

AI-Assisted Thermal Modeling and Predictive Management for Energy-Efficient Mobile Computing Devices

Jose Genaro Gonzalez-Hernandez^{1,2}, Loreto Gonzalez-Hernandez³,
Mariana Soledad Centeno-Sierra¹, Feibert Alirio Guzmán-Pérez⁴

¹Universidad Tecnológica de Altamira, México

²TecNM Campus Instituto Tecnológico de Ciudad Madero, México

³George Mason University, USA

⁴Corporación Universitaria Lasallista, Colombia

Abstract :

The increasing computational capability of smartphones has intensified thermal constraints in high-performance mobile applications. This study assesses whether the DroidRan Android hardware-behavior dataset enables sample-domain thermal prediction and compact dynamic modeling. Examination of all 36 workbooks and 180 worksheets identified 299,298 observations across 23 recorded variables, including battery voltage and temperature, three CPU thermal sensors, GPU temperature and frequency, eight CPU-core frequencies, memory, storage, network traffic, and attack stage. The dataset lacks timestamps, sampling intervals, ambient temperature, battery current, and measured power. To ensure leakage resistance, we held out nine complete workbooks and compared a stage-mean baseline, a second-order autoregressive model with exogenous inputs (ARX), a multilayer perceptron (MLP), and an ARX-plus-residual-MLP hybrid for one-observation-ahead battery-temperature prediction. On 70,389 held-out observations, the stage-mean baseline achieved RMSE = 3.684 °C, MAE = 3.086 °C, and $R^2 = 0.207$. The MLP and hybrid models yielded RMSE values of 3.718 and 3.717 °C, respectively, while the ARX model reached 3.932 °C. Median within-sheet lag-1 correlations for the five thermal channels ranged from -0.008 to 0.006, indicating that row order does not preserve measurable thermal continuity. DroidRan is therefore suitable for cross-sectional workload-stage characterization, but the current release is insufficient to identify physical RC parameters or validate current thermal forecasting. This negative result establishes the acquisition requirements for a robust predictive thermal-management experiment.

Keywords: Modeling of dynamic systems, mobile computing, smartphone thermal management, artificial intelligence, multilayer perceptron, neural networks, thermal throttling, energy efficiency.

Date of Submission: 17-09-2026

Date of Acceptance: 01-10-2026

I. INTRODUCTION

Modern smartphones integrate multicore CPUs, GPUs, neural processing units, high-speed memories, high-resolution displays, and multiple wireless interfaces within a thin enclosure. This integration enables computationally demanding applications but also concentrates heat generation in a small passive-cooling volume. Experimental studies show that virtual- and augmented-reality workloads can raise the temperature of smartphone batteries, processors, GPUs, and communication modules to levels that require thermal control [1]. Thermal throttling is therefore not only a reliability mechanism but also an important determinant of application-level performance. Bantock et al. demonstrated that conventional frequency capping can significantly degrade interactive performance [2], while thermal-aware workload management can recover useful performance when the thermal state is predicted before throttling occurs [3].

Recent research has consequently shifted from purely reactive thermal protection toward compact thermal modeling, predictive control, and workload-aware management. Chetoui and Reda coordinated thermal and electrical-current constraints in a cascaded controller for mobile devices [4]. Liu, Yu, and Wang demonstrated that compact models can support model predictive control under skin-temperature constraints [5]. At the device-modeling level, Phone-nomenon 2.0 showed that compact smartphone thermal models can reproduce steady-state and transient behavior with substantially lower computational cost than detailed CFD simulation [6]. Dynamic compact models based on convolution and fitting have also demonstrated practical accuracy for smartphone and laptop skin-temperature prediction [7].

Artificial intelligence offers a complementary approach because smartphone temperature depends on nonlinear interactions among processor utilization, clock frequency, battery state, wireless traffic, display

activity, ambient conditions, and thermal history. Machine-learning models have already been used with phone battery sensors for temperature inference [8], and LSTM architectures have shown that separating variables by thermal time scale can improve temperature prediction [9]. More recent electronic-device studies have demonstrated neural-network temperature prediction with experimental validation [10], while mobile gaming research has shown that machine-learning models can predict thermal sensation from a limited number of temperature monitoring points [11].

This study first evaluates whether DroidRan supports the temporal assumptions required by compact dynamic and recurrent models. We then perform grouped, one-observation-ahead prediction using models of varying complexity, ranging from a stage-conditioned mean to ARX and MLP variants. The contributions are both computational and methodological; we report performance metrics alongside an explicit dataset-suitability test to ensure that cross-sectional telemetry is not misrepresented as a continuous thermal trajectory.

II. THEORETICAL BACKGROUND

2.1 Thermal behavior of mobile devices

Heat generated by a mobile system-on-chip is transferred through package materials, printed circuit boards, thermal interface layers, heat spreaders, the battery, and the enclosure, before ultimately being rejected to the environment via natural convection and radiation. The limited thickness of modern devices constrains the area and volume available for conventional cooling solutions. Recent advancements include ultra-thin vapor chambers [12], phase-change materials [13], sorption-based reversible cooling [14], and microencapsulated phase-change materials for smartphone charging [15]. These technologies indicate that mobile thermal management constitutes a coupled hardware-software challenge: enhancing heat-transfer capability can postpone throttling, while predictive software control can more effectively utilize the available thermal capacity. For control-oriented analysis, the full three-dimensional heat-transfer field can be simplified to a lumped thermal network. Compact thermal simulators based on resistance-capacitance networks are widely adopted because they retain the dominant transient time constants while reducing computational requirements. PACT, for example, demonstrates the scalability of compact thermal-network simulation for electronic systems [16]. In the context of smartphones, a two-state approximation can represent both the primary heat-generating processor region and the slower battery or enclosure thermal mass.

2.2 Proposed second-order thermal model

Let T_c represent the equivalent CPU or system-on-chip (SoC) temperature, T_b the battery or enclosure temperature, T_a the ambient temperature, $P(t)$ the effective heat-generation input, R_{cb} the thermal resistance between the processor region and the body or battery region, R_{ba} the thermal resistance between the body and ambient, and C_c and C_b the corresponding effective thermal capacitances. The governing equations are as follows:

$$C_c \frac{dT_c}{dt} = P(t) - \frac{T_c - T_b}{R_{cb}} \quad (1)$$

$$C_b \frac{dT_b}{dt} = \frac{T_c - T_b}{R_{cb}} - \frac{T_b - T_a}{R_{ba}} \quad (2)$$

By defining $\theta_c = T_c - T_a$ and $\theta_b = T_b - T_a$, a two-state linear model is obtained. In state-space representation, as shown in Eq. (3). The model parameters $\theta = [R_{cb}, R_{ba}, C_c, C_b]$ are effective quantities that summarize multiple physical heat paths rather than representing a single material layer. This interpretation aligns with compact-model approaches commonly used for portable electronics [6], [7].

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (3)$$

where:

$$x = [\theta_c, \theta_b], \text{ and } u(t) = P(t)$$

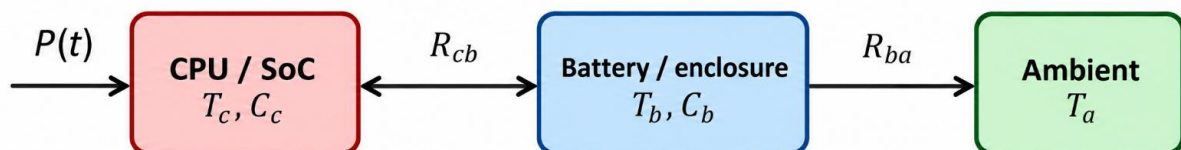


Figure 1. Second-order lumped RC thermal representation of the mobile device.

2.3 Data-driven temperature prediction

Physical and recurrent thermal models require measurements that maintain both acquisition order and sampling interval. In an appropriately designed future dataset, a multilayer perceptron (MLP) could be represented by Eq. (4), while a long short-term memory (LSTM) network could utilize Eq. (5) to estimate Eq. (6). These formulations are included as theoretical background because heating and cooling processes exhibit memory effects and multiple time constants [9]. However, the validity of their temporal assumptions must be evaluated prior to model training.

$$\hat{T}(k+h) = f_{MLP}(z(k)) \quad (4)$$

$$Z_k = \{z(k-m+1), \dots, z(k)\} \quad (5)$$

$$\hat{T}(k+h) = f_{LSTM}(Z_k) \quad (6)$$

A potential hybrid approach could predict the residual of a physical model, defined as shown in Eq. (7), and subsequently compute Eq. (8) $\hat{T}_{hybrid}(k+h) = TRC(k+h) + f_{LSTM}(Z_k)$. Previous research on physical-model-based machine learning has demonstrated promise for electronic temperature prediction [17]. In this study, the empirical lag test assesses this architecture's applicability to DroidRan.

$$e(k) = T_{meas}(k) - TRC(k) \quad (7)$$

$$\hat{T}_{hybrid}(k+h) = TRC(k+h) + f_{LSTM}(Z_k) \quad (8)$$

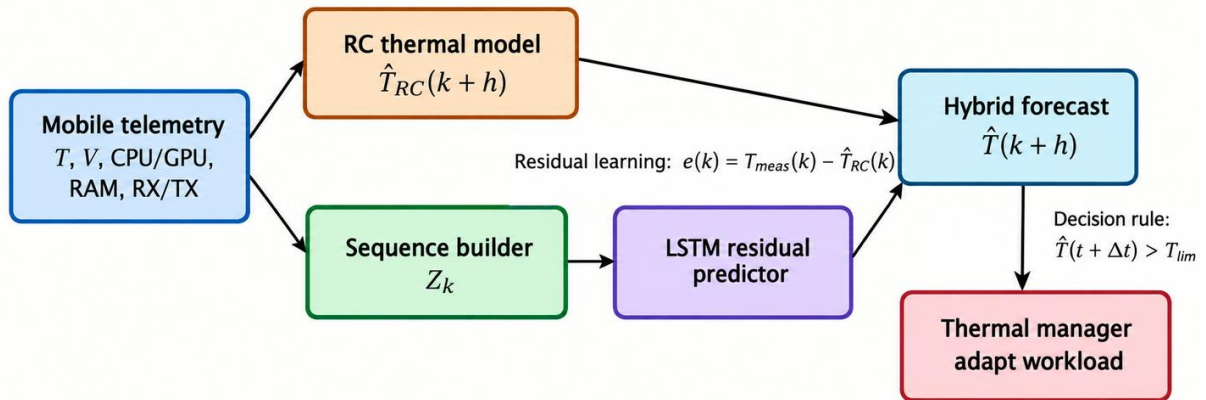


Figure 2. Conceptual RC + LSTM architecture for a future fixed-rate acquisition.

III. DEVELOPMENT

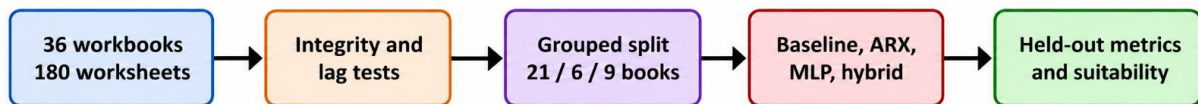


Figure 3. Verified DroidRan analysis workflow.

3.1 Data Sources

This study utilizes a single public dataset, DroidRan. The provided archive comprises 36 Excel workbooks, categorized into App Squatting, Email Squatting, and GUI Squatting, with 12 ransomware-family workbooks per category. Each workbook includes five worksheets corresponding to the stages: Before, Initial Access, Privilege Escalation, Command & Control, and Impact. Analysis of all worksheets resulted in 299,298 observations. Each row in the dataset contains 23 variables, including Total RAM, Free RAM, Available Internal Memory, Available Storage, Total Internal Memory, Received and Transmitted network rates, battery Voltage, battery Temperature, three CPU thermal sensors, GPU Temperature and Frequency, eight CPU-core frequencies, and Attack Stage [18].

The distribution of observations across stages is as follows: 43,756 for Before, 86,150 for Initial Access, 52,050 for Privilege Escalation, 64,400 for Command & Control, and 52,942 for Impact. App Squatting, Email Squatting, and GUI Squatting contribute 101,902, 102,904, and 94,492 observations, respectively. In this analysis, Attack Stage is considered an experimental-context variable rather than a temperature variable. A significant limitation, identified through direct inspection, is the absence of both a timestamp column and direct measurements of battery current or electrical power in the workbooks. Consequently, battery voltage and CPU/GPU activity variables may serve as predictors, but any activity proxy should not be described as measured power. Additionally, the sequence of rows may be interpreted as an observation sequence for machine learning experiments only after confirming the acquisition order from the dataset documentation. In the absence of a known sampling period, prediction horizons should initially be expressed in samples rather than seconds.

3.2 Data Preprocessing and Time-Series Construction

Preprocessing included indexing by workbook, category, family, worksheet, and within-sheet row; verification of the 23 original column names; numeric validation; descriptive analysis; and standardization based solely on training data. No synthetic timestamps or interpolation-based resampling were applied. In the diagnostic one-observation-ahead experiment, only consecutive rows within the same workbook and worksheet were used, ensuring that sequence boundaries were maintained.

To prevent data leakage, entire workbooks (rather than randomly selected rows) were allocated to training, validation, and test partitions. The deterministic grouped split consisted of 21 training workbooks (189,329 usable one-step samples), six validation workbooks (39,400 samples), and nine held-out test workbooks (70,389 samples). The test set included Covidlock, Jisut, and Slocker, each representing one of the three squatting categories. This design tests generalization to completely unseen category-family workbooks while preserving all preprocessing decisions within the training partition.

3.3 Discrete second-order thermal surrogate

Due to the absence of sampling interval and heat input data for DroidRan, the compact baseline was formulated as a second-order autoregressive model with exogenous variables, rather than relying on a physically identified RC network. This model forecasts the battery temperature at the subsequent observation point using the two preceding battery temperature values, along with the current GPU temperature, GPU frequency, battery voltage, and the frequencies of the eight CPU cores. Ridge regularization is applied to stabilize coefficient estimation. The resulting coefficients are regression parameters in the sample domain and should not be expressed in units such as K/W, J/K, or seconds, as shown in Eq. (9).

$$\hat{T}_b(k+1) = a_1 T_b(k) + a_2 T_b(k-1) + b^T u(k) \quad (9)$$

Prior to fitting the predictive models, temporal suitability was evaluated separately for each workbook-stage sequence. The lag-1 Pearson correlation was calculated for battery temperature, the three CPU thermal sensors, and GPU temperature. Because thermal inertia prevents temperature from changing independently between closely spaced samples, a genuine thermal trajectory typically exhibits positive short-lag persistence. Therefore, correlations near zero indicate that the order of the rows in the worksheet does not preserve the acquisition-time trajectory.

3.4 Comparative predicted models

Four models were evaluated. The stage-mean baseline assigns the mean battery temperature from the training set for the corresponding attack stage. The ARX surrogate incorporates two battery-temperature lags and exogenous hardware variables. The MLP utilizes current thermal, voltage, memory, network, GPU-frequency, and eight CPU-frequency measurements, employing two tanh hidden layers (32 and 16 units), L2 regularization, a deterministic seed of 17, and standardization applied only during training. The hybrid model adds an MLP estimate of the ARX residual to the ARX prediction. For all models, the prediction target is battery temperature one observation ahead.

An LSTM result is not reported. Although a recurrent architecture was initially considered, the empirical continuity test indicated that row order exhibits negligible lag-1 structure and the dataset release does not specify a sampling interval. Training an LSTM on these rows would therefore create the appearance of temporal forecasting without verified temporal information. Recurrent and physics-informed models are appropriate only for future datasets with fixed-rate acquisition, explicit timestamps, and continuous trajectories.

Model selection was performed using only the training and validation workbooks. The held-out test workbooks were evaluated once using RMSE, MAE, R^2 , and maximum absolute error. Because adjacent rows do not necessarily represent equal time increments, the prediction horizon is defined strictly as one subsequent observation rather than a specific number of seconds.

3.5 Evaluation metrics

Primary predictive metrics are shown in Eqs. (10)-(13) $RMSE = \sqrt{(1/N)\sum(T_k - \hat{T}_k)^2}$, $MAE = (1/N)\sum|T_k - \hat{T}_k|$, $R^2 = 1 - \sum(T_k - \hat{T}_k)^2 / \sum(T_k - \bar{T})^2$, and $E_{max} = \max|T_k - \hat{T}_k|$. Dataset suitability is additionally evaluated through within-sequence lag-1 correlation. This diagnostic distinguishes cross-sectional predictability from physically meaningful thermal forecasting.

$$RMSE = \sqrt{\frac{1}{N} \sum (T_k - \hat{T}_k)^2} \quad (10)$$

$$MAE = \frac{1}{N} \sum |T_k - \hat{T}_k| \quad (11)$$

$$R^2 = 1 - \frac{\sum (T_k - \hat{T}_k)^2}{\sum |T_k - \hat{T}_k|} \quad (12)$$

$$E_{max} = \max(|T_k - \hat{T}_k|) \quad (13)$$

3.6 Predictive thermal-management logic

The final control concept employs a forecast-based rule: if the predicted future temperature $\hat{T}(k+h)$ exceeds a device-dependent limit T_{lim} , the thermal manager reduces computational demand prior to reaching the threshold. Potential actions include CPU or GPU frequency capping, frame-rate adaptation, temporary reduction of neural-network inference frequency, and task offloading. In the DroidRan-only study, this approach remains a proposed control architecture rather than a closed-loop experimental result, as the dataset comprises observations rather than actuator commands or controlled intervention trials.

IV. RESULTS

Analysis of all 180 worksheets produced 299,298 observations, rather than the earlier partial count obtained from a subset of worksheets. The overall battery temperature was 40.72 ± 4.12 °C (34.03–48.51 °C). Mean temperatures were 40.33 °C for CPU Thermal Sensor 0, 40.40 °C for Sensor 1, 40.36 °C for Sensor 2, and 40.44 °C for the GPU sensor. Figure 4 shows stage-level battery-temperature profiles for all three squatting categories. The large sample counts make the stage means stable, but these aggregate profiles do not establish continuous-time heating or cooling trajectories.

Temporal continuity testing showed median within-sheet lag-1 correlations of 0.0047 for battery temperature, −0.0031, −0.0078, and 0.0057 for CPU Thermal Sensors 0–2, and 0.0041 for GPU temperature. These values are effectively zero. Figure 5 summarizes the workbook-stage correlation distributions and shows that consecutive rows behave as nearly independent thermal observations rather than samples from a smooth thermal process.

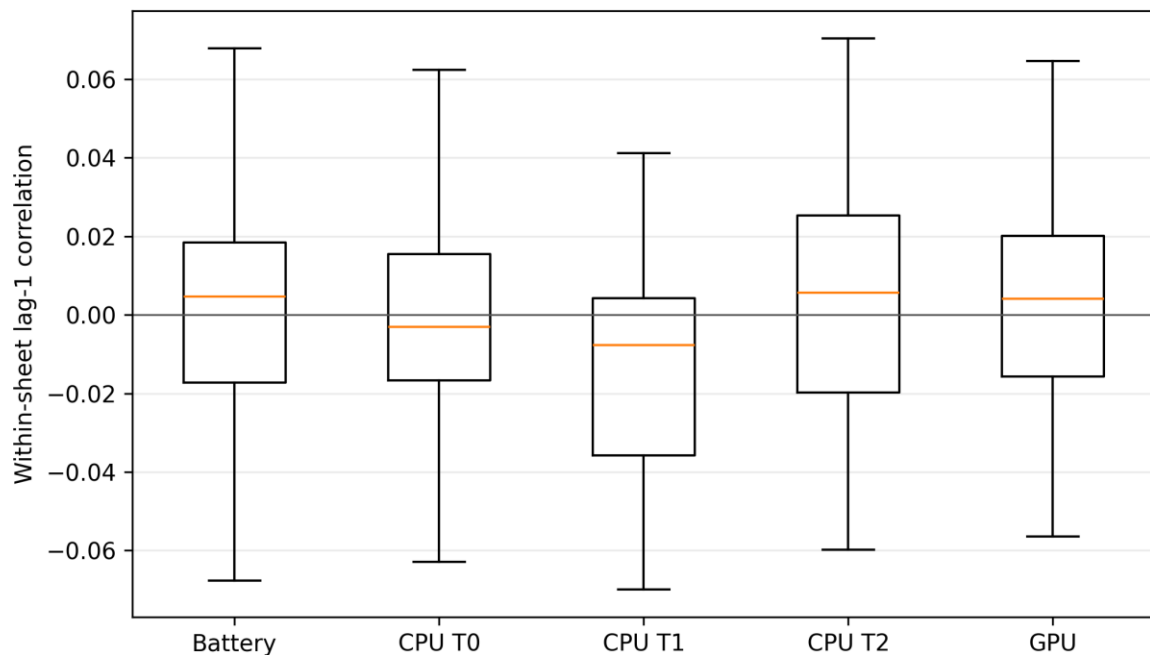


Figure 4. Distribution of within-sheet lag-1 correlations for the five temperature channels. Near-zero medians indicate no preserved short-lag thermal continuity.

Grouped held-out performance is reported in Table I. The simple stage-mean baseline achieved the lowest RMSE (3.684 °C) and the highest R^2 (0.207). The MLP and ARX + residual-MLP hybrid performed similarly, with RMSE values of 3.718 and 3.717 °C. The ARX surrogate was weaker (RMSE = 3.932 °C; R^2 = 0.097). Thus, model complexity did not improve one-observation-ahead generalization over a stage-conditioned mean. This result is consistent with the absence of short-lag continuity. On the other hand, Figure 5 shows the Measured and predicted battery temperature in a representative held-out sequence.

Table 1. One-observation-ahead battery-temperature prediction on nine held-out workbooks.

Model	RMSE (°C)	MAE (°C)	R ²	Max. error (°C)
Stage-mean baseline	3.684	3.086	0.207	7.077
Second-order ARX surrogate	3.932	3.366	0.097	9.705
MLP	3.718	3.075	0.193	15.207
ARX + residual MLP	3.717	3.058	0.193	11.132

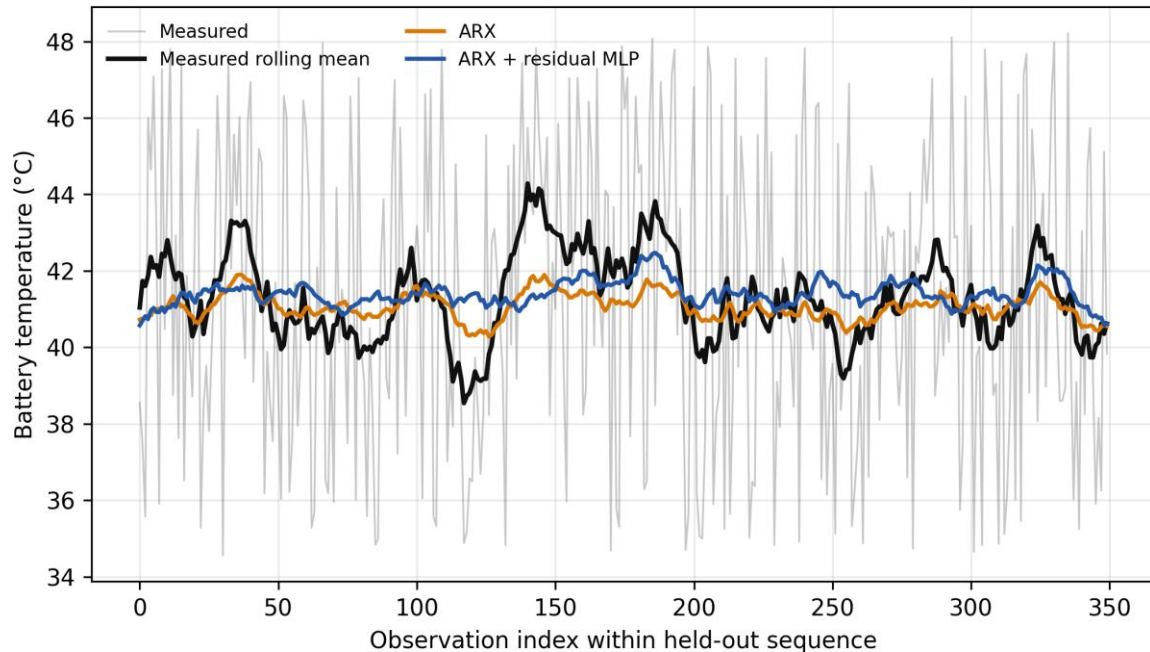


Figure 5. Measured and predicted battery temperature in a representative held-out sequence. Rolling means are shown for readability; the faint trace contains the raw measurements.

V. FINDINGS AND COMMENTS

The primary finding is not the relative performance of predictive models, but rather that the provided DroidRan release lacks the temporal structure necessary for dynamic thermal identification. Typically, a smartphone temperature signal sampled at a fixed short interval should exhibit significant local persistence, as thermal mass constrains rapid changes. The observed near-zero lag correlations and abrupt row-to-row variation suggest that the data rows were randomized, independently generated, or otherwise disconnected from the original acquisition chronology. Therefore, achieving a low numerical error with a recurrent model does not constitute validation of thermal dynamics.

DroidRan continues to offer value for cross-sectional analysis of hardware behavior across ransomware stages and delivery categories [18]. Its large dataset, consistent feature definitions, and extensive hardware-variable coverage facilitate stage characterization and potentially classification. However, its design for cybersecurity applications and the structure of its data limit its suitability for control-oriented thermal modeling. The grouped test results further underscore this limitation: the stage-mean baseline outperformed more complex predictors, indicating that stage-level distribution shifts provide more consistent information than row-to-row thermal history.

The lack of timestamps, ambient temperature, direct current, and electrical power measurements precludes the identification of physical thermal resistance, capacitance, and time constants. Without proper calibration, voltage and clock frequencies cannot be reliably converted into heat-generation power. Future controlled Android data acquisitions should therefore record timestamp, battery current and voltage, CPU and GPU utilization and frequency, thermal-zone readings, ambient temperature, workload transitions, and any throttling actions at a fixed sampling interval.

The proposed predictive controller currently represents a conceptual architecture rather than an outcome substantiated by DroidRan. When continuous data trajectories become available, the comparative framework can be reestablished using a physical RC model, multilayer perceptron (MLP), long short-term memory (LSTM), and a physics-assisted residual model, all evaluated under identical chronological partitions.

Metrics such as threshold precision, recall, and warning lead time can then complement RMSE and MAE to assess whether prediction enables effective workload adaptation.

VI. CONCLUSIONS

Analysis of the complete DroidRan archive identified 299,298 observations across 36 workbooks and 180 worksheets. The dataset provides extensive cross-sectional Android hardware telemetry; however, it lacks chronological continuity, defined sampling intervals, ambient condition data, and power input information necessary for dynamic thermal-system identification. Median lag-1 correlations for all five temperature channels were approximately zero.

Grouped validation on nine held-out workbooks showed that the stage-mean baseline achieved RMSE = 3.684 °C and $R^2 = 0.207$, outperforming the second-order ARX surrogate, MLP, and ARX plus residual-MLP hybrid in RMSE. These findings refute the initial hypothesis that the supplied row order supports recurrent thermal forecasting and demonstrate that random row splits or unverified LSTM sequences may produce misleading results.

DroidRan is appropriate for descriptive and cross-sectional workload-stage analyses; however, a separate fixed-rate measurement campaign is necessary for predictive thermal management. Future research should collect continuous, timestamped data under controlled workloads and record current, voltage, ambient temperature, CPU/GPU state, and intervention commands. Only with such data can RC parameters, physical time constants, LSTM horizons, and closed-loop thermal-management benefits be reported with scientific rigor.

REFERENCES

- [1] X. Zhang, H. Vadodaria, N. Li, K. D. Kang, and Y. Liu, "A smartphone thermal temperature analysis for virtual and augmented reality," in Proc. 2020 IEEE Int. Conf. Artificial Intelligence and Virtual Reality (AIVR), 2020, pp. 301–306, doi: 10.1109/AIVR50618.2020.00061.
- [2] J. R. B. Bantock, B. M. Al-Hashimi, and G. V. Merrett, "Mitigating interactive performance degradation from mobile device thermal throttling," IEEE Embedded Systems Letters, vol. 13, no. 1, pp. 5–8, 2021, doi: 10.1109/LES.2020.2991327.
- [3] G. P. Srinivasa, D. Werner, M. Hempstead, and G. Challen, "Thermal-aware overlocking for smartphones," in Proc. 2021 IEEE Int. Symp. Performance Analysis of Systems and Software (ISPASS), 2021, pp. 220–222, doi: 10.1109/ISPASS51385.2021.00039.
- [4] S. Chetoui and S. Reda, "CasCon: Cascaded thermal and electrical current throttling for mobile devices," IEEE Embedded Systems Letters, vol. 14, no. 1, pp. 3–6, 2022, doi: 10.1109/LES.2021.3079707.
- [5] H. Liu, J. Yu, and R. Wang, "Model predictive control of portable electronic devices under skin temperature constraints," Energy, vol. 260, Art. no. 125185, 2022, doi: 10.1016/j.energy.2022.125185.
- [6] Y.-M. Lee, H.-W. Chiou, S. Shiau, C.-W. Pan, and S.-H. Ting, "Phone-nomenon 2.0: A compact thermal model for smartphones," IET Computers & Digital Techniques, vol. 17, no. 2, pp. 43–59, 2023, doi: 10.1049/cdt2.12052.
- [7] H. Liu, J. Yu, and R. Wang, "Dynamic compact thermal models for skin temperature prediction of portable electronic devices based on convolution and fitting methods," International Journal of Heat and Mass Transfer, vol. 210, Art. no. 124170, 2023, doi: 10.1016/j.ijheatmasstransfer.2023.124170.
- [8] A. Trivedi, P. Bovornkeeratiroj, J. Breda, P. Shenoy, J. Taneja, and D. Irwin, "Phone-based ambient temperature sensing using opportunistic crowdsensing and machine learning," Sustainable Computing: Informatics and Systems, vol. 29, Art. no. 100479, 2021, doi: 10.1016/j.suscom.2020.100479.
- [9] Y. Kang, C. Ma, S. Wang, W. Wu, and K. Zhao, "A two-segment LSTM based data center temperature prediction model," IEICE Electronics Express, vol. 19, no. 21, Art. no. 20220291, 2022, doi: 10.1587/ele.19.20220291.
- [10] J. Nan, J. Chen, M. Li, Y. Li, Y. Ma, and X. Fan, "A temperature prediction model for flexible electronic devices based on GA-BP neural network and experimental verification," Micromachines, vol. 15, no. 4, Art. no. 430, 2024, doi: 10.3390/mi15040430.
- [11] Y. Liu, L. Dong, T. Chen, P.-L. P. Rau, P. Chen, Y. Shen, K. Luo, and J. Shi, "Assessing mobile gaming thermal experience using machine learning: A temperature monitoring approach," International Journal of Industrial Ergonomics, vol. 115, Art. no. 104014, 2026, doi: 10.1016/j.ergon.2026.104014.
- [12] C. Yan, H. Li, Y. Tang, X. Ding, X. Yuan, Y. Liang, and S. Zhang, "A novel ultra-thin vapor chamber with composite wick for portable electronics cooling," Applied Thermal Engineering, vol. 226, Art. no. 120340, 2023, doi: 10.1016/j.applthermaleng.2023.120340.
- [13] V. Bianco, M. De Rosa, and K. Vafai, "Phase-change materials for thermal management of electronic devices," Applied Thermal Engineering, vol. 214, Art. no. 118839, 2022, doi: 10.1016/j.applthermaleng.2022.118839.
- [14] H. R. Liu, C. X. Wang, B. J. Li, L. J. Hua, J. Q. Yu, and R. Z. Wang, "Reversible sweat cooling on mobile electronic devices by metal-organic frameworks-based moisture sorption-desorption process," Materials Today Nano, vol. 18, Art. no. 100198, 2022, doi: 10.1016/j.mtnano.2022.100198.
- [15] T. Hu, J. Zhang, W. Chan, R. Xiao, H. Xu, X. Zhao, G. Wang, and W. Yu, "Synthesis and optimization of GO/TiO₂@n-octadecane microcapsules for thermal management of smartphone charging," Applied Thermal Engineering, vol. 257, Art. no. 124409, 2024, doi: 10.1016/j.applthermaleng.2024.124409.
- [16] Z. Yuan, P. Shukla, S. Chetoui, S. Nemtsov, S. Reda, and A. K. Coskun, "PACT: An extensible parallel thermal simulator for emerging integration and cooling technologies," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 41, no. 4, pp. 1048–1061, 2022, doi: 10.1109/TCAD.2021.3079166.
- [17] Z. Wan, F. Wei, J. Peng, C. Deng, S. Ding, D. Xu, and X. Luo, "Application of physical model-based machine learning to the temperature prediction of electronic device in oil-gas exploration logging," Energy, vol. 282, Art. no. 128973, 2023, doi: 10.1016/j.energy.2023.128973.
- [18] K. Selvaraj and M. M. Singh, "DroidRan: A hardware behavior dataset for Android ransomware threats," Mendeley Data, Version 1, 2026, doi: 10.17632/xwnm862mbp.1.
- [19] H.J. Park and R. Ha, "Optimizing on-device VSR performance via adaptive thermal management," The Journal of Korean Institute of Communications and Information Sciences, vol. 50, no. 1, pp. 1–8, 2025, doi: 10.7840/kics.2025.50.1.1.
- [20] X.B. Tang, K.N. Wang, Z. Cui, B.C. Wang, and W. Shao, "Physics-informed LSTM for the dynamic response prediction of a nonlinear thermal management system," Applied Thermal Engineering, vol. 298, Art. no. 130929, 2026, doi: 10.1016/j.applthermaleng.2026.130929.