Available online at
https://jcrinn.com/
https://crinn.conferencehunter.com/

Journal of Computing Research and Innovation 9(2) 2024

Journal of Computing Research and Innovation

Deep Learning-Driven Predictive Modelling for Optimizing Stingless Beekeeping Yields

Noor Hafizah Khairul Anuar[1], Mohd Amri Md Yunus[2*],Muhammad Ariff Baharudin[3], Sallehuddin Ibrahim[4], Shafishuhaza Sahlan[5]

[1]*Electrical Engineering Studies, College of Engineering, Universiti Teknologi MARA Johor, Pasir Gudang Campus, Masai, Malaysia*
[2,3,4,5]*Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Skudai Johor, Malaysia*

ARTICLE INFO

Article history:
Received 11 June 2024
Revised 13 August 2024
Accepted 14 August 2024
Online first
Published 1 September 2024

Keywords:
Meliponiculture
Stingless Beekeeping
Deep Learning
LSTM
RNN

DOI:
10.24191/jcrinn.v9i2.451

ABSTRACT

Environmental factors like temperature, solar irradiance, and rain may influence the health and productivity of stingless bees. This paper aims to investigate the best approaches applied in meliponiculture to predict beehive health and products based on environmental variables and bee activity data. The data on temperature, humidity, rain, beehive weight, and bee activity traffic utilized in this project were monitored in real-time and saved on the Google Spreadsheet platform. The dataset extracted from the 6th of January 2024 to the 5th of February 2024, at a 15-minute time interval comprising a total of 2577 data points was analyzed using various deep learning approaches for best RMSE performance. A single-layer LSTM model with 50 units produced the best RMSE performance of 0.039, representing that the beehive weight was accurately predicted. This predictive capability can help farmers determine the optimum harvesting time based on weight forecasts, ensuring maximum yield and quality. Additionally, by providing early warnings of unwanted conditions such as swarming or potential attacks, this method significantly enhances the ability of beekeepers to take proactive measures to protect their colonies, safeguarding both bee populations and the livelihoods of farmers.

1. INTRODUCTION

The technical term for the beekeeping process of stingless bee 'Kelulut' is known as meliponiculture, whereas the traditional beekeeping of honeybee species is known as apiculture. The name of stingless bees' superfamily is Apoidea, and the family is Apidae, hence the subfamily is Meliponinae. This subfamily includes two main genera, Meliponi and Trigona, with over 500 species thriving in tropical regions for over 65 million years (Roubik, 2006). Environmental factors such as temperature, humidity, rain, and solar irradiance are most influenced by the health and productivity of bee colonies. Bees frequently regulate their hive temperature between 33°C and 36°C to prevent stress and overheating inside the beehive (Becker et al., 2018). Meanwhile, higher humidity may promote fungus and mold growth in the region with higher

[2*] Corresponding author. *E-mail address*: amri@utm.my
https://dx.doi.org/10.24191/jcrinn.v9i2.451

humidity air. This increases the risk of viruses and bacterial infections, leading bees to avoid affected areas for brood cells and food storage. The rising in global temperatures and unbalanced air moisture may lead to unwanted situations like colony death (Zacepins et al., 2011), swarming (Rybin et al., 2017), low-quality honey (Meitalovs et al., 2009), and colony disappearance (Kridi et al., 2016). This research aims to investigate a short-term forecasting model utilizing a deep learning approach to predict honey production, hence observing the beehive health conditions. The system may help beginner farmers conduct habitat surveys to improve and maintain colony quality and health.

1.1 Data Analysis

Deep and machine learning approaches are tools utilized in time series analysis. The algorithm can be categorized into three types: supervised, unsupervised, and reinforcement learning. Deep learning is a subset of machine learning and involves hierarchical learning to build complex concepts. Examples of supervised learning algorithms include Feedforward Neural Networks (FFNNs), Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-term Memory (LSTM) models, and Gated Recurrent Units (GRUs). Traditional statistical methods like Autoregressive Moving Average (ARMA) and Seasonal Autoregressive Integrated Moving Average (SARIMA) have been used for time series forecasting. However, Artificial Neural Networks (ANN) and recurrent Neural Networks (RNN) models are now being explored for their ability to capture spatial and temporal features from time-series data, with LSTM and GRU networks providing superior performance in learning long-term dependencies (Bontempi et al., 2013; Mahesh, 2018; Usama et al., 2019). In beekeeping, there is limited research applying deep and machine learning to data analysis especially in meliponiculture. Table 1 outlines the previous studies in beekeeping related to the deep and machine learning approach. Significant study in meliponiculture by (Gomes et al., 2020) performed forecasted of stingless bee forager activity patterns, by experimental up to 60 hours forecasting window size based on the bee activity data, video observations, and solar irradiance data near the hive entrance.

Table 1. The summary of the deep and machine learning approach applied in the research of beekeeping

Beekeeping species	Method	Research Output	Strength	Limitations	Reference
Meliponiculture	Deep learning	Bee activity forecast	Integrating local meteorological data with bee traffic data near the hive entrance to forecast bee activity with a window size of up to 60 hours	Utilizing RFID tags glued to the bees' thoraxes to capture bee traffic data; limited to short-term monitoring (bees' lifespan is short), reduce bees' flight capacity, and required regular maintenance.	Gomes et al. (2020)
Apiculture	Machine Learning	Queen bee classification	Contactless method to differentiate various queen bee situations inside the hive	Utilizes offline recorded data that does not accurately reflect the live conditions inside the hive.	Howard et al. (2013)
Apiculture	Machine Learning	Beehive health classification	utilized large-scale apiary data collected over 3 years and employed a classification model to detect the health status of hives, identifying them as healthy, unhealthy, or collapsed.	Used offline data to classify hive conditions, which do not reflect live conditions or forecast future events.	R. Braga, G. Gomes, Rogers, et al. (2020)

https://dx.doi.org/10.24191/jcrinn.v9i2.451

Apiculture	Machine Learning	Swarming and brood-rearing classification	Classify the honeybee state in the winter season	Depending on the temperature data only	Armands Kviesis, 2016; Kviesis et al. (2020)

2. METHODOLOGY

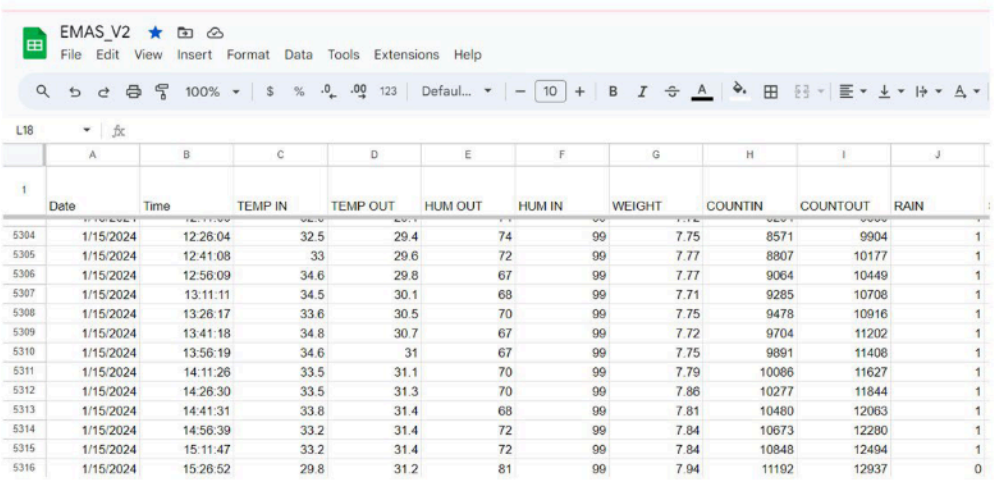
2.1 The Dataset

In this study, data on environmental and bee activities were collected from the 6th of January 2024 to the 5th of February 2024, at a 15-minute time interval comprising a total of 2577 data points. Variables consist of temperature, humidity, beehive weight, and bee traffic activity downloaded from the wireless stingless bee monitoring system located at the 'Kelulut Farm' of Kolej Dato' Onn Jaafar, Universiti Teknologi Malaysia in Johor, Malaysia (Latitude: 1.575534, Longitude: 103.61971). The developed meliponi monitoring system pushed and stored the observed variables to Google Spreadsheet. The selection criteria for the bee colony samples include factors such as species, health condition, and age. The chosen species for this study is the stingless bee, Heterotrigona itama. This species is preferred due to its common presence in the study area, its reputation for producing high-quality products, and its hardworking nature, making it an easily accessible sample pool. Fig. 1(a) presents the system environments, Fig. 1(b) shows the frontal view of the monitoring system, and Fig. 1(c) provides an inner view of the monitoring system used in this project. The Meliponi monitoring system utilizes sensors such as the DHT22 for temperature and humidity, Load Cell for beehive weight, and two pairs of infrared emitters and photoresistors for counting departing and arriving bees (representing bee activity). Fig. 2 shows an example of the Google spreadsheet page where the observed variables are being pushed and stored frequently. In addition to the variables extracted from the monitoring system, some environmental data from the Solcast Satellite system were downloaded for similar time intervals (15 minutes) to ensure data synchronization during the analysis (Solcast, 2019).

(a) (b) (c)

Fig. 1. (a) The monitoring system environments (b) The frontal view of the monitoring system (c) An inner view of monitoring hardware consisting of microcontroller and solar rechargeable battery power system

https://dx.doi.org/10.24191/jcrinn.v9i2.451


EMAS_V2

File Edit View Insert Format Data Tools Extensions Help

	Date	Time	TEMP IN	TEMP OUT	HUM OUT	HUM IN	WEIGHT	COUNTIN	COUNTOUT	RAIN
5304	1/15/2024	12:26:04	32.5	29.4	74	99	7.75	8571	9904	1
5305	1/15/2024	12:41:08	33	29.6	72	99	7.77	8807	10177	1
5306	1/15/2024	12:56:09	34.6	29.8	67	99	7.77	9064	10449	1
5307	1/15/2024	13:11:11	34.5	30.1	68	99	7.71	9285	10708	1
5308	1/15/2024	13:26:17	33.6	30.5	70	99	7.75	9478	10916	1
5309	1/15/2024	13:41:18	34.8	30.7	67	99	7.72	9704	11202	1
5310	1/15/2024	13:56:19	34.6	31	67	99	7.75	9891	11408	1
5311	1/15/2024	14:11:26	33.5	31.1	70	99	7.79	10086	11627	1
5312	1/15/2024	14:26:30	33.5	31.3	70	99	7.86	10277	11844	1
5313	1/15/2024	14:41:31	33.8	31.4	68	99	7.81	10480	12063	1
5314	1/15/2024	14:56:39	33.2	31.4	72	99	7.84	10673	12280	1
5315	1/15/2024	15:11:47	33.2	31.4	72	99	7.84	10848	12494	1
5316	1/15/2024	15:26:52	29.8	31.2	81	99	7.94	11192	12937	0

Fig. 2. The screenshot of the Google Spreadsheet page for saving the observed variables from the monitoring system

2.2 The Model

Time series are analyzed for various purposes, such as forecasting future trends based on past data, gaining insights into underlying phenomena, and providing concise summaries of key characteristics. In this study, the RNN, LSTM, and GRU time series models were rigorously tested to evaluate their performance and effectiveness. First, the input dataset undergoes data preparation, segmentation, and splitting into training and testing sets. Next, network design analysis is performed to tune the model and hyperparameters for optimal performance. Finally, the model is trained and evaluated using specific metrics to assess its performance. Initially, nine variables were chosen as input for this study: beehive weight (kg), outside hive temperature (°C), relative humidity (%), inside hive temperature (°C), inside hive relative humidity (%), the number of incoming and outgoing bees passing through the beehive funnel, rain status (0 for no rain and 1 for rain), and the GHI reading from the Solcast satellite to represent solar irradiance. After a feature selection process utilizing correlation analysis, only the most influential variables were retained as inputs to ensure better model performance. The data were then filtered to select the optimal inputs for the system. Subsequently, the models used in this study, including RNN (Recurrent Neural Network), LSTM (Long Short-Term Memory), and GRU (Gated Recurrent Unit), were tested and validated. For the RNN deployment, Keras (https://keras.io) with the Theano (http://deeplearning.net/software/theano) backend was utilized. Scikit-Learn (https://scikit-learn.org/stable) was also employed for obtaining metrics and methods for normalization. The RNN was built using Python 3.7.

2.2.1. The Linear Correlation Analysis

Linear correlation analysis is crucial for understanding the relationships between observed attributes, assessing data quality and reliability, and supporting predictive modeling, feature selection, and decision-making. To avoid 'not a number' (NaN) results during correlation analysis, it is important to address missing and constant values within each variable. In dataset cycle 1, the inside humidity variable consistently exhibited a constant value of 99 for most of its data. The `pandas.corr()` function in Python handles missing values through pairwise deletion, calculating correlations using only rows with non-missing values for both variables. When two variables have zero variance, their correlation cannot be computed, resulting in NaN values in the correlation matrix. Therefore, the inside humidity variable was removed for the correlation

https://dx.doi.org/10.24191/jcrinn.v9i2.451

analysis. The resulting correlation matrix for dataset cycle 1, after excluding the inside humidity attribute, is shown in Fig. 3.

	TEMP OUT	TEMP IN	HUM OUT	HIVE WEIGHT	COUNT IN	COUNT OUT	RAIN STATUS	GHI
TEMP OUT	1.00							
TEMP IN	0.74	1.00						
HUM OUT	-0.97	-0.76	1.00					
HIVE WEIGHT	0.15	0.26	-0.19	1.00				
COUNT IN	0.56	0.55	-0.54	0.02	1.00			
COUNT OUT	0.63	0.60	-0.62	0.10	0.98	1.00		
RAIN STATUS	0.29	0.22	-0.35	0.31	-0.01	0.04	1.00	
GHI	0.80	0.56	-0.78	0.09	0.63	0.66	0.19	1.00

Fig. 3. The correlation matrix heatmap of the dataset from 6th January 2024 to 5th January 2024

The correlation heatmap provides insights into the relationships between various variables. For the first column, outside and inside temperature variables show a strong positive correlation of 0.74, indicating that as outside temperature increases, inside temperature also tends to increase. Additionally, both outside and inside temperatures have strong negative correlations with outside humidity, at -0.97 and -0.76 respectively, suggesting that higher temperatures are associated with lower humidity levels. The beehive weight variable shows a weak positive correlation with outside temperature (0.15) and inside temperature (0.26), but a moderate negative correlation with outside humidity (-0.19), indicating that warmer and drier conditions may slightly increase hive weights. Both bee count variables (incoming and outgoing bees) exhibit moderate positive correlations with outside temperature (0.55 and 0.63) and moderate negative correlations with outside humidity (-0.54 and -0.62), suggesting that warmer temperatures and lower humidity levels may be associated with higher bee activity. The correlation between bee counts and rain status shows a weak positive correlation, indicating that rainy conditions may not much impact bee activity and environmental conditions. Lastly, the GHI (Global Horizontal Irradiance) has a strong positive correlation with temperature and a strong negative correlation with humidity, suggesting that higher solar irradiance is associated with warmer and drier conditions. Since all variables show weak to moderate correlations, the first model investigation will consider all variables to decide the best model topology for this study. In the investigation of the best network topology to forecast the beehive weight, some combinations of networks consisting of RNN, LSTM, and GRU were tested. Six different topologies were experimented with in this study consisting of 1-layer RNN networks with 50 units, 1-layer LSTM with 50 units, 2-layer LSTM networks with 50 units, 1-layer Stacked LSTM networks with 50 units, 1-layer Bidirectional LSTM with 50 units, and 1-Layer GRU networks with 50 units.

Fig. 4 illustrates an example of a model architecture using LSTM networks. Variations may include other networks such as RNN and GRU, as previously described. The model predicts the beehive's weight (kg) based on various input features. It consists of an input layer, two hidden LSTM layers, and an output-dense layer. The input layer accepts 9 features: hive weight (kg), inside hive temperature, inside hive humidity, outside hive temperature, outside hive humidity, rain status, incoming bee count, outgoing bee count, and solar irradiance. The datasets were segmented into train and test data approximately 70% and 30 % respectively. During the training process, the model was fitted to the training network, monitoring the convergence of the loss function over epochs (set to 50 times) to evaluate learning effectiveness and tuning hyperparameters such as batch size (16, 32, 64, and 128).

https://dx.doi.org/10.24191/jcrinn.v9i2.451

After training, the model evaluation process involved using unseen testing (remaining 30%) data to assess the model's generalization performance and calculating metrics such as Root Mean Square Error (RMSE), R-squared (R^2), Mean Absolute Percentage Error (MAPE), and Mean Absolute Error (MAE). R^2 measures the proportion of variance in the dependent variable that is predictable from the independent variables, with a higher R^2 (close to 1) indicating a better fit. RMSE measures the square root of the average squared differences between actual and predicted values. MAPE calculates the average percentage difference between actual and predicted values. MAE computes the average absolute difference between actual and predicted values, with lower RMSE, MAPE, and MAE values indicating better accuracy. This method is crucial for gauging the model's accuracy and performance. If the performance is unsatisfactory, adjusting hyperparameters or retraining the model may be necessary.

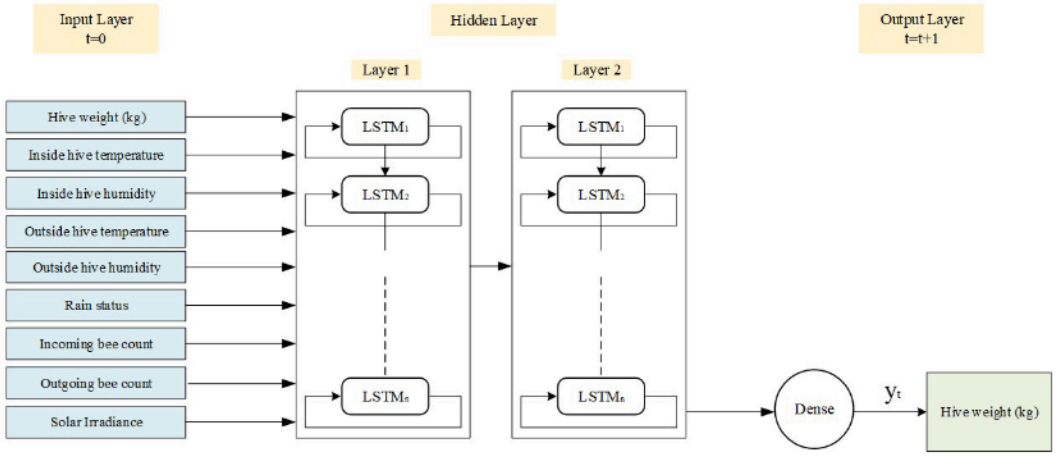


Fig. 4. The example of model architecture using two-layer LSTM networks

3. RESULT

In the first evaluations, various network designs were experimented with for dataset training and tests to find the best and lowest RMSE results. Table 2 presents a performance summary comparing various Multivariate Time Series Forecast Models, evaluated using Root Mean Square Error (RMSE). The LSTM networks, both single-layered and two-layered, achieve the lowest RMSE values, ranging from 0.040 to 0.049, with no significant difference between the two. The Bidirectional LSTM network also shows a low RMSE of 0.047. Based on these findings, this study has selected a single-layer LSTM network with 50 LSTM units for further analysis. This choice is motivated by the simplicity of the design, given the negligible variations in accuracy among the different architectures, thereby minimizing complexity. By evaluating Model 1, a 1-layer LSTM with 50 units, the loss function and optimizer were varied to tune and stabilize model performance. The experiment found that using the Mean Absolute Error (MAE) loss function and the Adam optimizer yielded the most accurate results for this dataset, achieving an RMSE of 0.038. The optimal batch size was determined to be 16.

Fig. 5(a) illustrates the epoch evaluation trends for both training and testing of Model 1. The decreasing trends in the plot indicate that the model is effectively learning from the data. The dashed line, representing testing loss, shows a slightly higher value before approximately 20 epochs, indicating the model's ability to generalize to new, unseen data. This suggests that the model is not significantly overfitting. After around 20 epochs, the model's performance stabilizes, indicating that additional training

https://dx.doi.org/10.24191/jcrinn.v9i2.451

beyond this point will not yield further improvements. This stabilization suggests that the model has reached its learning capacity. The exact point of stabilization may vary depending on changes in hyperparameters such as the learning rate, the number of LSTM units, and the batch size. Fig. 5(b) presents the forecasted beehive weight plotted against the actual data. The white region represents the training phase using approximately 1,800 data points (about 70% of the total dataset). The green region indicates the testing/forecast phase using the remaining 776 data points, which make up 30% of the total dataset of 2,577 data points.

The process resulted in an RMSE of 0.039, and the R-squared coefficient of determination was 0.924, which is close to one. An R-squared value near one indicates a near-perfect fit of the model, while a value of zero would suggest no explanatory power. The Mean Absolute Percentage Error (MAPE) for this forecast testing is 0.492, with lower MAPE values generally indicating greater predictive accuracy. Additionally, the Mean Absolute Error (MAE) for the forecast testing is 0.028, reflecting the average magnitude of errors between the forecasted and actual values. A lower MAE value signifies higher accuracy. In summary, the testing process of the developed model demonstrates strong performance, evidenced by low RMSE and MAE values, and a high R-squared value, indicating accurate and reliable predictions.

Table 2. The summary of training and testing performance between various multivariate time series forecast topology

Model	Networks	Layer	Units	RMSE
1	LSTM	1	50	**0.040**
2	LSTM	1	100	0.049
3	LSTM	2	50	0.047
4	Stacked LSTM	1	50	0.070
5	Bidirectional LSTM	2	50	0.047
6	RNN	2	50	0.087
7	GRU	2	50	0.051

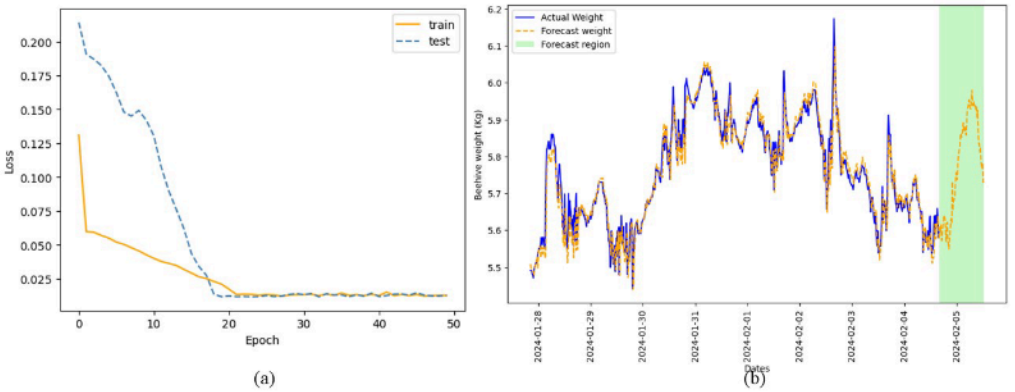


Fig. 5. (a) The epoch evaluation trends for both training and testing of Model 1 (b) The training and forecast of beehive weight from the saved models

https://dx.doi.org/10.24191/jcrinn.v9i2.451

4. CONCLUSION

This study focuses on using deep learning approaches to forecast the beehive weight at 15-minute time intervals before the events. Data was collected from a monitoring system in 'Kelulut Farm', located at Universiti Teknologi Malaysia, including environmental variables and bee activities data. Nine variables, including beehive weight, temperatures, humidity, bee count, rain status, and GHI reading were initially selected. After a feature selection process, only influential variables were retained. Various models, including RNN, LSTM, and GRU, were tested and validated. Correlation analysis revealed relationships between variables, such as temperature and humidity, bee activity, and solar irradiance. All variables will be considered for the first model investigation. The various models such as RNN, LSTM, and GRU were experimented with various network topologies such as 1-layer RNN, LSTM, stacked LSTM, bidirectional LSTM, and GRU networks with 50 units for beehive weight forecasting resulted in single-layer LSTM with 50 units produce best fits in RMSE 0.039, represented the beehive weight were predicts accurately before the events.

5. ACKNOWLEDGEMENTS/FUNDING

The author would like to express sincere gratitude to Universiti Teknologi Malaysia (UTM) for providing the facilities essential for data collection in this research.

6. CONFLICT OF INTEREST STATEMENT

The authors agree that this research was conducted in the absence of any self-benefits, commercial or financial conflicts and declare the absence of conflicting interests with the funders.

7. AUTHORS' CONTRIBUTIONS

All authors contributed equally to the research and writing of this paper.

8. REFERENCES

Armands Kviesis, A. Z. (2016). Application of neural networks for honey bee colony state identification. In *2016 17th International Carpathian Control Conference (ICCC)* (pp. 413-417). IEEE Xplore. https://doi.org/10.1109/CarpathianCC.2016.7501133

Becker, T., Pequeno, P. A. C. L., & Carvalho-Zilse, G. A. (2018). Impact of environmental temperatures on mortality, sex and caste ratios in Melipona interrupta Latreille (Hymenoptera, Apidae). *Science of Nature*, *105*(9), 55. https://doi.org/10.1007/s00114-018-1577-6

Bontempi, G., Ben Taieb, S., & Le Borgne, Y. A. (2013). Machine learning strategies for time series forecasting. *Applied Mechanics and Materials*, 263–266(PART 1), 62–77. https://doi.org/10.4028/www.scientific.net/AMM.263-266.171

Gomes, P. A. B., Suhara, Y., Nunes-Silva, P., Costa, L., Arruda, H., Venturieri, G., Imperatriz-Fonseca, V. L., Pentland, A., Souza, P. de, & Pessin, G. (2020). An Amazon stingless bee foraging activity predicted using recurrent artificial neural networks and attribute selection. *Scientific Reports*, *10*(1), 1–13. https://doi.org/10.1038/s41598-019-56352-8

Howard, D., Duran, O., Hunter, G., & Stebel, K. (2013). Signal processing the acoustics of Honeybees

(Apis Mellifera) to identify the "Queenless" State in Hives. In the *Proceedings of the Institute of Acoustics* (pp. 290-297).

Kridi, D. S., de Carvalho, C. G. N., & Gomes, D. G. (2016). Application of wireless sensor networks for beehive monitoring and in-hive thermal patterns detection. *Computers and Electronics in Agriculture*, *127*, 221–235. https://doi.org/10.1016/j.compag.2016.05.013

Kviesis, A., Komasilovs, V., Komasilova, O., & Zacepins, A. (2020). Application of fuzzy logic for honey bee colony state detection based on temperature data. *Biosystems Engineering*, *193*, 90–100. https://doi.org/10.1016/j.biosystemseng.2020.02.010

Mahesh, B. (2018). Machine learning algorithms - A review. International Journal of Science and Research, 9(1), 381-386. https://doi.org/10.21275/ART20203995

Meitalovs, J., Histjajevs, A., & Stalidzans, E. (2009). Automatic microclimate controlled beehive observation system. In *8th International Scientific Conference 'Engineering for Rural Development'* (pp. 265–271).

R. Braga, A., G. Gomes, D., M. Freitas, B., & A. Cazier, J. (2020). A cluster-classification method for accurate mining of seasonal honey bee patterns. *Ecological Informatics*, *59*, 101107. https://doi.org/10.1016/j.ecoinf.2020.101107

Roubik, D. W. (2006). Stingless bee nesting biology. *Apidologie*, *37*, 124–143. https://doi.org/10.1051/apido

Rybin, V. G., Butusov, D. N., Karimov, T. I., Belkin, D. A., & Kozak, M. N. (2017). Embedded data acquisition system for beehive monitoring. In *Proceedings of 2017 IEEE 2nd International Conference on Control in Technical Systems (CTS 2017)* (pp. 387–390). IEEE Xplore. https://doi.org/10.1109/CTSYS.2017.8109576

Solcast. (2019). *Global solar irradiance data and PV system power output data.* https://solcast.com/

Usama, M., Qadir, J., Raza, A., Arif, H., Yau, K. A., Elkhatib, Y., Hussain, A., & Al-Fuqaha, A. (2019). Unsupervised machine learning for networking: Techniques, applications and research challenges. *IEEE Access*, *7*, 65579–65615. https://doi.org/10.1109/ACCESS.2019.2916648

Zacepins, A., Meitalovs, J., Komasilovs, V., & Stalidzans, E. (2011). Temperature sensor network for prediction of possible start of brood rearing by indoor wintered honey bees. In *Proceedings of the 2011 12th International Carpathian Control Conference (ICCC' 2011)* (pp. 465–468). https://doi.org/10.1109/CarpathianCC.2011.5945901

https://dx.doi.org/10.24191/jcrinn.v9i2.451