Interpretable Telematics-Based Energy Anomaly Detection Using LIME

Ilyas Galiev, Daria Alexandrova, Kamilya Shakirova

February 12, 2026

1 Project Overview

We propose implementing LIME (Local Interpretable Model-Agnostic Explanations) to explain predictions of a Gradient Boosting model (XGBoost) in vehicle telematics and predictive maintenance. The project will include:

- Original implementation of the XAI method
- Application to vehicle energy anomaly detection
- Visual explanations and documentation
- Public GitHub repository with reproducible code

2 Components

2.1 Application Domain

- Domain: Vehicle telematics / predictive maintenance
- Importance: Modern vehicles generate large amounts of OBD telemetry data, but drivers cannot understand early signs of vehicle degradation. We aim to use machine learning to detect abnormal energy consumption patterns and provide explainable predictions to build user trust.
- Dataset: Vehicle Energy Dataset (VED)
  - Real-world OBD and GPS telematics
  - 383 vehicles
  - ∼600k km of driving data
  - Multivariate time-series (speed, acceleration, energy consumption)

2.2 Machine Learning Model

- Model Type: Gradient Boosting (XGBoost) for regression
- Architecture: We will train a model to predict expected (normal) energy consumption per km using aggregated trip features:
  - mean and variance of speed
  - acceleration statistics
  - stop-and-go ratio
  - idle time
  - time of day / seasonality

Anomaly definition: $Residual = actual\_energy - predicted\_energy$
A high residual indicates unexplained energy consumption and a potential vehicle issue.

- Need for XAI: The model is a black-box ensemble, so we need to explain why a trip was expected to be efficient and why high consumption is suspicious.

2.3 Proposed XAI Method

- Method: LIME (Local Interpretable Model-Agnostic Explanations)
- Implementation Plan:
  - Code LIME for tabular time-series features from scratch
  - Integrate LIME with the trained XGBoost regression model
  - Detect anomalous trips using a residual threshold
  - Generate perturbed samples around each selected trip
  - Query the black-box model on perturbed samples
  - Fit a local linear surrogate model for each anomaly
  - Extract local feature importance explanations
  - Generate visual visualizations for anomalous trips
- Key Equation:

$$\hat{g} = \arg\min_{g \in G} \sum_{i=1}^{N} \pi_x(z_i) \left(f(z_i) - g(z_i) \right)^2 + \Omega(g)$$

3 Deliverables

- Documented Python code on GitHub
- Trained anomaly detection model
- 3–5 LIME explanation examples
- Visualizations: predicted vs actual energy and feature importance plots
- Technical blog post explaining method/results
- Validation using regression metrics (MAE / RMSE)

4 Timeline

- Feb 12: Proposal submission
- March 12: Base implementation (ML model + anomaly detection)
- April 15: Visualizations and testing (LIME)
- April 21: Final presentation