Temporal convolutional network algorithm for streamflow predictions in a subtropical river

Y.L.Liu [a], **C.Deng** [b] **and H.Zhang** [b]

[a] School of Information Technology and Electrical Engineering, The University of Queensland, Brisbane, Queensland, [b] School of Engineering and Built Environment, Griffith University, Gold Coast, Queensland
Email: yilun.liu@uqconnect.edu.au

Abstract: Rainfall-runoff models have a high degree of uncertainty and stochasticity, and the relationship between them is non-linear. Conventional hydrology streamflow prediction models are mostly built for specific watersheds and specific prediction scales, which are poorly promoted and applied. Therefore, in some scenarios, data-driven machine learning predictive models are replacing traditional physical models. Long short-term memory (LSTM) network is a machine learning algorithm for predicting time series and has been applied in the field of streamflow prediction. Temporal convolutional network (TCN) is another machine learning algorithm that is gaining popularity in the field of time series forecasting. LSTM and TCN were implemented in this study to analyse the hourly streamflow prediction for the Nerang River at the Numinbah gauging site, and the predictive accuracy of the models on the test dataset was calculated based on the historical data of the study area. According to the results of the analysis, the TCN model achieved better performance for the hourly streamflow prediction with a coefficient of determination (R^2) of 0.9837 and Nash–Sutcliffe efficiency (NSE) of 0.9829 in the best scenario and the lag time for hourly streamflow generation is about three hours in the study area. Additionally, the maximum predicted lead time is six hours in the study area on the TCN model.

In summary, the accuracy of hourly streamflow prediction using the TCN algorithm is of a good level. Moreover, streamflow can be predicted for up to six hours. Finally, this study demonstrates that the novel TCN algorithm has enormous potential for solving streamflow prediction problems in comparison to the LSTM algorithm.

Keywords: *Hourly streamflow prediction, temporal convolutional network, long short-term memory network, machine learning, data-driven model*

1. INTRODUCTION

The Nerang River is the largest and most significant river system on the Gold Coast in the SEQ region, Australia. It provides multiple essential services to the public, and it has significant economic potential and the ability for flood mitigation. Hinze Dam is located in the middle of the Nerang River and is the primary source of water supply for the Gold Coast region, supplying an average volume of 169 ML water per day (Gold Coast City Council, 2005). The Gold Coast is a world-renowned tourist destination and also being one of the fastest-growing urban regions in Australia, placing heavy demands on water supplies. Historically, six flooding events have occurred since 1920 in the Nerang River (Bureau of Meteorology, 2019). However, there is a lack of research into the possible flash floods caused by short-duration rainfall in this area. Our study area is the Numinbah catchment (Figure 1), which is a sub-catchment of the Nerang River catchment with a covering area of around 70 km^2.

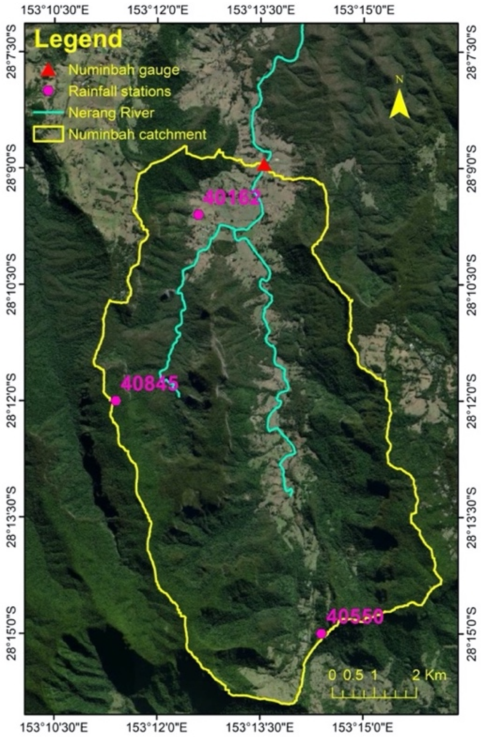


Figure 1. The headwaters catchment of the Nerang River.

This project is hospitable for both society and the economy. The significance of predicting short-term streamflow is to help water managers to make optimal decisions issues, such as drought mitigation, flood control, economic planning and keeping minimum streamflow (Maity et al., 2011; Sun et al., 2019). Additionally, the prediction model can help the Hinze Dam manage input discharge and flood migration.

The conventional hydrology rainfall-runoff model process is complex and wastes many human resources. For these reasons, data-driven models are being widely used for predicting streamflow, which is known as streamflow hydrology estimate using machine learning (SHEM) (Petty et al., 2018). When the historical data is adequate, using machine learning is much more streamlined than hydrological models. In this study, several machine learning algorithms were implemented to develop streamflow prediction models and then, which will be used to predict the hourly streamflow of the Nerang River. The streamflow series is a stochastic phenomenon, and it is affected by multi-variance, such as rainfall intensity, temperature, solid condition, and evapotranspiration. Artificial intelligence (AI) is a good option to solve this none-liner multivariate problem. Machine learning can solve non-linear multivariate problems, and deep learning models of machine learning, such as long short-memory (LSTM) neural networks, are widely used for time series prediction in various industries. Most recently, a temporal convolutional network (TCN) has been used in different fields for time series prediction. Many studies suggest that TCN perform better in time series prediction compared with LSTM. However, TCN has not yet been used in streamflow prediction tasks; therefore, this study will fill this gap.

2. METHODOLOGY

2.1. Model Concept and Implementation

Recurrent neural networks (RNNs) are more suitable for time series prediction than classical neural networks (Mhammedi et al., 2016). Input vectors of RNN are updated recurrently through the same operation cell, which means that the data retains previous information. However, as the gap between information grows, the conventional RNN model cannot connect them perfectly (Thapa et al., 2020). RNN is only able to inspect the relationship between inputs and outputs in a short input series. However, when the input series is longer, the drawback of RNN is appeared because of its simple recurrent network structure (Hu et al., 2020). Therefore, to avoid long-term dependencies, the LSTM neural network was designed. The other main advantage of LSTM compared to RNN is to avoid the exploding and vanishing gradient problems.

The entire LSTM cell includes input i_t, output o_t, and forget f_t gates. The four learnable gates encode the input groups of vectors and ensure they maintain the hidden information once they cross those cells. The operation process can be defined as:

$$i_t = \sigma(W_t \cdot [h_{t-1}, x_t] + b_t), \tag{1}$$

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right), \tag{2}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o), \tag{3}$$

$$\tilde{C} = tanh(W_C \cdot [h_{t-1}, x_t] + b_C), \tag{4}$$

$$C_t = f_t \odot C_{t-1} + i_t * \widetilde{C_t}, \tag{5}$$

$$h_t = o_t \odot \tanh(C_t). \tag{6}$$

Equations (1) – (6) are the fundamentals of the LSTM model operation. The forget layer converts the h_{t-1} and x_t into 0 to 1 utilising the sigmoid function, where 0 is data that should be omitted and 1 is data that should be reserved. The combination of h_{t-1} and x_t is updated to i_t and $\tilde{C}$ by passing across the sigmoid and tanh layers, and the new candidate values are achieved by $i_t * \tilde{C}$. Then, the new cell state C_t is generated by equation (6). Finally, the next hidden state h_t is calculated by the equation (3) and (6). The sigmoid layers ensure the output only retains the parts that the model decides to keep because the data will be deleted if the result is 0 after passing the sigmoid layer. These steps guarantee that LSTM can avoid long-term dependency.

The TCN network is based on a causal convolution network, a one-to-one structure with a strict causal relationship between each layer. The traditional convolutional operation is to convolute the sequence and encode the information as a new cell, which is named as pooling process. However, one of the main faults of the traditional convolutional networks is that the sequential information could be lost in the pooling process (He et al., 2016). The dilated convolutions are defined as:

$$\mathrm{F}(s) = (x * df)(s) = \sum_{\mathrm{i}=0}^{k-1} \mathrm{f(i)} \cdot \mathrm{x}_{\mathrm{s-d \cdot i}}, \tag{7}$$

where df is the dilation factor, k is the filter size and $s - d \cdot i$ accounts for the direction in the past. Compared with LSTM, the TCN network can control the memory size by adjusting the dilation factor and the kernel size. The receptive field can be defined as:

$$n_{out} = \left\lfloor \frac{n_{in} + 2p - k}{s} \right\rfloor + 1, \tag{8}$$

where n_{out} is the size of the output and n_{in} is the size of the input. For $n_{out} = n_{in}$, add padding p. k is the kernel size, and s is the stride step, which equals the dilation factor in each hidden layer.

The advantage of the dilated convolution network is that it abandons the pooling process and uses a series of dilated convolutions, which helps each output consist of adequate information for long-term tracking (Bai et al., 2018).

These models were implemented using TensorFlow in the Python platform. The basic parameters of the models are shown in Table 1.

Table 1. Details of hourly and daily streamflow prediction.

Model Type	LSTM	TCN
Regularisation	Robust	Min-Max
Loss function	mean squared error (MSE)	mean absolute error (MAE)
Learning rate	0.0001	0.0001
Epochs	500	500
Neurons	32	64
Batch size	32	32
Dilation factor (TCN)	--	[1, 2, 4, 8, 16, 32]

Hourly rainfall and streamflow data are observed at Numinbah gauging site during the period from 2015 to 2018, which was split into the training dataset (2015, 2016 and 2017) and the testing dataset (2018). Figure 2 shows the dataset which was obtained from the water monitoring information portal (WMIP).

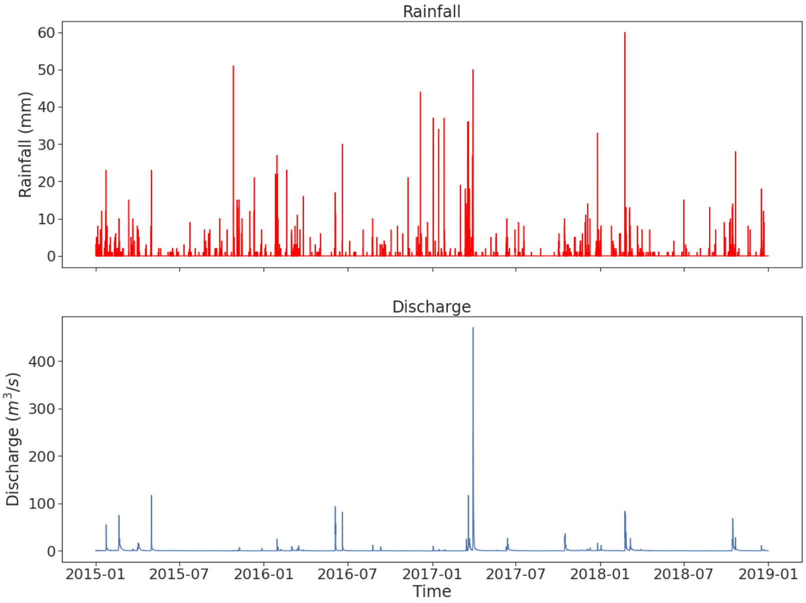


Figure 2. Hourly rainfall and streamflow at the Numinbah gauge (time range is from midnight on 1 January 2015 to midnight on 1 January 2019).

For the one-step perdition task, input parameters only consist of gauge rainfall (P) and discharge (D). The output is the streamflow discharge (D) in the target hour. Table 2 shows the input parameters of different models.

Table 2. Input parameters and output in each model.

Model	Input	Output
MH1	P_{t-1}, D_{t-1}	D_t
MH2	P_{t-1}, P_{t-2}, D_{t-1}, D_{t-2}	D_t
MH3	P_{t-1}, P_{t-2}, P_{t-3}, D_{t-1}, D_{t-2}, D_{t-3}	D_t
MH4	P_{t-1}, ..., P_{t-4}, D_{t-1}, ..., D_{t-4}	D_t
MH5	P_{t-1}, ..., P_{t-5}, D_{t-1}, ..., D_{t-5}	D_t
MH6	P_{t-1}, ..., P_{t-6}, D_{t-1}, ..., D_{t-6}	D_t

After finishing the one-step prediction, the optimal time-step of models was chosen to predict the multistep streamflow. In the multistep streamflow prediction, the lead time was extended to test the ability of the models to predict the future streamflow.

2.2. Evaluation Metrics

To evaluate the performance of the prediction results, there are two metrics: coefficient of determination (R^2) and Nash-Sutcliffe Efficiency index (NSE).

Table 3 is used to evaluate the rating of the predicted results.

Table 3. Performance rating for streamflow prediction used in this study (Van Liew, 2018).

Performance rating	NSE
Very good	0.75<NSE<1.0
Good	0.65<NSE<0.75
Satisfactory	0.50<NSE<0.65
Unsatisfactory	NSE<0.50

3. RESULTS AND DISCUSSIONS

3.1. Hourly Streamflow Prediction Results

Table 4 shows that the TCN models can achieve the best prediction level as the input time steps are three hours. Fewer time-steps not only improve the model training efficiency but also reduce the predicted input dataset costs. For instance, to predict the next hourly streamflow, the previous three-hour rainfall must be known for the TCN model to get the best results.

Table 4. The results of each model.

Model	NSE		R^2	
	LSTM	TCN	LSTM	TCN
MH1	0.8159	0.9724	0.9465	0.9730
MH2	0.9068	0.9738	0.9546	0.9776
MH3	0.9442	0.9829	0.9477	0.9837
MH4	0.9618	0.9829	0.9624	0.9829
MH5	0.9650	0.9803	0.9728	0.9818
MH6	0.9619	0.9820	0.9626	0.9822

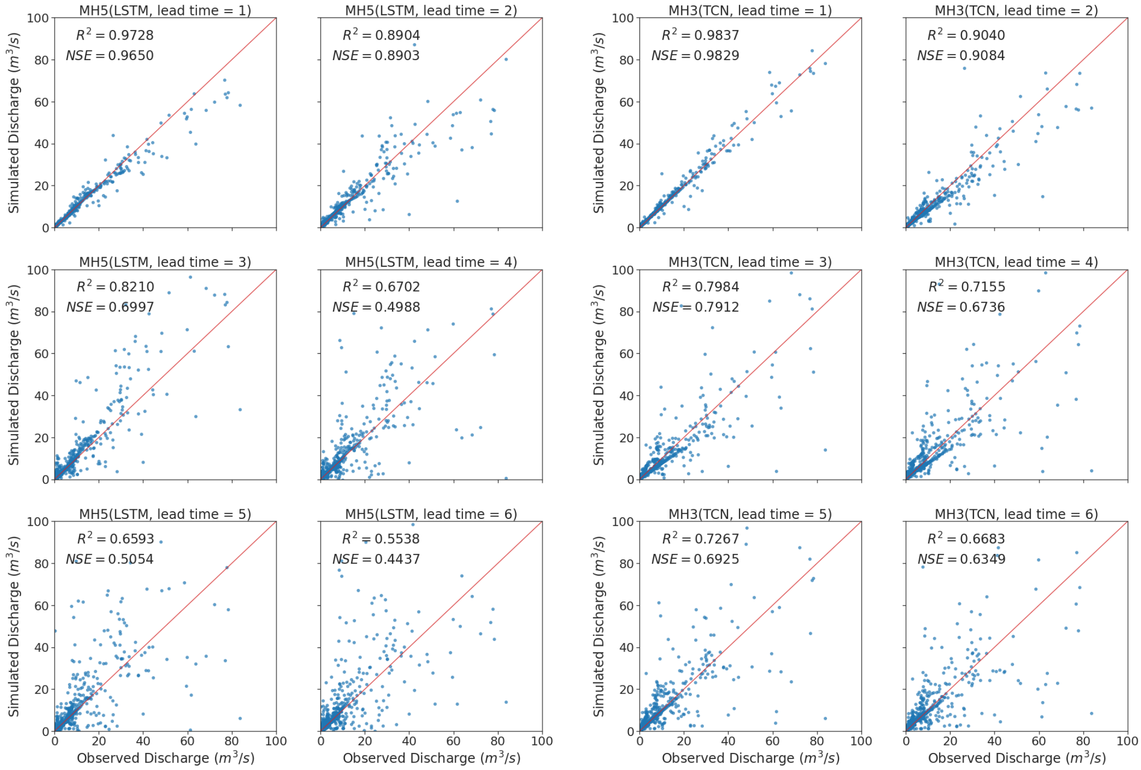
Reference to Table 4, the TCN model has a smaller time-step compared with the LSTM model. TCN can achieve the best level as the time steps are three hours, but the best time steps are five hours for the LSTM network. Fewer time-steps not only improve the model training efficiency but also reduce the predicted input dataset costs. For instance, to predict the next hourly streamflow, the previous three-hour rainfall must be known for the TCN model to get the best results, but the LSTM needs the previous five-hour rainfall data. Furthermore, the predictive accuracy of the LSTM and TCN models for multistep predictions is shown in Table 5.

Table 5. The hourly multistep streamflow predictions of the LSTM (MH5) and TCN (MH3) model.

Lead time (Hour)	NSE		R^2	
	LSTM	T*CN*	LSTM	TCN
1	0.9650	0.9829	0.9728	0.9837
2	0.8903	0.9084	0.8904	0.9040
3	0.6997	0.7912	0.8210	0.7984
4	0.4988	0.6736	0.6702	0.7155
5	0.5054	0.6925	0.6593	0.7267
6	0.4437	0.6349	0.5538	0.6683

The longest lead time for the LSTM hourly streamflow prediction model is three hours within a good level, but the TCN model can obtain a good level of prediction when the lead time extends to five hours. When the lead time exceeds a certain time, the accuracy is decreased dramatically. For instance, the NSE of the LSTM (MH5) model decreases a lot from a three-hour lead time to a four-hour lead time. That is, the LSTM can identify the relationship between the future fourth streamflow and the previous five-hour rainfall information. Figure 3 shows that the predictive accuracy changes as the lead time increases in the LSTM (MH5) model TCN (MH3) model, respectively.

As the lead time increases, the errors between predicted and observed streamflow data also increases. Despite the predictive accuracy still being within a good level, when the lead time extends to five hours, the scatter plots reveal high errors. The reason is that some high streamflow events are caused by sudden rainfall, whose streamflow generation lag time is less than the lead time. Therefore, these uncertainties lead to relatively big errors.



(a) (b)

Figure 3. (a) Multistep prediction analysis of the LSTM model. (b) Multistep prediction analysis of the TCN model.

As the lead time increases, the errors between predicted and observed streamflow data also increases. Despite the predictive accuracy still being within a good level, when the lead time extends to five hours, the scatter plots reveal high errors. The reason is that some high streamflow events are caused by sudden rainfall, whose streamflow generation lag time is less than the lead time. Therefore, these uncertainties lead to relatively big errors.

Compared with the multistep prediction of the TCN model, the LSTM has worse predictive accuracy. LSTM is a sequential model, and the calculation of LSTM is also a forward process. This mechanism leads to incomplete historical information. On the contrary, the TCN model is a parallel calculation model, and it can avoid historical information loss fundamentally. Additionally, the TCN algorithm is based on the dilated convolutional network, which is allowed to adjust memory sizes by changing the dilation factor. Because of the unique architecture, TCN has a better performance in multistep prediction problems.

Overall, using the TCN model to predict hourly streamflow can improve efficiency and decrease the cost. Despite the LSTM neuronal networks avoiding the effects of a long dependency on conventional RNN networks, the memory of LSTM is not eternal. The advantages of convolutional networks result in the TCN needing fewer time-steps to predict the streamflow and having better predictive accuracy in multistep prediction tasks.

4. CONCLUSION

This study focused on hourly streamflow predictions on the Nerang River at the Numinbah gauge site using LSTM and TCN algorithms. The LSTM architecture is based on RNN, which transforms data by the same calculating cell recurrently, while the structure of the TCN model is based on the dilated convolutional network, which has parallel computing capabilities. Therefore, the TCN model operates more efficiently than the LSTM model. Additionally, the TCN model can be determined how many inputs can be related to outputs by adjusting the dilation factor, but the memory size for outputs of the LSTM model is uncontrollable. As the result, the TCN model outperformed the LSTM model in the hourly streamflow prediction task.

Experimentally, lag time and input parameters for the hourly streamflow prediction were determined in this study area. The input parameter of the prediction model only includes rainfall information because the hourly streamflow has a quicker response. The result suggested that the optimal input size is three hours, which means the lag time of hourly streamflow generation is about three hours in this study area. Furthermore, the lead time of the hourly model can be extended to six hours.

This case study demonstrates that the TCN model has great accuracy in hourly streamflow predictions, which can be used for water supply management, flood alerting, migration, and completing missing streamflow data for certain monitoring stations. This study was limited by the datasets, but there are still spaces for improvement in the further research. For instance, the datasets of the hourly model can add more hourly rainfall stations as inputs to test the predictive accuracy.

ACKNOWLEDGEMENTS

We would like to thank WMIP for supporting the historical rainfall and streamflow data for this study and Google colab for providing the cloud computing resources.

REFERENCES

Bai, S., Kolter, J. Z., & Koltun, V. (2018). An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. http://arxiv.org/abs/1803.01271

Bureau of Meteorology. (2019). *Flood warning system for the Nerang River*. Bureau of Meteorology. http://www.bom.gov.au/qld/flood/brochures/south_coast/nerang.shtml#PreviousFlooding

Gold Coast City Council. (2005). Hinze Dam Stage 3 Environment Impact Statement.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. http://image-net.org/challenges/LSVRC/2015/

Hu, Y., Yan, L., Hang, T., & Feng, J. (2020). Stream-Flow Forecasting of Small Rivers Based on LSTM. http://arxiv.org/abs/2001.05681

Maity, R., & Kashid, S. S. (2011). Importance analysis of local and global climate inputs for basin-scale streamflow prediction. Water Resources Research, 47(11), 1–17. https://doi.org/10.1029/2010WR009742

Mhammedi, Z., Hellicar, A., Rahman, A., Kasfi, K., & Smethurst, P. (2016). Recurrent neural networks for one day ahead a of stream flow. ACM International Conference Proceeding Series, 25–31. https://doi.org/10.1145/3014340.3014345

Petty, T. R., & Dhingra, P. (2018). Streamflow Hydrology Estimate Using Machine Learning (SHEM). JAWRA Journal of the American Water Resources Association, 54(1), 55–68. https://doi.org/10.1111/1752-1688.12555

Sun, Y., Niu, J., & Sivakumar, B. (2019). A comparative study of models for short-term streamflow forecasting with emphasis on wavelet-based approach. Stochastic Environmental Research and Risk Assessment, 33(10), 1875–1891. https://doi.org/10.1007/s00477-019-01734-7

Thapa, S., Zhao, Z., Li, B., Lu, L., Fu, D., Shi, X., Tang, B., & Qi, H. (2020). Snowmelt-Driven Streamflow Prediction Using Machine Learning Techniques (LSTM, NARX, GPR, and SVR). Water, 12(6), 1734. https://doi.org/10.3390/w12061734

Van Liew, M. W., & Mittelstet, A. R. (2018). Comparison of three regionalization techniques for predicting streamflow in ungaged watersheds in nebraska, USA using SWAT model. International Journal of Agricultural and Biological Engineering, 11(3), 110–119. https://doi.org/10.25165/j.ijabe.20181103.3528