Testing

Sdmay22-10
Frankie Mago, Hailey Lucas, Christian W, James Gossling

Unit Testing

Machine Learning:
- Keyword recognition testing: How accurate the neural network identifies a keyword.
    - Keyword spotting accuracy first measured using randomly selected test set data on Edge Impulse to give theoretical accuracy.

Arduino Software:
- Microphone sensor code testing: Use arduino IDE and serial monitor to check that sampling data is correctly sampled/stored.
- Motor(s) code testing: visually inspect that motor working as it should when the arduino runs it
- EdgeImpulse library: Using arduino IDE and serial monitor to check that the model library is working, and returns a result within time limit.

Interface Testing

Interfaces:
Adafruit microphone library
Stepper motor library
EdgeImpulse library

The microphone unit will interact with the edge impulse library unit, we will test that the microphone delivers the proper data in the proper format to the edge impulse unit. The edge impulse unit will interact with the motor unit, we will test that the edge impulse library's return value is properly used to make the motor move or not move. This will be done visually and with the serial monitor on the arduino IDE.

Integration Testing

Machine Learning:
- Keyword recognition testing: How accurate the neural network identifies a keyword.
    - Will then test keyword spotting accuracy using real time data (spoken words) once deployed onto a microcontroller.

-Motor calibration and positioning: Ensure that the motor does not over/under rotate the deadbolt axel to make sure that the motor does not burn out and/or wear down gears.

-Simple Keyword test: When the keyword is spoken does the lock open.

System Testing

We will be saying words and visually confirming that the system acts appropriately to a keyword/non keyword. All the interface and integration tests will be applicable to the system test.

Regression Testing

After major implementation changes rerun system tests. If we see multiple of these large implementations in the future make a “test rig” to ensure functionality.

Acceptance Testing

Whole System:
- From the spoken keyword, the system accurately detects and identifies the keyword and opens the lock within a set amount of time.
    - Measure the time it takes from the spoken phrase until the lock finishes opening.

Machine Learning:
-Keyword Recognition Testing: How accurately the neural network identifies a keyword and opens the lock.

- Keyword spotting accuracy first measured using randomly selected test set data on Edge Impulse to give theoretical accuracy.
    - Will need an accuracy of >x%
- Will then test keyword spotting accuracy using real time data (spoken words) once deployed to a microcontroller with a lock system attached.
    - Will need an accuracy of >x%

Physical Testing:

-Endurance testing: make sure the lock operates after 1,000 lock cycles. This can be done by writing code that will cycle the lock.

-Lock Testing: Test to ensure the lock is able to open and can accurately and quickly. Once given the command to open the lock needs to open constantly. This can be do simultaneously to the endurance testing

Security Testing

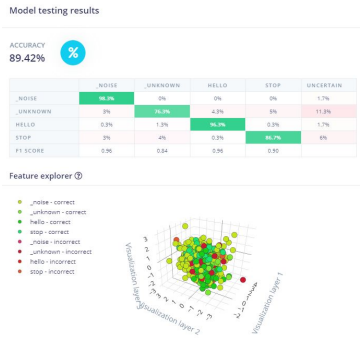
Test for false positives in our machine learning algorithm:

-Try and activate a keyword with someone who sounds similar

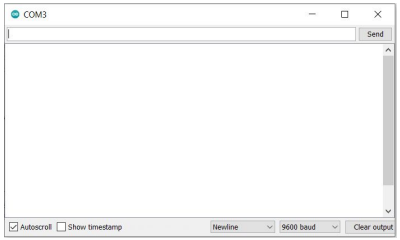
-Try and use a recording of someone's spoken keyword

Results

Example of a figure we can include to display model accuracy:



Edge Impulse has this and more graphs to display model characteristics. They will be used to help visualize machine learning testing results.



Tests involving the arduino and any hardware will be either inspected visually or will be inspected using the serial monitor (above) in the arduino IDE

Summary:
When a chosen user says the password, the system will always lock/unlock for that person using that word in the specified amount of time.
When anyone or any other sound besides the user saying the password is picked up, the system will not lock/unlock the door.