Sensor Fusion in Supervised Learning: When Do Multiple Sensors Help or Hurt in IoT Devices?

Project Proposal

Srihari Raman

Summer 2025

Problem Description

The rise of Internet of Things (IoT) devices has transformed the way data is collected, monitored, and analyzed across domains like healthcare, smart homes, fitness tracking, and transportation. These devices are embedded with sensors that continuously gather data about the physical world—such as motion, temperature, pressure, and location—and send it to centralized systems for decision-making. In the field of data science, IoT data offers rich opportunities for modeling human behavior, predicting events, and optimizing systems, but also introduces challenges related to data quality, volume, and sensor accuracy and variability.

One common strategy in IoT systems is sensor fusion, where data from multiple sensors is combined to improve the accuracy or robustness of predictions, with the intuition that multiple sensors provide a more complete picture of the underlying behavior. However, adding more data is not always beneficial—sometimes it leads to redundancy, noise, or overfitting, especially in supervised learning settings.

This project explores the practical and theoretical impact of sensor fusion in supervised machine learning using the UCI Human Activity Recognition (HAR) dataset. The dataset consists of preprocessed sensor data from smartphones, where subjects performed six different activities and features were extracted from both accelerometer and gyroscope signals.

Literature Review

Human Activity Recognition (HAR) using smartphone sensors is a well-studied problem in the machine learning and IoT communities. Most approaches rely on data collected from inertial sensors—primarily accelerometers and gyroscopes—which are embedded in mobile devices and wearables. To improve classification performance, many systems apply sensor fusion, the process of combining signals from multiple sensor modalities into a unified feature representation.

A comprehensive survey by Gravina et al. (2019) categorized sensor fusion techniques into three levels: data-level (raw signal fusion), feature-level (fusion after feature extraction), and decision-level (fusion of model outputs). Their analysis found that feature-level fusion is the most commonly applied method, especially in wearable HAR systems, due to its balance between performance and computational cost. They also noted that fusion of heterogeneous sensors (sensors with different types of readings) tends to yield higher accuracy than using a single sensor modality.

Similarly, in another study, Webber and Fernandez Rojas (2021) directly compared these three levels of fusion on four public HAR datasets. Their results showed that decision-level fusion produced the highest accuracy (74.4%) but required greater computational resources. Feature-level fusion offered a practical middle ground, achieving solid performance (67.4%) without significant overhead. This aligns with the approach taken in this project: using feature-level fusion to combine preprocessed accelerometer and gyroscope data from the UCI HAR dataset.

Several works have also explored specific benefits and limitations of sensor fusion. For instance, Huang et al. (2021) applied a method called "movelets" to combine accelerometer and gyroscope data, demonstrating improved classification for ambiguous activity pairs like standing vs. sitting.

In terms of model selection, Hossain et al. (2025) benchmarked classical and deep learning approaches for HAR. They found that models like logistic regression, random forest, and XGBoost remained competitive on the UCI HAR dataset, often achieving over 90% accuracy. Deep learning models performed better on larger or more complex datasets, but required more data and computational power. These findings support the use of baseline and tree-based models in this project.

Overall, the existing literature emphasizes the value of sensor fusion in HAR, but also leads to questions about when fusion truly helps and when it introduces redundancy or overfitting. This project addresses that gap by systematically evaluating how feature-level sensor fusion impacts performance in traditional supervised models. By utilizing 4 different models, each with their unique theoretical approaches, we aim to not only provide the best sensor fusion methodology but also generalize which model types perform the best with which sensor fusion setup.

Algorithms

With this in mind, the central question in this project is:

> *Does combining multiple sensor modalities always improve model performance in supervised learning?*

To answer this, I will compare the performance of various classifiers trained on:

- Accelerometer-only features
- Gyroscope-only features
- All features (sensor fusion)

We will use the following models for experimental analysis:

- **Logistic Regression** – A baseline model that is computationally efficient, interpretable, and commonly used for multiclass classification.

- **k-Nearest Neighbors (k-NN)** – A simple non-parametric model that classifies new data points based on the majority class of the k closest training examples.
- **XGBoost** – A gradient-boosted decision tree model known for its high performance on structured data and resistance to overfitting.
- **Gaussian Naive Bayes** – A lightweight probabilistic model that assumes feature independence and normal distribution, well-suited for multiclass classification.

Each model will be trained on the three sensor configurations above and compared based on predictive performance to identify which model-sensor combinations are most effective.

Datasets

This project uses the **UCI Human Activity Recognition (HAR)** dataset, a well-established benchmark for activity classification from smartphone sensor data. Originally introduced by Garcia-Gonzalez et al. (2020), the dataset was designed for real-world, mobile-based HAR applications.

The data was collected from 30 participants performing six daily activities:

- Walking
- Walking Upstairs
- Walking Downstairs
- Sitting
- Standing
- Lying

Each participant wore a smartphone on their waist, which recorded signals from two embedded sensors: an accelerometer and a gyroscope. For each observation, 561 precomputed time- and frequency-domain features were extracted and will be used for model training.

Libraries and Tools

This project will be developed locally using a Python-based machine learning environment on macOS (MacBook), leveraging the following tools:

- **pandas, numpy** – For data manipulation and numerical operations
- **scikit-learn, xgboost** – For implementing and evaluating machine learning models
- **Reflex** – A Python web app framework used to build a dashboard for analysis and visualization
- **Weights & Biases** – For experiment tracking, logging, and visualizing model performance across runs
- **knockknock** – A lightweight notification tool to send desktop alerts when training jobs finish

Expected Results

I expect that combining data from both accelerometer and gyroscope sensors will improve classification performance over using either sensor independently across all models. This improvement should be most noticeable in more complex models like XGBoost, which can capture non-linear interactions and are robust to noisy features.

However, not all models may benefit equally from sensor fusion. Simpler models like Logistic Regression and Gaussian Naive Bayes may show marginal or no improvement, and may even experience performance degradation due to redundancy or overfitting. This would demonstrate the importance of model complexity and regularization when applying sensor fusion.

Model performance will be evaluated using:

- Accuracy
- Weighted F1 Score (to handle potential class imbalance)
- Generalization Gap (difference between training and testing accuracy)
- Confusion Matrices (to analyze specific misclassifications)

Additional expectations:

- XGBoost will achieve the highest overall performance.
- Gaussian Naive Bayes will be most sensitive to noisy fusion due to its independence assumptions.
- Some activities (e.g., walking upstairs vs. downstairs) may be harder to distinguish and show up in confusion matrices.

By comparing results across all sensor configurations and model types, I aim to identify whether sensor fusion consistently improves performance or if it introduces a point of diminishing returns. Feature importance analysis from XGBoost and cross-model comparisons will help explain how each model responds to various sensor inputs.