

PROGRESS 4: MOBILE WEB APP

BIRD SONG IDENTIFICATION

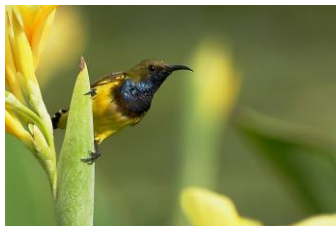
Thanyaporn Phinthuphan
21 Mar 2019

Outline

- Recap
- Methodology
- Preliminary Result
- Next step

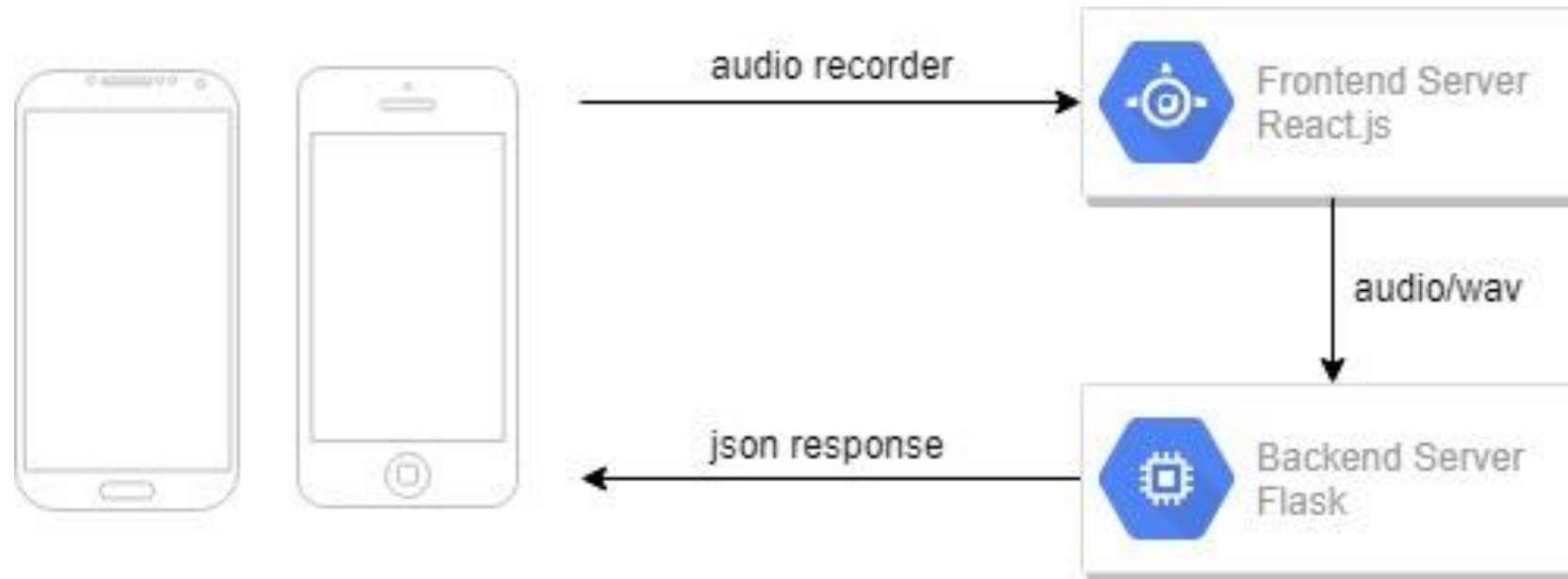
Recap

- trained on 11 bird sound class 1,847 file
- first CNN model cannot reduce batch loss → reduce learning rate
- want to write UI for mobile web app



Methodology

Web architecture



Methodology

Frontend server

- build the frontend with **react**
- using starter code from

[https://github.com/hackingbeauty/how to write a single page application course](https://github.com/hackingbeauty/how-to-write-a-single-page-application-course)



Fixing React-Mic issues.

**Mark M.
Muskardin**
hackingbeauty

Methodology

Frontend server

- record page : using react-mic
- history page : using localforage cache
- replay page : show bird song identification result

Methodology

Frontend server

- problem: audio recorder of starter code using react-mic
 - audio/webm → not compact to model process
 - cannot record on iphone (safari)

Methodology

Frontend server

- change recorder library to [RecordRTC](#)
- using StereoAudioRecorder → audio/wav
- axios post audio/wav via blob (FormData)

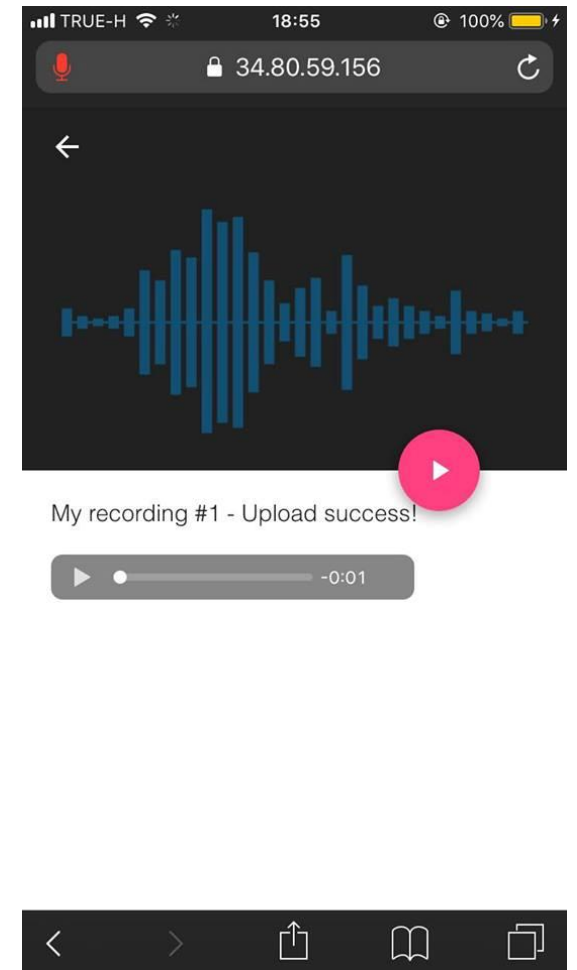
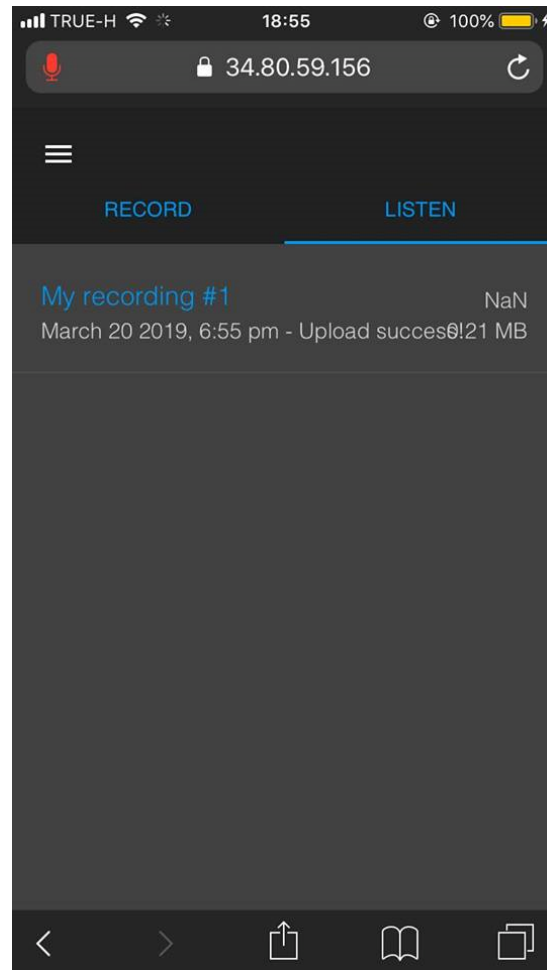
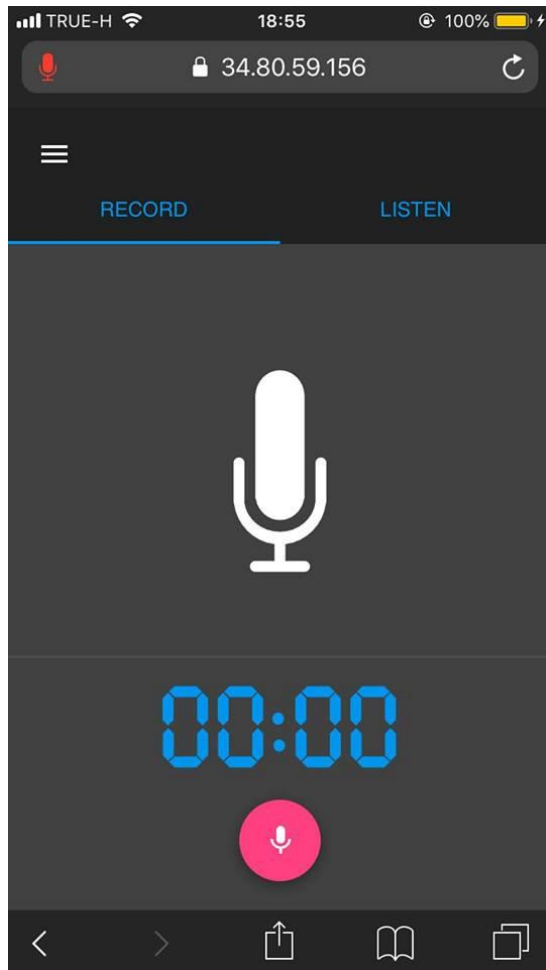
Methodology

Backend server

- build the backend with `flask`
- test application process on the old model
- return json contained bird name for tested song via API

Preliminary result

- running on <https://34.80.59.156:3000> (via google cloud)
- ✓ sound recorder on mobile phone (ios/android)
- ✓ view recording history
- ✓ replay the sounds
- ✓ get result from backend API



Next Step

- find the best model on 11 class: good AUC / recall / prediction time
- define new scope and test
- test with real data (manual record / data from other website)
- test in the park on mobile phone

Q&A