Clean-Label Backdoor Attacks on Time-Series Human Activity Recognition

Nikhil Chaitanya Anasmit

Departmental Project — DS603

November 2025

Outline

- Motivation Problem
- Key contributions
- Methodology (overview)
- Representative attack methods
- Selected results takeaways
- Limitations, implications, next steps

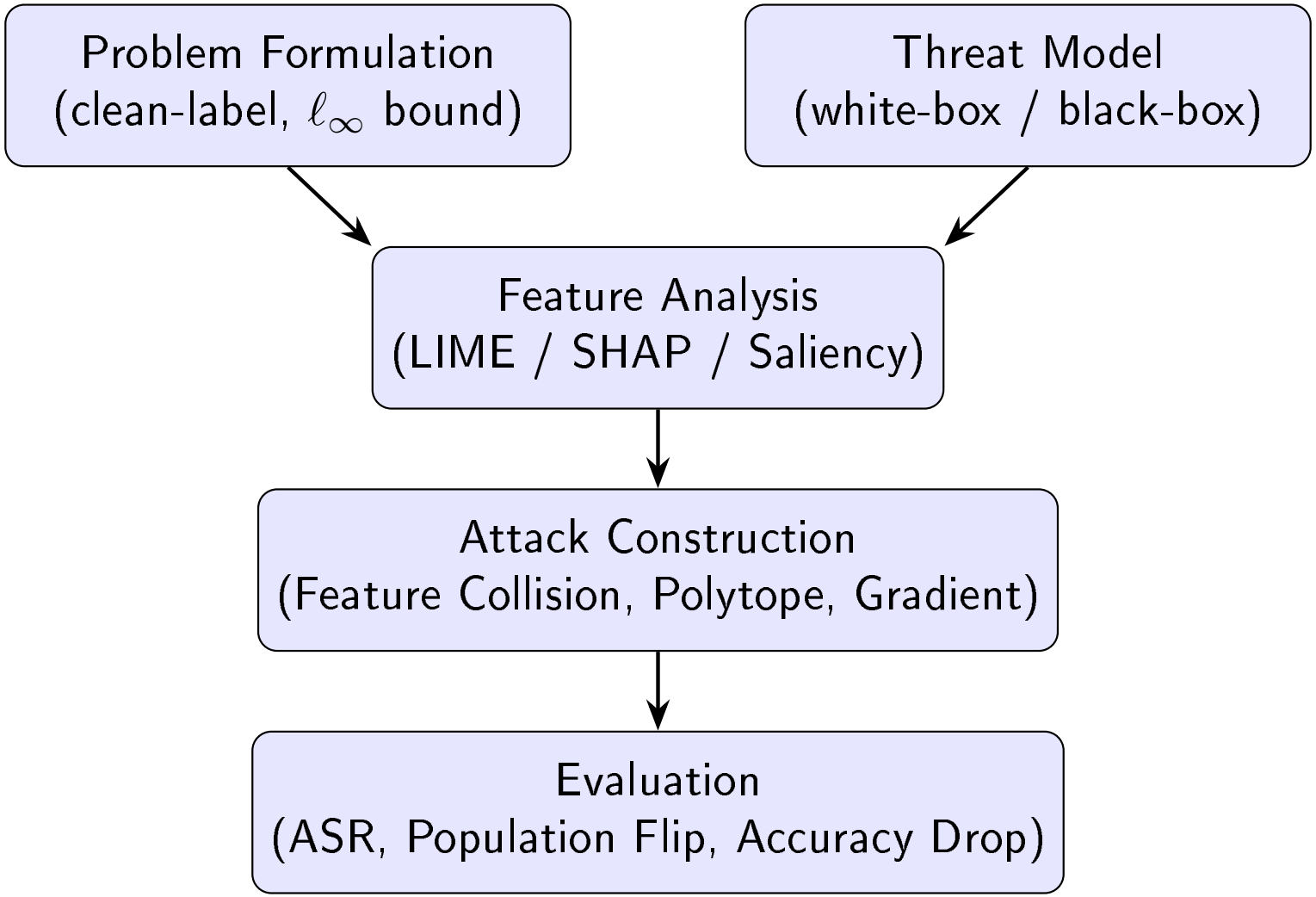
Motivation

- HAR systems are used in healthcare, wearables, and IoT — safety-critical domains.
- Clean-label poisoning: stealthy attacks that preserve true labels, evading simple checks.
- Question: Can explainability tools (LIME/SHAP) be used to guide effective, stealthy attacks on time-series models?

Key Contributions

- Integrated LIME/SHAP analysis with feature-space and gradient-based poisoning for time-series HAR.
- Developed reproducible saliency-driven and explainability-guided attack pipelines for UCI-HAR and WISDM.
- Comprehensive evaluation across deep and traditional ML models; documented vulnerabilities and defenses needed.

Methodology — High Level



Representative Attack: Saliency-Driven

- Use gradient-based saliency to identify high-influence timesteps/channels.
- Construct deterministic target signature and blend into selected source-class windows.
- Enforce temporal continuity (50% overlap) to keep samples plausible.
- Works well for identifying sensitive windows in CNNs.

Representative Attack: Explainability-Guided

- Use LIME/SHAP to compute channel/timestep importance matrix W.
- Optimize perturbations focusing on high-weight entries: feature collision, centroid shifts, polytope rings.
- Applicable in black-box scenarios (uses model-agnostic explainability or surrogate models).

Selected Results — Topline

- WISDM (traditional ML vulnerability): Ridge classifier achieved **22.7% ASR** with clean-label attacks.
- UCI-HAR (deep model robustness): CNN shows $<$ **4% ASR** (1.2% in our tests).
- Explainability: Y-acceleration emerged as a dominant feature on WISDM (~46% importance by SHAP).
- Attacks preserve model utility: poisoned accuracy within 1–3% of clean accuracy in most settings.

Implications and Recommendations

- Explainability tools can reveal attackable features — dual-use risk.
- Prefer robust architectures and preprocessing for safety-critical deployments.
- Defenses should include feature-space checks, temporal-consistency tests, and certified robustness where possible.

Limitations & Future Work

- Mostly single-target evaluations; multi-target and adaptive defenses remain to be explored.
- Need experiments on Transformer-based models and physical-sensor realizability.
- Next: implement defense baselines (spectral signatures, activation clustering) and cross-architecture transfer tests.

Appendix: Implementation Notes

- Code: PyTorch-based; scripts enforce clean-label constraint (labels copied, only features modified).
- Perturbation bounds: ℓ_∞ (UCI-HAR: $\varepsilon = 0.02$, WISDM: $\varepsilon = 0.3$).
- Reproducibility: deterministic seeding and overlapping-window constraints used across experiments.

Thank you / Contact

- Questions welcome — we can expand any slide.
- Authors: Nikhil, Chaitanya, Anasmit
- Project repo: (available on request)