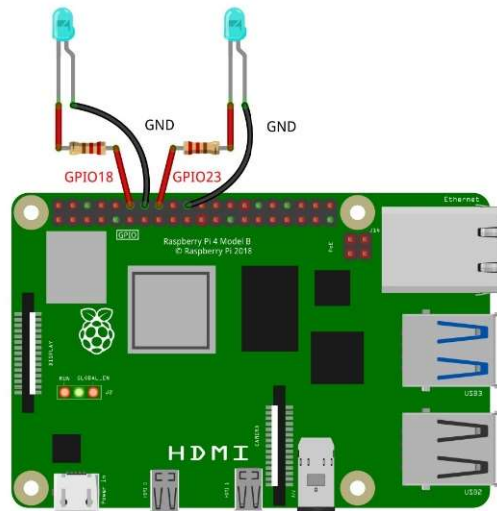

Create an OpenAI Account

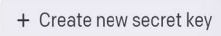Open a web browser on your PC and navigate to the OpenAI website - https://openai.com/.

In the upper right-hand corner, click on "API"

On the next screen, click on **SIGN UP** and then follow the prompts to create your account.

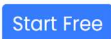## Obtain Your Secret OpenAI API Key

Next, on the logged-in screen, click on your Personal icon in the upper right-hand corner and then click on "View API keys".
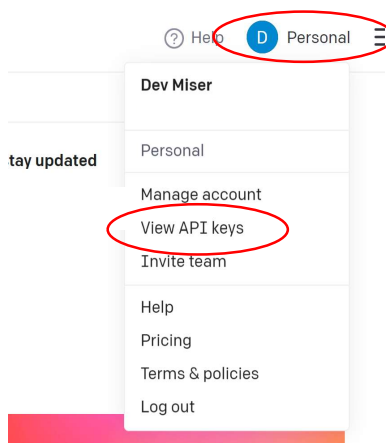
Then click on `+ Create new secret key` to create your new secret API key. Copy your API key and keep it in a secure location. You will need it in a later step.

Create a Picovoice Account

Create a free Picovoice account. Open a web browser and navigate to https://picovoice.ai/.

In the upper right-hand corner, click the `Start Free` button and follow the prompts to open your account.

Obtain Your Secret Picovoice Access Key

As soon as you complete your sign up, you will automatically be redirected to a page with your secret access key. Copy your 🔑 AccessKey and keep it in a secure location. You will need it in a later step.

Create an AWS Account

Create a free AWS account. Open a web browser and navigate to the following website to use the AWS Free Tier: https://aws.amazon.com/free/

In the upper right-hand corner, click on the `Complete Sign Up` button and follow the prompts to open your account.

Create an AWS IAM User Account and Obtain Your AWS Access Key and Secret Access Key

After you create your AWS account, you will next need to create an IAM User Account on AWS. To do so, follow AWS's instructions here: https://docs.aws.amazon.com/IAM/latest/UserGuide/getting-set-up.html#create-an-admin

Or you can try the detailed steps in Addendum I. The process is revised occasionally by AWS and may no longer be the same as in the Addendum.

Copy your "Access key" and "Secret access key" and keep them in a secure location. You will need them in a later step.

Prepare Your Raspberry Pi 4 to Run the Program

These instructions assume you already have a Raspberry Pi 4 set up and running. If not, set up your Raspberry Pi using the instructions found here: https://www.raspberrypi.com/tutorials/how-to-set-up-raspberry-pi/

Be certain to load the 64-bit (not the 32-bit) version of the Raspberry Pi OS when setting up your Raspberry Pi. If you use the 32-bit version, you are likely to get memory errors when running the program.

To prepare your Raspberry Pi to run DaVinci – The ChatGPT Virtual Assistant, do the following:

1. Edit the bashrc file on your Raspberry Pi as follows:

a. Open a terminal >_ and enter the following command to open the bashrc file:

   sudo nano ~/.bashrc

b. Scroll to the bottom of the file using your keyboard and add the following lines at the end (be certain to include the #s):

   # sets a location where the Raspberry Pi OS and Python can look for
   # executable/configuration files
   export PATH="$HOME/.local/bin:$PATH"

c. Press the CTRL and X keys simultaneously on your keyboard, then press Y and then press Enter to save the revisions to the file.

d. Then enter the following command:

   sudo reboot

This will reboot your Raspberry Pi. After the reboot is completed, log back in.

2. Open a terminal and enter the following commands in the following order:

   sudo apt update

   sudo apt full-upgrade

   pip3 install --upgrade pip

   sudo apt-get install portaudio19-dev [When asked if you want to continue, enter Y and then press Enter]

   pip3 install pyaudio

   pip3 install pvrecorder

   pip3 install pvporcupine

pip3 install pvcobra

pip3 install pvleopard

pip3 install --upgrade openai

pip3 install boto3

pip3 install awscli

sudo reboot – will reboot your Raspberry Pi; log back in after the reboot.

3. Configure the AWS command line interface by opening a terminal and entering the following command:

aws configure

The following lines will appear one-by-one. Take the actions specified in the brackets:

AWS Access Key ID [None]: [type in your access key ID and press enter]

AWS Secret Access Key [None]: [type in your secret access key and press enter]

Default region name [None]: [check for the name of the region closest to you here: https://docs.aws.amazon.com/general/latest/gr/rande.html#pol_region – mine is us-east-1 – then type it in and press enter]

Default output format [None]: [leave this blank and press enter]
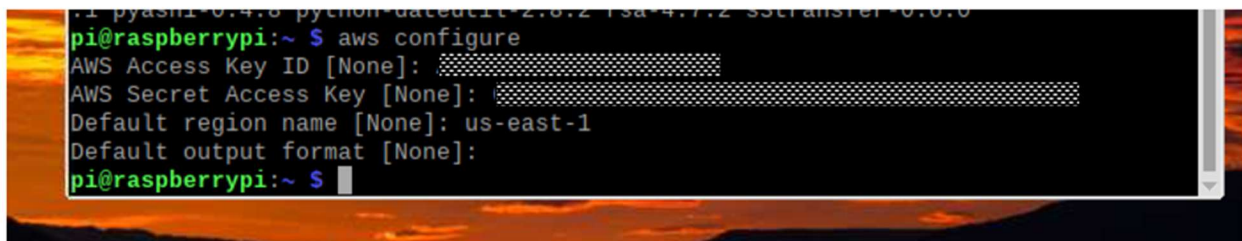
4. Download the DaVinci.py program and associated files by opening a terminal and entering the following command:

git clone https://github.com/DevMiser/DaVinci.git

5. Modify DaVinci.py to include the secret API key that you previously created in your OpenAI account and the secret access key that you created in your Picovoice account by doing the following:

a. Open a terminal and enter the following commands:

cd /home/pi/Davinci

sudo nano DaVinci.py

b. Use your keyboard to scroll down to the lines that say:

openai.api_key = "*put your secret API key between these quotation marks*"
pv_access_key= "*put your secret access key between these quotation marks*"

Now modify those lines to replace the portions that are italicized above with your secret OpenAI API key and your secret Picovoice access key, respectively.

c. Press the CTRL and X keys simultaneously on your keyboard, then press Y and then press Enter to save the revisions to the file.

6. Move the DaVinci keyword file to the Porcupine raspberry-pi folder by entering the following command:

mv /home/pi/DaVinci/DaVinci_raspberry-pi.ppn /home/pi/.local/lib/
python3.9/site-packages/pvporcupine/resources/keyword_files/raspberry-pi

7. Reboot your Raspberry Pi.

Run the Program

Plug your microphone and speaker into USB ports on your Raspberry Pi 4.

To run the program, open a terminal and enter the following commands:

cd /home/pi/DaVinci

python3 DaVinci.py

You can then wake up your ChatGPT Virtual Assistant by saying one of the following wake words: DaVinci, computer or Jarvis.

DaVinci – The ChatGPT Virtual Assistant will respond by saying "How may I assist you?", "I'm listening." or something similar.

Then make your request. For example, say:

Describe purple.

What is the difference between pumas and leopards?

What are the most popular tourist attractions in Tel Aviv?

What would happen if the moon disappeared?

What can you do?

Write a silly dialogue between a stapler and a coffee mug.

What is the plot of To Kill a Mockingbird?

What is the meaning of the song Stairway to Heaven by Led Zeppelin.

Tell me a story about a mouse named Mike who is looking for his lost cheese in the Astrodome.

What are the top 5 foods that should be included in a healthy diet.

Make a joke about dancing robots.

How can I avoid procrastinating today?

*Write a short poem about artificial intelligence.*

When you are finished with the program, press the CTRL and C keys simultaneously on your keyboard and you will exit the program.

Addendum I

AWS Identity and Management Access (IAM) Dashboard Instructions

The following instructions are how to to create an IAM User Account on AWS. This process is revised occasionally by AWS and may no longer be exactly the same as in this Addendum. You can check AWS's documentation for the most current instructions.

After you create your AWS account, sign in and go to the Identify and Manage Access (IAM) dashboard at this url: https://console.aws.amazon.com/iamv2
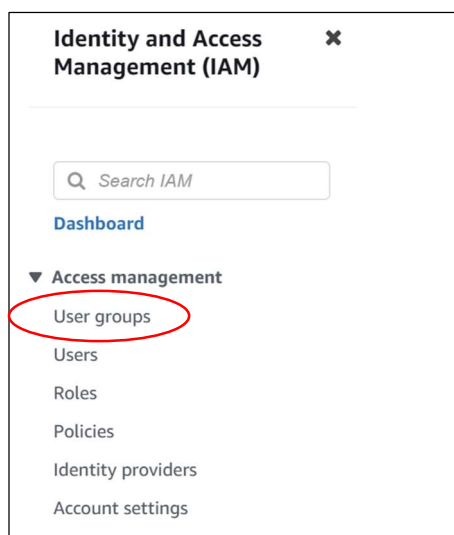
1. Select "User groups" from the menu on the left-hand side and then click [Create group] in the upper right-hand corner.

2. On the next page, enter "Polly" into the box underneath "User group name ".

3. Further down on the same page in the box that says "Attach permissions policies – Optional", search for "Polly" and then select "AmazonPollyFullAccess" from the search results.

4. Next select "Users" from the left-hand menu.

5. Click [Add users] from the upper right-hand corner and type "Polly_User" in the box under "User name" and click "Next".

6. On the next page under Permission options, select "Add user to group" and check the box next to "Polly" under "User groups", and then click "Next".

7. On the next page, click [Create user] .

8. On the next page, click the "View User" button:

IAM > Users > Polly_User

Polly_User

Delete

Summary

ARN	Console access	Access key 1
▨▨▨▨▨▨▨▨▨▨	Disabled	Not enabled
Created	Last console sign-in	Access key 2
March 07, 2023, 11:33 (UTC-05:00)	-	Not enabled

Permissions   Groups   Tags   **Security credentials**   Access Advisor

Console sign-in   Enable console access

Console sign-in link	Console password
▨▨▨▨▨▨▨▨▨▨	Not enabled

Multi-factor authentication (MFA) (0)
Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. Learn more

Remove   Resync   Assign MFA device

Device type	Identifier	Created on

No MFA devices. Assign an MFA device to improve the security of your AWS environment
Assign MFA device

Access keys (0)
Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. Learn more

Create access key

---

9. On the next page, click the "Security credentials" tab that is part way down the page.

And then scroll further down on the page and select the "Create access key" button.

Access key best practices & alternatives

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alterna

○ **Command Line Interface (CLI)**
You plan to use this access key to enable the AWS CLI to access your AWS account.

○ **Local code**
You plan to use this access key to enable application code in a local development environment to access your AWS account.

○ **Application running on an AWS compute service**
You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.

○ **Third-party service**
You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.

○ **Application running outside AWS**
You plan to use this access key to enable an application running on an on-premises host, or to use a local AWS client or third-party AWS plugin.

○ **Other**
Your use case is not listed here.

⚠ **Alternatives recommended**
• Use AWS CloudShell, a browser-based CLI, to run commands. Learn more ☐
• Use the AWS CLI V2 and enable authentication through a user in IAM Identity Center. Learn more ☐

☑ I understand the above recommendation and want to proceed to create an access key.

---

10. On the next page, select the Command Line Interface (CLI) option, check the "I understand…" box at the bottom, and then click "Next".

11. On the next page, leave it blank and click **Create access key** .

12. The next page will be headed "Retrieve access keys".

Copy both the "Access key" and "Secret access key" and keep them in a secure location. This is the only time that the secret access key can be viewed or downloaded. If you lose it, you will need to create a new access key and make the old one inactive.