babblelabs

Ultra-Low-Power Command Recognition for Ubiquitous Devices

DAC Talk 7.1

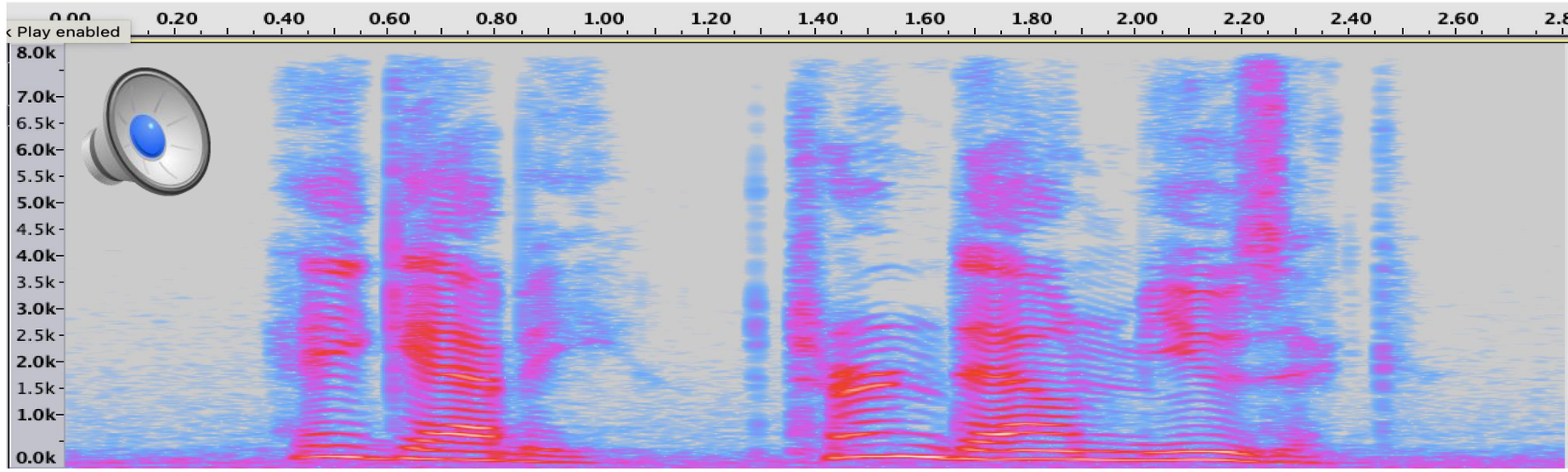
Chris Rowen

BabbleLabs Inc.

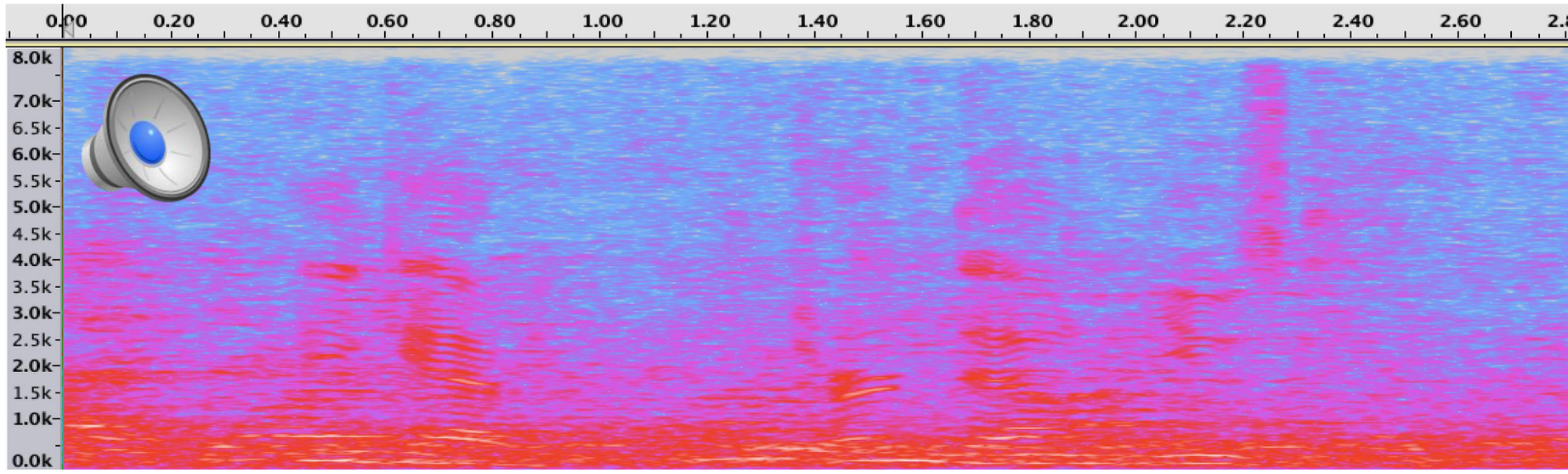
June, 2019

The Noisy Speech Problem

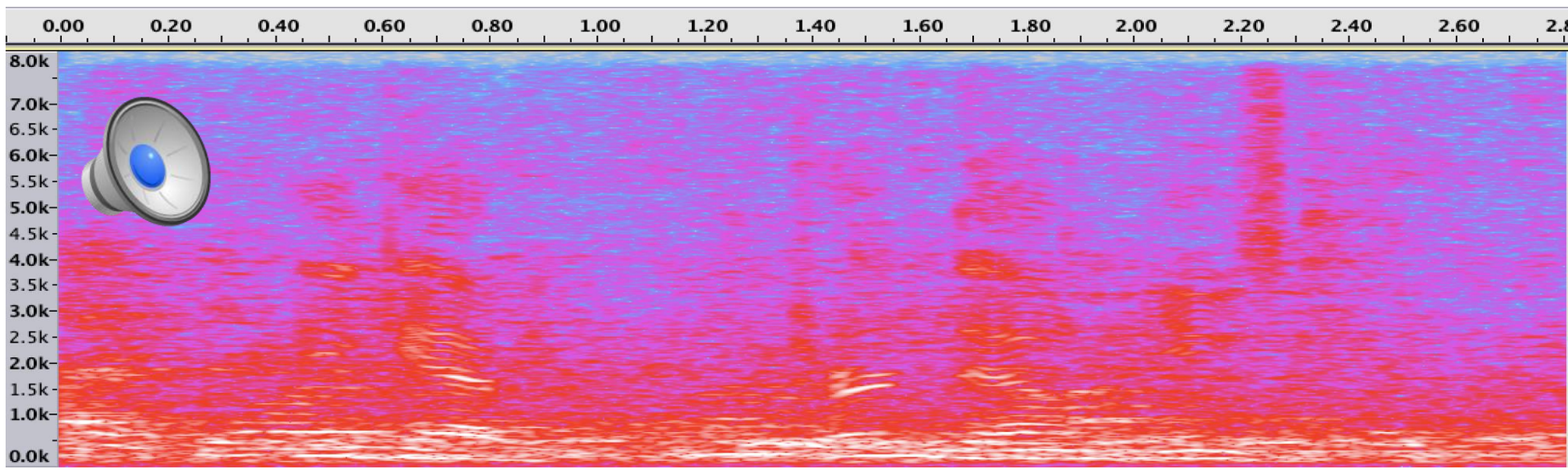
>20dB Signal to Noise Ratio (SNR)



0dB SNR



-10dB SNR

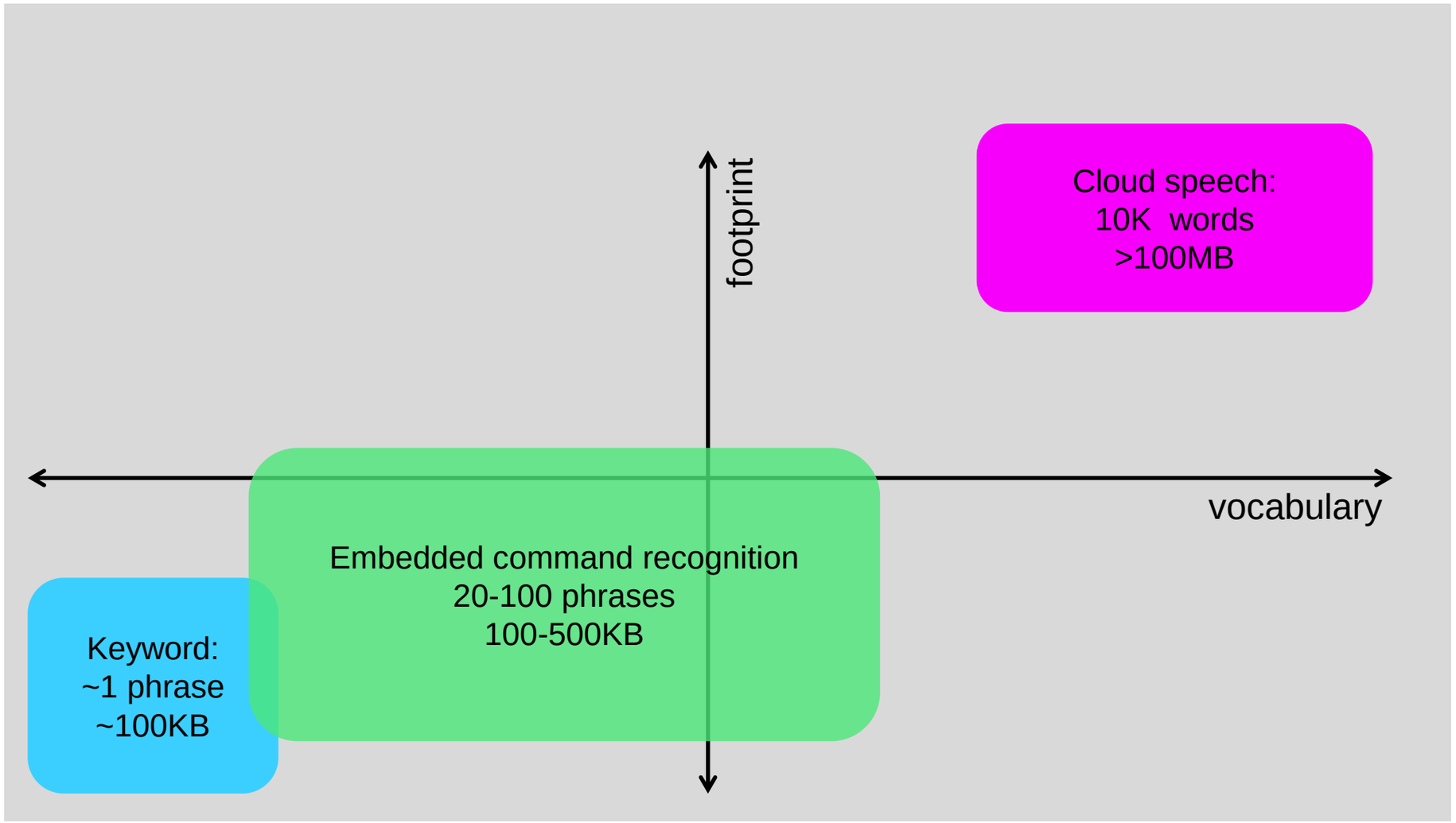


babblelabs

Command Recognition System

Goals:

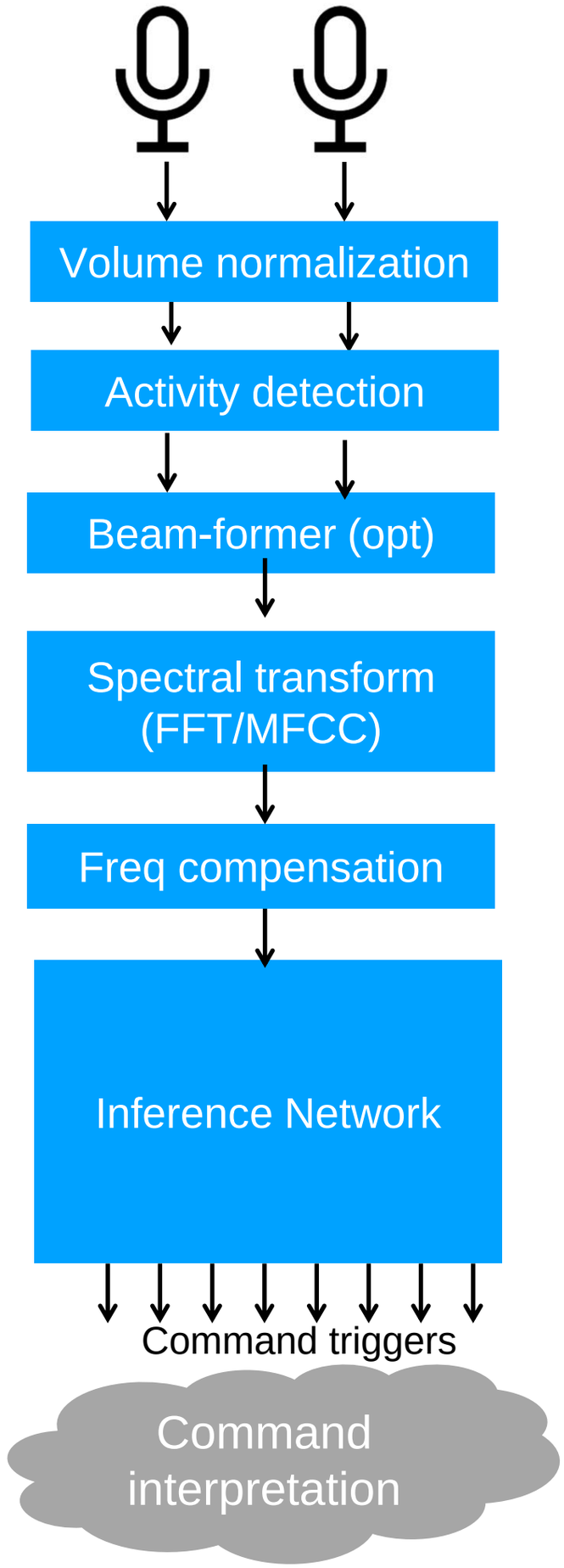
- Tiny footprint in memory, compute, power
- 3-5x more robust to noise
- Span range of command set size: up to about 100 phrases
- Rapid vocabulary training
- Support both trigger-phrase prefix and non-trigger systems



babblelabs

The Core Functions

- Optional multi-microphone front-end extracts multiple candidate beams via cross correlation to find speech and noise sources
- Optional frequency domain compensation for microphone characteristics
- Spectral domain processing (FFT/MFCC)
- Keep inference model as small as possible for necessary classification capacity
  - Convolutions with minimal full-connected back-end
  - Cascaded Inception/SqueezeNet-like small separable convolutions: 3x1, 1x3, 1x1
  - Minimal full-connected back-end on pooled results
  - Medium deep: ~20 layers
  - Scale network with
    - Utterance length adaptation
    - Accuracy vs. cost tradeoff knob
- Implementations in fp32, int16, int8

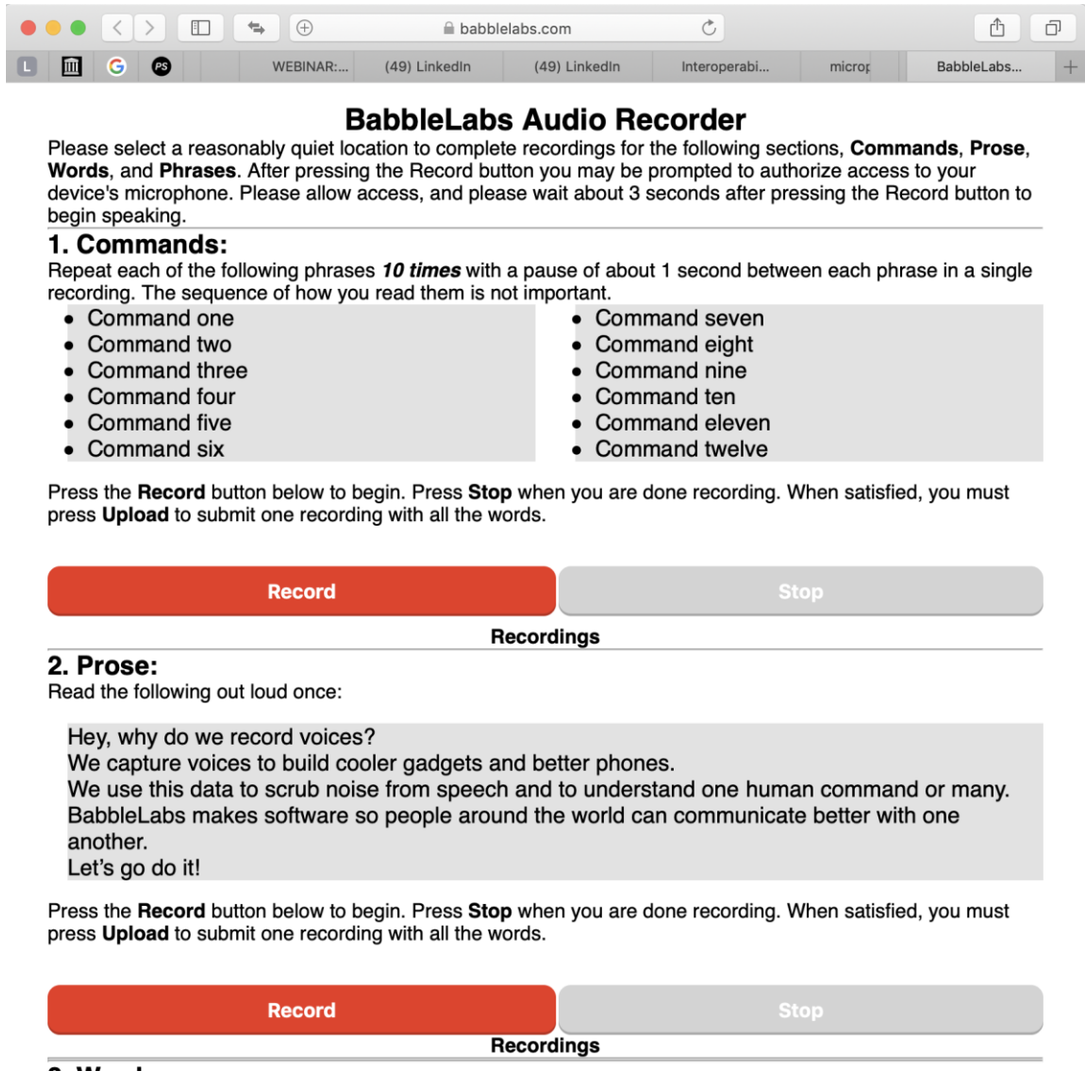


babblelabs

Training for Commands

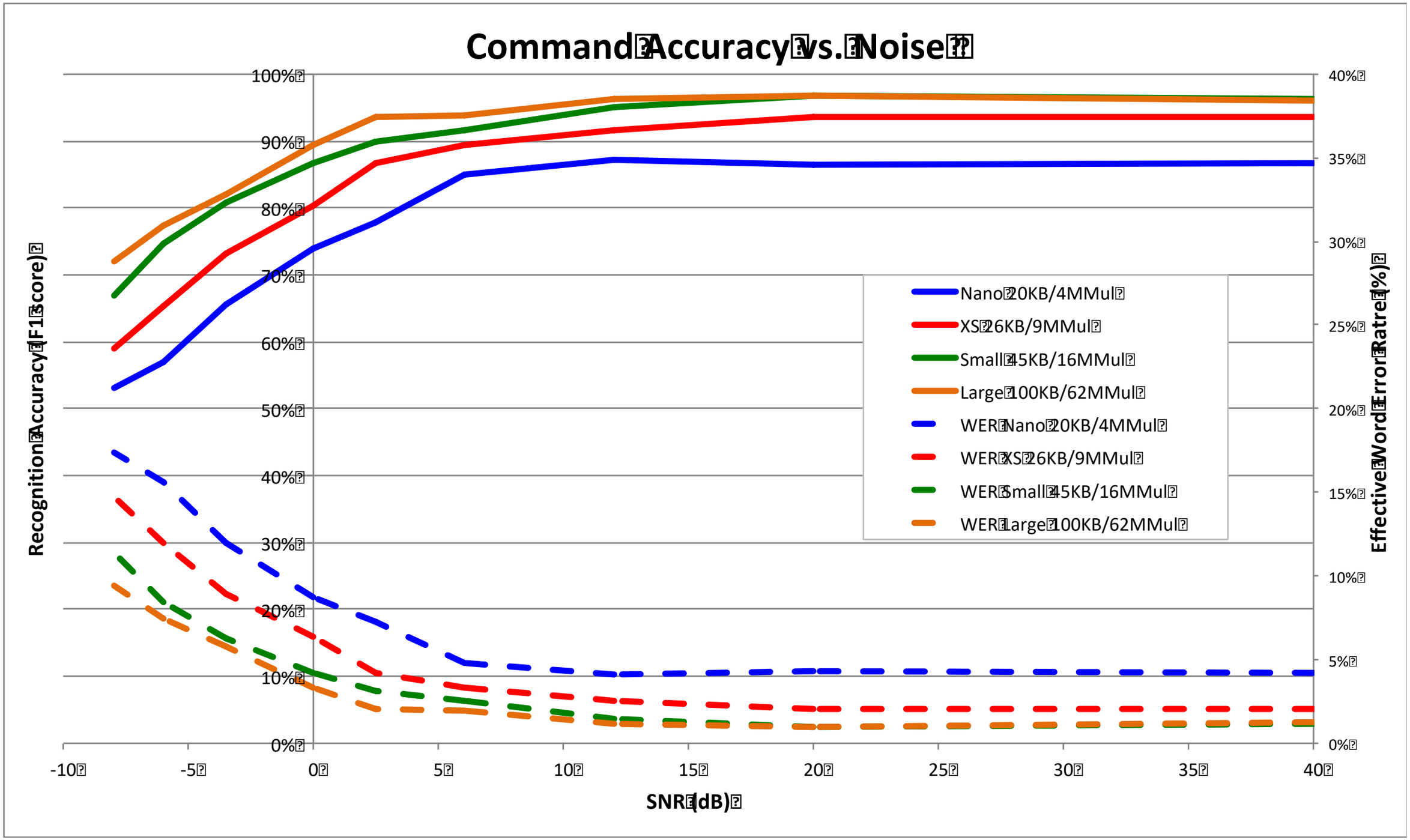
- Direct training for specific command vocabulary requires efficient training corpus generation
- Automated system for data collection and scrubbing:
  - Browser-based capture interface
  - Crowd-sourced workers speak script of target and non-target phrases
  - Cleaning, segmentation and labeling using cloud ASR
- Multi-dimensional speech augmentation for added diversity
- Leverage BabbleLabs unique noise corpus: 15,000 hours, mostly non-stationary
- Two week turn-around from command specification to installed binary

Raw target utterances	11,000
Total raw target+non-target speech per vocabulary	50,000s
Unique augmented utterances	1M
Total training utterances	100M



babblelabs

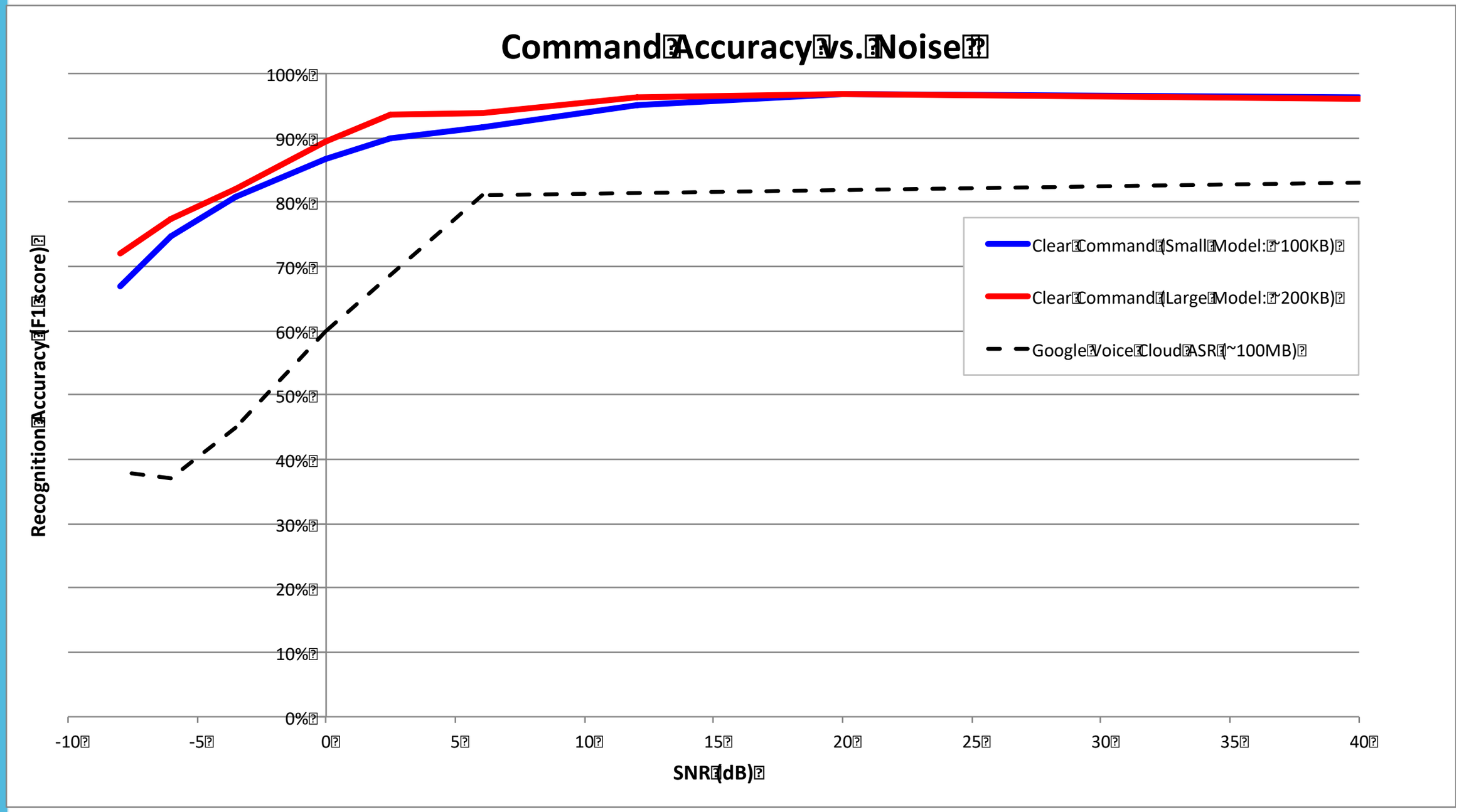
Command Recognition Results



babblelabs

Clear Command Speech Recognition

Noise-immune hands-free UI commands



babblelabs

Example Command Set

- BabbleLabs Reference Command Set
  - 35 common function commands
  - 80 phrases (2-5 words each)

Command ID	Example: 80 phrases for 35 commands		
0	turn on the TV	turn on the television	
1	turn off the TV	turn off the television	
2	turn up the TV	turn up the television	
3	turn down the TV	turn down the television	
4	turn on the AC	turn on the air conditioner	turn on the air conditioning
5	turn off the AC	turn off the air conditioner	turn off the air conditioning
6	turn up the AC	turn up the air conditioner	turn up the air conditioning
7	turn down the AC	turn down the air conditioner	turn down the air conditioning
8	turn on the lights		
9	turn off the lights		
10	turn up the lights		
11	turn down the lights		
12	turn on music	turn on the music	turn on the sound
13	turn off the music	turn off music	turn off the sound
14	turn up music	turn up the music	turn up the sound
15	turn down music	turn down the music	turn down the sound
16	turn on the heat		
17	turn off the heat		
18	turn up the heat		
19	turn down the heat		
20	open menu	open the menu	show the menu
21	open music	show music	
22	open maps	show maps	
23	open Facebook	show Facebook	
24	open Twitter	show Twitter	
25	open Instagram	show Instagram	
26	open browser	open a browser	open the browser
27	open weather	show weather	
28	open messages	show messages	
29	open photos		
30	open WeChat	show WeChat	
31	what time is it?	what's the time?	
32	what's the weather?		
33	answer the phone	answer phone	answer telephone
34	show the news	open the news	show news

Implementation on Tiny Hardware

- Network developed and trained in TensorFlow
- Custom quantizer directly generates C data structures
- Scalable C implementation works across network configuration space
- Leverages DNN or DSP libraries where available

Current example platforms		Compute requirements for reference command set (“small model”)
NXP: i.MX RT1060	ARM Cortex M7 MCU	25MHz
Ambiq: Apollo 3 Blue	ARM Cortex M4 MCU	45MHz
Cadence	Tensilica HiFi Fusion F1 DSP	12.5MHz

Memory footprint : reference command set on NXP i.MX RT1060 “small model”	
Code	5KB
Model	45KB
Memory Buffers	50KB
Total RAM +flash	100KB

Low power implementations

Core power example – reference command set	
Energy requirements (Fusion F1 in TSMC 16FF 9T):	18 μW/MHz
Core frequency	12.5MHz
Core computer power	225μW
Other power including local memory – est.	150μW
Leakage	100μW
Total Power	**>500μW**

Example Target: NXP i.MX RT MCU-based AVS solution kit

babblelabs

Implications

- Command recognition plays an important role in speech-powered systems:
  - More noise-robust
  - More private
  - Less sensitive to network outage
  - Lower energy
- Command recognition complements or replaces heavyweight continuous speech recognition
- Careful co-design of
  - Signal processing stack
  - Networks
  - Implementation
  - Data collection and training system

  enables rich functionality in a tiny footprint space
- Further refinement not just possible but likely:
  - Leverage hardware for energy-minimized DNN inference engines
  - Pushing the envelope on vocabulary richness
  - New uses for rich word spotting

s p e a k y o u r m i n d