# Car Engine Audio Classifier

**Team Members:** Rayden Dodd, Ryan Hua, Valeria Pozo, Dylan Green, Austin Hall
**Mentor:** Dr. Daniel Connors | **Sponsors:** Scott Patterson, Ethan Brooks
**SME:** Dr. Sook Shin Ha

## Overview

- Proof of Concept
- Can an engine's make be identified from its sound?
- Applications in signal/audio processing
- Will be used in military applications
- Detecting submarines while underwater

## Objectives

A US Navy submarine's mission is to remain as silent as possible to complete their objectives. Sailors must continuously measure sound output within the submarine. Using computer assistance for this task would greatly benefit the boat's safety. The goal is to apply machine learning techniques to identify car engines as a proof of concept.

## Requirements

- Program has minimum accuracy of 80%
- Constantly listens to its environment
- System only activates when engine sound is detected
- Uses machine learning techniques to determine car brand
- Displays top 3 predicted car brands in GUI
- Engine identification is complete within 5 minutes
- Program must run completely offline

## Acknowledgements

Our team would like to thank the following people for their support throughout the project:
- Dr. Daniel Connors
- Dr. Sook Shin Ha
- Scott Patterson
- Ethan Brooks

## Solution

- Engine classifier program written in Python
- Can record audio from microphone or read from file
- Mel Frequency Cepstral Coefficients (MFCCs) extracted from audio
- MFCCs fed into to binary classifier for engine detection
- Deep Neural Network used to classify audio
- Top 3 predicted car brands shown in GUI

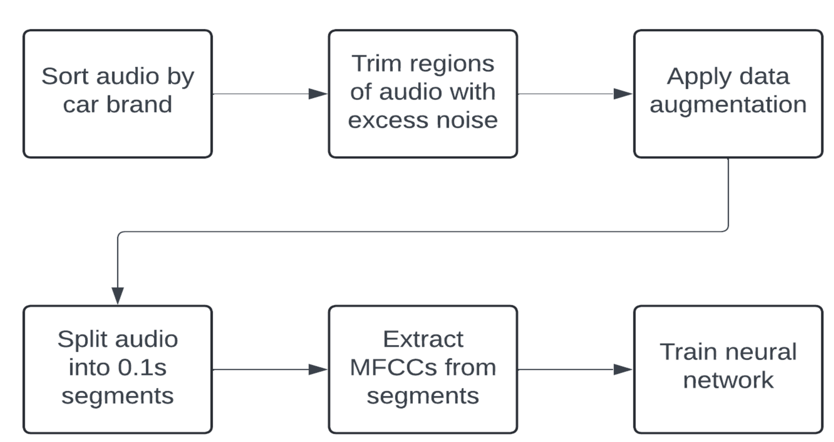


Fig. 1: Neural network training process

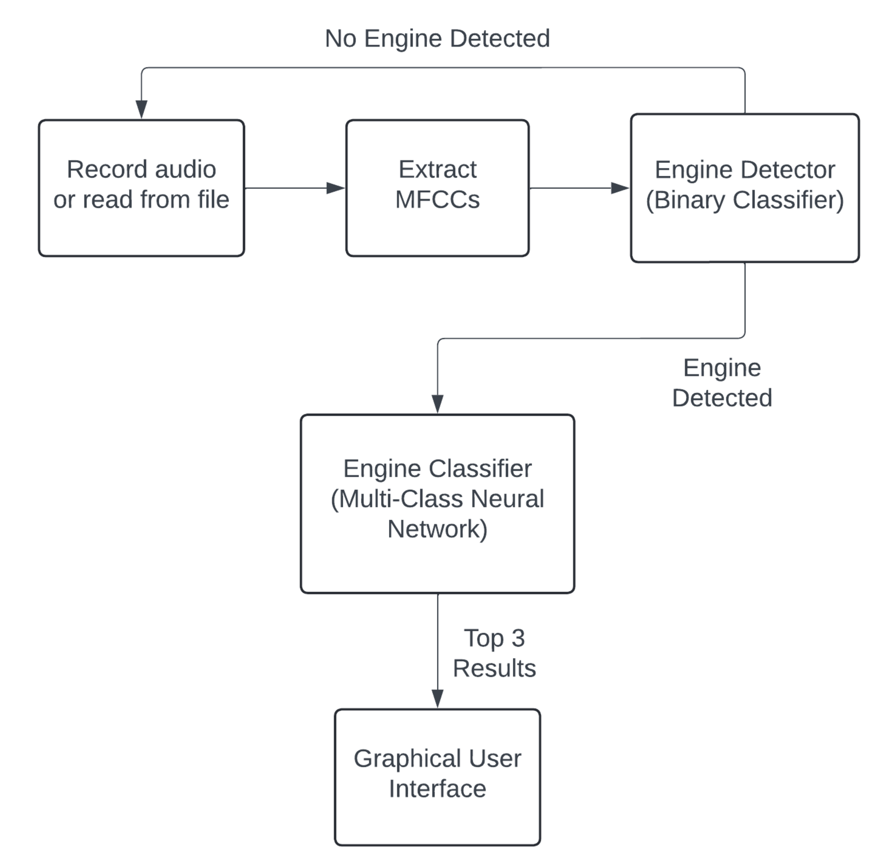


Fig. 2: Software flowchart

## Results

- System runs completely offline
- Neural network classifies audio with 96% accuracy
- System activates only when engine is detected
- Average time per engine identification: 8.5s

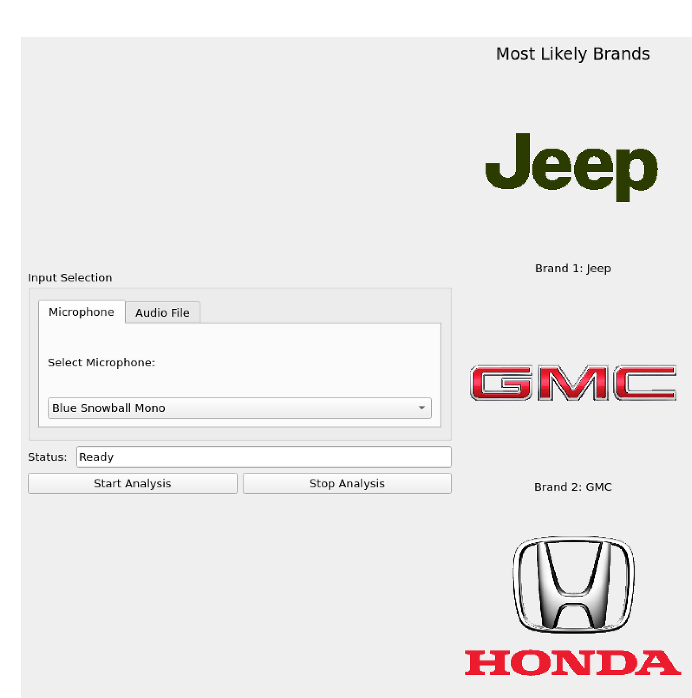


Fig. 3: Program GUI

Predicted Brand (columns) / Actual Brand (rows)

| | BMW | Ford | GMC | Honda | Hyundai | Jeep | Kia | Nissan | Subaru | Toyota |
|---|---|---|---|---|---|---|---|---|---|---|
| BMW | 91 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 2 |
| Ford | 0 | 69 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| GMC | 0 | 0 | 102 | 1 | 0 | 0 | 0 | 0 | 2 | 5 |
| Honda | 0 | 0 | 1 | 122 | 1 | 0 | 1 | 0 | 0 | 0 |
| Hyundai | 1 | 0 | 0 | 1 | 84 | 0 | 4 | 0 | 0 | 0 |
| Jeep | 0 | 0 | 0 | 1 | 0 | 84 | 1 | 1 | 0 | 3 |
| Kia | 1 | 0 | 0 | 0 | 0 | 0 | 89 | 0 | 0 | 0 |
| Nissan | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 107 | 0 | 0 |
| Subaru | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 107 | 0 |
| Toyota | 0 | 0 | 0 | 5 | 2 | 0 | 1 | 0 | 0 | 222 |

Fig. 4: Engine classifier confusion matrix

## Lessons Learned

1. Data collection
   a. Very difficult to find usable car engine audio online
   b. Balanced dataset is difficult to obtain
2. Machine learning model performance
   a. Prone to overfitting
   b. Longer audio segments leads to high input dimensionality
   c. Distance from microphone affects classification accuracy
3. Model preparation
   a. Tuning of model parameters is very time consuming
   b. Difficult to work with large audio datasets
   c. Data augmentation greatly improved performance

## Future Plans

1. Dataset
   a. Balance total audio length between brands
   b. Add class for engines that are not cars (motorcycles, planes, etc.)
2. Machine learning model
   a. Add ability to determine car model in addition to brand
   b. Test different neural network architectures
   c. Test effects of microphone distance on classification accuracy

Fig. 5: Audio recording setup