Voice Interfaces on a Budget: Building Real-Time Speech Recognition on Low-Cost Hardware

Pete Warden

CEO

Useful Sensors

USEFUL SENSORS

What Is This Talk About?

- Voice interfaces in the past:
  - Cost $$$
  - Only available to big tech co's
  - Required specialists
  - Took years to build
- Voice interfaces now:
  - Open source
  - Available to everyone
  - Usable by any software engineer

What You'll Leave With

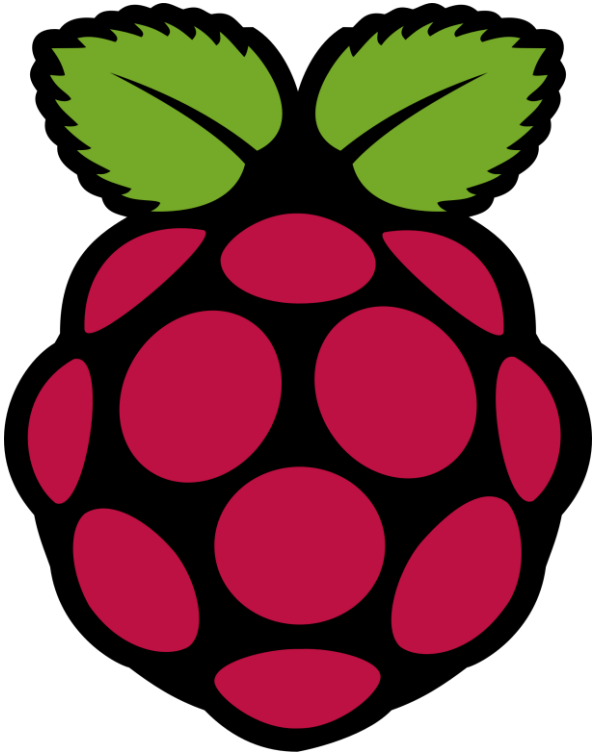
- How to build a simple voice app
  - Running on low-cost hardware (Raspberry Pi)
  - Without a cloud API or network connectivity
  - Using open-source, freely available software and models

Does Anyone Use Voice Interfaces?

- Siri, Alexa, Ok Google?
  - No, except for timers and baby shark
- However, a thought experiment:
  - You're next to your significant other on a couch
  - Do you text them to decide what show to watch?
  - So, we do like voice interfaces, but the current ones aren't good enough

What Do You Need to Start?

- Cortex A CPU or equivalent
  - MCUs soon, hopefully
- Open source frameworks
- Open weights speech recognition models



Speech Recognition Models

- OpenAI's Whisper
  - First production-quality open weights ASR model
  - Smallest version is 40 million parameters
  - Can run on APUs, but hard to get real-time
  - Always processes 30 seconds of audio at once, very wasteful for interactive use cases
- Useful Sensors' Moonshine
  - Open weights
  - Achieves same accuracy as Whisper for tiny and base models
  - Smallest version is 26 million parameters
  - Able to run well on most modern APUs
  - Flexible input window, so you only compute what you need

Moonshine Options

- Tiny version:
  - 26 million parameters
  - Word error rate of 4.51%
- Base version:
  - 52 million parameters
  - Word error rate of 3.29%
- Tutorial uses Tiny

- Many frameworks supported:
  - PyTorch
  - Keras
  - TensorFlow
  - ONNX
- We're using ONNX
- Quantized versions available:
  - 26 MB / 56 MB file sizes
  - 1.6x faster than float

Tutorial

Now I'll show you how to run an interactive speech application using Moonshine on a Raspberry Pi 5.

Based on material from my Stanford EE292D Edge AI course:

https://github.com/ee292d/labs/tree/main/lab4

You'll need a Pi 5, some way to connect to it, and a USB microphone.

Live Coding Demo

Next Steps

- How can you take action based on speech?
  - Plain old string matching can work for simple uses
  - Recognizing a natural speaking style needs speech to intent
  - Still a research problem
- https://github.com/AIWintermuteAI/Speech-to-Intent-Micro

- What about text to speech?
  - Speakers are a cheaper alternative to displays
  - PiperTTS is very efficient, runs on Pi's, and sounds good
  - Hyper-realistic models are emerging, but they use a lot of resources, won't work on a Pi (yet)

Conclusion

- It’s never been easier to build a voice-driven product
- It’s still early days for voice, don’t write it off because Siri isn’t popular
- You’ve got this!

Resources

Whisper:
https://openai.com/index/whisper/

Moonshine:
https://github.com/usefulsensors/moonshine

PiperTTS: https://piper.ttstool.com/

Speech-to-intent Micro:
https://github.com/AIWintermuteAI/Speech-to-Intent-Micro

EE292D Tutorial:
https://github.com/ee292d/labs/tree/main/lab4

Me: pete@usefulsensors.com