Embedded Machine Learning Lesson Plan

Main objective:

This lesson plan intends to be an introductory project (or perhaps course) for students interested in machine learning. Students should go through the process of collecting data, creating a model, then using that model to perform a task on an embedded system.

Lab Activities:

Lab Activity 1: Record and understand voice data
Prerequisite: None
Students will be given a set of words to record a specific number of times. Students will use a given microphone and the Voice Recorder app on Windows to do the recordings. They will name their recordings and package them in a folder to be processed.

Lab Activity 2: Process and visualize voice data
Prerequisite: Lab Activity 1
Students will use Audacity to view their recordings and gain an understanding of .wav files. They will learn how to manipulate sampling rate and bit compression. They will get a visual understanding of the wave forms of different words and how they differ, as well as what differs between samples of the same word, and what patterns remain between samples of the same word. Students will cut the sound file to be the correct length (1 second for example) and package and submit these processed samples. The processed samples will go into a large repository for the whole class to be used later on with machine learning model training. This repository will grow as each section completes activities 1 and 2.

Lab Activity 3: Learn about Edge Impulse and upload data
Recommended after Lab Activity 2 (The coursera course introduces voice data)
Students will be asked to complete a Coursera Course on embedded machine learning with Edge Impulse: [Introduction to Embedded Machine Learning | Coursera]().
There are 3 sections:
Part 1 consists of 13 videos (107mins), 14 readings and 5 quizzes.
Part 2 consists of 10 videos (77mins), 10 readings, and 5 quizzes
Part 3 consists of 9 videos (75mins), 7 readings, and 4 quizzes.
Students will start the course during a lab session, and then be asked to complete the course by the next lab session. This course will show them the basics of using edge impulse, recording data, feature extraction, and model training/testing. It will also familiarize them with the Arduino Nano 33 BLE Sense which will be used for the entirety of the lab activities.

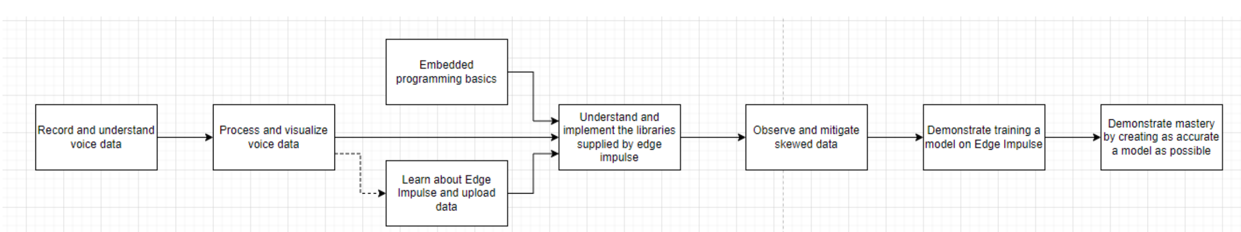
Lab Activity 4: Embedded programming basics
Prerequisite: None
Students will use the Arduino Nano 33 BLE Sense along with 2 LEDs. They will learn how to wire up the LEDs properly to the Arduino pins and to a common ground. They will then create a program that uses the serial monitor to send commands to the board to turn a certain light on or off. This activity will give them the basics of hardware and circuitry, and the Arduino IDE and serial monitor usage.

Lab Activity 5: Understand and implement libraries supplied by Edge Impulse
Prerequisite: Lab Activities 3 & 4
In this activity students will take their knowledge from activities 3 and 4 and create a program where their LEDs turn on and off when given the proper voice command. This will give the students a chance to use the supplied Edge Impulse libraries and examples to create a program of their own that performs the given task.

Lab Activity 6: Observe and mitigate skewed data
Prerequisite: Lab Activity 5
Students will be given data that is decidedly skewed. For example, a model that was trained on all the same voice, or a model that has many duplicates of the same exact data. Students can also be given data and parameters to over or under-fit a specific set of known data. Students can then experience what skewed data results in, when actually put to an embedded system.

Lab Activity 7: Demonstrate training a model on Edge Impulse
Prerequisite: Lab Activity 6
Students will upload data to Edge impulse, as was taught in lab activity 3. This time, with the data supplied by the current class, and classes past. With the data, train a model adjusting parameters to get the model confidence in an acceptable range.

Lab Activity 8: Demonstrate mastery by optimizing a model for best performance
Prerequisite: Lab Activity 7
Students will demonstrate their understanding in a final project using the data from previous classes. With the same dataset, students will train a model and tweak parameters or add synthetic data to demonstrate the highest confidence value. Students can then use their own model on the equipment in the lab. Then, students will go around to each different team and try out every other team's model. Students with the highest success rate could be rewarded to incentivise further effort into the project.



Preliminary lab outline (for reference only)

Lab 1: Familiarize students with the (relatively smaller) embedded part of the project, getting a motor to move. Have students give voice data to add to the class voice dataset.

At this point, it would be expected that students are not familiar with machine learning. Thus, the first lab will be reviewed for the non-machine learning aspects of the project until students are introduced.

Lab 2: Familiarize students with how voice data is recorded and stored. Use the embedded portion of the project yet again to have students record their voices, and make a .wav file. Students can then take their voice data and combine it with data supplied by the class. Supplied data can be the compilation of data from past semesters. The combined data can then be uploaded to edge impulse for creation of a model.

Students are still likely to be unfamiliar with machine learning, as it will only be some of the first weeks, so lab 2 will be to help refresh students with storing data from an arduino. Additionally, a program could be used to visualize the .wav file that students record. This would help visual learners to think about what their machine learning model will ultimately be looking at to fit a curve to.

Lab 3: Students should now be able to grasp the concepts of machine learning, and begin a model. This lab should carry students through the process of using synthetic data to create a model for future use. Synthetic data can be provided by the class (Data from our project and google?). Our suggestion would be to use Edge Impulse to create the students' models, as this is how we created our own during the course of our senior project.

Lab 4: Students will take the model they created in Lab 3, compile it to a library for the Arduino Nano, add the library to their code, and upload it to their board. Learning outcomes will be knowing how to compile and download ML libraries from Edge Impulse, knowing how to import a library into the Arduino Libraries and how to integrate that library into existing code. Deliverables will be an arduino nano with the ML library on board and a serial monitor output showing the confidence levels of the prediction.

Lab 5: Students will be using new models for this lab. Students will be supplied with sets of data, and they will then add the voice data recorded in lab 2 to the data supplied. Students will then duplicate their data multiple times, so that there is an overabundance of their own voices in the training process.

Alternatively, students can be supplied with two models that only require parameter changes to demonstrate over and under fitting. This could reduce cases where students cannot get the desired results just by adding more of their own voices.

Lab 6: In this lab students will take what they learned from Lab 5, and work with the edge impulse parameters and dataset partitioning to create as accurate of a model as possible. There could be an informal competition to see which team has the best test accuracy on Edge Impulse, and which team's Arduino nano has the best accuracy. This will showcase the difference between EI accuracy and real world accuracy. Learning outcomes will be how to troubleshoot and modify ML Models to increase test and real world accuracy. Deliverables will be an Arduino nano/ motor setup that works properly and has at least decent accuracy.