EECE-4710 IoT and Tiny Machine Learning

Syllabus – Spring 2024

Course info

EECE-4710 IoT and Tiny Machine Learning, 3 credits
Prerequisites: EECE-3015 Digital Electronics Lab
Lecture: Tue, Thu, 9:30-10:45AM, Haggerty 388
Course website #1 on D2L: http://d2l.mu.edu
Course website #2 public: http://dejazzer.com/eece4710/index.html

Instructor

Cristinel Ababei, cristinel.ababei@marquette.edu
Phone: 414-288-5720
Office: Haggerty Hall 220
Office hours: Tue, Thu 8:30-9:20AM or stop by any time
Web: www.dejazzer.com

Catalog description

This course introduces students to applied tiny machine learning (TinyML) for embedded Internet of Things (IoT) devices. The emergence of embedded machine learning (ML) frameworks like TensorFlow Lite for microcontrollers, coupled with the pervasiveness of low-power embedded devices will usher the proliferation of artificial intelligence (AI)-powered IoT devices. Covered topics will include low-power design, energy harvesting, sensor data fusion and ML methods for lifetime prediction.

Textbook(s)

[1] Pete Warden and Daniel Situnayake, TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers, O'Reilly Media, 1st edition, 2020. ISBN-10: 1492052043.

Additional materials (sensor datasheets, research articles, and hands-on activities descriptions) will be provided in class.

Hardware kit for hand on in-class activities and assignments

This course uses a learning approach based on hands-on activities and assignments. To verify IoT programs on real hardware, a course HW kit is used. The kits includes a microcontroller board (Arduino Nano 33 BLE Sense – as of the time of drafting this syllabus, but this may be changed later) and several sensor modules.

Course objectives

The goal of this course is to teach electrical and computer engineering students theoretical concepts and practical skills of embedded Internet of Things (IoT). Emphasis is put on machine learning technique applied to problems from the IoT area. Specific objectives include the following:

1. Learn techniques to design embedded systems used in IoT.
2. Utilize sensor modules to design environment monitoring IoT devices.
3. Develop machine learning models and techniques for the IoT. Program such techniques in Python and embedded C/C++ using Tensorflow Lite.
4. Deploy and verify these techniques on real hardware.
5. Learn low power methods for microcontroller-based systems.
6. Learn the type of energy harvesting techniques appropriate for specific IoT applications.
7. Apply testing and debugging strategies.
8. Prepare informative and organized lab reports that describe the methodologies employed, the results obtained, and the conclusions made in a laboratory experiment.

Relationship of course objectives to ABET student outcomes

The course objectives partially fulfill the following ABET student outcomes: 1, 2, and 6.

Course policies

Grading: Grading is based on the following components:

- Class participation: 20%
- Homework assignments: 80%

Grading scale:
A [94-100]%; A- [90-94)%; B+ [87-90)%; B [83-87)%; B- [80-83)%; C+ [77-80)%; C [73-77)%; C- [70-73)%; D+ [67-70)%; D [60-67); F [0-59)%.

Regrading: Regrading requests should be made within a week from the date of the graded item (homework assignment, exam, or project) becoming available.

Homework policy: Homework assignments will be assigned throughout the semester. **Late submissions will not be accepted.** All homework assignments should be submitted before the deadline (the beginning of the class on that day). Collaboration on homework is ok (**I encourage you to study in groups**), copying is not ok; a separate solution is required from each student. If you are absent from class or you know that you will be absent from class, you should as soon as possible arrange with the instructor for any missed work. It is the student's responsibility to contact the instructor in such a case. Arrangements made in advance of an absence (if approved) may allow full credit to be given for late work. Include your name only (**not MUID**) on all homework assignments, reports, exams, etc. Turn in solutions that are written clearly and neatly; turning in disorganized or esthetically-ugly solutions with scratched-out text, figures, and formulas, etc. is penalized by deducting grade-points even if the final answer is correct. All assignments must be turned in via D2L. The handwritten portions must be electronically scanned and converted to PDF.

Academic integrity: All submitted work should be your own. Instances of plagiarism and cheating will result in all students involved getting an automatic zero on the assignment/exam/project and potentially a failing grade based on the severity of the case. Please refer to the Marquette University's academic integrity/honesty policies and procedures: http://bulletin.marquette.edu/undergrad/academicregulations/#academicintegrity

Attendance: I encourage everyone to attend all the lectures and actively participate in class discussions. While attendance will not be taken, note that students may be withdrawn from a course as a result of excessive absences (i.e., when more than 6 classes have been missed in a 3 credit course). Please read the Marquette University's undergraduate attendance policy: http://bulletin.marquette.edu/undergrad/academicregulations/#attendance

Class participation: You are expected to (pro)actively participate in this course. Active participation means to study the assigned reading items before class, ask and answer questions, and participate in discussions during lectures. Proactive participation means to search and read – on your own – additional information (e.g., online articles, research papers, textbooks, etc.) related to the topics of this course and to share it with the other students and the instructor during in class or outside discussions.

Lab attendance and grading: Laboratory attendance is mandatory. Missing labs without prior approval from the instructor will automatically result into a grade Incomplete or F. The grade of each lab includes: 1) Verification of provided lab examples and described in the lab document on the day the lab takes place (30%), 2) Verification of implemented lab assignment(s), due on day of next lab (30%), and 3) Written report for the lab, due on the day of next lab (40%).

Special needs: If you have a disability and require accommodations, please contact your instructor early in the semester so that you learning needs may be appropriately met. You will need to provide documentation of your disability to the Office of Disability Services (OSD). If you are unsure of what you need to qualify for services, visit the OSD website at http://www.marquette.edu/disability-services/forms.shtml or contact the Office of Disability Services at 414-288-1645. The Office of Disability Services is located in Marquette Hall 05.

Homework assignments

Most of the homework assignments involve writing code in Python and embedded C to implement various embedded IoT systems and verification of written programs either via simulations or deployment and testing on the hardware kit used in the course. This course uses a hardware kit, which includes a microcontroller board and several different types of sensors. It is assumed that each student has a personal laptop, which shall be used in the classroom activities and hands-on exercises. Additional assignments may include research-paper reviews and presentations.

Tentative course outline and schedule (Note: This is subject to change during the term)

Week	Topics	Important Dates
1, 2,3,4,5	**Part 1: Foundations of Machine Learning (ML)** About the course and Syllabus Introduction to IoT and TinyML The Machine Learning Paradigm Building Blocks of Deep Learning (DL) - Introduction Building Blocks of DL - Regression with Dense NN Building Blocks of DL - Classification with Dense NN Image Classification using CNN Introduction to Edge Impulse – CNN with Cifar-10 Datasets and Model Performance Metrics Preventing Overfitting	
6,7	**Part 2: Sensors** Sensors and IoT applications Wireless sensor networks Sensor data collection Sensor fusion	
8,9,10,11,12	**Part 3: Applications and Deployment to Microcontrollers** TFLite and TFLite-Micro TinyML Kit Overview TFL-Micro Hello World Hello World Code Walkthrough Motion Classification - Introdution Motion Classification using MCU (Nano 33) K-means Clustering and Anomaly Detection Anomaly Detection Hands-On Lab Keyword Spotting - Introduction KWS using MCU Image Classification - Introduction Image Classification using Edge Impulse Studio	Week 9 (March 11-15): Midterm break – no classes Week 11-12 (March 28-April 1): Easter break – no classes
13,14	**Part 4: Energy Harvesting Techniques** Energy harvesting techniques Solar, electromagnetic, mechanical (piezo) Simulations in Spice	
15	**Part 5: Security in IoT** Security aspects	
16	Conclusion	