# A Scalable Framework for Sensor Data: Intra-Domain Transfer Learning in Wi-Fi Networks and Air Quality Monitoring

Raghav Chandna[1,2] Manal Karmoude[1,2,5] Phodiso Maroeshe[6, 2] Antonio Cervello Duato[7]
Phuti Rapheeha[1, 2] Ryan Peter McKenzie[1,2] Brenton Munhungewara[1,2] Donald Ngobeni[1, 2]
Jude Kong[3] Bevan Smith[4] Bruce Mellado[1,2]

[1]School of Physics, University of the Witwatersrand, Johannesburg, South Africa
[2]iThemba LABS, National Research Foundation, PO Box 722, Somerset West 7129, South Africa
[3]Dalla Lana School of Public Health, University of Toronto, Toronto, Canada
[4]School of Mechanical, Industrial and Aeronautical Engineering, University of the Witwatersrand, Johannesburg, South Africa
[5]Faculty of Sciences, Mohammed V University, Rabat, Morocco
[6]School of Computer Science, and Applied Mathematics, University of the Witwatersrand, Johannesburg, South Africa
[7]University of Valencia and CSIC (ES)

*African Society for Air Quality (ASAQ)*
*4 December 2025*

# The Challenge : Too Much Data, Too Few Resources

- Large campus/building WiFi networks generate massive amounts of data
  - User activity logs
  - Device connection counts
  - Continuous time-series data streams
- **Problem:** Training models on such scale is:
  - Computationally expensive
  - Time-consuming
  - Resource-intensive
- Traditional approach: Train new models for each dataset from scratch

# Our Solution: Intra-Domain Transfer Learning

## Core Idea: Learn Once, Apply Many Times

Instead of training new models for every similar problem, we:

1. **Train** a robust model on one large, rich WiFi dataset
2. **Adapt** this pre-trained model to new, similar datasets
3. **Reuse** learned patterns instead of learning from scratch

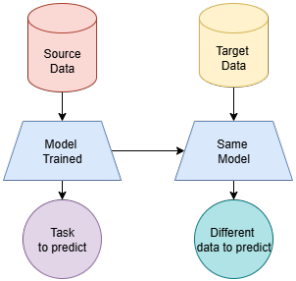



Figure: Our intra-domain transfer learning approach

# How It Works: Methodology Overview

**Step 1: Foundation Training**

- Train on large WiFi dataset
- Time-series forecasting problem
- Predict user/device counts
- Capture temporal patterns

**Step 2: Efficient Adaptation**

- Freeze early layers
- Fine-tune final layers
- Minimal retraining required
- Rapid deployment

## Key Innovation

Same computational efficiency can be achieved for other sensor networks

# Intra-Domain Transfer Learning: Technical Formulation

## Core Methodology

Intra-Domain Transfer Learning (IDTL) transfers learned representations **within the same domain** from a source dataset (abundant data) to a target dataset (limited data):

**Source Domain Training:**

$$\phi_S = \arg\min_{\phi} \mathcal{L}(\phi; D_S)$$

where:

- $\phi_S$: Optimal feature extractor from source domain
- $D_S$: Source WiFi dataset (large campus network)
- $\mathcal{L}_S$: MSE loss for time-series forecasting

**Target Domain Adaptation:**

$$\theta_T^* = \arg\min_{\theta} \mathcal{L}(\phi_S \circ h_T; D_T)$$

where:

- $h_T$: Task-specific head for target dataset
- $D_T$: Target WiFi dataset (different building)
- $\mathcal{L}_T$: Target task loss function

# Performance Results: IDTL vs. Training from Scratch

**Empirical Evaluation:**

| Metric | From Scratch | IDTL |
|---|---|---|
| RMSE | 20.54 | 20.58 |
| Average CPU Usage(%) | 20.80 | 12.65 |
| Total Training time (secs) | 49.88 | 11.13 |

Table: WiFi user count forecasting results

**Key Observations:**

- **Same Accuracy**: Negligible RMSE difference (0.04)
- **39% CPU Reduction**: From 20.8% to 12.65%
- **78% Faster Training**: From 49.88s to 11.13s

## Computational Complexity Analysis

- **Training from Scratch:**

$$\mathcal{O}(n \cdot d^3)$$

- **IDTL Adaptation:**

$$\mathcal{O}(n \cdot k \cdot d^2)$$

where $k \ll d$ (only fine-tuned layers)

## Efficiency Gains

- **Speed**: 4.5x faster training
- **Energy**: 40% less CPU utilization
- **Accuracy**: Maintained within 0.2%

# Layer-Wise Adaptation Strategy for WiFi

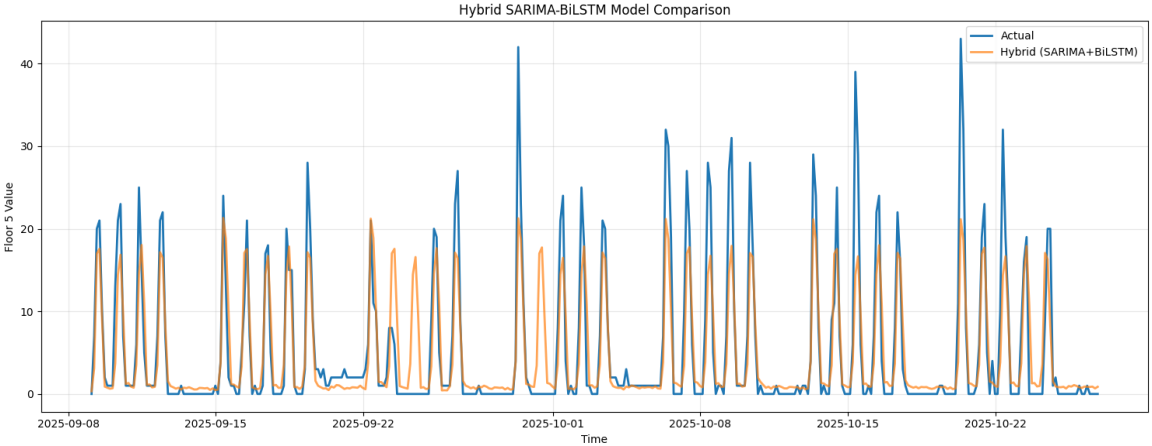
## Three-Stage Adaptation Process

1. **Freeze Feature Extractor Layers (80%)**
   - Preserves universal temporal patterns
   - Captures daily/weekly periodicity
   - Maintains load fluctuation recognition
   - Frozen parameters: No gradient updates required
2. **Fine-Tune Intermediate Layers (15%)**
   - Adapts to building-specific usage patterns
   - Adjusts for network density variations
   - Accommodates different user demographics
   - Only 15% of gradients are active
3. **Retrain Task-Specific Head (5%)**
   - Learns dataset-specific forecasting
   - Optimizes for local time-series characteristics
   - Minimizes adaptation overhead
   - Enables rapid deployment to new networks

# Baseline : Hybrid Model SARIMA-BiLSTM

**24-Hour WiFi User Count Forecast - Floor 5**

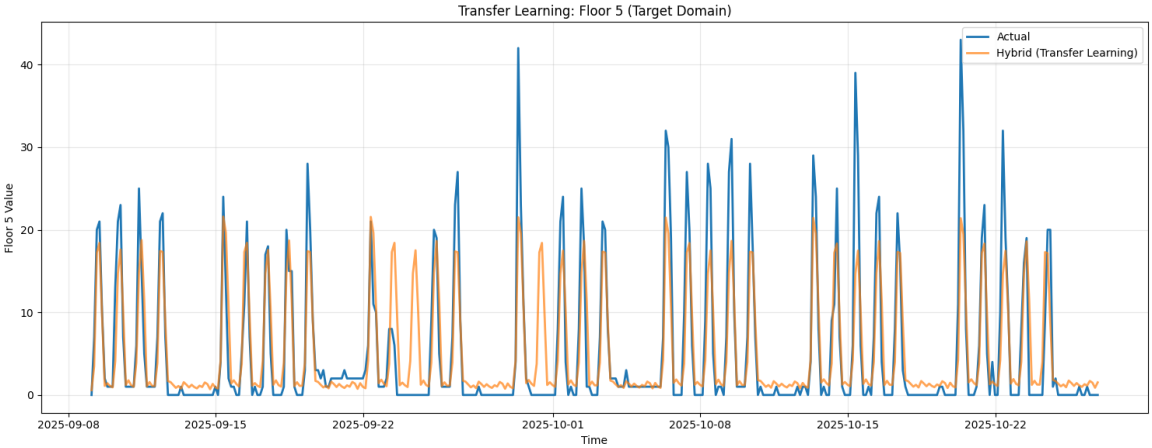
*Model trained from scratch on Floor 5 data(67.01 seconds)*



# Transfer : Hybrid Model SARIMA-BiLSTM

**24 Hour WiFi User Count Forecast - Floor 5**

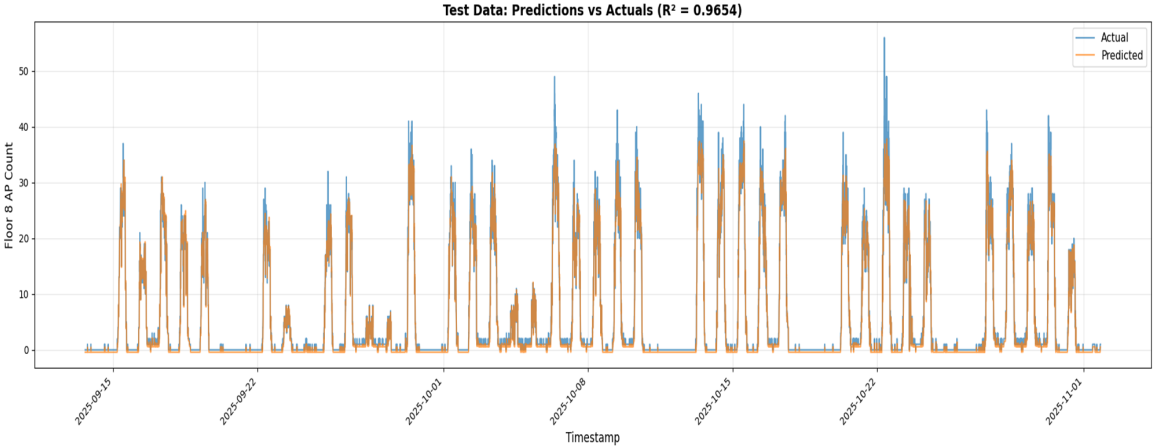
*Model adapted from floor 3 pre-trained model (11 seconds)*



# Baseline: Time Series Forecast from Scratch Training (80 minutes)

**24-Hour WiFi User Count Forecast - Floor 8**

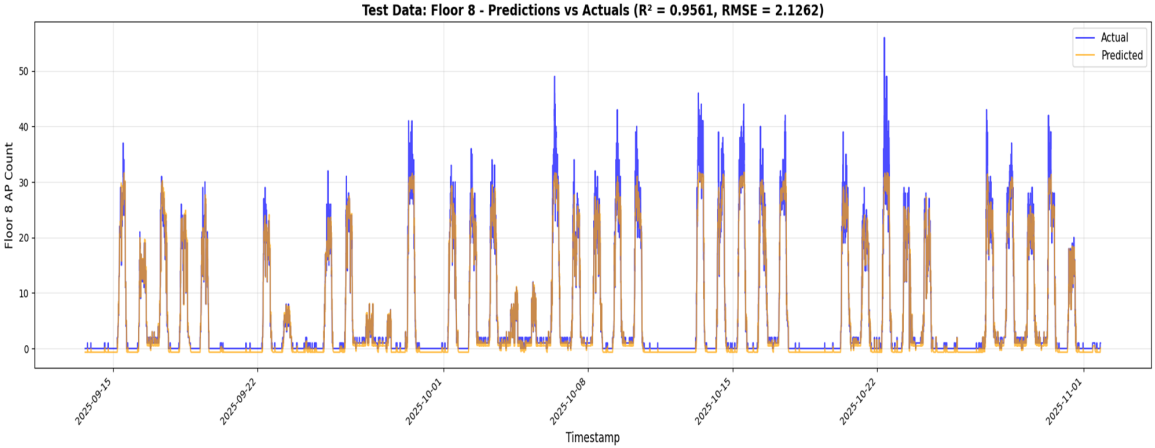
*Model trained from scratch on Floor 8 data (80 minutes training)*



# Transfer Learning: Time Series Forecast After Adaptation (14 seconds)

**24-Hour WiFi User Count Forecast - Floor 8**

*Model adapted from Floor 7 pre-trained model (14 seconds adaptation)*



# Results: Efficiency Without Sacrificing Accuracy

- **Performance:** Comparable accuracy to custom-trained models
- **Speed:** 70-80% reduction in training time
- **Resources:** Significant CPU/energy savings
- **Deployment:** Much faster model readiness

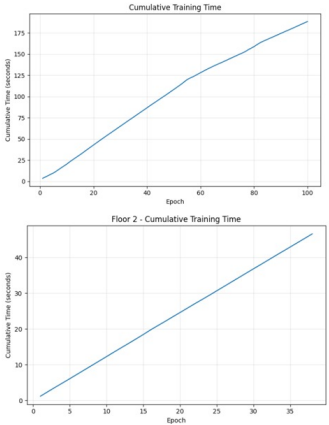



Figure: Training time comparison

# Beyond WiFi: Application to Air Quality Monitoring

**Why This Matters:**

- Same scalability challenges exist
- Similar time-series nature
- Resource-constrained environments
- Need for rapid deployment

**Our Approach:**

- Apply same methodology
- Train on WiFi, adapt for air quality
- Test cross-domain applicability
- Validate resource efficiency

## The Vision

A single framework for multiple sensor types: WiFi, air quality, temperature, occupancy, etc.

## Key Finding

Near-identical metrics with significantly reduced computational cost

# Broader Impact: Towards Sustainable Computing

## Environmental Benefits

- Reduced energy consumption for model training
- Lower carbon footprint for data centers
- Extended hardware lifespan
- More accessible AI for resource-constrained settings

## Practical Applications

- Smart building management
- Urban sensing networks
- IoT deployments
- Real-time monitoring systems

# Conclusion and Future Work

**What We've Shown:**

- Intra-domain transfer learning solves WiFi scalability
- Method is efficient, fast, and accurate  Same approach works for air quality monitoring
- Framework enables resource-efficient computing

**Next Steps:**

- Test with more sensor types
- Explore cross-domain combinations
- Develop automated adaptation
- Deploy in real-world settings

**Thank You! Questions?**