ECE2500Y: M.Eng. Project

Department of Electrical & Computer Engineering
University of Toronto

Project Supervisors:
Prof. Deepa Kundur

dkundur@ece.utoronto.ca

Mentor:
Dr. Ahmad Abdelsamie

a.abdelsamie@mail.utoronto.ca

Student: Student

Project Title: Explaining Unsupervised Anomaly Detectors in Cyber-Physical Systems

Focus Area: Explainable AI from an ML perspective

Project Overview (research + prototype)

Most CPS security pipelines rely on unsupervised/semi-supervised anomaly detection.

Goal: produce understandable “why anomalous” explanations for these detectors.

Scope (what you will do)

- Implement 1-2 unsupervised detectors (e.g., autoencoder, isolation forest, one-class SVM).
- Design explanations: feature contributions, reconstruction-based cues, or rule-like summaries.
- Evaluate explanation faithfulness and whether it helps locate root-cause sensors/features.
- Document limitations and failure modes.

Suggested tools (offline-friendly)

- Python; PyTorch or sklearn; offline plots and saved HTML reports if desired.
- Optional: simple counterfactual search for anomalies.

Major milestones (Preliminary)

Note: milestone dates and scope are preliminary and may be adjusted during the term.

Milestone 1 (Literature Review)

- Review ~10-15 key papers and produce a short summary of what is most relevant to this project.
- Define the problem statement, dataset(s)/scenario, and evaluation plan (metrics + baselines).

Deliverable: literature review note + evaluation plan + baseline repo (runnable offline).

Milestone 2 (Implementation + Experiments)

- Implement the main method/idea and 1-2 strong baselines; run controlled experiments.
- Analyze results and failure cases; iterate with supervisor/mentor on what is useful in OT/CPS settings.

Deliverable: working prototype + results note (plots/tables) + reproducible experiment scripts.

Milestone 3 (Final Report + Presentation + Code)

- Finalize experiments and discussion; write the final report.
- Prepare the final presentation/demo and clean the repository for handover (README + run steps).

Deliverable: final report + presentation/demo + final code repository.

End-of-term deliverables (must include)

- Literature review + references list.
- Implementation/prototype with reproducible scripts (offline).
- Final technical report.
- Final presentation + demo.
- Clean code repository (README + environment + run instructions).

Evaluation (typical weighting)

20%	Milestone 1 (Literature Review)
35%	Milestone 2 (Implementation + Experiments)
30%	Milestone 3 (Final Report + Code)
15%	Final presentation + demo