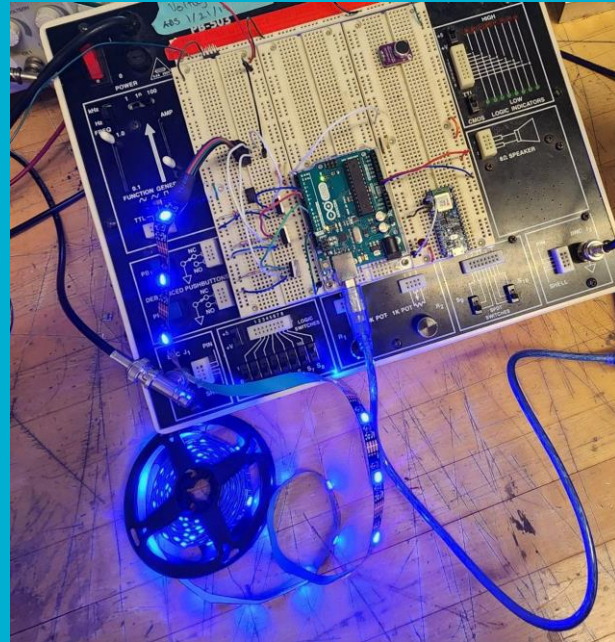# Voice Activated LED Strip using ML

PY371 Final Project
Ivanna Morales-Roman

Abstract: This device uses machine learning to run inference on active audio input in order to recognize commands to control an LED strip. The ML will occur on one microprocessor, which will signal a second board to change the LEDs accordingly

A Voice Command LED lights Using Machine Learning

Using Edge Impulse development platform: I trained an audio classification model to recognize spoken color commands like "blue" with audio clips of myself saying the words in different settings.

Step 1:

➔ Microprocessor actively **listens** to its surroundings

➔ **recognizes** a spoken color command
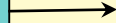
➔ **sends a signal** to the second microprocessor

Step 2:

➔ Second microprocessor will **interpret the signal**

➔ **activates the LED** strip according to the command received.

Block Diagram

Step 1:

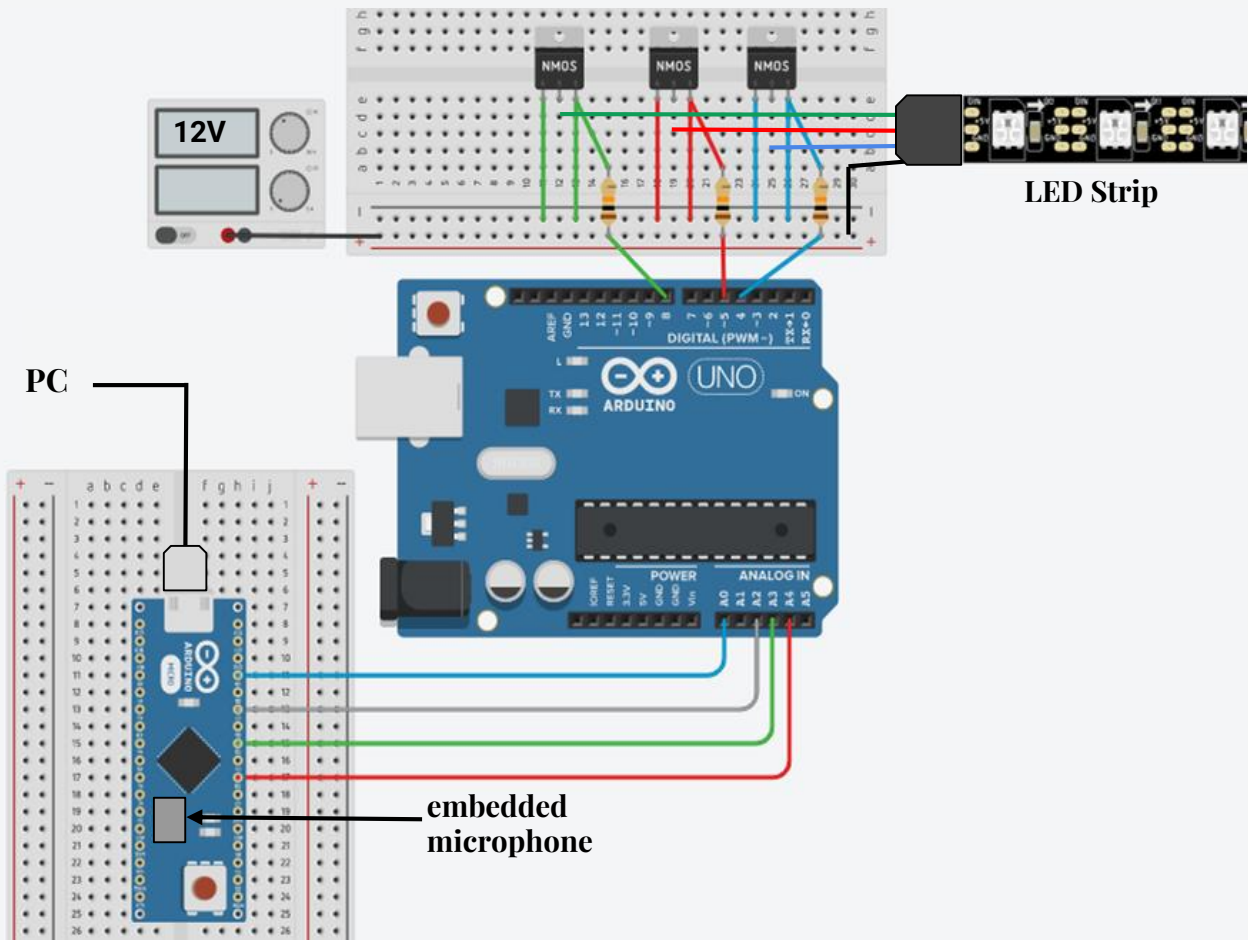**Step 2:**

```
Embedded mic  →  Arduino Nano for ML voice recognition  →  Arduino UNO for LED control  →  LED Strip
```
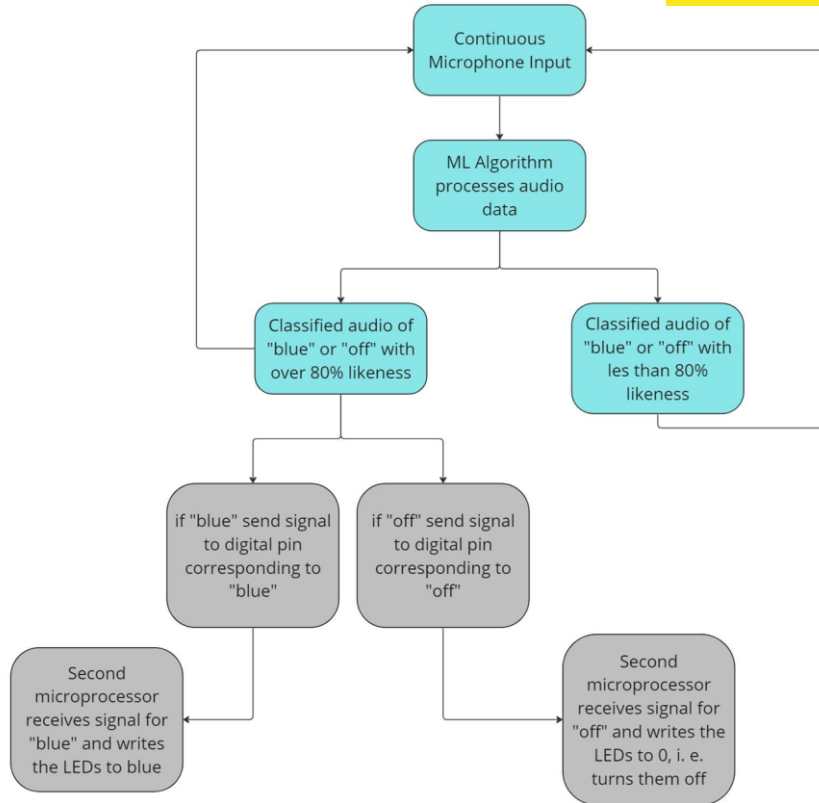
12V

LED Strip

PC

embedded
microphone

Software Flowchart

Continuous Microphone Input → **ML Algorithm processes audio data**

- Classified audio of "blue" or "off" with over 80% likeness
- Classified audio of "blue" or "off" with les than 80% likeness

- if "blue" send signal to digital pin corresponding to "blue"
- if "off" send signal to digital pin corresponding to "off"

- Second microprocessor receives signal for "blue" and writes the LEDs to blue
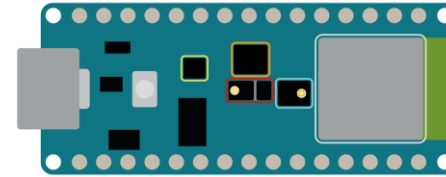- Second microprocessor receives signal for "off" and writes the LEDs to 0, i. e. turns them off

In blue: First microprocessor (arduino nano) with 1MB CPU flash memory and embedded microphone

NANO 33 BLE SENSE

- ◆ Color, brightness, proximity and gesture sensor
- ◆ Digital microphone
- ◆ Motion, vibration and orientation sensor
- ◆ Temperature, humidity and pressure sensor
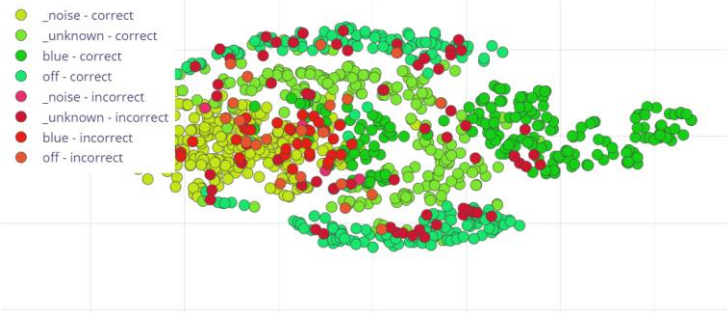- ◆ Arm Cortex-M4 microcontroller and BLE module

In red: Second microprocessor (arduino UNO) with 32 kB flash memory

miro

Training Statistics On The Testing Set

Confusion matrix (validation set)

	_NOISE	_UNKNOWN	BLUE	OFF
_NOISE	**96.5%**	0.9%	1.8%	0.9%
_UNKNOWN	2.6%	**82.5%**	5.7%	9.2%
BLUE	8.9%	0%	**91.1%**	0%
OFF	2.3%	3.9%	1.2%	**92.7%**
F1 SCORE	0.91	0.88	0.91	0.92

Data explorer (full training set) ⑦

- _noise - correct
- _unknown - correct
- blue - correct
- off - correct
- _noise - incorrect
- _unknown - incorrect
- blue - incorrect
- off - incorrect

Next Improvements:

★ Add more colors for the device to recognize
★ Tweak parameters/ add layers to the model for better recognition
★ Curate more audio data to increase accuracy + get it to only recognize my voice
    ○ For this prototype I recorded 50 samples and augmented them into 1500 total for the command "blue"
    ○ For "off" i used a large open source dataset from google

More information

- Arduino Nano specsheet https://store-usa.arduino.cc/products/arduino-nano-33-ble-sense
- Edge Impulse Classification (keras) model documentation https://docs.edgeimpulse.com/docs/edge-impulse-studio/learning-blocks/classification
- Edge Impulse feature extraction: https://docs.edgeimpulse.com/docs/edge-impulse-studio/processing-blocks/audio-mfcc
- https://studio.edgeimpulse.com/studio/205783/learning/keras/7