Proceedings of the 4th International Conference on Fluid Flow and Thermal Science (ICFFTS'23)
Lisbon, Portugal- December 07 - 09, 2023
Paper No. 158
DOI: 10.11159/icffts23.158

Comparative Analysis of Machine Learning Approaches for Boiling ONB Prediction

Adrián Cabarcos[1], Concepción Paz[1], Miguel Concheiro[1], Marcos Conde-Fontenla[1], Eduardo Suárez[1]
[1]CINTECX
Universidade de Vigo, 36310 Vigo, España
acabarcos@uvigo.gal; cpaz@uvigo.gal; mconcheiro@uvigo.gal; mfontenla@uvigo.gal; suarez@uvigo.gal

Abstract **-** This study investigates the use of Machine Learning models for predicting both wall temperature and heat flux at the Onset of Nucleate Boiling (ONB). The dataset used in this work was obtained from an experimental test bench using Joule heating for boiling generation. Furthermore, five models, including Artificial Neural Networks (ANN), XGBoost, Support Vector Regression, AdaBoost, and Random Forest, were trained and evaluated. Results reveal that AdaBoost performed the worst in both wall temperature and heat flux predictions, indicating limitations in its ability to accurately forecast the ONB parameters. Conversely, the Random Forest model showed signs of overfitting in both predictions, suggesting that it may struggle to generalize to unseen data. In contrast, ANN demonstrated superior performance in predicting wall temperature (with mean square errors of 3.79 °C² and 3.84 °C² for training and testing), while XGBoost outperformed other models in heat flux prediction. Both models successfully captured the complex relationships between inputs (bulk temperature, pressure, channel inclination and velocity) and ONB parameters, leading to accurate predictions.

Keywords**:** Boiling, ONB, Machine Learning, Decision Trees, Support Vector Regression, Artificial Neural Network

1. Introduction

The prediction of wall temperature and heat flux during boiling phenomena is a critical aspect in numerous engineering applications, such as in the design and operation of heat transfer systems like heat exchangers or water-cooled nuclear reactors [1–3]. This prediction is particularly significant at the Onset of Nucleate Boiling (ONB), which occurs when vapor bubbles start to form and grow on a heated surface, resulting in a complex heat transfer phenomenon that has been extensively studied but is still not fully understood [4–6]. These bubbles, as they initiate and expand, greatly enhance the heat transfer capabilities, effectively extracting heat from the surface [7]. Consequently, nucleate boiling is widely used in various engineering fields due to its high heat transfer coefficients [8], [9]. However, the presence of bubbles can also lead to undesirable effects, such as the formation of vapor film, bubble coalescence, or a significant decrease in heat transfer efficiency once the Critical Heat Flux (CHF) is reached [10], [11]. Therefore, accurately estimating boiling processes becomes crucial for optimizing heat transfer and enabling the efficient design and performance of thermal systems.

Traditionally, researchers have proposed physics-based models and empirical correlations to predict the onset of nucleate boiling based on specific input conditions, often relying on experimental data. For instance, Hsu [12] conducted pioneering research on the conditions that enable nucleate boiling to occur, proposing a correlation that mainly considers the superheat and physical properties of the liquid. Qu and Mudawar [13] conducted experiments to measure the incipient boiling heat flux in micro-channel heat sinks and developed a model that considers both mechanical and thermal factors, including the force balance on the bubble, through a bubble departure criterion. Similarly, Liu et al. [14] formulated an analytical model based on experimental work to predict heat flux and bubble size at the ONB, incorporating various parameters such as fluid inlet conditions, subcooling, contact angle, microchannel dimensions, and fluid exit pressure. More recently, Lim et al. [15] explored the onset of nucleate boiling in a swirl tube using dimensional analysis, while Al-Yahia and Jo [16] investigated the influence of mass flux on nucleation site density, bubble departure, and heat flux, proposing a new correlation with an error of ±16.5%.

However, in spite of the substantial contributions of previous research, and due to the complex nature of boiling phenomena, the accuracy of traditional correlations may decrease when faced with novel applications, geometries or operating conditions [17]. Thus, the study of boiling and the development of new models will remain of significant interest for the scientific community [18]. In fact, Machine Learning (ML) approaches have recently merged as promising tools for boiling prediction due to their capability to capture nonlinear relationships between input parameters and output target values, thereby enhancing prediction accuracy. For example, Liu et al. [19] utilized data from numerical simulations to train a deep feed-forward neural network for predicting heat transfer in pool boiling. Alotaibi et al. [20] applied a decision tree algorithm for minimum film boiling temperature and, in another study, Cai [21] applied a support vector machine (SVM) to predict CHF in a concentric-tube open thermosiphon. These studies demonstrate the successful application of supervised learning algorithms for predicting boiling phenomena. However, a recent review conducted by Rashidi et al. [22] emphasizes the need for a more extensive investigation into the influence of different model architectures and inputs on the accuracy of predictions in boiling applications. Precisely, this work is framed within this scientific context.

In fact, this study aims to explore the performance of machine learning algorithms, specifically Artificial Neural Networks (ANN), Random Forest (RF), XGBoost, AdaBoost, and Support Vector Regression (SVR), in predicting the onset of nucleate boiling. The data used to train these algorithms were obtained from an experimental setup consisting of a heated plate embedded in a rectangular channel with adjustable orientations. Hyperparameter tuning was performed to optimize these algorithms during training, and the Mean Square Error (MSE) was used as the evaluation metric to compare these approaches.

2. Experimental setup and methodology

The experimental setup depicted in Fig. 1(a) was used to obtain the data utilized for training the machine learning algorithms. This facility, extensively described in previous studies [23], [24], comprises two clearly distinct parts: a study section outlined by a dashed line in Fig. 1(a), and additional subsystems that enable the proper adjustment and measurement of bulk temperature, pressure, and volumetric flow rate. As shown in Fig. 1(a), the study section consists of a rectangular channel with effective dimensions of 10 x 25 x 320 mm with adjustable inclination. The heated area within this channel has dimensions of 10 mm in width and 75 mm in length. In this specific study, heat flux on this area was generated by Joule heating of a 1 mm-thin stainless steel plate using two 10 kW high-intensity power supplies. Previous and detailed examples where this heating method was used for boiling purposes can be found in works performed by Jo et al. [25] and Kim and Bang [26], among other examples.

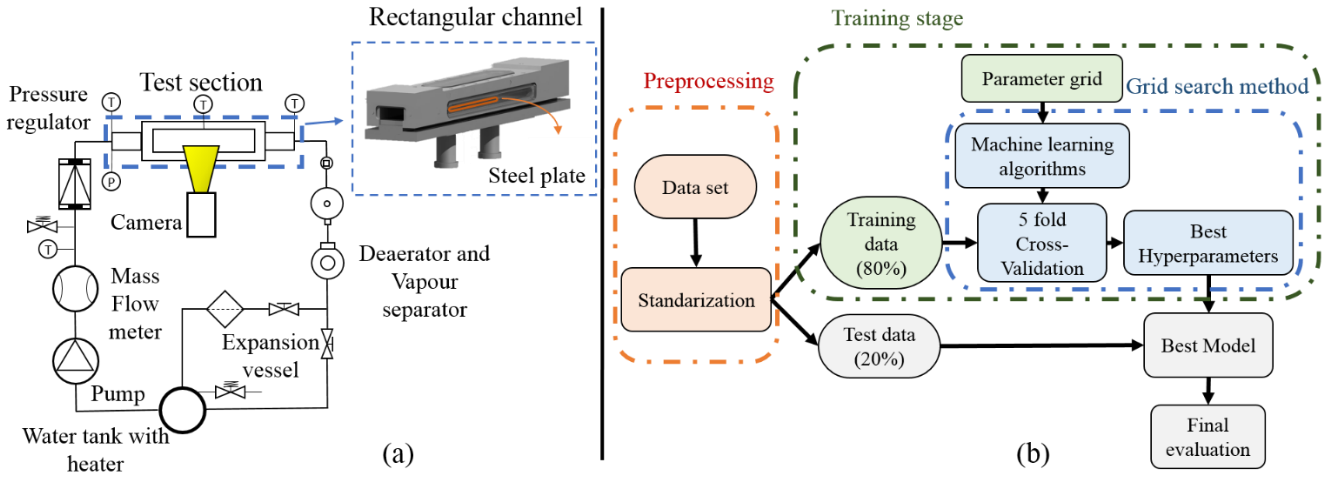

Fig. 1: (a) Sketch of the experimental test bench; (b) Flowchart of the Machine learning training process.

Specifically, 228 tests were conducted, varying inlet velocities (0.2–0.8 [m·s-^{1}]), inlet temperatures (20–95 [°C]), pressures (130–220 [kPa]), and channel inclination (0°, 180°, 225°, and 270°). In all of them, a 50% ethylene glycol – water mixture was used as the working fluid. Specifically, each of this tests was performed increasing the electrical load to the heating surface in steps of 0.5 A until bubbles were clearly formed. According to Jo et al. [25], from electrical current (I), voltage (V) and the heating area (A), heat flux (q) was calculated via Eq. 1, while wall temperature was calculated using the measured electrical resistance and calibration charts of this specific test bench. Finally, from the resulting boiling curves of each test, the ONB was obtained as the inflection point where the slope of the boiling curve changes.

$$q = I\frac{V}{A} \tag{1}$$

3. Machine learning models

From the experimental tests outlined in section 2, a collection of input conditions is obtained along with the corresponding results of heat flux and wall temperature. This dataset enables the utilization of machine learning algorithms to autonomously uncover patterns on this data and subsequently make predictions based on this information. In this specific study, a comparative analysis is proposed, focusing on five different machine learning algorithms: Support Vector Regression, Artificial Neural Networks, Random Forest, AdaBoost, and XGBoost. The basis of each algorithm and its implementation are succinctly described below.

3.1. Overview of algorithms

All the algorithms employed in this work are under the domain of supervised learning, which involves using labeled training data to build predictive models. In this context, the algorithms are provided with input-output pairs, allowing them to identify underlying patterns within the data. One of the algorithms considered in this study are the Artificial Neural Networks. ANNs are composed of interconnected artificial neurons, which reproduce the behavior of biological neurons. Furthermore, by employing a layered structure and activation functions, ANNs are able to learn from input features through non-linear transformations [27]. Specifically, these algorithms employ a process known as backpropagation to iteratively adjust the connection weights between neurons [28]. This adjustment is based on the comparison between the predicted output and target values. By minimizing a suitable loss function, often using the MSE, ANNs aim to optimize their predictive performance. Furthermore, the addition of penalty terms proportional to the square of the network weights is also common for ANNs [29]. By adding these terms (a process known as regularization “L^2”), smaller weights are promoted during training, which helps to prevent single weights from dominating the learning process.

Regarding the AdaBoost model, it is an ensemble learning technique that combines weak models to construct a more robust one [30]. In regression tasks, it trains these weaker models iteratively, giving more importance to complex cases by adjusting their sample weights. The ensemble prediction is determined by considering the weighted contributions of these individual, weaker learners. In fact, its main objective is to reduce prediction errors by minimizing a loss function, usually defined using exponential, linear, or square functions, which measures the disparities between the ensemble model's predictions and the actual labeled values. Therefore, the key idea of this supervised learning algorithm is to sequentially train a series of weak learners on weighted versions of the training data.

Similarly, the Random Forest algorithm also utilizes weak learners. However, in RF, multiple decision trees are built independently using bootstrapped samples from the training data [31]. Each tree is constructed using a random subset of features, where this randomness tends to enhance the model's generalization ability [32]. The final prediction from the Random Forest is obtained by aggregating the results given by all decision trees, commonly by means of the average value. Therefore, in the RF approach, decision trees are constructed independently, but they collectively contribute to the final prediction through aggregation. Additionally, the XGBoost algorithm extends the boosting framework of AdaBoost and the ensemble construction of Random Forest to achieve a better performance. Specifically, XGBoost incorporates a shrinkage parameter that scales the contribution of each learner in the ensemble and minimizes the loss function using gradient-based optimization [33].

Finally, the SVR algorithm is based on the principles of support vector machines and involves solving a mathematical optimization problem to find the optimal hyperplane that maximizes the margin while minimizing the error between predicted and target values. Specifically, given a training set x, SVR aims to find a regression function $f(x) = w^T \phi(x) + b$, where w is the weight vector and $\varphi(x)$ is the feature mapping using a given kernel function [34]. The SVR optimization problem can be formulated by means of the minimization of the cost function given by Eq. 2.

$$J = \frac{\|w\|^2}{2} + C \sum_{i=1}^{n} |\varepsilon_k| \tag{2}$$

Where, n denotes the number of independent variables, C is a trade-off hyperparameter to adjust regularization between margin size and the amount of error allowed within that margin, and ε_k denotes the slack margin variables – tolerances for data points to fall within a certain distance from the hyperplane –.

3.2. Implementation

The algorithms mentioned in section 3.1 were trained using Python 3.9.13 and the Keras module. Fig. 1(b) illustrates the schematic of the process, which begins with data preprocessing. In this step, the input data was standardized, transforming the distributions to have a mean of zero and a standard deviation of one [35]. This process ensures that all features are on a comparable scale, avoiding any particular feature from overpowering the model due to its larger values.

After the preprocessing stage, the dataset was partitioned into two parts: 80% for training and 20% for testing. Then, to ensure the robustness of the training stage, cross-validation with 5 folds was applied. This approach involves dividing the training data into multiple subsets or folds (5 in this specific work). Consequently, the model underwent training and evaluation five times, with each fold serving as the validation set once, while the remaining folds were utilized for training [36]. This approach tends to enhance the ability of the final model to generalize to unseen data [37].

In addition to cross-validation, the Search Grid method was also employed. This approach is a commonly used technique for hyperparameter tuning, which involves selecting the optimal combination of parameters that controls the model [38]. It systematically explores different hyperparameter values by defining a grid of possible combinations. By evaluating the performance of the model using those different parameters, it is possibly to identify the optimal set that yields the best performance [36]. The specific hyperparameters considered in this work are presented in Table 1, for each of the ML algorithms evaluated.

Table 1. Hyperparameters considered during training.

Artificial Neural Network

Hyperparameter	*Values*
Activation Function	ReLu
Number of hidden layers	2
Neurons in the first hidden layer	[25,35,45,50,55,75,100,125,150]
Neurons in the second hidden layer	[5,15,20,25,30,35]
Optimizer	Adaptive Moment Estimation
Kernel Regularizer	“L2”
Batch size	[4, 8, 12]

Support Vector Regressor

Hyperparameter	*Values*
Kernel function	[“sigmoid”, “rbf”, “linear”]
Kernel coefficient	[0.01, 0.05, 0.1, 0.5, 1]
C	[0.1, 1, 10]

AdaBoost

Hyperparameter	*Values*
Number of weak estimators	[10,20,50,75,100,120,125,130, 150,175,200,225,250,275]
Learning rate	[0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 1.5]
Loss function	[“exponential”, “square”, “linear”]

XGBoost

Hyperparameter	*Values*
Number of weak estimators	[10,20,50,75,100,120,125,130, 150,175,200,225,250,275]
Maximum depth of the weak estimator	[1,4,8,12,24,32]
Learning rate	[0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 1.5]

Random Forest

Hyperparameter	*Values*
Number of weak estimators	[10,25,50,150,175,200,225,250,275,300]
Max features	[“sqrt”,”log2”]

Furthermore, during the final test phase, MSE was used as the metric to evaluate the final model's performance with unseen data. In this sense, MSE is a widely used metric for regression problems, representing the mean squared difference between the predicted values and the target values. By squaring the differences, this metric mainly emphasizes the larger errors.

4. Results and discussion

Table 2 showcases the MSE values for training and testing of each algorithm discussed in section 3, along with the corresponding best hyperparameters identified during training. These results correspond to the predictions made for wall temperatures using both the training and testing datasets, as outlined in Fig. 1(b) of the methodology section.

Table 2: Best hyperparameters obtained using Search Grid method.

Algorithm	***Hyperparameters***	***MSE*** [°C^2] Training	***MSE*** [°C^2] Testing
ANN	Neurons in the first hidden layer: 50 Neurons in the second hidden layer: 25 Batch size: 8	3.79	3.84
RF	Number of weak estimators: 175 Max features: sqrt	1.92	5.86
SVR	Kernel function: rbf Kernel coefficient: 0.01 C: 10	4.19	4.26
AdaBoost	Number of weak estimators: 130 Learning rate: 1.5 Loss function: linear	5.74	9.34
XGBoost	Number of weak estimators: 50 Maximum depth of the weak estimator: 1 Learning rate: 0.5	4.65	5.91

As observed from Table 2, the Random Forest model achieved the lowest training MSE value (1.92 °C^2). However, there is a notable discrepancy between the training and testing MSE results, indicating a higher risk of overfitting. This suggests that the model may have overly adjusted to the training data, affecting its performance when encountering unfamiliar data. On the other hand, the ANN model demonstrated comparable performance between training and testing, with MSE values of 3.79 °C^2 and 3.84 °C^2, respectively. This similarity implies that the ANN model exhibits better generalization ability compared to the Random Forest model.

The AdaBoost algorithm yielded the worst performance, with MSE values of 5.74 °C^2 for training and 9.34 °C^2 for testing. These results suggest that the model struggled to effectively fit the training data, resulting in poorer generalization compared to the ANN model when applied to the testing set. The higher testing MSE also indicates potential overfitting. In contrast, the SVR and XGBoost algorithms showed consistent MSE values between both datasets. However, these results were higher than those obtained for the ANN model, suggesting a lesser ability to generalize to unseen cases. Furthermore, in order to visually depict the performance of those five models, Fig. 2 shows the relation between predicted values and experimental results for the testing dataset.

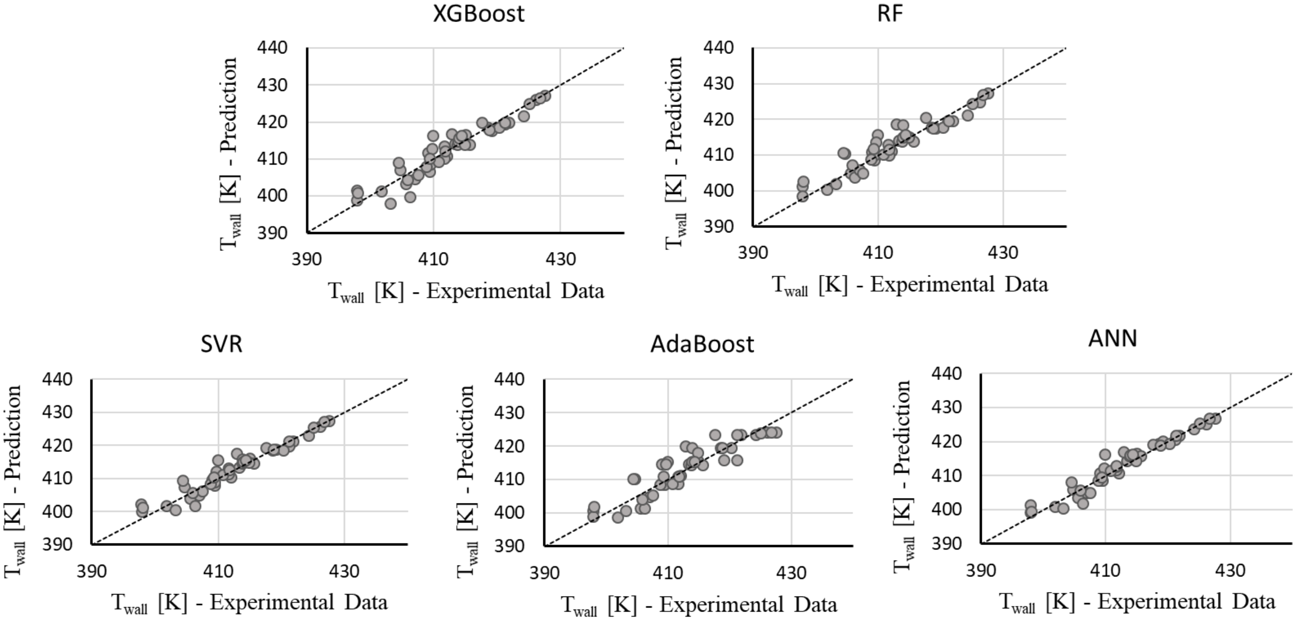

Fig. 2: Comparison between experimental data and Machine Learning predictions regarding wall temperatures.

Specifically, the x-axis represents the experimental data, while the y-axis represents the corresponding predicted values. The diagonal black lines observed in those panels represent the geometrical place where the predicted values perfectly match the experimental measurements, thus, resulting in null error. In all cases, as evidenced by the proximity of the predicted wall temperatures to the diagonal lines, the machine learning algorithms are able to effectively capture the underlying patterns of ONB points. In fact, all individual errors are in the same range, where the markers in the panel corresponding to the ANN graph cluster more clearly around the diagonal black line, which is coherent with the lower MSE shown in Table 2.

Regarding heat flux results, Table 3 shows the training and testing errors, along with the corresponding best hyperparameters obtained during training. In this sense, the XGBoost algorithm demonstrates relatively low MSE values for both the training and testing datasets (0.54 x 10^{-3} [MW/m^2]^2 and 0.56 x 10^{-3} [MW/m^2]^2 respectively), indicating good performance and generalization ability. The similar values for both errors suggest that the model is not overfitting and could perform consistently on unseen data. On the other hand, the Random Forest algorithm achieves a significantly lower MSE on the training dataset compared to the testing dataset. This discrepancy, also found for wall temperature predictions, suggests potential overfitting, where the model may struggle to generalize to new data. In addition, the AdaBoost algorithm exhibits the higher MSE values, indicating a lesser prediction capability compared with the other methods.

Table 3: Best hyperparameters obtained using Search Grid method.

Algorithm	***Hyperparameters***	***MSE*** [MW/m^2]^2 Training	***MSE*** [MW/m^2]^2 Testing
ANN	Neurons in the first hidden layer: 100 Neurons in the second hidden layer: 5 Batch size: 8	0.31 x 10^{-3}	0.61 x 10^{-3}
RF	Number of weak estimators: 150 Max features: log2	0.16 x 10^{-3}	0.85 x 10^{-3}
SVR	Kernel function: rbf Kernel coefficient: 0.05 C: 10	0.31 x 10^{-3}	0.74 x 10^{-3}
AdaBoost	Number of weak estimators: 125 Learning rate: 1.5 Loss function: square	0.87 x 10^{-3}	0.89 x 10^{-3}
XGBoost	Number of weak estimators: 75 Maximum depth of the weak estimator: 4 Learning rate: 0.1	0.54 x 10^{-3}	0.56 x 10^{-3}

Moreover, in order to visually illustrate the effectiveness of the five models in predicting heat flux, Fig. 3 presents the correlation between the predicted values and the experimental results for the testing dataset. It seems that the machine learning algorithms successfully capture the inherent patterns of the ONB points, as indicated by the close proximity of the predicted heat fluxes to the diagonal lines. Remarkably, all individual errors fall within a similar range. Notably, the markers in the XGBoost graph panel seem to exhibit a clearer clustering along the diagonal black line, also aligning with the lower MSE values depicted in Table 3.

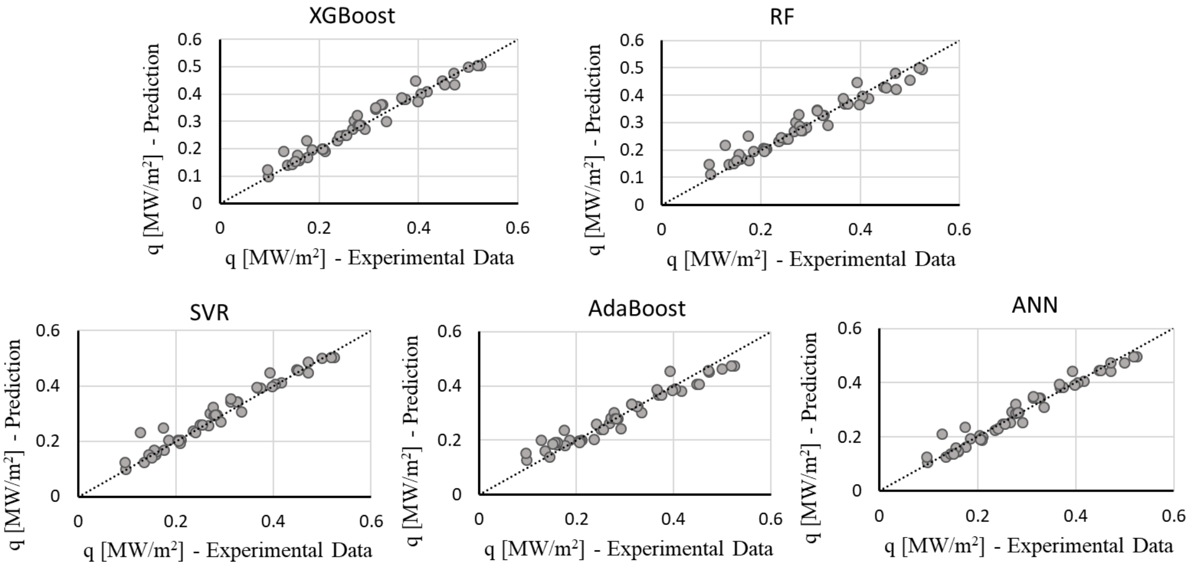

Fig. 3: Comparison between experimental data and Machine Learning predictions regarding heat flux.

4. Conclusion

In this work, Machine Learning techniques were utilized to predict the wall temperature and heat flux associated with the onset of nucleate boiling. The training and testing data were obtained from an experimental setup where the channel inclination, flow velocity, pressure, and inlet temperature of the fluid were systematically modified. Furthermore, the boiling process was induced by the Joule effect.

Five Machine Learning algorithms, including Artificial Neural Networks, Support Vector Regression, Random Forest, AdaBoost, and XGBoost, were evaluated to assess their performance. Specifically, in terms of wall temperature prediction, ANN was found as the most balanced method, with a training and testing Mean Squared Error of 3.79 $°C^2$ and 3.84 $°C^2$, respectively. On the other hand, Random Forest exhibited the largest disparity between training and testing, suggesting potential overfitting, while AdaBoost involves the worst performance. In fact, AdaBoost has also shown the worst MSE in predicting the heat flux, whereas XGBoost has demonstrated superior performance compared to ANN for that specific variable. Overall, these results demonstrate the effectiveness of these models in accurately forecasting the ONB, where future studies will focus on exploring in-depth the prediction of additional significant points along the boiling curve.

Acknowledgements

Authors are grateful to the Spanish Ministry for Science and Innovation for the financial support though the RTC2019-006955-4 and PID2020-114742RB-I00 projects.

References

[1] R. Zubair, A. Ullah, A. Khan, and M. H. Inayat, “Critical heat flux prediction for safety analysis of nuclear reactors using machine learning,” in *2022 19th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, 2022, pp. 314–318.

[2] B. T. Jiang, J. Zhou, X. B. Huang, and P. F. Wang, “Prediction of critical heat flux using Gaussian process regression and ant colony optimization,” *Annals of Nuclear Energy*, vol. 149, p. 107765, 2020.

[3] P. A. Kew and K. Cornwell, “Correlations for the prediction of boiling heat transfer in small-diameter channels,” *Applied Thermal Engineering*, vol. 17, no. 8, pp. 705–715, 1997.

[4] S. K. Saha, G. P. Celata, and S. G. Kandlikar, “Thermofluid Dynamics of Boiling in Microchannels,” in *Advances in Heat Transfer*, vol. 43, Y. I. Cho and G. A. Greene, Eds. Elsevier, 2011, pp. 77–226.

[5] D. P. Ghosh, R. Raj, D. Mohanty, and S. K. Saha, “Onset of Nucleate Boiling, Void Fraction, and Liquid Film Thickness,” in *Microchannel Phase Change Transport Phenomena*, S. K. Saha, Ed. Butterworth-Heinemann, 2016, pp. 5–90.

[6] Y. Koizumi, M. Shoji, M. Monde, Y. Takata, and N. Nagai, Eds., “Nucleate boiling,” in *Boiling*, Boston: Elsevier, 2017, pp. 13–144.

[7] A. K. Sadaghiani, R. Altay, H. Noh, H. J. Kwak, K. Şendur, B. Mısırlıoğlu, H. S. Park, and A. Koşar, “Effects of bubble coalescence on pool boiling heat transfer and critical heat flux – A parametric study based on artificial cavity geometry and surface wettability,” *International Journal of Heat and Mass Transfer*, vol. 147, p. 118952, 2020.

[8] G. Liang and I. Mudawar, "Review of spray cooling – Part 1: Single-phase and nucleate boiling regimes, and critical heat flux," *International Journal of Heat and Mass Transfer*, vol. 115, pp. 1174–1205, 2017.

[9] A. Faghri and Y. Zhang, "Boiling," in *Transport Phenomena in Multiphase Systems*, A. Faghri and Y. Zhang, Eds. Boston: Academic Press, 2006, pp. 765–852.

[10] N. Cheng, S. Yu, J. Xiao, and C.-H. Peng, "Experimental Study of Onset of Nucleate Boiling in Vertical Rectangular Channels with Different Flow Path Heights," *Science and Technology of Nuclear Installations*, vol. 2022, no. 7760569, 2022.

[11] G. Liang and I. Mudawar, "Pool boiling critical heat flux (CHF) – Part 1: Review of mechanisms, models, and correlations," *International Journal of Heat and Mass Transfer*, vol. 117, pp. 1352–1367, 2018.

[12] Y. Y. Hsu, "On the Size Range of Active Nucleation Cavities on a Heating Surface," *Journal of Heat Transfer*, vol. 84, no. 3, pp. 207–213, 1962.

[13] W. Qu and I. Mudawar, "Prediction and measurement of incipient boiling heat flux in micro-channel heat sinks," *International Journal of Heat and Mass Transfer*, vol. 45, no. 19, pp. 3933–3945, 2002.

[14] D. Liu, P.-S. Lee, and S. V. Garimella, "Prediction of the onset of nucleate boiling in microchannel flow," *International Journal of Heat and Mass Transfer*, vol. 48, no. 25, pp. 5134–5149, 2005.

[15] J. H. Lim, M. Park, S. M. Shin, and S. S. Chung, "New correlations for the prediction of incipient nucleate boiling in a one-side heated swirl tube," *Applied Thermal Engineering*, vol. 209, p. 118300, 2022.

[16] O. S. Al-Yahia and D. Jo, "Onset of Nucleate Boiling for subcooled flow through a one-side heated narrow rectangular channel," *Annals of Nuclear Energy*, vol. 109, pp. 30–40, 2017.

[17] C. Paz, M. Conde, J. Porteiro, and M. Concheiro, "Effect of heating surface morphology on the size of bubbles during the subcooled flow boiling of water at low pressure," *International Journal of Heat and Mass Transfer*, vol. 89, pp. 770–782, 2015.

[18] Z. Xu, P. Zhang, C. Yu, W. Miao, Q. Chang, M. Qiu, Y. Li, Y. Tian, and L. Jiang, "Liquid-Superspreading-Boosted High-Performance Jet-Flow Boiling for Enhancement of Phase-Change Cooling," *Advanced Materials*, vol. n/a, no. n/a, p. 2210557.

[19] Y. Liu, N. Dinh, Y. Sato, and B. Niceno, "Data-driven modeling for boiling heat transfer: Using deep neural networks and high-fidelity simulation results," *Applied Thermal Engineering*, vol. 144, pp. 305–320, 2018.

[20] S. Alotaibi, S. Ebrahim, and A. Salman, "Prediction of the Minimum Film Boiling Temperature of Quenching Vertical Rods in Water Using Random Forest Machine Learning Algorithm," *Frontiers in Energy Research*, vol. 9, 2021.

[21] J. Cai, "Applying support vector machine to predict the critical heat flux in concentric-tube open thermosiphon," *Annals of Nuclear Energy*, vol. 43, pp. 114–122, 2012.

[22] M. M. Rashidi, M. A. Nazari, C. Harley, E. Momoniat, I. Mahariq, and N. Ali, "Applications of machine learning methods for boiling modeling and prediction: A comprehensive review," *Chemical Thermodynamics and Thermal Analysis*, vol. 8, p. 100081, 2022.

[23] M. Conde-Fontenla, C. Paz, M. Concheiro, and G. Ribatski, "On the width and mean value of bubble size distributions under subcooled flow boiling," *Experimental Thermal and Fluid Science*, vol. 124, p. 110368, 2021.

[24] M. C. Paz, M. Conde, E. Suárez, and M. Concheiro, "On the effect of surface roughness and material on the subcooled flow boiling of water: Experimental study and global correlation," *Experimental Thermal and Fluid Science*, vol. 64, pp. 114–124, 2015.

[25] H. Jo, H. S. Ahn, S. Kang, and M. H. Kim, "A study of nucleate boiling heat transfer on hydrophilic, hydrophobic and heterogeneous wetting surfaces," *International Journal of Heat and Mass Transfer*, vol. 54, no. 25, pp. 5643–5652, 2011.

[26] H.-T. Kim and K.-H. Bang, "An experimental study of flow boiling from downward-facing heated wall in inclined channels," *International Journal of Heat and Mass Transfer*, vol. 133, pp. 920–929, 2019.

[27] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, 2022.

[28] S. Agatonovic-Kustrin and R. Beresford, "Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research," *Journal of Pharmaceutical and Biomedical Analysis*, vol. 22, no. 5, pp. 717–727, 2000.

[29] K. Santosh, N. Das, and S. Ghosh, "Chapter 2 - Deep learning: a review," in *Deep Learning Models for Medical Imaging*, K. Santosh, N. Das, and S. Ghosh, Eds. Academic Press, 2022, pp. 29–63.

[30] R. E. Schapire, "The Boosting Approach to Machine Learning: An Overview," in *Nonlinear Estimation and Classification*, D. D. Denison, M. H. Hansen, C. C. Holmes, B. Mallick, and B. Yu, Eds. New York, NY: Springer New York, 2003, pp. 149–171.

[31] Y. Xia, "Chapter Eleven - Correlation and association analyses in microbiome study integrating multiomics in health and disease," in *The Microbiome in Health and Disease*, vol. 171, J. Sun, Ed. Academic Press, 2020, pp. 309–491.

[32] U. A. Umoh, I. J. Eyoh, V. S. Murugesan, and E. E. Nyoho, "Fuzzy-machine learning models for the prediction of fire outbreaks: A comparative analysis," in *Artificial Intelligence and Machine Learning for EDGE Computing*, R. Pandey, S. K. Khatri, N. kumar Singh, and P. Verma, Eds. Academic Press, 2022, pp. 207–233.

[33] A. I. A. Osman, A. N. Ahmed, M. F. Chow, Y. F. Huang, and A. El-Shafie, "Extreme gradient boosting (Xgboost) model to predict the groundwater levels in Selangor Malaysia," *Ain Shams Engineering Journal*, vol. 12, no. 2, pp. 1545–1556, 2021.

[34] X. Du, H. Xu, and F. Zhu, "Understanding the Effect of Hyperparameter Optimization on Machine Learning Models for Structure Design Problems," *Computer-Aided Design*, vol. 135, p. 103013, 2021.

[35] H. Belyadi and A. Haghighat, "Chapter 3 - Machine learning workflows and types," in *Machine Learning Guide for Oil and Gas Using Python*, H. Belyadi and A. Haghighat, Eds. Gulf Professional Publishing, 2021, pp. 97–123.

[36] V. H. Kamble and M. P. Dale, "Chapter 1 - Machine learning approach for longitudinal face recognition of children," in *Machine Learning for Biometrics*, P. P. Sarangi, M. Panda, S. Mishra, B. S. P. Mishra, and B. Majhi, Eds. Academic Press, 2022, pp. 1–27.

[37] J. Delgadillo and D. Atzil-Slonim, "Artificial intelligence, machine learning and mental health," in *Reference Module in Neuroscience and Biobehavioral Psychology*, Elsevier, 2022.

[38] J. Zhang, Q. Wang, and W. Shen, "Hyper-parameter optimization of multiple machine learning algorithms for molecular property prediction using hyperopt library," *Chinese Journal of Chemical Engineering*, vol. 52, pp. 115–125, 2022.