Customizable Speech Recognition for Internet of Things

Mohammad Hasanzadeh Mofrad[1], Omid Kashefi[2], and Daniel Mosse[1]

[1] Department of Computer Science, School of Computing and Information, University of Pittsburgh

[2] Intelligent Systems Program, School of Computing and Information, University of Pittsburgh

Motivations and Contributions

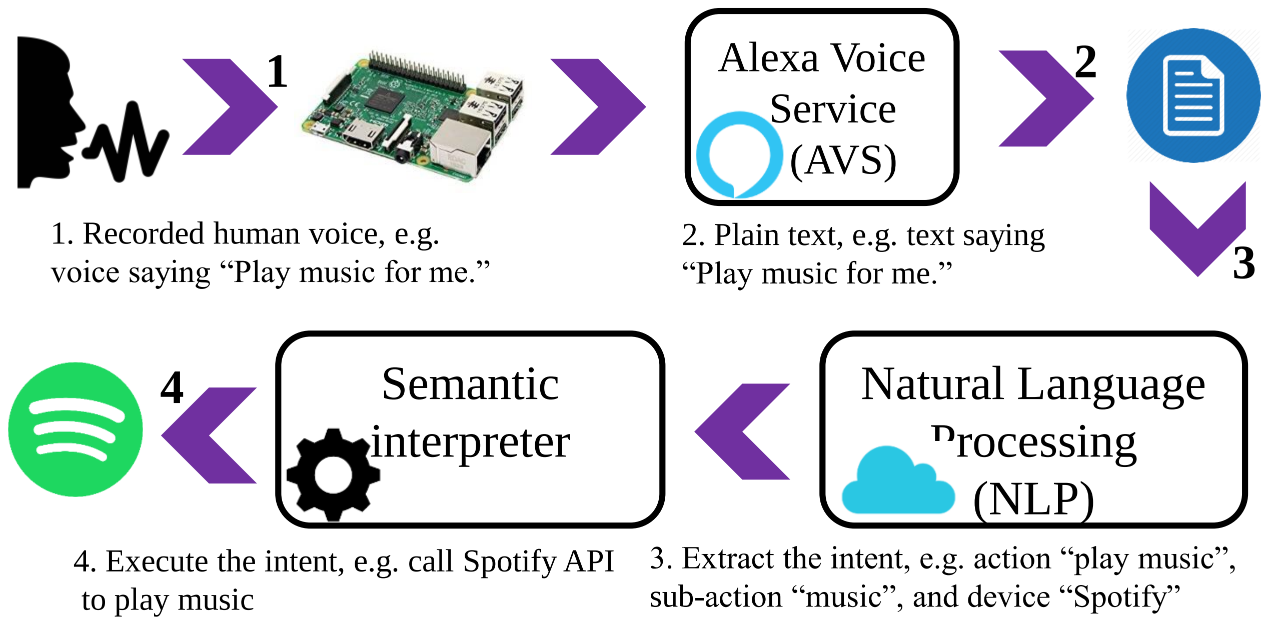
- Internet of Things (IoT) devices are all around:
  - **Smart Speakers**: Amazon Echo & Google Home
  - **Heating & Cooling**: Nest Learning Thermostat & Ecobee4
  - **Smart Locks & Doors**: August Smart Lock & SkyBell HD
  - **Smart Lightening**: Philips Hue & Lifx Color
  - **Cleaning**: iRobot Roomba, Neato Botvac
  - **Smart Shades**: Lutron's Serena Shades
  - **Security Cameras**: Amazon Cloud Cam & Nest Protect

- Ways to communicate with IoT devices:
  - **Graphical User Interface (GUI)**
    - ✓ Pushing buttons
    - ✓ Clicking on icons
  - **Speech Interfaces**
    - ✓ Just talking to the device which is more intuitive and efficient
    - ✓ Speech processing and natural language processing empowers these interfaces
- Current smart home devices **limitations**:
  - They are not customizable and customers cannot extend them to have their customized voice commands
  - They are designed to serve the company's specific purposes such as making it easy to buy products in Amazon Echo which contributes to Amazon's profit
- **Contribution** are:
  - Creating a cheap, customizable and programmable IoT infrastructure using Raspberry Pi.
  - Using free or open source speech recognition APIs like Alexa Voice Service (AVS) or Google Cloud Speech API for converting speech to text
  - Writing a customized language models on top of the text output of AVS to capture an accurate intent of the user
  - Writing a wrapper that convert the final extracted intent to actions and apply them to the IoT device

Customizable Speech Recognition Model

- The proposed model consists of the following components:

1. **Recorder component (Acoustic Model)**
   - ✓ We use a small USB microphone that captures the incoming voice
2. **Speech to text component (Acoustic Model)**
   - ✓ We use Alexa Voice Service (AVS)
3. **Text to intent component (Language Model)**
   - ✓ We use Natural Language Processing (NLP) to create a language model that returns the intent of the extracted sentence.
4. **Intent to action component (Language Model)**
   - ✓ Execute the inferred intent on the target device



1. Recorded human voice, e.g. voice saying "Play music for me."

2. Plain text, e.g. text saying "Play music for me."

3. Extract the intent, e.g. action "play music", sub-action "music", and device "Spotify"

4. Execute the intent, e.g. call Spotify API to play music

Customizable Speech Recognition Model

- The **Recorder component** has two parts:
  - A USB microphone mounted on the Raspberry Pi and records the voice
  - A program written in Python which send the recorded voice to AVS and store the transcribed text
- The **Speech to text component**:
  - We use the AVS as the choice of speech to text component because it is already trained with tons of voice data by Amazon
  - We use the AVS device SDK as a tool to transcribe the recorded voice by the recorder component
  - The text to speech processing part is done in Amazon's Cloud and not the Raspberry Pi so we are saving energy for the computation that is being done in the Cloud
  - The recorded voice is discarded after being transcribed in order to save space and memory; the extracted text is stored for further use of the application

Language Model (Text to Intent and Intent to Action Components)

- Noisy Channel Model
  - Find intended sentence, given a sentence where the words are scrambled
  - E.g. "I've ordered [two/too/to] of it."
  - **Approximation**

$$P(sentence \,|\, acoustic) = \underbrace{P(acoustic|sentence)}_{\text{Likelihood}} \underbrace{P(sentence)}_{\text{Prior}}$$

  - **Likelihood**

SR/Acoustic Model

"I've ordered [two/too/to] of it."

  - **Prior**

I've ordered [two/too/to] of it.

$$\frac{Language\ Model}{P(w_i^n) = \prod_{k-1}^{n} P(w_k|w_{k-N-1}^{k-1})}$$

I've ordered two of it.

- **Text to intent component** extract the intent from the constructed sentence
- **Intent to action component** execute the inferred intent on the device

Customized prototype

- The core hardware components of the proposed customized IoT prototype costs us $68.42 and consists of the following building blocks:
  - **Raspberry Pi 3 Model B Motherboard** – Quad core, 1GB RAM, Wireless 802.11, and Bluetooth 4.0 – $35.80
  - **Kinobo USB 2.0 Mini Microphone** – $4.65
  - **Samsung 64GB Micro SD Card** – $19.99
  - **Raspberry Pi Case** – $7.98
- Other hardware :
  - A 2.5A power adaptor (Your mobile adaptor)
  - A monitor and HDMI cableA USB keyboard and mouse
  - Wired/Bluetooth speaker
- The software components of the prototype are:
  - Language model developed using example sentences
  - AVS API which is free
  - Raspbian OS
  - Python 3.5