CSC-495/693 - TinyML

- **Course:** CSE 495/693 - TinyML
- **Schedule:** Tuesday and Thursday 3:30 pm - 4:45 pm
- **Instructor:** Drs. Somya D. Mohanty and Evan Goldstein
- **Location:** 217 Stone Building

- **Class Discussions:** https://discord.gg/ktdrsUH4Fd (use #tinyml channel for class discussions)
- **Office Hrs:** Thursday 2:00 pm - 3:00 pm via Zoom only (email for appointment and Zoom link)
- **Email:** [sdmohant@uncg.edu](mailto:sdmohant@uncg.edu and mohanty.somya@uncg.edu)

Course Description

The TinyML course covers topics in the domain of machine learning on edge devices. We will learn about how to develop classical and deep learning approaches to be deployed on small microcontrollers which can sense the world around you. You will learn about model development, compression techniques, efficient memory and CPU management, and long range connectivity via LoRA-WAN. The course is hands-on with embedded devices that are provided for every student to develop their experiments. This is sponsored by Google TensorFlow Research.

The core topics addressed by the course will be:

- Introduction to Micro-Controllers
- Data Collection and Analysis with Micro-Controllers
- Machine Learning and Model Compression
- Deployment on IoT devices

Prerequisites

Programming experience (C, C++, or Python)

Textbooks

TinyML : Machine Learning with TensorFlow Lite on Arduino and ultra-low-power Microcontrollers, Pete Warden and Daniel Situnayake

- https://uncg.on.worldcat.org/oclc/1135326041

Requirements

- Laptop for connecting to micro-controllers (provided in the class)
    - Fill out the Loan Agreement form: https://forms.gle/9VuvqJrwv7yDSUXb8
- Micro-USB and USB-C cables

Course Topics and Schedule (Tentative)

1. **Introduction: (Week 1)**
    - Class Syllabus
    - Introduction to Micro-Controllers
2. **Data Collection (Weeks 2-3)**
    - Human Sensing
        1. Accelerometer, Magnetometer, Gyroscope
    - Environment Sensing
        1. Temperature, Audio, Vison
    - Final Project Discussions - Goals and Requirements
3. **Machine Learning (Week 4-9)**
    - Machine Learning
    - Deep Learning
    - Model Tuning
    - End to End Development
4. **Model Compression (Week 10-11)**
    - Sparsification
    - Quantization
    - Size vs Speed vs Accuracy
    - Optimizers
5. **Deployment on Micro-Controllers (Week 12-13)**
    - Develop a gesture sensor - Nano
    - Develop a object detector - Portenta
    - Develop a wake word detector - Nano
6. **Final Project (Week 14-15)**
    - In class debug
    - Presentation

Grading

Grade	**Max% to Min%**
A	100% to 94%
A-	< 94% to 90%
B+	< 90% to 87%
B	< 87% to 84%
B-	< 84% to 80%
C+	< 80% to 77%
C	< 77% to 74%

Grade Max% to Min%

C- < 74% to 70%

D+ < 70% to 67%

D < 67% to 64%

D- < 64% to 60%

F < 60% to 59%

1. **Class / Homework Assignments (4):** 60%
    - https://github.com/UNCG-DAISY/CSC-495-693-TinyML/Assignment
    - *Week 1:*
        1. Connect with you embedded device
            1. Arduino IDE
                - Sinewave LED
            2. OpenMV IDE
                - Take a selfie with camera
    - *Week 3:*
        1. Collect data - Convert into a proper format for ML
            1. IMU - Arduino Nano - IDE (Wave and Fist Bump)
            2. Audio - Edge Impulse - (Up and Down/ Stop and Go)
            3. Images - OpenMV - (apple, banana, orange)
    - *Week 5:*
        1. Create a model for IMU
    - *Week 7:*
        1. Create a model for Audio
    - *Week 10:*
        1. Create a model for Images.
        2. Sparsify the model.
        3. Quantize the model.
    - *Week 12:*
        1. Deploy to Gesture to Nano
        2. Object Detection to Portenta
2. **Final Project:** 40%
    - https://github.com/UNCG-DAISY/CSC-495-693-TinyML/Project

    The final project of the class will be a team-based effort. In first week of the course the students split into teams of 3-4 students. The students will come up with a project idea and pitch it in the class for approval.

    - *Week 6 - Final Project Pitch*
        1. Group project 3-4 members - 10 min presentation
    - *Week 9 - Final Project Progress/Check-in*
    - *Week 11 - Final Project Progress/Check-in*
    - *Week 13 - Final Project Progress/Check-in*
    - *Week 14-15*
        1. In Class debug

2. Final Presentation

After completing each check-in, the teams will be asked to report on the progress of their project. The projects will be open-source and the teams will have to use GitHub as their code repository. Upon completion of the project the teams will present their software along with the results in form of a presentation (20 minutes).

7. Final Project has 100 points. The deliverables are:
    1. Code Jupyter/IPython Notebooks - Explaining the project development to deployment. (40 pts)
    2. Presentation - Present your approach and the results. (20 pts)
    3. Demonstration - Demonstration of your approach on a microcontroller. (40 pts)
8. ***Graduate Students Only:*** In addition to the aforementioned project deliverable, graduate students will create a project report. The report will be for 20 points and other deliverables amount to 80 points. Minimum 5 pages for single author, 8 for 2 authors, and 12 for 3 authors (figures and references included). Template:. Example: (Due:)

Total: 100%

Deadlines

Category	Sub-Category	Deadline
Assignment	* Assignment 1	
	* Assignment 2	
	* Assignment 3	
	* Assignment 4	
	* Assignment 5	
	* Assignment 6	
Project	* Presentation	
	* Report	

Submissions

- **Assignments**:
    - Create a ***private Github repository*** (under your own account) and call it --- CSC-495-693-TinyML.
    - Send me and our TA access to the repository,
        - My username: somyamohanty
        - Our TA is: ()
    - For each assignment create a folder within the repository i.e. /Assignment_1
        - Create two sub-folders /src and /data
        - Work on your assignment (under /src)

        - Provide a readme of you assignment.
        - IPython notebooks (.ipynb), C code, arduino sketches
        - Comment your code appropriately in Markdown.
    - Enter the link to your assignment solution in the assignment text entry (on canvas) once you are done with your solution.
    - No collaboration at all in assignments
- **Project**:
    - Your code and documentation will reside in a ***project repository***.
    - The structure of the repository should be maintained as such.
        - /src - code and notebooks
        - /data - data folder for the repository
            - /stage_X
        - /utility - utility or scripts
        - /doc - documentation - project reports and presentations
        - Readme.MD - Description of project.
        - all src files (notebooks) should use relative path.
    - Use Issues to track progress.

Communication

Discord channel for class discussions and team creation: https://discord.gg/ktdrsUH4Fd. The channel should be used for discussing general questions related to assignments and projects. Use this channel to ask questions and find anwsers to already responded quesitons. If the question has been already answered in the channel, I will not be responding to emails. Emails are a one-to-one conversation which takes a lot of time hence the channel is there to broadcast information and have more community oriented discussion. Do not share code or screenshots of code in the channel. Email should be the last step and can be used to ask student specific questions.

Project Teams:

Team 1:

-

Academic Honesty Policy

The instructor will deal strictly with any violations of academic honesty and integrity in this course. See Academic Integrity Policy (Link). for more details. ***Absolutely no discussion, collaboration, copying, and sharing on assignments. This includes coping from the internet. Any student who violates this policy will receive "F" in the course. The instructor will report the case to the university.***

Attendance Policy

Attendance is required for all the class meetings. If you will be absent for any class it is your responsibility to catch up on class materials.

Special Needs and/or Disabilities

Students with disabilities should have documentation from the Office of Accessibility Resources & Services (Link). This documentation should be provided to the instructor for review. In the case of major provisions such as separate testing environment or test-readers, the student must make arrangements with Office of Accessibility Resources & Services so that suitable accommodations can be provided.

COVID Statement

As we return for spring 2022, all students, faculty, and staff are required to uphold UNCG's culture of care by actively engaging in behaviors that limit the spread of COVID-19. These actions include, but are not limited to:

- Following face-covering guidelines
- Engaging in proper hand-washing hygiene
- Self-monitoring for symptoms of COVID-19
- Staying home when ill
- Complying with directions from health care providers or public health officials to quarantine or isolate if ill or exposed to someone who is ill
- Completing a self-report when experiencing COVID-19 symptoms, testing positive for COVID-19, or being identified as a close contact of someone who has tested positive
- Staying informed about the University's policies and announcements via the COVID-19 website

Instructors will have seating charts for their classes. These are important for facilitating contact tracing should there be a confirmed case of COVID-19. Students must sit in their assigned seats at every class meeting. Students may move their chairs in class to facilitate group work, as long as instructors keep seating chart record.Students should not eat or drink during class time.

A limited number of disposable masks will be available in classrooms for students who have forgotten theirs. Face coverings are also available for purchase in the UNCG Campus Bookstore. Students who do not follow masking requirements will be asked to put on a face covering or leave the classroom to retrieve one and only return when they follow the basic standards of safety and care for the UNCG community. Once students have a face covering, they are permitted to re-enter a class already in progress. Repeated issues may result in conduct action. The course policies regarding attendance and academics remain in effect for partial or full absence from class due to lack of adherence with face covering and other requirements.

For instances where the Office of Accessibility Resources and Services (OARS) has granted accommodations regarding wearing face coverings, students should contact their instructors to develop appropriate alternatives to class participation and/or activities as needed. Instructors or

the student may also contact OARS (336.334.5440) who, in consultation with Student Health services, will review requests for accommodations.

Super Useful Links :)

MicroPython

- Getting Started with MicroPython
- MicroPython Docs
- OpenMV and MicroPython

C and Arduino

- Tutorial - Arduino Programming in C
- Arduino Reference

Python

- The official web page of the Python programming language.
- https://developers.google.com/edu/python/ - Google Python Class
- http://www.python.org/dev/peps/pep-0008 - Style guide for Python programming. Highly recommended.
- http://www.greenteapress.com/thinkpython/ - A free book on Python programming.
- Python Essential Reference - A good reference book on Python programming.

Jupyter Notebooks

- Install Jupyter on Mac
- Install Jupyter on Windows
- Jupyter/IPython Notebook Quick Start Guide
- Jupyter Notebook: An Introduction
- How To Enhance Jupyter Notebooks for Data Science?
- 28 Jupyter Notebook Tips, Tricks, and Shortcuts
- Optimizing Jupyter Notebook: Tips, Tricks, and nbextensions
- 28 Jupyter Notebook Tips, Tricks, and Shortcuts

TinyML

- Screencasts and other content related to TinyML at [www.youtube.com/user/petewarden]
- TinyML Tensorflow example [www.github.com/tensorflow/tflite-micro/tree/main/tensorflow/lite/micro/examples]