THE UNIVERSITY OF TEXAS AT DALLAS

CS 6301 Special Topics: Machine Learning in Mobile Computing

(Fall 2025)

Yi Ding

Department of Computer Science

Picture generated by ChatGPT

Recap

- A team of 3 (around 15 teams in total)
- In the project, we aim to develop an APP on the smartphone that:
  - meets specific need (e.g., healthcare, wellness, social, entertainment);
  - collects data from different sensors (e.g., camera, microphone, GPS, IMU)
  - utilize LLM/VLM/VLA (e.g., ChatGPT, Llama, Claude) via API or deployment on the mobile device;
  - with efficient system design (i.e., latency, CPU load, memory usage, bandwidth usage)

Highlights:

- Must involve sensing data (vision, acoustic, GPS, IMU, Wi-Fi, Bluetooth).
- Must apply a machine learning model (classical ML, deep learning, or large language models).
- Deployment can be on mobile devices or in the cloud, but sensing + ML are both essential.

Examples

✅ Good Example:

- Multimodal sensing for assistive navigation.
  - Combine camera + acoustic + IMU.
  - Detect obstacles and guide visually impaired users.
  - Uses real sensing + ML model.

❌ Less Ideal Example:

- Pure computer vision image classification.
  - Only uses static vision dataset.
  - No mobile sensing or multimodal integration.
  - Limited alignment with course focus.

Project Workflow Template

1.Idea & Motivation

- Identify a sensing problem relevant to mobile computing
- Define why it matters (use-case driven)

2.Data Collection

- Choose one or more sensing modalities (vision, acoustic, GPS, IMU, Wi-Fi, Bluetooth)
- Collect or reuse datasets

3.Model Development

- Apply a machine learning model (classical ML, deep learning, or large model)
- Consider deployment constraints (mobile vs cloud)

4.Evaluation

- Define metrics (accuracy, latency, energy, robustness)
- Compare baseline vs improved model

5.Demo & Presentation

- Show a working prototype or simulation
- Present findings clearly in class

Useful Resources

Mobile Development Frameworks(IDE):
Android studio:
https://developer.android.com/studio?gad_source=1&gclid=Cj0KCQiAy8K8BhCZARIsAKJ8sfRVoWJf0aJL08g80660JQTtBEv3bWJUqYqRAwdA64NZyg5hxZsI4MIaAlRLEALw_wcB&gclsrc=aw.ds
Android development language:
- Java:
https://developer.android.com/build/jdks
- Kotlin:
https://developer.android.com/kotlin

Mobile LLM Framework:
- MediaPipe(recommand)
https://ai.google.dev/edge/mediapipe/solutions/guide
- ExecuTorch(optional):
https://pytorch.org/executorch-overview

Fundamental LLM Knowledge
https://github.com/mlabonne/llm-course

https://www.bing.com/videos/riverview/relatedvideo?&q=Fundamental+LLM+Knowledge&qpvt=Fundamental+LLM+Knowledge&mid=6B7834A2CF670380E3E76B7834A2CF670380E3E7&&FORM=VRDGAR

Simple Deep Learning / Pre-Training Model Implementation
https://github.com/changruijie/simple_LLM_pretrain_learning_model

🎯 Key Takeaway

- Projects must connect sensing with ML.
- Aim for creativity, practicality, and technical depth.
- Think: 'What can mobile devices sense, and how can ML make it useful?'

Questions?