Ambient Temperature Prediction for Embedded Systems using Machine Learning

Selma Rahman[1 [0009-0008-9870-348X]], Mattias Olausson[1 [0009-0000-6964-2253]], Carlo Vitucci[1,2 [0000-0003-2598-6796]], and Ioannis Avgouleas[1 [0000-0001-8960-0544]]

[1] Ericsson AB, Sweden name.surname@ericsson.com
[2] Mälardalens Univeristy, Sweden name.surname@mdu.se

Abstract. In this work, we use two well-established machine learning algorithms i.e., Random Forest (RF) and XGBoost, to predict ambient temperature for a baseband's board. After providing an overview of the related work, we describe how we train the two ML models and identify the optimal training and test datasets to avoid the problems of data under- and over-fitting. Given this train/test split, the trained RF and XGBoost models provide temperature predictions with an accuracy lower than one degree Celsius, i.e., far better than any other approach that we used in the past. Our feature importance assessments reveal that the temperature sensors contribute significantly more towards predicting the ambient temperature compared to the power and voltage readings. Furthermore, the RF model appears less volatile than XGBoost using our training data. As the results demonstrate, our predictive temperature models allow for an accurate error prediction as a function of baseband board sensors.

Keywords: Predictive Maintenance, Temperature prediction, Radio Access Network

1 Introduction

The development of fifth-generation telecommunications, the so-called 5G, was not driven by technological evolution but by a commercial necessity. In fact, with the advent of smartphones, the value of the network has progressively shifted from connectivity to the data. 5G represents the opportunity for the operators to enter the rich market of services, making their business model and investment in network infrastructure sustainable. The core business shifts from connectivity to service deployment, and operators can generate profits by hosting a broad set of services in their infrastructure, close to the end user. However, 5G has led to increased infrastructure complexity due to:

- increased throughput and delay requirements [1],
- widespread computing capacity deployment (especially for dense urban areas) [9], and

– intelligent self-monitoring and easily-maintained configuration system to decrease CAPEX and OPEX [2].

Consequently, the need for a fault management framework that is strongly oriented towards the centrality of the recovery action has also grown in tandem with the complexity of the infrastructure [24]. Fault prediction [11, 8] and predictive maintenance [12] derive from the need of increasing the infrastructure sustainability.

1.1 Context Description

Our research focuses on the ability to do predictive maintenance for products in the Radio Access Network (RAN) domain. The "cloudification" of the network suggests a technological convergence with data center hardware products, but the environmental conditions are very different. A RAN solution, for example, must rely on something other than the cooling systems available for data centers due to cost, space, and noise constraints. Furthermore, RAN products should work under very different circumstances, e.g., their operating temperature spans a more demanding range than the typical for data center products. The above scenario exemplifies how research results that investigate the correlation between environmental parameters and system reliability depend on the domain of interest. Another characteristic of RAN products is that they poorly tolerate disturbances and interruptions. The data acquisition process must be unique regarding environmental and work parameters, i.e., the use of system resources. Furthermore, data collection is crucial for network access systems since they are often called for hosting soft real-time systems. The latter exhibit stringent requirements in terms of the reaction time and execution of a particular task such as the reception and decoding of traffic packets. The collection of data must therefore be as least intrusive as possible so as not to compromise the functionality of the node and the availability of bandwidth when transmitting the collected data.

1.2 Problem Statement

The more distributed computing and high data traffic capacity also involve a considerable workload. The evolution of hardware design on the nanoscale has been the response to this growth in data processing for both the latest generation processors and memory devices (DDR5). The reliability of hardware components has indeed increased in recent years [21], but it is equally valid that the complexity of the design has also increased. And, with the nanoscale hardware design, the probability of temporary or permanent fault conditions is higher due to power fluctuations, excessive operating temperatures, or cosmic radiation. Eventually, the hardware will end its life due to aging issues, and the system reliability will enter a critical phase where the failure rate will increase exponentially. The hardware repair process is costly: maintenance activities on-site, packaging, transportation, board troubleshooting, and test to confirm the failure condition diagnosis for the component, and faulty hardware replacement, if applicable. In telecommunication networks, multi-chip packages, robotics, automotive, and, more generally speaking, in an increasingly widespread distributed system, the hardware devices must work and inter-work properly, react to external disturbances promptly, and operate as long as possible. However, it must use an appropriate error prediction action by analyzing the data available from the system. Without this fundamental prediction action, the maintenance costs could be relatively high. Thus, it is essential to know how to identify a possible failure condition before it happens. Understanding how the state and use of resources affect their life cycle allows planning appropriate recovery actions in time, whether an actual replacement of the component or preventive isolation to enable an operational state in full or degraded function mode. Predicting the hardware fault is, therefore, fundamental for the sustainability of the future network. Without it, the unsustainable maintenance cost would compromise developing innovative services for industry 5.0 [10]. Machine learning and Artificial Intelligence can be the technology enabler for a fault prediction based on system data [5].

1.3 Research Objective

The paper assumption is that the likelihood of a system error depends on the environmental parameters, like temperature and humidity. Those environment parameters drive the entire life cycle of the hardware devices: board working continuously under stressful environment condition will have a shorter lifetime. Our research objective is to devise a model capable of predicting the ambient temperature of the board, i.e., the temperature of the immediate surroundings of the board. The latter has a direct impact on the board's operating temperature so an accurate ambient temperature model will allow for:

- implementing operations e.g., thermal throttling, that maintain the temperature of the device below a critical threshold, and
- forecasting the component's life cycle according to the ambient temperature for optimal maintenance planning.

1.4 Research Methodology

The paper is a quantitative engineering study [14] that aims to examine the relationships between environment parameters and resource usage using machine learning approach. For the evaluation of temperature prediction algorithms, the research used two types of data: environmental (i.e.: temperature and humidity) and resource use (number of cores used and their load). The data refer exclusively to industrial baseband boards, and this paper used them in respect of a confidential agreement. We have also used a thermal chamber to simulate different temperature working environments. We have verified the temperature prediction algorithms' validity by comparing them with other solutions proposed in the literature. Baseband board designers have reviewed the research outcomes and evaluated implementation feasibility and sustainability in the radio access network domain. With this approach, the advantage for the industrial partner is the possibility of reducing OPEX and the maintenance cost in the next generation of telecommunications systems.

2 Related Works

The ability to have a thermal model for any system is a well-known need because it is clear that, as the operating temperature increases, the reliability of the CMOS-based ICs decreases exponentially [23]. Yang et al. [27], for example, provides an interesting analysis of all those factors that negatively influence both the aging and the reliability of electronic components, such as the effects of voltage (Hot carrier injection) and temperature (Bias Temperature instability). Even considering the system as a non-divisible entity, the system's failure rate doubles for every ten Celsius degrees increase above twenty-one Celsius degrees [18]. Research on the thermal model mainly focuses on two types of algorithms [25]: those based on the thermodynamic laws and the physical characteristics of the components to find a thermodynamic model of the device [26, 16, 19] and those which, recognizing the limited capacity of a thermodynamic physical model to be representative for different types of installations, prefer algorithms that have data-driven solutions [22, 15]. The latter has received more attention from researchers recently, especially concerning the progress of AI/ML as a mechanism for evaluating predictive models. AI/ML methods have stood the test of time concerning temperature prediction by providing very accurate models for applications such as weather forecasting and temperature control in industrial environments, among others. For example, Ma et al. study demonstrates a spatiotemporal correlation for fault prediction algorithms using graph convolutional recurrent neural networks (GCRNN), which seems promising to replicate beyond the meteorological domain. In the networking domain, only a few researchers have dealt with temperature prediction in the RAN domain. On the contrary, most research works considered temperature prediction in data centers and High-Performance Computers (HPC). Therein, temperature prediction allows the intelligent implementation of energy saving utilizing workload management [17, 28], effective heat dissipation [13], and improved cooling efficiency [20]. Previous works considered the operational data of the board, such as the number of cycles per CPU or the cache metrics, and the physical characteristics of the system, such as the number of CPUs, the size and type of memory or traffic devices [29, 15]. One of the used algorithms is the long short-term memory-based temperature prediction (LSTM), an improved version of the more traditional recurrent neural network (RNN), more suitable for solving time series prediction problems. In the most significant works that have used LSTM, we point out the work of Cheng et al. [7] in the multicore and Network on Chip (NoC) domain. Neural networks are computationally demanding, and our research focuses on temperature prediction through less complex algorithms and less costly solutions to meet the requirements described in the context description section. There is an inevitable divergence in the research results we have considered. XGBoost is the algorithm frequently used in applied machine learning for structured data due to its fast speed compared with other gradient-boosting implementations [6].

3 Temperature Prediction Process

3.1 Design description

This chapter presents the design description of a machine-learning model that predicts ambient temperature, i.e., the target value based on lab measurements. We train the model using board temperature, rail, and board power sensors as independent variables while controlling computing load, environment humidity, and fan speed to simulate different board operating conditions. We evaluated XGBoost Regressor (XGB) [6] and Random Forest Regressor (RF) [4] (with and without cross-validation [3]) models to determine the most suitable for the RAN domain. We performed hyperparameter optimization for both the tree-based models to fine-tune their performance and promote better generalization. By searching for the optimal hyperparameter values, our approach is to effectively regularize the models to mitigate the risk of overfitting and enhance their ability to generalize to unseen data. We placed the radio access network boards inside a climate chamber in the lab. The climate chamber allows the simulation of all possible humidities and temperature levels that the baseband is likely to encounter in the field. We collected data for different computing loads by simulating no network traffic, minimal activity, or peak traffic conditions. Since the baseband board is a multiprocessor system, we have modified the active processing units' number and computing load to simulate different working conditions. Additionally, to simulate the environmental conditions of the installation site on the baseband board, we varied the fan speed of the cooling system. Following the well-established ML principles, we split the data into two distinct data sets:

1) the **training set** that is used to train the ML model. The input features include temperature sensors, watts and power levels measured at different points of the baseband board, the relative humidity and the ambient temperature of the climate chamber, among others, and

2) the **test set** that is used to assess the model's performance.

The training set is assigned a splitting ratio of 80%, while the test set receives 20%. Consequently, the collected data sets encompass the distinctive patterns that characterize the baseband board in various environmental and radio traffic conditions. We trained the ML model using the training data set to create an accurate and scalable model, making it possible to use the model for future versions of RAN boards without compromising its validity. Our evaluation metric regarding which ML model to use for environmental temperature prediction is based on the mean absolute error (MAE) i.e., the absolute value of the difference between the predictions and the targets, and R-squared (R^2). Residual analysis between the predicted and the measured ambient temperatures is considered as well.

3.2 Execution

Variable	Value	Distribution [X/Total]
DSP	Low	9/18
	Mid	7/18
	High	2/18
CPU load [%]	0	1/18
	20	1/18
	30	8/18
	100	8/18
Fan speed [%]	30	2/18
	40	1/18
	50	1/18
	70	4/18
	100	10/18
Relative Humidity [%]	0	8/18
	20-80	2/18
	30-80	8/18
Temperature Ranges [°C]	0-35	8/18
	20-55	8/18
	50-60	2/18

Table 1: The distribution of the dataset, for each setting of the controlled variables.

As described in the previous section, we continuously test the baseband in the climate chamber. Thus, the training runs with a new data set after each successful run. The current training for the ML models contains 18 datasets, each collected from their respective laboratory tests. Table 1 shows the data distribution of the various combinations of the controlled variables (DSP, fan speed, CPU load, relative humidity, and ambient temperature). For example, out of eighteen datasets, nine have DSP set to "Low", seven have DSP set to "Mid", and two have DSP set to "High", etc. The data collected is then explored and handled appropriately for the models to process. We also analyzed how to impute missing values and decided to use linear interpolation after investigating a few other methods, such as rolling mean or dropping entire rows containing at least one missing value. For the training of the models, we randomly divided the whole dataset into a training and testing set using the train-test split-function in Python ($train\_test\_split()^3$) by specifying the splitting ratio to be $80-$

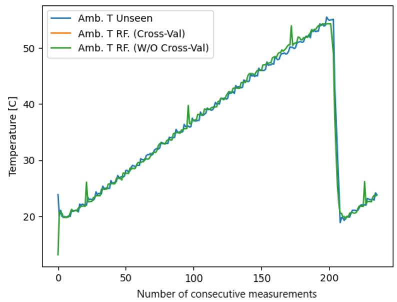
[3] sklearn.model_selection.train_test_split,
https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

20% respectively. The purpose of the testing set is to assess and evaluate the performance of the trained model by comparing the model's predictions with the actual values from the testing set. The performance evaluation described above allows us to measure metrics such as accuracy and residuals, which provide insights into how well the model generalizes to unseen data and, thus, performs in real-world scenarios. For the sake of presentation and to provide an efficient way to compare the predicted with the measured ambient temperature values side by side, we decided to introduce a data set referred to as unseen data. The unseen dataset contains a continuous baseband run in the climate chamber i.e. with the temperatures increasing with every measurement and it is completely excluded from the training and testing phase of the ML models. The data from the features (all variables except the target variable) is then used as an input to the models to acquire their predictions. This allows us to further evaluate the models' predictive ability of new and unseen data.

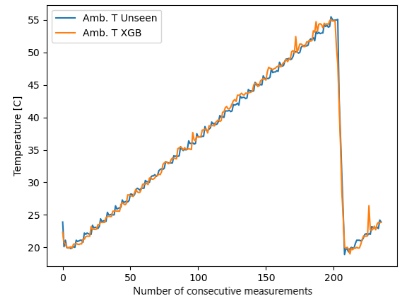
3.3 Results

We set the CPU and the fan speed maximum value (100%) as the test set of the baseband unit under evaluation. The prediction outcomes of this unseen data can be observed in Fig. 1a and Fig. 1b for both the Random Forest Regressor (with and without cross-validation) and the XGBoost Regressor, respectively.The reason to why cross-validation was not applied for the XGBoost regressor was because XGBoost generally performs well with smaller datasets where on the contrary Random Forest would benefit from cross-validation. The blue graph in both Fig. 1a and Fig. 1b shows the measured ambient temperature values obtained from an ambient temperature sensor during laboratory tests. It is the target value we want to predict successfully. The primary objective of the models is to predict this value accurately. Note that the Random Forest regressor with and without cross-validation overlays each other i.e., it did not matter whether we performed cross-validation on the training set or not. A well-performing model should exhibit residuals, i.e., the difference between the measured (actual) value and the predicted value, scattered randomly around the horizontal line at zero on the y-axis, with no apparent patterns or trends. The absence of patterns or trends indicate that the model effectively captures the relationship between the features and the target variable and that there is no further information that it could employ to enhance its predictions. On the other hand, if the residual plot displays patterns or trends, such as a U-shape or a curve, the model fails to satisfactorily capture the underlying relationships between the features and the target variable.

Including additional information could improve the models' predictions avoiding underfitting or overfitting. Underfitting occurs when a model or algorithm fails to capture the underlying trend of the data, resulting in poor performance on training and testing data. Underfitting occurs when the training dataset is too small, the model needs to be more complex, or the data needs to be more precise. Overfitting happens when a model is too complex and learns from noise or inaccurate data entries in the training set, leading to poor performance on



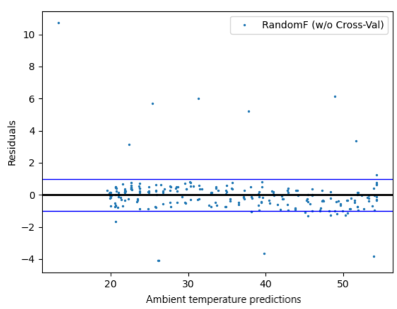
(a) RF predictions



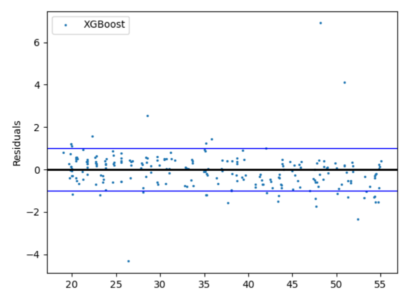
(b) XGB predictions

Fig. 1: Ambient Temperature Predictions, CPU=100% and fan=100%

testing data. An over-fitted model indicates the need to explore the reduction of the model complexity, use early stopping during training, or implement regularization, among others. Upon observing the graphs in Fig. 2a and Fig. 2b along with the graphs in Fig. 1a and Fig. 1b, it is evident that the Random Forest and XGBoost regressors are capable of making predictions with a high degree of accuracy, without under- or overfitting and exhibiting errors between the range of $\pm 1°C$.



(a) RF prediction (no cross validation)



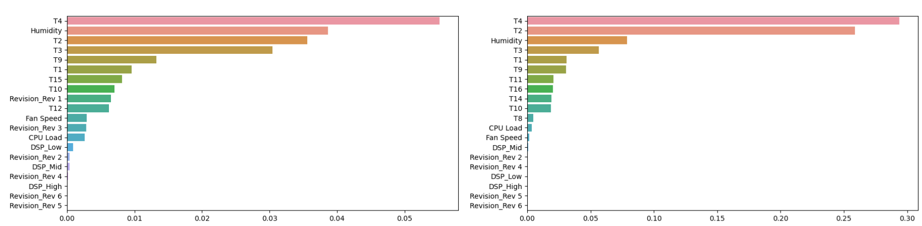
(b) XGB prediction

Fig. 2: Scatter plot of residuals between predictions and the measured value for a baseband with CPU=100%, fan=100%, and $\pm 1°C$ threshold displayed

To further evaluate the accuracy of the predictions, we calculated and compared the mean absolute error (MAE) and R-squared (R^2) between the model's prediction and the measured ambient temperature of either the testing or the unseen set. These metrics provide insight into how well the model is performing and how much of the variation in the data can be explained by the model. For instance, a low MAE suggests that the average difference between the predicted and actual values is small. In contrast, a high R^2 value indicates that the model explains a large proportion of the variance in the target variable - and vice versa. Table 2 shows the result. The models are trained successfully with relatively low error and high accuracy based on the metrics' values for the testing data, suggesting that the model fits the test data well and can make reliable predictions. Moving on to the metrics for the unseen data, it suggests that the model can generalize well and make accurate predictions on data that it has not seen before. The fact that the MAE value is lower for the unseen data than the testing data suggests that the model has not overfitted to the testing data and is not capturing noise or irrelevant information. Overall, these metrics indicate that the model has high accuracy and can be considered a reliable model for predicting ambient temperature.

Metric	**Random Forest (CROSS-VAL)**	**Random Forest (w/o CROSS-VAL)**	**XGBoost**
Test MAE	0.795	0.791	0.613
Test R^2	0.984	0.984	0.987
Unseen MAE	0.654	0.687	0.595
Unseen R^2	0.987	0.988	0.994

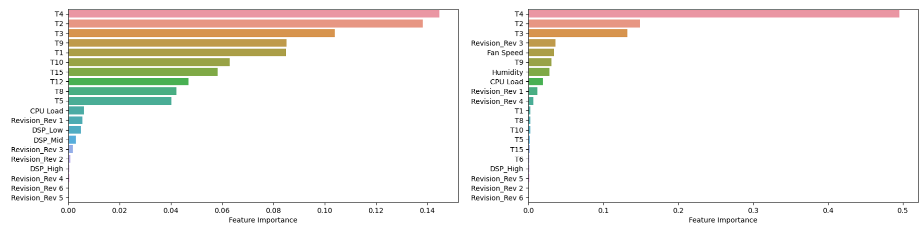
Table 2: MAE and R^2 values for different models when predicting baseband ambient temperature at CPU=100% and fan=100%



(a) RF model (b) XGB model

Fig. 3: Top and bottom 10 features based on permutation importance, predicting baseband ambient temperature at CPU=100% and fan=100%.

The results available in our paper show temperature prediction using baseband temperature sensors and the controlled variables as features, excluding temperature as it is the target variable. The permutation (Figs. 3a and 3b) and feature importance (Figs. 4a and 4b) indicate that our features' choice is correct. Permutation importance is a technique for evaluating feature importance based on a model's performance decrease during the permutation of a feature. It measures how much each feature contributes to the model's accuracy on the



(a) RF model (b) XGB model

Fig. 4: Top and bottom 10 features based on importance, predicting baseband ambient temperature at CPU=100% and fan=100%.

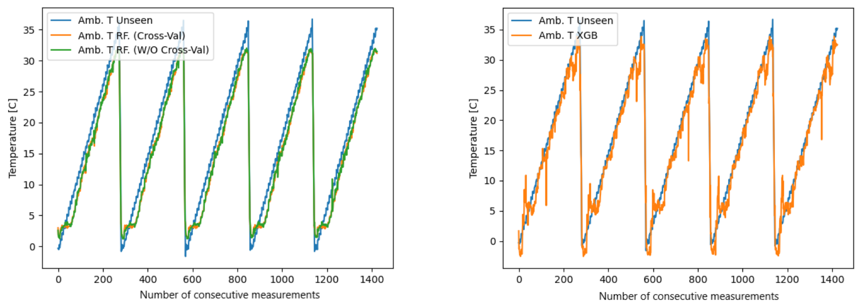
training set. On the other hand, feature importance is a metric that ranks features according to their importance for making predictions on new, unseen data. Figs. 3a, 3b, 4a, and 4b indicate that it is the temperature sensors that primarily contribute to the model's performance and,hence predictions' accuracy. Moreover, they show that power and voltage readings can be excluded without any loss of prediction accuracy.

3.4 Predictions on under-represented training data

To assess the performance of our trained model on data that is under-represented we tested our models' predictions on a dataset for which the unseen data are: CPU load = 30%, fan speed = 70%, DSP = Low, ambient temperature range $= 0 - 35°C$ and relative humidity range = 0%. The predictions can be seen in Figs. 5a and 5b. Insufficient dataset refers to a situation where the prediction test case lacks adequate representation in the training dataset concerning the parameter settings. Note that the increased number of "triangles" in the Figures only indicates consecutive test execution at the same temperature. Figures 5a and 5b clearly show a case of overfitting. Possible reasons for overfitting could be:

- Insufficient training data: When the training dataset is small, the model may learn the noise or specific patterns present in the limited data. Increasing the amount of training data can help alleviate this issue.
- Feature overfitting: When the model has access to irrelevant or noisy features with no predictive power for the target variable, it may overfit by learning patterns specific to the training data. Feature selection or dimensionality reduction techniques can help address this issue.
- Complex model architecture: Models with high complexity, such as those with a large number of parameters, have a higher tendency to overfit. Simplifying the model architecture, reducing the number of parameters, or using regularization techniques can mitigate overfitting.

Table 3 shows how the value of MAE in the case of prediction based on an insufficient training dataset is higher than that obtained with an adequate



(a) RF predictions (b) XGB predictions

Fig. 5: Ambient Temperature Predictions, CPU=30% and Fan=70%

number of variables in the training dataset (compare with Table 2) for both test and unseen data. An MAE greater than two indicates that, on average, the model's predictions deviate from the actual temperature by more than two degrees Celsius. This level is unacceptable; the goal is to keep the error below one degree Celsius. An R^2 of 0.94 indicates that the model is still explaining 94% of the variance in the data, which is still relatively high, but not as high as the previous value of 0.98.

Metric	**Random Forest (CROSS-VAL)**	**Random Forest (w/o CROSS-VAL)**	**XGBoost**
Test MAE	0.919	0.897	0.633
Test R^2	0.985	0.985	0.986
Unseen MAE	2.252	2.144	1.524
Unseen R^2	0.942	0.947	0.959

Table 3: MAE and R^2 values for different models when predicting baseband ambient temperature at CPU=30% and Fan=70%

4 Conclusion and Future Works

In this paper, we use two well-established machine learning algorithms to predict the ambient temperature of a baseband board; Random Forest and XGBoost Regressors. The hypothesis is that we can achieve accurate ambient temperature prediction for baseband boards without using neural-network-based solutions. In fact, tree-based models are considered more suitable for regression tasks involving the prediction of continuous numerical values, and produces accurate result for the baseband domain described in 1.1. These models capture the relationships and patterns within the data to make accurate temperature predictions. Both tree-based models were hyperparameter optimized. Additionally, we performed cross-validation for the Random Forest regressor to evaluate its performance. The trained Random Forest and XGBoost models provide temperature predictions with an accuracy lower than one degree Celsius, i.e., far better than any other approach we used in the past. We observe MAE of at least 0.59 and R^2 values of around 0.99 on completely unseen data. The evaluation of our metrics (see Tables 2 and 3) indicate accurate predictions. Based on the generated permutation and feature importance measurements, we can further conclude that the temperature sensors are the most critical contributors to the model's performance. At the same time, the power and voltage readings don't contribute significantly, and the prediction can safely ignore them. When evaluating the models on unseen data where the test case is not well-represented, the MAE increases to approximately 2, and the R^2 decreases to around 0.95. The robustness of the models is underscored by the enhanced value for MAE and R^2, indicating their high confidence levels. This signifies that the models have successfully captured the intricacies of the data and minimized the potential for overfitting. Moreover, the comprehensive examination of the prediction graphs will not only yield further valuable insights but also solidify the overall findings of the study. Finally, predicting the ambient temperature is the first step to putting into practice those thermal throttling and preventive maintenance policies that we have indicated as the primary objective of our research (compare with Section 1.3). Pursuing the research's goals requires future study in two different but parallel domains:

- Use the ambient temperature prediction along with system resources (computer, networking, and memory) to obtain a hardware fault prediction.
- Use the prediction of ambient temperature as a critical variable in the runtime product's life cycle evaluation as a function of the environmental parameters.

References

1. 3GPP: Ts 22 261 - v19.1.0 - 3rd generation partnership project; technical specification group services and system aspects; service requirements for the 5g system; stage 1 (release 19) (2022)
2. Al-Dulaimi, A., Wang, X., Chih-Lin, I.: 5G networks: Fundamental requirements, enabling technologies, and operations management (2018). https://doi.org/10.1002/9781119333142
3. Bates, S., Hastie, T., Tibshirani, R.: Cross-validation: What does it estimate and how well does it do it? (2022). https://doi.org/10.48550/arXiv.2104.00673
4. Breiman, L.: Random forests. Machine Learning **45**(1), 5–32 (2001). https://doi.org/10.1023/A:1010933404324
5. Camps-Mur, D., Gavras, A., Ghoraishi, M., Hrasnica, H., Kaloxylos, A., Anastasopoulos, M., Tzanakaki, A., Srinivasan, G., Antevski, K., Baranda, J., Schepper, K., Casetti, C., Chiasserini, C., Garcia-Saavedra, A., Guimares, C., Kondepu, K., Li, X., Magoula, L., Malinverno, M., Cogalan, T.: Ai and ml – enablers for beyond 5g networks (05 2021). https://doi.org/10.5281/zenodo.4299895

6. Chen, T., Guestrin, C.: Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 785–794 (2016). https://doi.org/10.1145/2939672.2939785
7. Cheng, T., Du, H., Li, L., Fu, Y.: Lstm-based temperature prediction and hotspot tracking for thermal-aware 3d noc system. 2021 18th International SoC Design Conference (ISOCC) pp. 286–287 (2021)
8. Chigurupati, A., Thibaux, R., Lassar, N.: Predicting hardware failure using machine learning. 2016 Annual Reliability and Maintainability Symposium (RAMS) pp. 1–6 (2016). https://doi.org/10.1109/RAMS.2016.7448033
9. Chih-Lin, I., Kukliński, S., Chen, T., Ladid, L.L.: A perspective of o-ran integration with mec, son, and network slicing in the 5g era. IEEE Network **34**, 3–4 (11 2020). https://doi.org/10.1109/MNET.2020.9277891
10. Cotta, J., Breque, M., Nul, L.D., Petridis, A.: Industry 5.0 towards a sustainable, human-centric and resilient european industry. European Commission Research and Innovation (R&I) Series Policy Brief (2021). https://doi.org/10.2777/308407, https://ec.europa.eu/eurostat/statistics-
11. Das, A., Mueller, F., Rountree, B.: Aarohi: Making real-time node failure prediction feasible. 2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS) pp. 1092–1101 (2020). https://doi.org/10.1109/IPDPS47924.2020.00115
12. Das, M.K., Rangarajan, K.: Performance monitoring and failure prediction of industrial equipments using artificial intelligence and machine learning methods: A survey. Proceedings of the 4th International Conference on Computing Methodologies and Communication, ICCMC 2020 pp. 595–602 (3 2020). https://doi.org/10.1109/ICCMC48092.2020.ICCMC-0000111
13. Durgam, S., Bhosale, A., Bhosale, V., Deshpande, R., Sutar, P., Kamble, S.: Ensemble learning for predicting temperature of heat sources for minimizing electronic failures (2021). https://doi.org/10.1109/ICNTE51185.2021.9487663
14. Escudero-Mancebo, D., Fernández-Villalobos, N., Óscar Martín-Llorente, Martínez-Monés, A.: Research methods in engineering design: a synthesis of recent studies using a systematic literature review. Research in Engineering Design (2023). https://doi.org/10.1007/s00163-022-00406-y
15. Ilager, S., Ramamohanarao, K., Buyya, R.: Thermal prediction for efficient energy management of clouds using machine learning. IEEE Transactions on Parallel and Distributed Systems **32** (2021). https://doi.org/10.1109/TPDS.2020.3040800
16. Lyu, N., Jin, Y., Xiong, R., Miao, S., Gao, J.: Real-time overcharge warning and early thermal runaway prediction of li-ion battery by online impedance measurement. IEEE Transactions on Industrial Electronics (2021). https://doi.org/10.1109/TIE.2021.3062267
17. Nisce, I., Jiang, X., Vishnu, S.P.: Machine learning based thermal prediction for energy-efficient cloud computing (2023). https://doi.org/10.1109/ccnc51644.2023.10060079
18. O'connor, P.D.: Arrhenius and electronics reliability. Quality and Reliability Engineering International **5** (1989). https://doi.org/10.1002/qre.4680050402
19. Ozceylan, B., Haverkort, B.R., Graaf, M.D., Gerards, M.E.: Improving temperature prediction accuracy using kalman and particle filtering methods (2020). https://doi.org/10.1109/THERMINIC49743.2020.9420535
20. Peng, Y.H., Lee, C.M., Tung, K.Y., Chen, R.: Rack inlet temperature prediction based on deep learning (2022). https://doi.org/10.1109/ICMT56556.2022.9997747
21. Phuyal, S., Bista, D., Bista, R.: Challenges, opportunities and future directions of smart manufacturing: A state of art review. Sustainable Futures **2**, 100023 (1 2020). https://doi.org/10.1016/J.SFTR.2020.100023

22. Prisacaru, A., Gromala, P.J., Han, B., Zhang, G.Q.: Degradation estimation and prediction of electronic packages using data-driven approach. IEEE Transactions on Industrial Electronics **69** (2022). https://doi.org/10.1109/TIE.2021.3068681
23. Spory, E.M.: Increased high-temperature ic packaging reliability using die extraction and additive manufacturing assembly (2016). https://doi.org/10.4071/2016-hitec-18
24. Vitucci, C., Sundmark, D., Jägemar, M., Danielsson, J., Larsson, A., Nolte, T.: Fault management framework and multi-layer recovery methodology for resilient system. Proceeding IEEE 6th International Conference on System Reliability and Safety (ICSRS) pp. 32–39 (11 2022)
25. Wang, N., Li, J.Y.: Efficient multi-channel thermal monitoring and temperature prediction based on improved linear regression. IEEE Transactions on Instrumentation and Measurement **71** (2022). https://doi.org/10.1109/TIM.2021.3139659
26. Wang, N., Zhang, J.N., Liu, Z.Y., Ding, C., Sui, G.R., Jia, H.Z., Gao, X.M.: An enhanced thermoelectric collaborative cooling system with thermoelectric generator serving as a supplementary power source. IEEE Transactions on Electron Devices **68** (2021). https://doi.org/10.1109/TED.2021.3059183
27. Yang, X., Sang, Q., Wang, C., Yu, M., Zhao, Y.: Development and challenges of reliability modeling from transistors to circuits. IEEE Journal of the Electron Devices Society (2023). https://doi.org/10.1109/JEDS.2023.3253081
28. Yao, X., Omori, M., Nishi, H.: Load balancing method using server temperature prediction considering multiple internal heat sources in data centers (2021). https://doi.org/10.1109/ICM46511.2021.9385604
29. Zhang, K., Ogrenci-Memik, S., Memik, G., Yoshii, K., Sankaran, R., Beckman, P.: Minimizing thermal variation across system components (2015). https://doi.org/10.1109/IPDPS.2015.37