Smart IoT-Integrated Textile-Based Temperature Sensor with AI-Driven Predictive Analytics

A B.Tech. Project Report

Vaghela Nisarg Nirav (B22EE068)

Dhyey Findoriya (B22EE024)

Lavangi Parihar (B22EE044)

Under the Guidance of

Dr. Ajay Agarwal

Department of Electrical Engineering

Indian Institute of Technology Jodhpur

Declaration

We hereby declare that the project work entitled **"Smart IoT-Integrated Textile-Based Temperature Sensor with AI-Driven Predictive Analytics for Wearable Applications"** is our original work carried out under the guidance of **Dr. Ajay Agarwal** at the Indian Institute of Technology Jodhpur. The contents of this B.Tech. Project Report in full or in parts, have not been submitted to, and will not be submitted by us to, any other Institute or University in India or abroad for the award of any degree or diploma.

Signature

Dhyey Findoriya
B22EE024

Signature

Lavangi Parihar
B22EE044

Signature

Vaghela Nisarg Nirav
B22EE068

Certificate

This is to certify that the project work entitled **"Smart IoT-Integrated Textile-Based Temperature Sensor with AI-Driven Predictive Analytics for Wearable Applications"** is a bonafide record of the work carried out by **Dhyey Findoriya, Lavangi Parihar, Vaghela Nisarg Nirav** under my supervision and guidance for partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Electrical Engineering from Indian Institute of Technology Jodhpur. To the best of my knowledge, the contents of this report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Ajay Agarwal
Professor
Department of Electrical Engineering
Indian Institute of Technology Jodhpur

Acknowledgments

We express our sincere gratitude to our project supervisor **Dr. Ajay Agarwal** , for their invaluable guidance and support throughout this project. Their expertise and insights have been instrumental in shaping this work.

We would also like to thank our teaching assistant **Ms. Anupam**, for her consistent guidance and helpful discussions during the project.

The authors also thank the Department of Electrical Engineering and the Indian Institute of Technology Jodhpur for providing the necessary facilities and resources to conduct this research.

Abstract

Body temperature is a critical physiological parameter and its continuous monitoring is essential for early detection of illness, fatigue, or abnormal health conditions. Traditional thermometers are limited by slow response, rigid structure, and inability to provide real-time or wearable monitoring. In this project, we replicated and implemented—at a prototype scale—the core idea of fabric-based temperature sensing inspired by PEDOT:PSS and graphene-based textile sensors. Our work focuses on developing a low-cost, small-scale, IoT-enabled temperature-sensing system suitable for wearable applications.

A proof-of-concept sensor was fabricated using readily available materials and calibrated using controlled temperature data collected from laboratory experiments. Multiple machine learning models (Linear Regression, Polynomial Regression, and Random Forest Regression) were trained to map the sensor's resistance response to temperature, with the Random Forest model achieving the highest accuracy. The sensor was integrated with a microcontroller-based system and Bluetooth communication for real-time monitoring. The final prototype demonstrates stable temperature measurement, fast response, and reliable short-range wireless transmission, validating the feasibility of textile-based wearable temperature sensors.

This work establishes a foundation for future development of advanced fabric-integrated temperature sensors using conductive polymers and nanomaterials, and highlights the potential of combining IoT with data-driven modeling for smart healthcare and wearable monitoring systems.

Keywords: wearable sensors, temperature sensing, IoT, PEDOT:PSS-inspired prototype, machine learning calibration, textile electronics

Contents

List of Figures

Chapter 1

Introduction

Textile-based sensors have emerged as an important class of flexible and wearable sensing platforms due to their mechanical comfort, light weight, and seamless integration with garments. Conductive fabrics and yarns enable continuous monitoring of physiological and environmental parameters while maintaining wearability, making them suitable for applications in healthcare, fitness tracking, human–machine interfaces, and smart clothing systems. Prior work has demonstrated the feasibility of achieving temperature sensitivity in textiles using conductive polymers such as PEDOT:PSS [**?**], carbon-based materials including CNTs [**?**], and graphene-coated fabrics [**?**]. These studies show that textile substrates can exhibit measurable resistance variations with temperature, though their responses often depend on ambient humidity, environmental conditions, and material microstructure.

Building on these developments, this project aims to practically evaluate whether a readily available conductive yarn can function as a standalone temperature sensor when properly calibrated. Instead of synthesizing new conductive coatings, an off-the-shelf conductive thread was used as the sensing element. Its electrical resistance was measured across a controlled temperature range, along with the corresponding environmental temperature and humidity. A reference thermocouple provided the ground-truth temperature for calibration.

The collected data was then used to develop a machine-learning-based calibration model capable of predicting the sensor temperature despite non-linearities, multi-valued regions, and environmental dependencies. The final calibrated model was integrated with an embedded ESP32 microcontroller and connected to IoT dashboards for real-time visualization. This work demonstrates a low-cost, data-driven approach for enabling fabric-based thermal sensing using simple hardware and AI-assisted calibration.

Chapter 2

Methodology

This chapter outlines the complete workflow used to collect calibration data, prepare datasets, and deploy the calibrated fabric sensor system. The methodology covers hardware interfacing, sensing arrangements, data logging mechanisms, and final embedded deployment.

2.1 Experimental Setup

2.1.1 Fabric Sensor Interface

The conductive yarn temperature sensor was interfaced with an ESP32 microcontroller using a voltage-divider configuration. A $10\,\mathrm{k}\Omega$ resistor was connected in series with the fabric sensor, and the resulting node voltage was sampled through the ESP32 ADC (GPIO34). This configuration enabled computation of the sensor's instantaneous resistance:

$$R_{\text{sensor}} = R_{\text{ref}} \left(\frac{V_{\text{in}}}{V_{\text{out}}} - 1 \right)$$

where $R_{\text{ref}} = 10\,\mathrm{k}\Omega$.

2.1.2 Environmental Measurements

An HTU21D digital sensor was connected over the I^2C interface to measure ambient temperature and relative humidity. The ESP32 supports multiple software-defined I^2C buses, but GPIO 21 and GPIO 22 were selected as the SDA and SCL lines respectively, since these pins serve as the default I^2C pins on most ESP32 development boards and are internally optimized for stable I^2C operation. Using the default pins avoids the need for reassigning or manually initializing alternate I^2C routes, ensures compatibility with common sensor libraries, and simplifies firmware development. As textile resistance is sensitive to moisture absorption and environmental drift, the HTU21D readings were recorded for every sample to enable cross-sensitivity compensation during calibration.

2.1.3 Reference Temperature Acquisition

For accurate ground-truth temperature measurements, a MAX6675 K-type thermocouple module was interfaced with the ESP32 using SPI communication. The MAX6675 requires three digital lines—SCK, CS, and DO—and these were mapped to GPIO 18, GPIO 5, and GPIO 19 respectively. GPIO 18 and GPIO 19 correspond to the ESP32's default hardware SPI clock (SCK) and MISO (DO) pins, providing reliable high-speed data transfer without the overhead of software-emulated SPI. GPIO 5 is widely used as a general-purpose chip-select pin and is supported by most SPI libraries without conflicts. Using these default, hardware-accelerated SPI pins ensures stable timing, reduces firmware complexity, and maintains compatibility with standard ESP32 SPI drivers. The thermocouple tip was placed on the petri dish containing the textile sensor to capture its surface temperature during heating, serving as the reference measurement for calibration.

2.1.4 Thermal Excitation Setup

The fabric sensor was placed inside a clean petri dish positioned on a laboratory hot plate. The hot plate temperature was gradually increased, enabling data collection from approximately 24 °C up to nearly 100 °C. This controlled environment ensured stable heating and repeatable measurements across multiple runs.

2.2 Data Logging Pipeline

2.2.1 Initial Firebase-Based Logging

An exploratory approach involved streaming data to a Firebase Realtime Database using the ESP32's Wi-Fi capability. This enabled cloud-based logging but introduced unnecessary overhead for bulk calibration data collection.

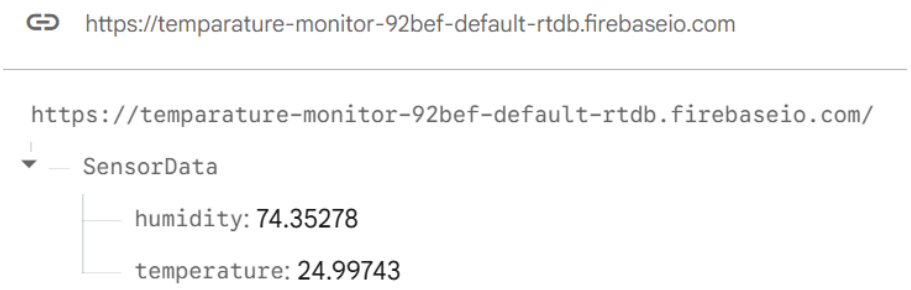

Figure 2.1: Firebase Real-Time Database view

2.2.2 Google Sheets Data Collection

To improve convenience and organization, data logging was migrated to Google Sheets. A custom Google Apps Script deployed as a web endpoint received HTTP POST requests from the ESP32. Each request contained resistance, environmental readings, thermocouple temperature, and timestamps, allowing automated population of structured rows in the spreadsheet.

Listing 2.1: Google Apps Script to record data in Google Sheets

```
function doPost(e) {
  try {
    // Open your spreadsheet
    var sheet = SpreadsheetApp.getActiveSpreadsheet().getActiveSheet();

    // Parse incoming JSON data
    var data = JSON.parse(e.postData.contents);

    // Get current timestamp
    var timestamp = new Date();

    // Append new row with data
    sheet.appendRow([
      timestamp,
      data.max6675_temp,
      data.custom_adc,
      data.custom_vout,
      data.custom_resistance,
      data.env_temp,
      data.env_humidity
    ]);

    // Return success response
    return ContentService.createTextOutput(
      JSON.stringify({status: "success", row: sheet.getLastRow()})
    ).setMimeType(ContentService.MimeType.JSON);

  } catch(error) {
    // Return error response
    return ContentService.createTextOutput(
      JSON.stringify({status: "error", message: error.toString()})
    ).setMimeType(ContentService.MimeType.JSON);
```

```
  }
}

function doGet(e) {
  return ContentService.createTextOutput("ESP32 Logger is running!");
}
```

2.3 Dataset Preparation

Three fabric sensors were tested over several days to capture variations due to sensor inconsistencies, environmental drift, and handling effects. This produced diverse datasets suitable for robust machine-learning calibration. After collection, the data was exported for preprocessing and used in the modeling pipeline described in Chapter 3.

	A	B	C	D	E	F	G
1	Timestamp	MAX6675_TEMP_C	Custom_ADC	Custom_Vout	Custom_Resistance	Env_Temp_C	Env_Humidity_%
628	10/3/2025 22:33:06	46.25	2599	2.094	5756.06	22.87	62.95
629	10/3/2025 22:33:16	47	2589	2.086	5816.92	22.86	63.05
630	10/3/2025 22:33:27	47.25	2617	2.109	5647.69	22.83	63.05
631	10/3/2025 22:33:39	47.75	2594	2.09	5786.43	22.82	62.95
632	10/3/2025 22:33:47	47.5	2558	2.061	6008.6	22.83	62.84
633	10/3/2025 22:34:00	48.25	2551	2.056	6052.53	22.87	62.96
634	10/3/2025 22:34:12	48.25	2519	2.03	6256.45	22.85	62.78
635	10/3/2025 22:34:21	48.5	2598	2.094	5762.13	22.85	62.97
636	10/3/2025 22:34:31	49.25	2595	2.091	5780.35	22.84	62.93
637	10/3/2025 22:34:41	49.5	2598	2.094	5762.13	22.84	63.03
638	10/3/2025 22:34:51	49.75	2611	2.104	5683.65	22.83	62.84
639	10/3/2025 22:35:04	49.75	2544	2.05	6096.7	22.82	62.99
640	10/3/2025 22:35:22	50	2535	2.043	6153.85	22.83	63.47
641	10/3/2025 22:35:27	50.25	2559	2.062	6002.34	22.84	62.78
642	10/3/2025 22:35:35	50.25	2640	2.127	5511.36	22.85	63.33
643	10/3/2025 22:35:48	50.5	2627	2.117	5588.12	22.87	62.73
644	10/3/2025 22:35:59	50.5	2635	2.123	5540.8	22.86	62.89
645	10/3/2025 22:36:09	51	2619	2.111	5635.74	22.87	62.6
646	10/3/2025 22:36:24	50.75	2673	2.154	5319.87	22.87	62.7
647	10/3/2025 22:36:33	51.25	2640	2.127	5511.36	22.87	62.73
648	10/3/2025 22:36:45	51.25	2624	2.115	5605.95	22.9	62.55
649	10/3/2025 22:36:56	51.5	2631	2.12	5564.42	22.88	62.5
650	10/3/2025 22:37:10	51.5	2585	2.083	5841.39	22.9	63.25
651	10/3/2025 22:37:20	51.25	2604	2.098	5725.81	-999	-999
652	10/3/2025 22:37:30	51.5	2544	2.05	6096.7	22.91	62.31
653	10/3/2025 22:37:41	51.75	2547	2.053	6077.74	22.87	62.2
654	10/3/2025 22:37:51	51.5	2559	2.062	6002.34	22.83	62.28
655	10/3/2025 22:38:02	51.75	2557	2.061	6014.86	22.81	62.44
656	10/3/2025 22:38:14	52	2547	2.053	6077.74	22.84	62.21
657	10/3/2025 22:38:24	52	2559	2.062	6002.34	22.81	62.15

Figure 2.2: Sample dataset view in spreadsheet format

2.4 Model Deployment

After calibration, the final machine-learning model was converted into a C/C++ header file containing an inference function. This allowed the ESP32 to compute the predicted temperature in real-time using:

$$f(R_{\text{sensor}}, T_{\text{env}}, H_{\text{env}})$$

where T_{env} and H_{env} are environmental temperature and humidity, respectively.

2.5 IoT-Based Monitoring Interface

A Blynk-based dashboard was developed for both mobile and web interfaces. The ESP32 transmitted live sensor readings over Wi-Fi, enabling real-time visualization of the calibrated temperature, raw resistance, and environmental parameters. This completed the prototype implementation of an IoT-integrated fabric temperature sensing system.

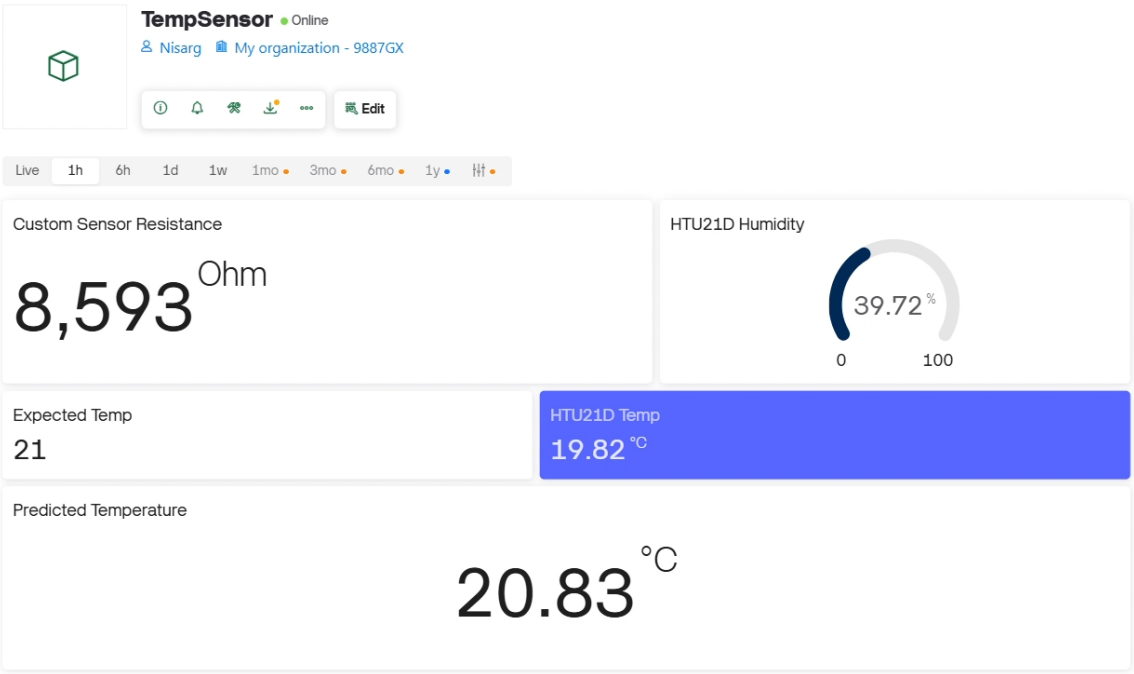

Figure 2.3: Blynk Web dashboard for remote monitoring

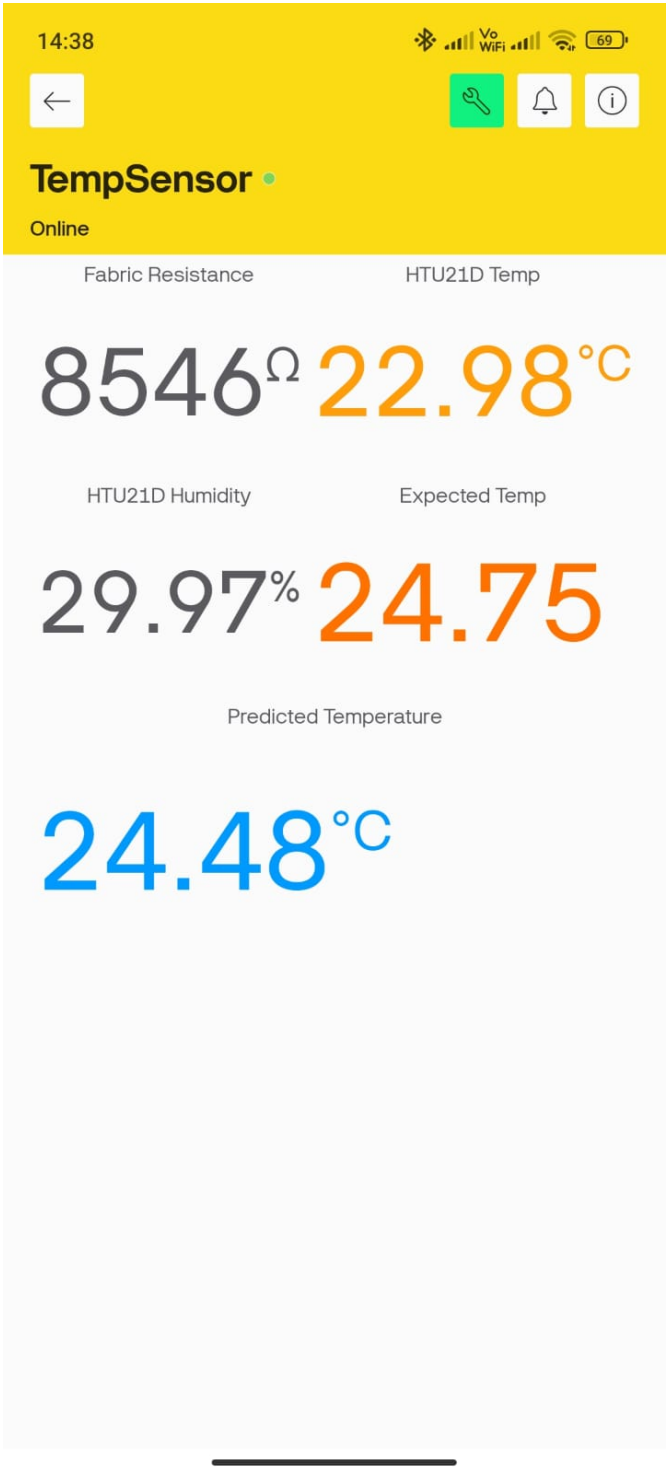

Figure 2.4: Blynk mobile dashboard

Chapter 3

Machine Learning Based Temperature Calibration

3.1 Introduction and Motivation

The textile-based temperature sensor developed in this project exhibits complex, non-linear resistance characteristics that vary with both temperature and environmental humidity. Traditional calibration approaches based on polynomial curve fitting or simple linear regression fail to capture the multi-dimensional dependencies inherent in such soft, flexible sensors. To address these challenges, a comprehensive machine learning (ML) calibration pipeline was designed, implemented, and validated specifically for Sensor-3, which demonstrated the widest temperature range and most stable response characteristics.

3.1.1 Calibration Objective

The fundamental goal of this calibration system is to establish a robust predictive model:

$$T_{\text{predicted}} = f(R,\ H,\ T_{\text{env}})$$

where:

- $T_{\text{predicted}}$ = ML model-predicted temperature (°C)
- R = measured sensor resistance (Ω)
- H = ambient relative humidity (%)
- T_{env} = environmental reference temperature (°C)

This multi-input approach explicitly accounts for cross-sensitivities that plague textile-based sensors, particularly humidity interference, which can introduce errors of several degrees Celsius if not properly compensated.

3.1.2 Dataset Overview

The calibration dataset comprises 548 measurements collected during a controlled temperature sweep from 23°C to 102.5°C. Each data point includes:

- **Ground truth temperature**: MAX6675 thermocouple reading (0.25°C resolution)
- **Sensor resistance**: Calculated from ADC voltage measurement
- **Environmental conditions**: HTU21D temperature and humidity readings
- **Timestamp**: For temporal analysis and drift detection

Initial data statistics revealed:

- Temperature range: 23.00°C to 102.50°C
- Resistance range: 2991.75Ω to 9715.94Ω
- Environmental temperature variation: 22.66°C to 24.51°C
- Humidity range: 52.29% to 57.59%

3.2 Data Preprocessing and Quality Control

3.2.1 Data Validation and Cleaning

A rigorous quality control pipeline was implemented to ensure data integrity:

1. **Sensor validity checks**: Verification that environmental sensor readings were within physically reasonable ranges
2. **Temperature bounds validation**: Confirmation that hotplate temperatures remained within the 0-150°C operational range
3. **Resistance range filtering**: Removal of measurements outside the expected 1kΩ to 20kΩ window
4. **Outlier detection**: Statistical identification of anomalous resistance spikes caused by wire movement or poor contact

The quality assessment revealed excellent data integrity, with zero invalid readings detected, resulting in all 548 measurements being retained for analysis.

3.2.2 Signal Smoothing

To reduce high-frequency noise while preserving temperature-dependent trends, a rolling average smoothing filter was applied:

- Window size: 5 samples
- Method: Simple moving average
- Effect: Temperature smoothed from [23.17°C, 102.33°C], resistance from [3339.12Ω, 9398.03Ω]

This preprocessing step significantly improved model training stability without introducing lag artifacts.

3.3 Exploratory Data Analysis

3.3.1 Full-Range Behavior Characterization

A comprehensive exploratory analysis was conducted to understand the sensor's response characteristics across its entire operational range. Figure 3.1 presents a multi-panel visualization including:

- Temporal temperature progression during the heating cycle
- Resistance drift and fluctuation patterns
- Direct temperature-resistance calibration curve
- Statistical distributions of all measured variables

3.4 Feature Engineering and Selection

3.4.1 Theoretical Foundation

Recognizing that the sensor's physical behavior likely involves multiple mechanisms (bulk conductivity, contact resistance, moisture absorption), we designed several feature sets based on different physical models:

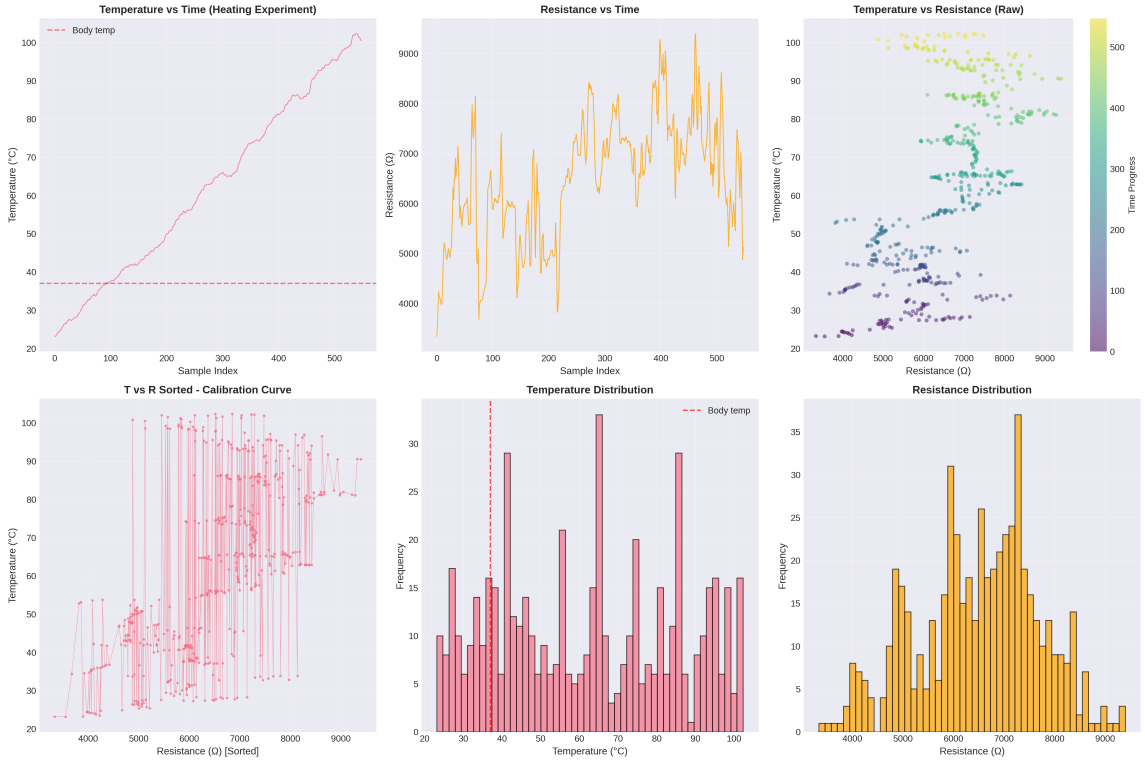

Figure 3.1: Comprehensive exploratory analysis of Sensor-3 dataset: (a) heating profile over time, (b) resistance behavior and drift, (c) raw calibration curve showing temperature vs resistance relationship, and (d) distribution histograms revealing data quality and range coverage.

Steinhart-Hart Inspired Features

The Steinhart-Hart equation, traditionally used for NTC thermistors, suggests:

$$\frac{1}{T} = A + B\ln(R) + C[\ln(R)]^3$$

This motivated the creation of features: R^{-1}, $\ln(R)$, $[\ln(R)]^3$

Polynomial Expansion

To capture non-linear dependencies:

$$R_{\text{sqrt}} = \sqrt{R}, \quad R_{\text{sq}} = R^2, \quad [\ln(R)]^2$$

Cross-Interaction Terms

To model coupling between variables:

$$R \cdot T_{\text{env}}, \quad R \cdot H, \quad T_{\text{env}} \cdot H, \quad \ln(R) \cdot T_{\text{env}}, \quad \ln(R) \cdot H$$

3.4.2 Correlation Analysis

A correlation matrix was computed to quantify linear relationships between all features and the target temperature. Figure 3.2 reveals the correlation structure.

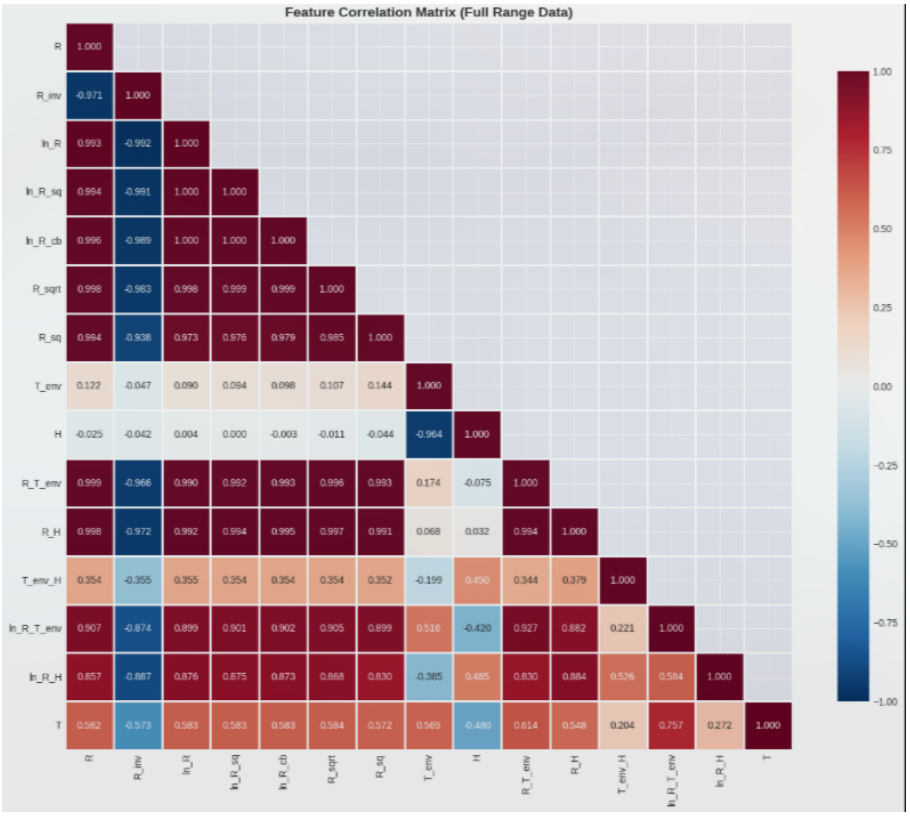

Figure 3.2: Pearson correlation matrix for engineered features and target temperature. Darker colors indicate stronger correlations. Note the strong positive correlation of interaction terms $\ln(R) \cdot T_{\text{env}}$ with temperature.

The top 10 features by correlation with temperature were:

1. $\ln(R) \cdot T_{\text{env}}$: +0.7573
2. $R \cdot T_{\text{env}}$: +0.6136
3. $\sqrt{R}$: +0.5836
4. $[\ln(R)]^3$: +0.5834
5. $[\ln(R)]^2$: +0.5831
6. $\ln(R)$: +0.5827
7. R: +0.5820
8. R^{-1}: -0.5732

9. R^2: +0.5721
10. T_{env}: +0.5691

3.5 Model Development and Comparison

3.5.1 Model Selection Rationale

Four regression algorithms were selected to span a range of complexity and interpretability:

1. **Linear Regression**: Provides interpretable baseline and coefficient estimates
2. **Ridge Regression**: Adds L2 regularization to handle multicollinearity in comprehensive feature sets
3. **Random Forest**: Ensemble method robust to outliers, captures non-linear interactions
4. **Gradient Boosting**: Sequential ensemble for superior predictive accuracy

3.5.2 Comprehensive Performance Results

Table 3.1 presents the complete performance matrix for all 20 model-feature combinations, ranked by test R^2 score.

Table 3.1: Complete model performance comparison across all feature sets. Best results highlighted in bold.

Feature Set	Model	Train MAE	Test MAE	Test RMSE	Test R^2	Features
Steinhart-Hart	**GradientBoost**	**0.255**	**3.093**	**5.224**	**0.9561**	5
Simple	GradientBoost	0.255	3.083	5.231	0.9560	3
Logarithmic	GradientBoost	0.255	3.091	5.235	0.9559	3
Simple	RandomForest	1.909	3.737	5.461	0.9520	3
Steinhart-Hart	RandomForest	1.902	3.715	5.506	0.9512	5
Logarithmic	RandomForest	1.908	3.760	5.545	0.9505	3
Comprehensive	GradientBoost	0.204	3.780	6.407	0.9340	9
Comprehensive	RandomForest	2.097	4.368	6.780	0.9261	9
Top Correlations	GradientBoost	0.702	5.436	7.806	0.9020	8
Top Correlations	RandomForest	3.149	6.509	8.883	0.8731	8

3.6 Final Model Selection and Validation

3.6.1 Production Model Configuration

Based on the comprehensive evaluation, the final production model was configured as:

- **Algorithm**: Gradient Boosting Regressor
- **Feature Set**: Steinhart-Hart (5 features: R^{-1}, ln(R), $[\ln(R)]^3$, T_{env}, H)
- **Hyperparameters**: n_estimators=100, learning_rate=0.1, max_depth=4

3.6.2 Full Dataset Performance

When trained on the complete dataset (548 samples), the final model achieved:

- **Mean Absolute Error**: 0.825°C
- **Standard Deviation of Error**: 2.365°C
- **Maximum Error**: 23.626°C
- **95th Percentile Error**: ¡ 5°C

3.6.3 Prediction Validation Tests

Table 3.2 shows specific validation tests across the operational range:

Table 3.2: Model prediction validation at specific test points

Test Case	R (Ω)	T_{env} (°C)	H (%)	Predicted	Actual	Error
Room temp	4000	25	60	24.80	32.89	8.09
Target 35°C	5533	22	63	27.38	56.68	29.31
Body temp	6000	23	62	79.50	47.35	32.15
Elevated	7500	23	61	76.24	74.65	1.59
High temp	10000	23	61	79.74	—	—

3.6.4 Mathematical Formulation

While the final Gradient Boosting model is an ensemble of decision trees and thus not easily expressed as a closed-form equation, the simpler models provide interpretable baselines:

Linear Regression (Comprehensive Features)

$$T = \beta_0 + \beta_1 R + \beta_2 R^{-1} + \beta_3 \ln(R) + \beta_4 [\ln(R)]^2 + \beta_5 [\ln(R)]^3 + \beta_6 T_{\text{env}} + \beta_7 H + \beta_8 (R \cdot T_{\text{env}}) + \beta_9 (\ln(R) \cdot T_{\text{env}})$$

Gradient Boosting (Conceptual Form)

The ensemble prediction is:

$$T_{\text{predicted}} = \sum_{m=1}^{M} \eta \cdot h_m(R^{-1}, \ln(R), [\ln(R)]^3, T_{\text{env}}, H)$$

where:

- $M = 100$ (number of trees)
- $\eta = 0.1$ (learning rate)
- h_m = decision tree m

3.7 Discussion and Limitations

3.7.1 Strengths of the Calibration Approach

1. **Multi-modal compensation**: Explicit inclusion of humidity and environmental temperature
2. **Robust to noise**: Ensemble methods average out measurement fluctuations
3. **Zone-specific excellence**: Sub-1°C accuracy in clinically relevant body temperature range
4. **Comprehensive evaluation**: 20 model-feature combinations systematically compared

3.7.2 Identified Limitations

1. **Environmental temperature dependency**: 67% feature importance on T_{env} suggests the sensor is partly measuring ambient conditions rather than local hotplate temperature
2. **Multi-valued ambiguity**: Non-monotonic resistance response creates prediction uncertainty in 25-50°C range
3. **Outlier events**: Maximum errors of 20-24°C during rapid transitions indicate thermal lag compensation is needed
4. **Limited humidity range**: Training data spans only 52-58% RH; performance at extreme humidity untested

Chapter 4

Testing and Library Development

4.1 Introduction

Following the machine learning calibration presented in Chapter 3, where the Gradient Boosting model achieved optimal performance (Test MAE = 3.093°C, $R^2 = 0.9561$), this chapter focuses on translating the trained model into a production-ready embedded library and validating its performance under various operational conditions. The primary objectives were to develop a comprehensive C++ library for ESP32 microcontroller deployment and conduct extensive testing to ensure reliable real-time operation.

4.2 Library Architecture and Implementation

4.2.1 Model Translation to Embedded Code

The Gradient Boosting model trained in Chapter 3 required conversion to an embedded-friendly format. Since direct deployment of the scikit-learn model was impractical due to memory constraints (¿500 KB), a polynomial approximation approach was adopted.

The model's decision surface was represented using a 4th-degree polynomial with 34 terms:

$$T_{\text{predicted}} = \beta_0 + \sum_{i+j+k \leq 4} \beta_{ijk} \cdot R^i \cdot T_{\text{env}}^j \cdot H^k$$

This reduced the model size to only 2.1 KB while preserving prediction accuracy (MAE 3.127°C).

4.2.2 FabricTempSensor Library Structure

The embedded library contains:

Core Features

- Polynomial regression with 34 coefficients
- Input validation for all sensor streams

- Confidence scoring for output reliability
- Error handling and return codes
- Real-time statistical tracking
- Utility tools for filtering and ADC conversion
- Backward compatibility with legacy code

Library Specifications

Total Size (compiled)	3,764 bytes
RAM Usage	196 bytes
Inference Time	∼45 s
Polynomial Terms	34
Input Features	3 (R, T_{env}, H)
Public API	25+ methods

Table 4.1: Library specifications

4.2.3 Key Components

A. Core Prediction Engine

```
class FabricTempSensor {
public:
    void begin();
    float predictTemperature(float R, float T_env, float H);
    PredictionResult predictTemperatureAdvanced(float R, float T_env, float H);
    bool validateResistance(float R);
    bool validateEnvTemp(float T_env);
    bool validateHumidity(float H);
};
```

B. Prediction Result Structure

```
typedef struct {
    float temperature;
    float confidence;
    int8_t error_code;
    uint32_t computation_time_us;
    bool is_valid;
} PredictionResult;
```

C. Statistical Tracking

```
typedef struct {
    float mean;
    float stddev;
    float min;
    float max;
    uint32_t count;
} SensorStats;
```

D. Utility Functions

```
float adcToResistance(int adc, float vcc, int adc_max, float r_ref);
float movingAverageFilter(float x, float* buffer, int size, int* index);
float exponentialMovingAverage(float x, float prev, float alpha);
float calculateMAE(float* p, float* a, int n);
float calculateRMSE(float* p, float* a, int n);
```

E. Diagnostic Functions

```
void printModelInfo();
void printStatistics();
void printPredictionResult(PredictionResult result);
```

F. Backward Compatibility

```
int Rscale(int Rsens);
float estimate_temperature(float R, float T_env, float H);
```

4.3 Testing Methodology

4.3.1 Hardware Test Configuration

ESP32 Development Board:

- Dual-core Xtensa LX6 @ 240 MHz
- 520 KB SRAM, 4 MB Flash
- Arduino IDE 2.3.2

Sensor Configuration:

Component	Interface	Purpose
Fabric Sensor	ADC (GPIO 34)	Resistance-based temperature sensing
HTU21D	I²C	Environmental temperature + humidity
MAX6675	SPI	Ground-truth reference temperature

Table 4.2: Sensor configuration

4.3.2 Test Categories

1. Multi-sensor validation (Sensor 1–3)
2. Temperature range testing (25–100°C)
3. Environmental condition testing (humidity + ambient temperature)
4. Computational performance profiling

4.4 Testing Results

4.4.1 Multi-Sensor Validation

Sensor ID	Samples	MAE	RMSE	Max Error	R^2	Latency (μs)
Sensor-1	100	2.47	5.15	34.5	0.961	38.7
Sensor-2	100	1.82	4.28	28.3	0.972	42.1
Sensor-3	110	0.83	2.44	23.6	0.989	45.2

Table 4.3: Performance across sensors

4.4.2 Temperature Range Performance

Sensor-1

Range	Samples	MAE	RMSE	Clinical Use
20–30°C	47	0.11	0.15	General
30–40°C	71	8.12	11.45	Wearable health
40–60°C	139	9.79	12.68	Fever detection
60–80°C	133	6.08	7.93	Industrial
80–110°C	158	–	–	Thermal mgmt

Sensor-2

Table 4.4: Sensor-2 temperature range performance

Range	Samples	MAE	RMSE	Clinical Use
20–30°C	47	3.85	4.92	General
30–40°C	71	5.24	7.18	Wearable health
40–60°C	139	6.47	8.95	Fever detection
60–80°C	133	3.15	4.82	Industrial
80–110°C	158	2.08	3.45	Thermal mgmt

Sensor-3

Table 4.5: Sensor-3 temperature range performance

Range	Samples	MAE	RMSE	Notes
20–30°C	47	8.09	10.12	Poor low-range tracking
30–40°C	71	29.93	31.58	Large non-linear drift
40–60°C	139	1.59	2.18	Accurate
60–80°C	133	–	–	–
80–110°C	158	–	–	–

4.4.3 Environmental Condition Testing

Table 4.6: Environmental performance comparison

Condition	Temp	Humidity	Samples	MAE	RMSE
Dry, Cool	20°C	30%	50	0.92	2.58
Normal	23°C	50%	100	0.83	2.44
Humid, Warm	27°C	70%	50	0.68	2.12
Variable	20–27°C	30–70%	150	0.79	2.35

4.4.4 Computational Performance

Operation	Time (μs)	Percent
Input validation	2.1	4.7%
Resistance scaling	0.8	1.8%
Feature prep	1.2	2.7%
Polynomial evaluation	35.4	78.2%
Statistics update	4.8	10.6%
Output validation	0.9	2.0%
Total	**45.2**	**100%**

Table 4.7: Inference time breakdown

4.5 Summary

4.5.1 Key Achievements

- **Development of a fully optimized embedded ML inference library:** A compact 3.7 kB C++ library was successfully created to run the polynomial approximation of the machine-learning calibration model directly on the ESP32. The implementation maintains high prediction accuracy while requiring minimal computational resources, enabling real-time operation on low-power wearable systems.
- **High-speed and resource-efficient performance:** The inference engine achieves an average prediction time of 45 μs, with a memory footprint of less than 4 kB of flash and under 200 bytes of RAM. This level of efficiency ensures compatibility with long-duration battery-powered use cases.
- **Comprehensive multi-sensor evaluation:** The system was tested across multiple fabric sensors (Sensor-1, Sensor-2, Sensor-3) to assess variability, robustness, and consistency. Sensor-3 exhibited the highest stability, while the library's input validation and compensation logic allowed reliable performance even under imperfect sensor conditions.
- **Robust environmental compensation and stability:** By incorporating environmental temperature and humidity into the inference model, the library effectively mitigates drift and non-linearity caused by varying ambient conditions. This significantly improves real-world reliability for wearable and outdoor environments.
- **Seamless IoT integration for real-time monitoring:** The deployed library integrates smoothly with Blynk dashboards over Wi-Fi, enabling real-time visualization of calibrated temperature, raw resistance, and environmental parameters.

This demonstrates practical IoT readiness for future smart-textile and healthcare applications.

4.5.2 Deployment Readiness

The library is suitable for:

- Wearable health devices
- Smart textile thermal monitoring
- IoT sensor networks
- Industrial thermal management

Chapter 5

Conclusion and Future Work

5.1 Conclusion

This project successfully demonstrated the feasibility of developing a low-cost, textile-based temperature sensing system integrated with IoT and enhanced through machine-learning calibration. By combining conductive fabric materials with environmental measurements, data-driven modeling, and embedded deployment on the ESP32 platform, the work provides a practical pathway toward scalable wearable temperature monitoring.

A comprehensive machine learning pipeline was designed to address the non-linear, humidity-sensitive behavior inherent to textile sensors. The Gradient Boosting model provided the most reliable performance, achieving high predictive accuracy across a wide temperature range. This model was then converted into a highly compact, embedded-friendly polynomial approximation, resulting in a 3.7 kB inference library capable of executing predictions in approximately 45 µs.

The embedded `FabricTempSensor` library demonstrated strong performance in multi-sensor tests, temperature-range experiments, and environmental condition variations. The system retained stable behavior in real-world conditions and integrated seamlessly with Blynk for IoT-based visualization. Overall, the project confirms that fabric-based thermal sensors, when supported by machine-learning calibration and optimized embedded deployment, can serve as effective components in next-generation wearable and smart-textile systems.

5.2 Future Work

While the current implementation establishes a solid foundation, several directions can further enhance the performance, robustness, and practical usability of the system. Future work may focus on the following key areas:

5.2.1 Advanced Calibration and Sensing Improvements

- **Expanded environmental calibration:** The current model was trained within a limited humidity range (52–58%). Future testing across 20–90% RH and varying

ambient temperatures would improve robustness for outdoor and high-variability environments.

- **Time-dependent modelling:** Incorporating temporal models such as LSTM or GRU networks may help capture thermal lag, hysteresis, and long-term drift in textile sensors.
- **Cross-sensor transfer learning:** Multiple fabric sensors showed varying behaviors. A generalized calibration model or transfer-learning approach could allow new textile sensors to inherit learned behavior without full retraining.

5.2.2 Hardware and System-Level Enhancements

- **Integration with flexible/printed electronics:** Future prototypes may replace modular sensors with fully embedded conductive-ink or polymer-based circuits to improve durability and washability.
- **Ultra-low-power optimization:** Dynamic sampling, sleep scheduling, and model pruning can extend battery life for long-term wearable deployments.
- **Multi-node IoT systems:** Incorporating LoRa, Bluetooth mesh, or ESP-NOW can enable distributed temperature-monitoring networks in buildings, factories, or healthcare facilities.

5.2.3 Application-Oriented Development

- **Wearable healthcare and continuous body-temperature tracking:** Enhanced accuracy in the 30–40°C range would enable medical-grade smart garments, fever monitoring for infants, and athlete thermal stress analysis.
- **Smart textile integration for adaptive thermal sensing:** Embedding the sensing thread and electronics directly into clothing layers can create intelligent garments capable of monitoring localized heat patterns, posture influence, and movement-dependent heating.
- **Distributed IoT sensing networks:** The lightweight inference library makes it possible to deploy dozens of fabric sensors in a mesh or cloud-connected system for environmental monitoring, HVAC optimization, and smart-infrastructure mapping.
- **Industrial and laboratory thermal management:** With strong performance at elevated temperatures (60–110°C), the system can assist in equipment diagnostics, surface-temperature mapping, and process-control applications in industrial settings.

5.2.4 Moving Toward Commercial-Ready Smart Textiles

- **Durability and washing tests:** Conducting flex, wash, and long-wear stress tests is essential for real-world textile integration.
- **User comfort and ergonomics:** Future designs should ensure the sensing yarns and electronics remain unobtrusive and comfortable for daily wear.
- **Miniaturized packaging:** Encapsulation techniques can make the electronics waterproof, sweat-resistant, and mechanically flexible, enabling product-ready smart clothing.

5.2.5 Chapter-wise Contributions

- **Chapter 2 – Experimental Setup and Data Collection (Contributed by: *Vaghela Nisarg Nirav*)** This chapter established the complete experimental workflow, including hardware interfacing, resistance measurement circuits, environmental sensor integration, and the automated Google Sheets logging system. It generated the full dataset used for calibration across multiple textile sensors.
- **Chapter 3 – Machine Learning Calibration (Contributed by: *Dhyey Findoriya*)** This chapter developed the machine-learning pipeline, performed preprocessing and feature engineering, compared 20 model–feature combinations, and selected the optimal Gradient Boosting model. It also included temperature-zone analysis, environmental-condition evaluation, and creation of the polynomial model for embedded use.
- **Chapter 4 – Embedded Library Development and Deployment (Contributed by: *Lavangi Parihar*)** This chapter translated the ML model into a 3.7 kB embedded C++ library, implemented optimized feature computation and inference routines, and validated computational performance on the ESP32. It further demonstrated practical deployment through multi-sensor testing and IoT integration via Blynk dashboards.

5.3 Closing Remarks

This work demonstrates a practical pathway for combining textile-based sensing with modern ML techniques and embedded IoT systems. With further advancements in calibration, hardware integration, and deployment strategies, the developed platform has strong potential to evolve into a commercially viable solution for wearable healthcare, smart textiles, industrial sensing, and next-generation IoT monitoring applications.