

Dokumentation

Diese Dokumentation beschreibt die Nutzung von Mycroft.ai im Rahmen des Komplexpraktikums Mensch-Computer-Interaktion. Mycroft ist ein Open Source Sprachassistent und ist auf Linux sowie Android und insbesondere dem Raspberry Pi nutzbar. Diese Dokumentation betrachtet nur die Nutzung auf Linux und dem Raspberry Pi.

Die offizielle Dokumentation für Mycroft findet sich unter <https://mycroft.ai/documentation/>

Setup

Linux

Unter Linux lässt sich Mycroft relativ leicht installieren. Dafür gibt es zwei Möglichkeiten. Zum einen kann Mycroft in vielen Distributionen über den jeweiligen Paketmanager heruntergeladen und installiert werden und zum anderen ist auch eine manuelle Installation über das Git-Repository (<https://github.com/MycroftAI/mycroft-core/blob/dev/README.md>) möglich.

Raspberry Pi

Um einen Pi als Sprachassistenten einzurichten, benötigt man einen Raspberry Pi 3 (oder besser), einen USB oder 3.5 mm Klinke Lautsprecher oder einen Bildschirm mit Lautsprechern und ein USB-Mikrofon. Außerdem werden eine Tastatur und ein Bildschirm benötigt, um das Setup auszuführen. Die Raspberry Pi Version von Mycroft (Picroft <https://github.com/MycroftAI/enclosure-picroft>) kommt als gewöhnliches Image, was auf eine SD Karte geflasht werden kann.

Wenn der Pi das erste Mal startet, folgt ein Dialog der das Einrichten einer Internetverbindung ermöglicht. Danach kann ein Setup ausgeführt werden, was das Einrichten von Lautsprecher und Mikrofon vereinfacht. Dies funktioniert für den Lautsprecher gut, da bei diesem nur der Anschlusstyp entscheidend ist. Für das Mikrofon funktioniert das Setup nur gut, wenn es eines der von Mycroft empfohlenen Mikrofone ist. Wenn man ein anderes angeschlossen hat muss man über "alsamixer", was schon vorinstalliert ist oder "pulseaudio" das richtige Gerät auswählen.

Nach dem Setup wird der Mycroft-Client zum ersten Mal gestartet und das Pairing erfolgt.

Bei der Verwendung eines Raspberry Pis ist darauf zu achten, dass das Netzteil genug Ampere zur Verfügung stellt. Mycroft selbst empfiehlt 2,5A, erfolgreich getestet wurde es auch mit 2A. Steht zu wenig zur Verfügung wird die Erkennung der Sprachbefehle schlechter oder manche Geräte wie Mikrofon oder Lautsprecher werden nicht mehr erkannt oder funktionieren nicht richtig.

Empfohlene Reihenfolge für das Setup:

1. Image auf SD-Karte flashen und Pi starten
2. Setup durchführen und mit Internet verbinden

3. Picroft aktualisieren ("sudo apt-get update" und "sudo apt-get upgrade")
4. Mikrofon mit alsamixer einrichten oder pulseaudio installieren (sudo apt-get install pulseaudio)
5. Pi neu starten
6. Picroft mit Mycroft-Home verbinden

Android

Um Mycroft unter Android nutzen zu können wird ein bereits eingestelltes Mycroft- Gerät, zum Beispiel ein Raspberry Pi benötigt.

1. Android Studio installieren
2. Git-Repo klonen (<https://github.com/MycroftAI/Mycroft-Android>)
3. App auf einem Android Gerät mit Android Studio installieren und starten
4. in General die IP-Adresse des Mycroft-Gerätes (zum Beispiel ein Raspberry Pi) einstellen ("Hey Mycroft, what is your IP address")
5. Befehle über Sprache oder Text eingeben

Pairing

Nach der Installation von Mycroft sollte dieses mit einem Mycroft-Home Account verknüpft werden. Dadurch kann Mycroft API-Keys besser verwalten und zusätzliche Services anbieten. Theoretisch ist ein Betrieb von Mycroft auch ohne einen Account und mit eigenen API-Keys möglich.

Das Pairing erfolgt über einen sechsstelligen Code, der nach dem erstmaligen Starten von Mycroft ausgegeben wird. Dieser muss online dem Konto hinzugefügt werden.

Danach ist die vollständige Nutzung von Mycroft möglich.

Installation von Skills

Skills können nach der Verknüpfung mit einem Mycroft Account einfach über den Mycroft Marketplace installiert werden.

Neben dieser Art der Installation gibt es auch noch den Mycroft Skills Manager, welcher das Herunterladen und Installieren von Skills über die Kommandozeile ermöglicht. Skills können außerdem direkt in den zugehörigen Ordner (/opt/mycroft/skills/) gelegt werden.

Eigene Skills auf Pi:

1. SSH aktivieren (sudo raspi-config → Interfacing Options → SSH)
2. Skill auf Pi kopieren mit: `scp -r path_to_skill/skill_name/pi@ip_address:/opt/mycroft/skills/skill_name.mycroftai/`
3. Pi neustarten

Weitere Systeme

Für Windows und MacOS gibt es momentan noch keine Mycroft Versionen. Für Android gibt es zwar eine Companion-App, diese funktioniert aber erst ab Android SDK24. Die Installation würde hier über Android Studio erfolgen, eine fertige, direkt installierbare, APK gibt es nicht.

Entwicklung von Skills

Skills für Mycroft basieren auf Python, weshalb hier ein gewisses Grundverständnis unabdingbar ist.

Für das Verständnis von Skills in Mycroft sind drei Begriffe besonders wichtig:

- **dialog:** Ein Dialog ist eine Phrase, die von Mycroft ausgesprochen wird, beispielsweise als Antwort auf eine Frage.
- **utterance:** Ein Utterance ist ein Satz der von einem Nutzer gesprochen wird nach dem dieser das Wake-Word gesagt hat.
- **intent:** Ein Intent ist ein identifizierter Aufruf eines Skills durch einen Nutzer. Utterances werden mit existierenden Intents verglichen und dadurch werden die passenden Skills aktiviert.

Für Skills gibt es bereits ein vorgefertigtes Repository, in welchem sich zu einem viele Skills aus der Community und zum anderen auch ein Template für neue Skills finden lassen (<https://github.com/MycroftAI/mycroft-skills>).

Die Struktur der Skill-Repositories ist relativ strikt und setzt sich wie folgt zusammen:

- **dialog-Ordner:** Hier landen alle Dialog-Dateien in den verschiedenen Sprachen, die der Skill ausgeben können soll. Dialog-Dateien enden mit `.dialog`
- **vocab-Ordner:** Hier landen alle Intent-Dateien in den verschiedenen Sprachen, auf die der Skill reagieren soll. Intent-Dateien enden mit `.intent`. Intents können auch reguläre Ausdrücke enthalten. Außerdem finden sich hier auch `.voc` Dateien. Diese definieren bestimmte Vokabeln, die in einem Intent vorkommen können, also beispielsweise Synonyme die vom Nutzer äquivalent verwendet werden.
- **regex-Ordner:** Dieser Ordner ist optional und enthält reguläre Ausdrücke, welche in Intents vorkommen können. Regex-Dateien enden mit `.rx`
- **init.py:** In dieser Python-Datei werden die Aufrufe der Intents behandelt, verarbeitet und die Ergebnisse an den Nutzer zurückgegeben.

Innerhalb der Python-Datei werden die Intents mit Dekoratoren auf Funktionen abgebildet. Vor einer Funktion die einen Intent verarbeitet muss deshalb immer der dazugehörige Dekorator stehen. Beispiel:

```
@intent_file_handler("hello.world.intent")
def handle_hello_world(self):
    return self.speak_dialog("hello.world") # hello.world ist eine .dialog Datei
```

Eine ausführlichere Erklärung über die notwendigen Python-Packages und das Vorgehen bei der Entwicklung von Skills findet sich unter <https://mycroft.ai/documentation/skills/introduction-developing-skills/>. Außerdem kann man sich sehr gut an den bereits existierenden Skills orientieren.

Beispiel Skill

Der "health care" skill ist ein Beispiel Skill um einige Funktionen von Mycroft vorzustellen. Mit diesem ist es möglich Patienten über ihren Namen in eine Datenstruktur zu speichern. Außerdem können über den Skill jedem Patienten eine Reihe von Herzraten zugeordnet und ausgelesen werden.

Nutzung

nutze Ausdrücke wie diese ...

Patienten hinzufügen:

- "create *name lastname*"
- "Add patient *name lastname*"

Patienten löschen:

- "remove *name lastname*"
- "delete patient *name lastname*"

Herzraten hinzufügen:

- "add heartrate *heartrate* to *name lastname*"
- "create a heartrate of *heartrate* for patient *name lastname*"

durchschnittliche Herzrate von einem Patienten:

- "show the average heartrate of *name lastname*"

die Datenstruktur löschen:

- "delete data"

Um alle Befehle nutzen zu können muss natürlich erst ein Patient hinzugefügt werden. Ein valider Name muss immer aus einem Vor- und Nachname bestehen, dies teilt einem der Skill aber auch mit. Falls die Datenstruktur gelöscht werden soll, fragt der Skill nach einer Bestätigung. Mitunter muss man ein paar Sekunden warten, bis man die Antwort gibt, in der CLI sieht man wann Mycroft wieder zuhört. Als Bestätigung wird nur "yes" akzeptiert, alles andere bricht den Prozess ab.