TELKOMNIKA Telecommunication, Computing, Electronics and Control
Vol. 22, No. 4, August 2024, pp. 976∼984
ISSN: 1693-6930, DOI: 10.12928/TELKOMNIKA.v22i4.25868

IoT-based flood disaster early detection system using hybrid fuzzy logic and neural networks

Muhammad Adib Kamali[1], Mochamad Nizar Palefi Ma'ady[2]
[1]Department of Information Technology, Faculty of Computing, Telkom University, Surabaya, Indonesia
[2]Department of Information Systems, School of Industrial Engineering, Telkom University, Surabaya, Indonesia

Article Info

Article history:

Received Nov 17, 2023
Revised Mar 23, 2024
Accepted Mar 29, 2024

Keywords:

Flood detection
Fuzzy model
Neural networks
Social media
Wireless sensor network

ABSTRACT

A flood stands as one of the most common natural occurrences, often resulting in substantial financial losses to property and possessions, as well as affecting human lives adversely. Implementing measures to prevent such floods becomes crucial, offering inhabitants ample time to evacuate vulnerable areas before flood events occur. In addressing the flood issue, numerous scholars have put forth various solutions, such as the development of fuzzy system models and the establishment of suitable infrastructure. However, when applying a fuzzy system, it often results in a loss of interpretability of the fuzzy rules. To address this issue effectively, we propose to reframe the optimization problem by incorporating stage costs alongside the terminal cost. Results show the proposed model called hybrid fuzzy logic and neural networks (NNs) can mitigate the loss of interpretability. Results also show that the proposed method was employed in a flood early detection system aligned with integrating into Twitter social media. The proposed concepts are validated through case studies, showcasing their effectiveness in tasks such as XOR-classification problems.

Corresponding Author:

Muhammad Adib Kamali
Department of Information Technology, Faculty of Computing, Telkom University
60231, Surabaya, Indonesia
Email: adibmkamali@telkomuniversity.ac.id

1. INTRODUCTION

Floods often occur in Indonesia, especially during the rainy season [1], [2]. This information is obtained from the National Disaster Management Agency (BNPB), which informs that disasters in Indonesia frequently transpire from early January to the end of October, resulting in a total of 1,238 flood events [2]. These floods can have significant impacts on communities, causing damage to infrastructure, disrupting livelihoods, and posing threats to human safety [3]. Efforts to manage and mitigate the effects of flooding are crucial to safeguarding both property and lives in these vulnerable regions [4]. Advanced systems like early warning systems [5], real-time monitoring of weather patterns [6], and flood modeling have been implemented to enhance the accuracy of flood predictions and provide residents with actionable insights [7]. These technological advancements play a pivotal role in enhancing disaster preparedness and response, ultimately contributing to the reduction of both human and material losses caused by flooding [8]. Acknowledging Indonesia's susceptibility to floods, we are at the forefront of implementing IoT-powered flood early detection technology. By seamlessly integrating real-time data from diverse sources including water flow and water levels [9], our advanced predictive models will proactively issue alerts [10], concentrating initially on regions with high vulnerability such as Surabaya, Jakarta and many more [11]. The area under our observation was the Rungkut district area in Surabaya. Our focus on a district in Surabaya city stemmed from information sourced from online news, which indicated that the flooding issue, particularly along the roadway, was attributed to damaged culverts [12]. Driven by these identified issues, our objective is to establish and deploy an advanced early detection system for flood disasters across Indonesia. This system will primarily focus on detecting rapid increases in water levels [13], aiming to facilitate timely anticipatory measures and early evacuation through a bespoke IoT-driven solution [14].

Furthermore, The proposed system utilized a modified fuzzy model. Takagi-Sugeno-Kang (TSK) fuzzy model was developed in the 1980s with popularity in wide contexts using linear system [15]. The algorithm of "if-then" rules has enjoyed popularity since it includes the domain expert. However, the learning scheme turns out to be not useful when the rules may differ in a complex real-world problem, and then the interpretability of fuzzy rules may be lost [16]. The related research [17] compared the performance evaluation of backpropagation (BP) neural networks (NNs), fuzzy logic, and neuro-fuzzy system for flood prediction.

Based on the above-aforementioned problem so as to optimize a large model, this paper proposes a nonlinear actuator with online hidden-node teaching for the BP procedure. Nonlinear actuator was also successfully identified by Bounemeur and Chemachema [18] for fault-tolerant application. In summary, the main contribution of our works is three folds;

- Hidden-node teaching of multi-layer NNs remedies the interpretability of fuzzy logic by minimizing a stage cost, instead of relying on an expert's "if-then" rules.
- Decision-boundary separating planes in creating NN architecture help to optimize the initial weight parameters instead of randomly-weights setting [19].
- An intelligent wireless sensor is considered in a hybrid system, which is proposed to cast the disaster information automatically to Twitter social media [20].

2. SYSTEM OVERVIEW

This section describes the proposed system workflow to solve the problem defined in section 1. Figure 1 shows the algorithm workflow. The proposed method starts with collecting valuable data such as water level and water flow. The collected data are normalized to transform features to be on a similar scale. Furthermore, the hybrid fuzzy logic and NNs called neuro-fuzzy hidden-node teaching NNs are prepared. This process involves the decision-boundary separating plane process, neuro-model fuzzy modeling, four-step hidden-node teaching formulation, and incorporating with the fuzzy system [21].

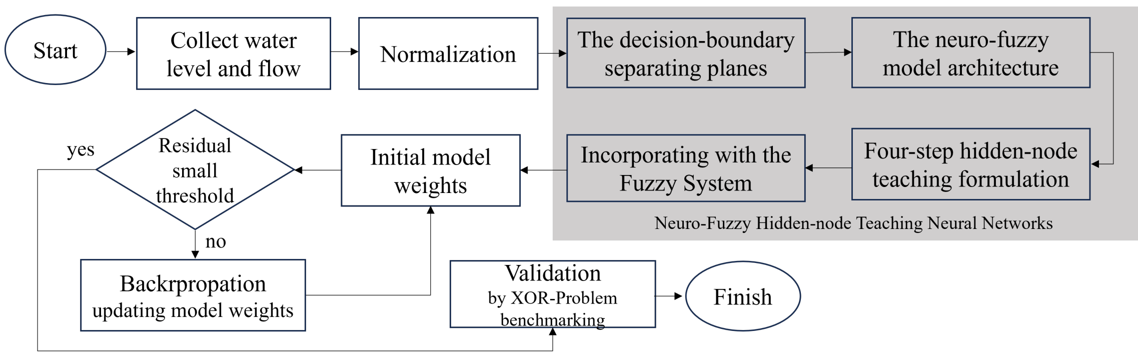

Figure 1. Flow of the proposed algorithm

2.1. Decision-boundary separating planes

Artificial neural-network learning involving multiple layers (i.e., deep learning) is a nonlinear, discrete multi-stage optimal-control problem. In fact, the roots of BP for deep learning can be traced back to the work of *multi-stage optimal-control gradient procedures* developed in early 1960s notably by H. Kelley (1960), A. Bryson (1961), and S. Dreyfus (1962). BP facilitates efficient stage-wise implementation of chain rules for differentiation by exploiting multiple-layered structure. Our objective is to find an optimal control sequences (θ_i) so as to minimize a general criterion J as the sum of stage costs L_i (at stage i) plus a terminal state cost ϕ.

The decision-boundary separating planes is inspired by the context of XOR problem since it differs the fuzzy logic that unable to learn XOR data, which called a linear separator. Hence, our two decision-boundary lines produce the desired outputs in the model. Any data points that falls exactly on the boundaries should be ignored. Points on line could be 0.5 as denoted by white points. This approach is creating appropriate initial weights for speeding up epoch running process. The BP is a simplified form of the gradient procedure. Table 1 shows a multi-layer neural-network learning when the criterion is denoted by J. For our convenience, ξ_i is called output sensitivity whereas δ_i is net-input sensitivity.

Table 1. Notation definition

Notation	NN learning
θ	Weight parameters
y	Net outputs
x	Net inputs
f(x)	Nonlinear node
s	Layer
L_s	Hidden-node teaching
xi_s	(Psi-rule) output sensitivity
δ_s	(Delta-rule) input sensitivity

2.2. Four-step hidden-node teaching formulation

The early flood detection-gate mapping can be realized by combining AND, OR, and NAND gates. For our 2-2-1 NN learning, suppose that we want to make two hidden-node outputs *close* to the NAND and OR gates outputs. Then, we must use those NAND and OR responses as *target signals* at two hidden nodes; this scheme is what we call hidden-node teaching. Accordingly, the objective function J must be so modified as to include a stage cost of the sum of squared hidden residuals, as shown (1):

$$J = \sum_{d=1}^{D} J(d), \text{ with } J(d) = \underbrace{\frac{1}{2}\left[y_1^{(1)}(d) - t_1^{(1)}(d)\right]^2}_{\text{first hidden-node cost}} + \underbrace{\frac{1}{2}\left[y_2^{(1)}(d) - t_2^{(1)}(d)\right]^2}_{\text{second hidden-node cost}} + \underbrace{\frac{1}{2}\left[y_1^{(2)}(d) - t(d)\right]^2}_{\text{terminal-state cost}} \tag{1}$$

Here, $t_k^{(1)}(d)$ is the teacher signal for hidden node k ($k = 1, 2$) at (hidden) layer 1 on each datum d, $d = 1, ..., D$ where $D = 15$ for the early flood detection problem. This is a two-stage ($N = 2$) optimal-control problem of the following form:

$$J(d) = L_1\left(y_k^{(1)}(d), -\right) + \phi\left((y)_1^{(2)}(d)\right) = \frac{1}{2}\left|\left|y_k^{(1)}(d) - t_k^{(1)}(d)\right|\right|^2 + \frac{1}{2}\left[y_1^{(2)}(d) - t(d)\right]^2 \tag{2}$$

where the stage cost L_1 (at stage 1) is independent of the *control vector* $\theta^{0,1}$, depending only on the state vector $y^{(1)}$. We have derived the BP procedure for $J = \phi\left(y_1^{(2)}\right)$, the terminal-state cost only.

- Definition of nominal cost-to-go action-value-function
  $T_s(y^s, \theta_+^{s,s+1})$ nominal cost-to-go starting, stage s in state y^s using a guessed control sequence $\theta_+^{s,s+1}$, $\theta_+^{s+1,s+2}$, ..., $\theta_+^{N-1,N}$.
- Recurrence relation
  L_s denotes the immediate cost.

$$T_s(y^s, \theta_+^{s,s+1}) = L_s(y^s, \theta_+^{s,s+1}) + T_{s+1}(y^{s+1}, \theta^{s+1,s+2}) \tag{3}$$

- Boundary condition
  In this setting, the sensitivity δ at terminal stage is given by:

$$T_N(y^N, -) = \phi(y^N) \tag{4}$$

- Backward-pass
  Departing from terminal layer N, it begins with:

$$\xi^N = \frac{\partial T_N}{\partial y^N} = \frac{\partial \phi}{\partial y^N} \tag{5}$$

then it is backpropagated one after another by delta-rule.

2.3. Incorporating with the fuzzy system

Online BP algorithms can be enticing when dealing with large-scale problems. However, their application in optimizing both the "if" and "then" parts of fuzzy rules can sometimes hinder the extraction of meaningful knowledge post-learning. In the initial setup of fuzzy rules, the linear parameters of the "then" part are typically initialized within a small range, following a similar practice to MLP learning. While aiming for improved precision in input-output mapping, online BP may excessively adapt the parameters of initial fuzzy rules, leading to the creation of fuzzy rules that lack meaningful interpretation. This situation presents a dilemma between achieving mapping precision and preserving interpretability.

Returning to our motivating example, Figure 2 vividly illustrates this architecture. This approach can be attributed to the fact that standard supervised learning relies solely on a terminal cost. Consequently, it primarily encourages the final combined output to closely match the desired output, aligning with the principles of complementary learning. However, within the context of fuzzy systems, the linear output of the "then" part for each rule should ideally align with the desired output when its firing strength is substantial. This aligns with the original assertion made for a TSK fuzzy system [22].

The degree of membership within the set of inundated pixels is iteratively adjusted, taking into account the relational dynamics among neighboring pixels. This adjustment process factors in several probabilities: the likelihood of an isolated flooded pixel occurring amidst a region predominantly composed of non-flooded pixels (or vice versa) is low; the probability of encountering a non-inundated pixel in close proximity to flooded pixels situated at higher elevations is minimal; similarly, the likelihood of an inundated pixel coexisting with non-inundated ones positioned at lower elevations is also remote. Ultimately, the map delineating the flooded regions is derived through a procedure known as defuzzification. This process involves a straightforward application of a threshold to the membership degree. Specifically, pixels exhibiting a membership degree surpassing a predetermined threshold value of 0.5 are classified as flooded.

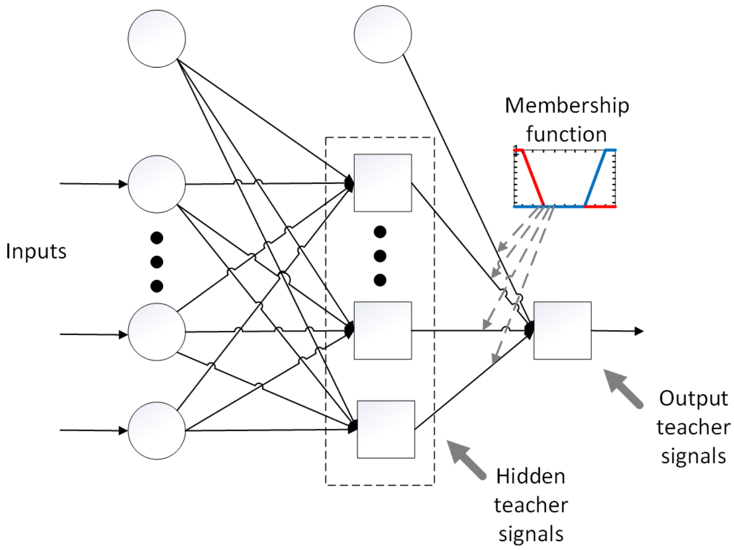

Figure 2. NN architecture with hidden-node teaching

3. EXPERIMENTAL SETUP AND RESULT

The flood detection system is comprised of several essential hardware components such as a microcontroller, sensors, solar panel system, and dry battery electrical energy as shown in Figure 3. At its core, the system relies on an Arduino Nano, which serves as the central processing unit, responsible for data collection, analysis, and decision-making [23]. Additionally, an ESP8266 module facilitates wireless connectivity, enabling the device to transmit data and alerts over the internet, thus allowing for remote monitoring and response. To monitor the river conditions, the system is equipped with a water flow sensor, which measures the rate of water flow, providing vital data for flood detection and monitoring purposes. An ultrasonic sensor is also employed, utilizing sound waves to measure the distance between the device and the water's surface, aiding in assessing flood levels accurately.

Furthermore, for immediate alerts and notifications, the system features a buzzer that functions as an audible alert mechanism, producing sound signals in response to flood-related events or warnings. An LED indicator complements this, providing visual feedback regarding the system's status and potential flood alerts. These integrated hardware components work in tandem to create a robust flood detection system, capable of real-time monitoring and timely alerts in the event of flood threats [24]. The Figure 3 shows the device testing process that is carried out in a river area in Surabaya, Indonesia. In this testing environment, the flood detection system is being evaluated under real-world conditions, allowing us to assess its performance and reliability in an area susceptible to flooding. This critical testing phase ensures that the system can effectively serve its purpose in providing early flood warnings and helping to mitigate the impact of flooding on the local community and infrastructure in Surabaya [25].

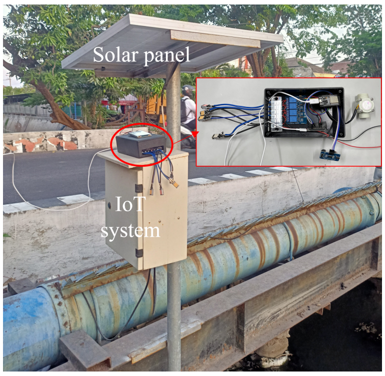

Figure 3. Device testing

The Figure 4 represents the outcomes observed on social media following the activation of the device, which automatically generates posts when it detects significant increases in water levels that enter the hazardous category. This innovative system provides real-time updates to the online community, offering crucial information about potential flood threats and the need for precautionary measures [26]. By leveraging social media as a communication channel [27], it enhances public awareness and safety, enabling individuals and authorities to respond promptly to evolving flood situations. This proactive approach to information dissemination not only aids in minimizing risks but also fosters a sense of community resilience in the face of environmental challenges.

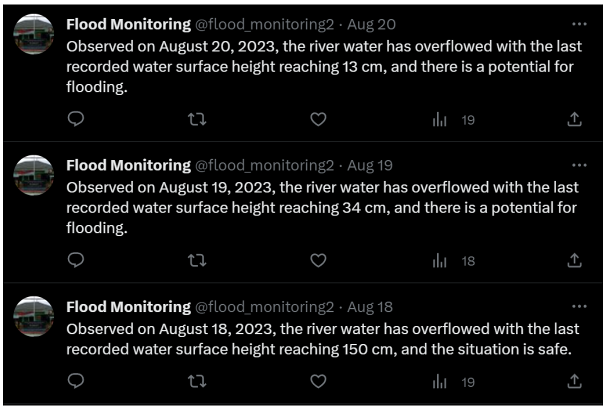

Figure 4. Monitoring results on social media

The Figure 5 represents the water level graph obtained from our testing. It is evident from the graph that there is a discernible upward trend in the water level as time unfolds. This trend indicates an increase in the water level over the course of the observed period. Such graphical representations are instrumental in tracking and understanding changes in water levels, particularly during events such as floods. They enable us to identify the timing, magnitude, and duration of rising water levels, which, in turn, is crucial for flood forecasting, disaster management, and decision-making. This data provides valuable insights for authorities, emergency responders, and community members to prepare and respond effectively to potential flood events, ultimately contributing to enhanced flood resilience and safety measures [28].

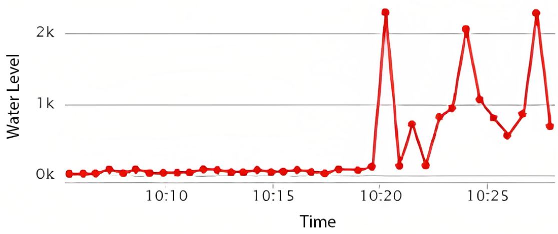

Figure 5. Water level graph

The Figure 6 depicts the water flow graph obtained through our testing procedures. It is distinctly evident from the graph that there is a noticeable upward trajectory in water levels as time advances. This pattern signifies a consistent increase in water flow throughout the observation period. Such graphical representations play a crucial role in monitoring and comprehending fluctuations in water flow, particularly during events like floods or rapid changes in river conditions. They empower us to ascertain the timing, magnitude, and duration of rising water levels, thus facilitating flood forecasting, effective disaster management, and informed decision-making. This data provides invaluable insights for authorities, emergency response teams, and the community, aiding in proactive measures and responses to potential flood events, ultimately enhancing flood resilience and safety measures [29].

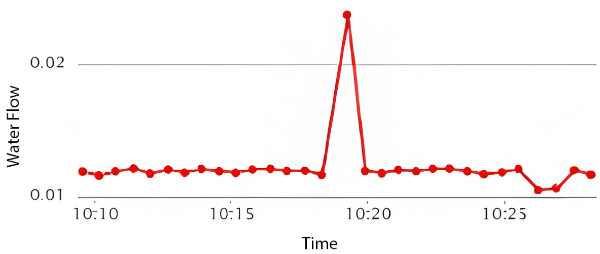

Figure 6. Water flow graph

3.1. Performance measurement

In the system evaluation subsection, we will delve into the sensor data obtained through the testing process. This section will provide an in-depth analysis of the data collected by our sensors, highlighting key metrics, trends, and patterns. By examining this information, we aim to assess the performance and reliability of our system in accurately detecting and responding to changes in environmental conditions. The sensor data will serve as a valuable resource for improving our system's effectiveness and ensuring its readiness for real-world scenarios. Additionally, it will contribute to enhancing our understanding of the dynamic factors that influence the system's functionality. During the testing process, the data that has been collected where comprises several variables such as sensor distance, water flow rate, and LED lights, each representing different flood categories. This dataset serves as a valuable resource for our flood detection system evaluation, allowing us to analyze and correlate the sensor readings with specific flood conditions [30]. By examining these variables, we aim to gain insights into the system's ability to accurately categorize and respond to varying flood levels, ultimately enhancing its reliability and effectiveness in providing early warnings and flood management.

As shown in Figure 7, this observation affirms that the proposed neuro-fuzzy algorithm, employing just two linear rules, is capable of flawlessly solving the XOR problem. Figures 7(a)-(d) shows that the 7-4-1 and 7-7-1 MLP-structured NNs possess significant nonlinear mapping capabilities, even with just two linear hidden nodes. This capacity arises from the data-dependent terminal weights and the firing strengths associated with these two linear rules. In contrast, within the framework of standard MLP learning, terminal weights remain consistent across all data points. Consequently, an MLP featuring ”linear” hidden nodes would be incapable of solving the XOR problem. This underscores a fundamental distinction between an MLP-structured NN and a conventional MLP. In detail, Figure 7(a) shows that the RMSE in batch-mode training without hidden-node teaching is quite small decreased for 7-4-1 MLP architecture, comparing to with hidden-node teaching as shown in Figure 7(b). While 7-7-1 NN with hidden-node teaching in Figure 7(d) shows a significantly decreased of RMSE, instead of 7-7-1 NN without hidden-node teaching Figure 7(c), with a less number of epochs.

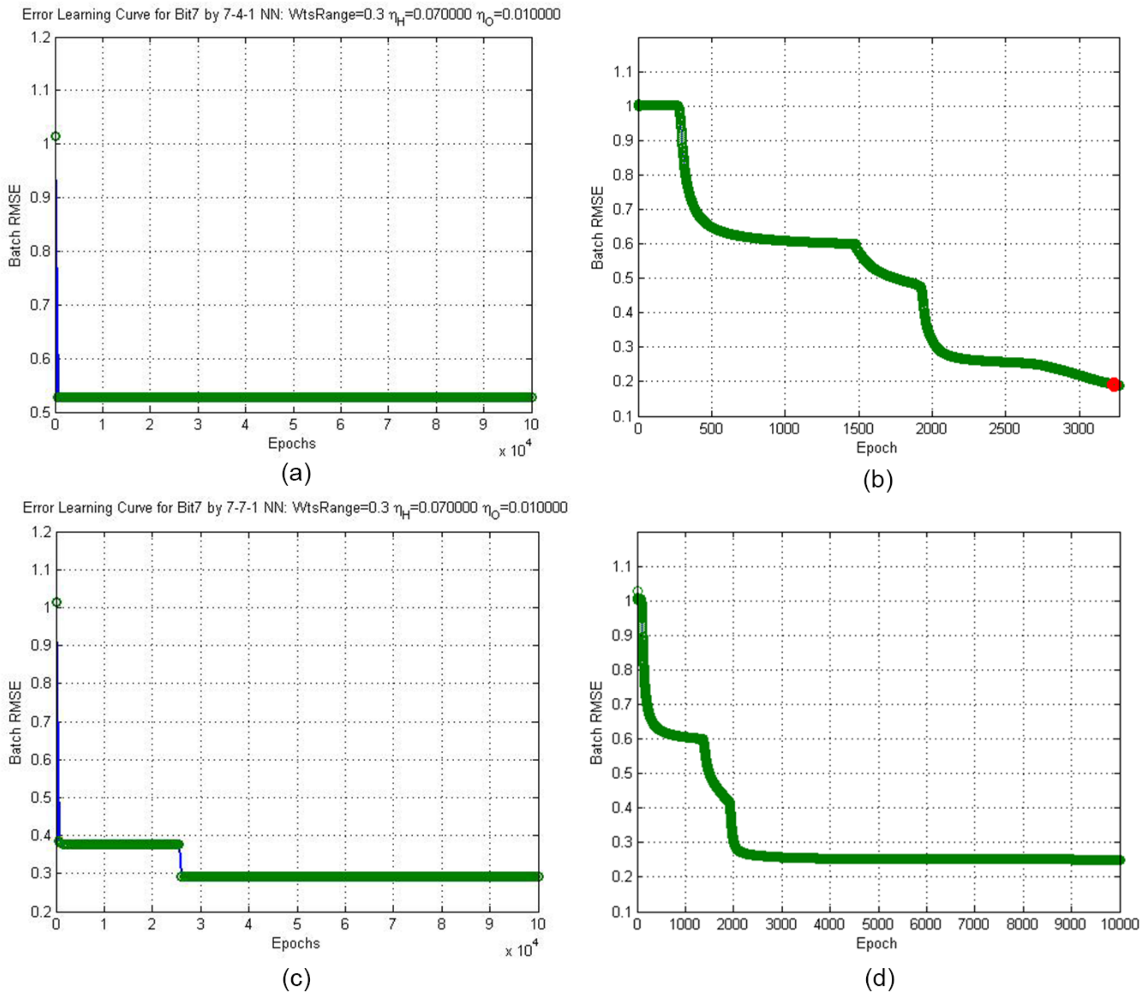

Figure 7. Algorithm evaluation using MATLAB: (a) 7-4-1 NN without hidden-node teaching, (b) 7-4-1 NN with hidden-node teaching, (c) 7-7-1 NN without hidden-node teaching, and (d) 7-7-1 NN with hidden-node teaching

4. CONCLUSION

The proposed IoT-based flood disaster early detection system, seamlessly integrated with social media, has been successfully implemented in the Rungkut area. This technological solution serves as a valuable tool in providing real-time information about potential flooding events. Through the integration with social media platforms, all residents in the Rungkut area can access the latest updates on flood conditions, fostering a safer and more comfortable environment. Furthermore, in in-depth analysis of the system’s performance reveals its effectiveness in accurately detecting and responding to changes in environmental conditions. The data collected through sensor readings, including variables such as sensor distance, water flow rate, and LED lights, has provided valuable insights. This information serves as a foundation for ongoing improvements, ensuring the system's readiness for real-world scenarios.

ACKNOWLEDGEMENT

This project was supported by the research department of Telkom university Surabaya and the Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia.

REFERENCES

[1] P. Wijayanti, X. Zhu, P. Hellegers, Y. Budiyono, and E. C. van Ierland, "Estimation of river flood damages in jakarta, indonesia," *Natural Hazards*, vol. 86, no. 3, p. 1059–1079, 2016, doi: 10.1007/s11069-016-2730-1.

[2] S. Sufri, F. Dwirahmadi, D. Phung, and S. Rutherford, "Enhancing community engagement in disaster early warning system in aceh, indonesia: opportunities and challenges," *Natural Hazards*, vol. 103, no. 3, pp. 2691–2709, Jun 2020, doi: 10.1007/s11069-020-04098-2.

[3] S. Fuchs *et al.*, "Flood risk perception and adaptation capacity: A contribution to the socio-hydrology debate," *Hydrology and Earth System Sciences*, vol. 21, no. 6, pp. 3183–3198, Jun. 2017, doi:10.5194/hess-21-3183-2017.

[4] M. Moishin, R. C. Deo, R. Prasad, N. Raj, and S. Abdulla, "Designing deep-based learning flood forecast model with ConvLSTM hybrid algorithm," *IEEE Access*, vol. 9, pp. 50982–50993, 2021. doi:10.1109/access.2021.3065939.

[5] V. Krzhizhanovskaya *et al.*, "Flood early warning system: design, implementation and computational modules," *International Conference on Computational Science*, vol. 4, pp. 106-115, 2011, doi: 10.1016/j.procs.2011.04.012.

[6] S. Parjuangan, R. Ali, and A. Purnama, "Real-Time Monitoring and Warning System in Urban Rivers," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 17, no. 3, pp. 1521–1525, Jun. 2019, doi: 10.12928/telkomnika.v17i3.10397.

[7] A. Wicaksono and H. Herdiansyah, "The internet of things (iot) for flood disaster early warning in jakarta: prospect and community preparedness," *IOP Publishing Ltd*, vol. 683, no. 1, Mar 2021, doi: 10.1088/1755-1315/683/1/012103.

[8] M. Khalaf *et al.*, "Iot-enabled flood severity prediction via ensemble machine learning models," *IEEE Access*, vol. 8, pp. 70375-70386, 2020, doi: 10.1109/ACCESS.2020.2986090.

[9] M. Khairudin, S. Riyanto, and Z. Mohammed, "Development of fuzzy logic control for indoor lighting using leds group," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 16, no. 3, pp. 1165-1173, Jun. 2018, doi: 10.12928/telkomnika.v16i3.8048.

[10] O. Bodhe, S. Ekawankar, R. Gawande, and A. Jadhao, "Iot based flood detection and alert system," *Proceedings of the International Conference on Smart Data Intelligence (ICSMDI 2021)*, May 2022.

[11] R.-S. Wu *et al.*, "Real-time flood warning system application," *Water*, vol. 14, no. 12, p. 1866, Jun. 2022, doi: 10.3390/w14121866.

[12] R. Byali, B. Divya, and S. V. Maskikar, "Iot based Early Flood Detection using Machine Learning," *International Journal of Research Publication and Reviews* , vol. 3, no. 7, pp. 3780–3783, Jul. 2022, doi: 10.55248/gengpi.2022.3.7.30.

[13] A. A. Soebroto, I. Cholissodin, M. T. Frestantiya, and Z. El Arief," Integration method of local-global svr and parallel time variant pso in water level forecasting for flood early warning system, *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 16, no. 3, pp. 1193-1200, Jun. 2018, doi: 10.12928/telkomnika.v16i3.6772.

[14] I. F. Ashari, "Analysis and implementation monitoring flood system based on iot using sugeno fuzzy logic," *Komputika Jurnal Sistem Komputer*, vol. 12, no. 1, pp. 25-34, May 2023, doi: 10.34010/komputika.v12i1.7089.

[15] H. Nasution, H. Jamaluddin, and J. M. Syeriff, "Energy analysis for air conditioning system using fuzzy logic controller," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 9, no. 1, pp. 139-150, Apr. 2011, doi: 10.12928/telkomnika.v9i1.680.

[16] B. Wang, F. Qin, X. Zhao, X. Ni, and X. Dongji, "Equalization of series connected lithium-ion batteries based on back propagation neural network and fuzzy logic control," *International Journal of Energy Research*, vol. 44, no. 6, pp. 4812-4826, Mar. 2020, doi: 10.1002/er.5274.

[17] R. Tabbussum and A. Q. Dar, "Performance evaluation of artificial intelligence paradigms—artificial neural networks, fuzzy logic, and adaptive neuro-fuzzy inference system for flood prediction," *Springer Science and Business Media LLC*, vol. 28, no. 20, pp. 25265–25282, Jan 2021, doi: 10.1007/s11356-021-12410-1.

[18] A. Bounemeur and M. Chemachema, "Fuzzy Fault-Tolerant Control Applied on Two Inverted Pendulums with nonaffine nonlinear actuator failures," *International Journal of Robotics and Control Systems*, vol. 3, no. 2, p. 144, Mar 2023, doi: 10.31763/ijrcs.v3i2.917.

[19] T.-J. Su, T.-S. Pan, Y.-L. Chang, S.-S. Lin, and M.-J. Hao, "A hybrid fuzzy and k-nearest neighbor approach for debris flow disaster prevention," in *IEEE Access*, vol. 10, pp. 21787-21797, 2022, doi: 10.1109/ACCESS.2022.3152906.

[20] A. Alhloul, "Early crisis warning system using community engagement through social network sites in times of disaster," *Scientia et Securitas*, vol. 3, no. 4, pp. 358–371, Jun. 2023, doi: 10.1556/112.2022.00127.

[21] M. H. H. Ichsan, W. Kurniawan, and M. Hud, "Water quality monitoring with fuzzy logic control based on graphical programming," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 14, no. 4, pp. 1446-1453, Dec. 2016, doi: 10.12928/telkomnika.v14i4.3505.

[22] T. Wang, R. Gault, and D. Greer, "A novel data-driven fuzzy aggregation method for takagi-sugeno-kang fuzzy neural network system using ensemble learning," *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Jul. 2021, doi:10.1109/fuzz45933.2021.9494396.

[23] C. Prakash, A. Barthwal, and D. Acharya, "Floodwall: A real-time flash flood monitoring and forecasting system using IoT," *IEEE Sensors Journal*, vol. 23, no. 1, pp. 787-799, Jan. 2023, doi: 10.1109/JSEN.2022.3223671.

[24] I. Ahmed, M. Ahmad, G. Jeon, and A. Chehri, "An internet of things and ai-powered framework for long-term flood risk evaluation,"

in *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 3812-3819, Feb. 2024, doi: 10.1109/JIOT.2023.3308564.
[25] M. Braveen, K. Anusha, M. A. Jerlin, R. Seetha, and N. Sampath, "IDISense: Iot-based dam water disaster sensing and prevention system," *IEEE Sensors Journal*, vol. 23, no. 23, pp. 29451-29457, Dec. 2023, doi: 10.1109/JSEN.2023.3322290.
[26] M. Anbarasan *et al.*, "Detection of flood disaster system based on iot, big data and convolutional deep neural network," *Computer Communications*, vol. 150, pp. 150-157, Jan. 2020, doi: 10.1016/j.comcom.2019.11.022.
[27] E. Ginting, N. L. Husni, T. Dewi, A. S. Handayani, and M. R. Harahap, "Application of fuzzy logic and social media on flood hazards early warning system," *VOLT: Jurnal Ilmiah Pendidikan Teknik Elektro*, vol. 3, no. 1, pp. 42–48, 2018, doi: 10.30870/volt.v3i1.3294.
[28] B. Arshad, R. Ogie, J. Barthelemy, B. Pradhan, N. Verstaevel, and P. Perez, "Computer vision and iot-based sensors in flood monitoring and mapping: A systematic review," *Sensors*, vol. 19, no. 22, 2019, doi: 10.3390/s19225012.
[29] B.-J. Jang and I. Jung, "Development of high-precision urban flood-monitoring technology for sustainable smart cities" *Sensors*, vol. 23, no. 22, Nov. 2023, doi: 10.3390/s23229167.
[30] A. R. Ildoromi, M. Vojtek, J. Vojtekov´a, Q. Pham, A. Kuriqi, and M. Sepehri, "Flood prioritization integrating picture fuzzy-analytic hierarchy and fuzzy-linear assignment model," *Arabian Journal of Geosciences*, vol. 15, p. 1185, 2022, doi: /10.1007/s12517-022-10404-y.

BIOGRAPHIES OF AUTHORS

Muhammad Adib Kamali earned a Bachelor of Engineering (B.Eng.) degree in Telecommunication Engineering from Telkom University, Bandung, Indonesia, in 2019. Following this, Adib pursued a Master of Engineering (M.Eng.) degree in Electronic Engineering at Kumoh National Institute of Technology, Gyeongsangbuk-do, Republic of Korea, and successfully completed his studies in 2022. His current research focus centers on the integration of artificial intelligence within embedded systems. He can be contacted at email: adibmkamali@telkomuniversity.ac.id.

Mochamad Nizar Palefi Ma'ady received the M.IM. degree in Information Management from National Taiwan University of Science and Technology, Taiwan. He is a lecturer in the department of Information Systems at Telkom Surabaya Institute of Technology. His research interests include neural network learning, visual analytics, dynamic programming, Markov decision processes, and text mining. He can be contacted at email: mnizarpm@telkomuniversity.ac.id.