TinyML for Manufacturing

Lesson Curriculum Outline

1. Unit
   1. **Global Learning Goal**
      1. Lesson Title
         1. **Learning Objective**
            1. Activities to achieve
2. Foundation
   1. **Recognize core concepts, processes, properties, and benefits of TinyML to applied manufacturing**
      1. ML vs. AI
         1. **Differentiate ML from Traditional programming, and its relationship to AI**
            1. Demonstration/Illustration of our human brain learning while driving a car and how it parallels the ML paradigm
            2. Provide a simple example of the TP paradigm in contrast
            3. Explain why AI is not ML and analyze different applications of AI vs. ML
      2. Embedded Systems
         1. **Describe why *Tiny*ML presents unique opportunities and challenges**
            1. Introduce what an embedded system is
            2. Compare micro controllers to micro processors
            3. Match scenarios where a micro controller is best suited and where a micro processor is best suited
      3. Responsible AI
         1. **Correlate the Responsible AI benefits to both society and corporations**
            1. Learn about different applications in the subdomains of environment, humans, and industry
            2. Explain the concept of Industry 4.0
            3. Introduce a ‘what-if’ scenario where an unethical ML application detriments both the user and the company
            4. Identify 10 different scenarios where ML could be implemented during the course of normal everyday life
3. Problem definition, scope, requirements
   1. **Identify different components in a holistic picture of TinyML that define the problem to be solved**
      1. Introduction, why we are already so familiar with this technology
         1. **Apply the Human Brain analogy to the extended environment**
            1. Reflect and write about how our environment biases our decisions in everyday life
            2. Provide sets of data at different scales and show how bias comes from datasets just as bias comes from *how we look, live, or experience life*
      2. The ML Workflow and Lifecycle
         1. **Modify a diagram to correctly illustrate the ML Workflow**
         2. **Create an outer skeleton beyond the *code* of ML**
            1. Brainstorm different aspects of the ML workflow and where they fit outside of the Neural Networks/ML Code
      3. Human-Centered Design
         1. **Argue a stance for or against a Human-Centered Design Approach**
            1. Show connections and define terms such as human-centered design, responsible AI, ethical AI, and “partnership in AI,” Type 1 vs Type 2 errors
            2. Introduce a case-study of an ML application
            3. Piece out answers to: *What* is the purpose of the application? *Who* are the stakeholders? and *What if* the application fails?
4. Data Preparation
   1. **Prepare a dataset to be used in the model by properly using data cleaning methods**
      1. Sources of Data
         1. **Evaluate the benefits of some data sources over others**
            1. Identify pros and cons of data from active sensors, public databases, private databases, crowdsourced data, paid contributors, or copyrighted and protected databases
         2. **Prioritize different datasets and/or repurpose a dataset for a *Tiny*ML application**
            1. Provide a *purpose* for an ML app. and task students to choose from a selection of datasets that would be best suited for the purpose
            2. Provide a *purpose* for an ML app. and ask students to repurpose an existing dataset for the specific purpose
      2. Data Cleaning
         1. **Parse a dataset by using different data cleaning techniques**
            1. Explain and demonstrate how to choose a sample to train from and keep an equal sample across sub-datasets for *Training, Validating, and Testing*
            2. Define the importance and difference between a training set, validating set, and testing set
            3. Utilize a case study to identify different target variables
            4. Practice labeling data in different contexts for different problems
         2. **Select, label, and choose data that prioritizes different fairness metrics**
            1. Compare and contrast Group Unawareness & Group Threshold and Equal Opportunity & Equal Accuracy
      3. Bias in Data
         1. **Apply the three key features of data protection to a case study**
            1. Define Identifiability, Data Minimization, Notice and Consent
            2. Discuss a benefit and downside to each of those data protection tenets
         2. **Evaluate if a provided dataset is biased** 
            1. Utilize bias testing toolkits? (3-9-3)
5. Train the model
   1. **Evaluate the accuracy and precision of a model using different model training methods**
      1. Neural Networks
         1. **Identify different features of a Neural Network**
            1. Show examples of single layer networks
            2. Show examples of two or three layer networks
            3. Demonstrate and define loss
         2. **Recognize the required code needed to set up a basic NN**
            1. Provide samples of Google Colab code that will be used to define and train an ML model
      2. Training Techniques
         1. **Evaluate the effectiveness of using different training techniques across multiple problems**
            1. Contrast the differences between Dense NN and Convoluted NN
            2. Introduce concepts of weights, biases, filters
      3. Anomaly Detection
         1. **Identify features of ML needed for anomaly detection**
            1. Introduce unsupervised learning
            2. Define terms, K-means, autoencoders
         2. **Choose which NN structure is best suited for anomaly detection**
            1. Provide students with a selection of NN structures
            2. Walk through which features are most effective for anomaly detection
            3. Demonstrate TensorFlow code features used for anomaly detection
      4. Model conversion
         1. **Explain the process and necessity for converting a model to TFLite**
            1. Reiterate TinyML challenges
            2. Provide information on TFLite vs. TF
         2. **Draw a complete picture of the ML pipeline, up to training a model for a TinyML application**
            1. Tie in previous sections to understand their place in Training