Deep Learning of Complex Pipe Leakages Events in Drinking Water Distribution Networks for Effective Spatiotemporal Pre-Detections and Isolations of Leak Conditions

Cheng Ann TAN[1]; Veradej PHIPATANASUPHORN[2]; Adrian Chun Hin LAI[3]
[1,2,3] DHI Water & Environment (Singapore) Pte Ltd, 2 Venture Dr, #18-18 Vision Exchange, Singapore 608526
[1]tca@dhigroup.com

EXTENDED ABSTRACT

Background & Problem Statement:

The drinking water distribution network in L-TOWN is built upon 42.6km of underground pipes which serves a total population size of 10,000 in both the residential and commercial domains. The entire network is generally divided into 3 key area: (a) Area A comprises of the residential and commercial facilities; (b) Area B comprises of a pressure reduction valve to reduce the background leakages; and (c) Area C comprises of a pump and water tank to supply enough pressure to the consumers in that area, and is also installed with 82 water demand reading sensors. 33 pressure sensors are also deployed across the most sensitive node locations within the network, as illustrated in Figure 1a, for maximizing the collective sensitivity of the sensors to detect pipes leakages over space and time. Figure 1b illustrates the respective distributions of the reported pipe leakages throughout the water distribution network in L-TOWN within the year of 2018.

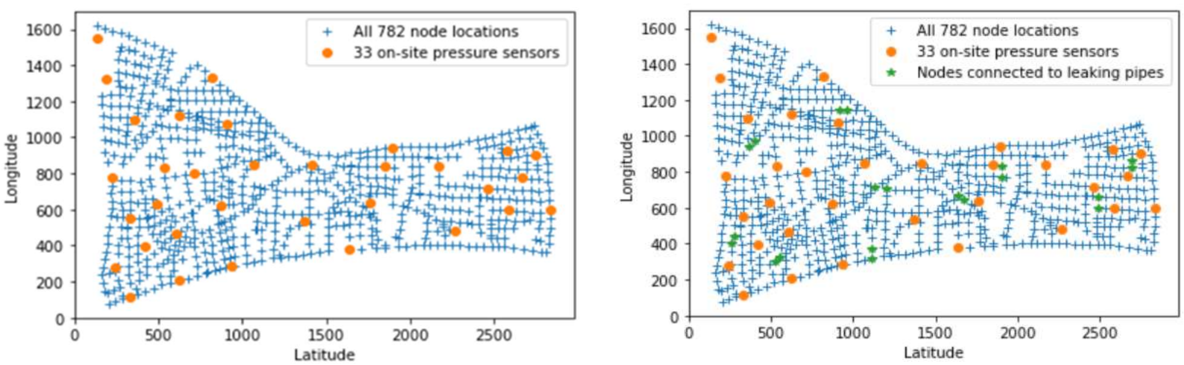

Figure 1: (1a, left) spatial distribution of 33 on-site pressure sensors/nodes in L-TOWN; (1b, right) spatial distribution of pipe leakages in L-TOWN for year of 2018

Till present, it remains challenging for operators in the local water utility company of L-TOWN to pre-detect spatial pipe leakages, of small to big in their sizes, as fast as possible over time. In short, not every leakage event, localized at its exact coordinates and/or proximity, can be pre-detected before and after its occurrence. In addition, there are also concerns in the overall accuracy of the model's numerical predictions due to physical uncertainties in the pipes' roughness, diameters, and seasonality coefficients for the commercial and residential demand patterns.

Objective of study:

To address the above-discussed problem statement, this research study develops an alternative engineering tool, by combining the numerical capabilities of genetic algorithm and deep learning, which can pre-detect near and/or exact locations of pipe leakages within the water distribution network in L-TOWN over time. The genetic algorithm is programmed using an open-source **Water Network Tool for Resilience** (WNTR) in Python package. WNTR is an EPANET compatible Python version and is designed to simulate and analyze resilience of water distribution networks. For the deep learning component, a personalized feed-forward deep neural network (DNN) is built on Tensorflow platform to develop a trained predictive model using volumes of calibrated simulation data derived from WNTR based on the physical characteristics of the water distribution network in L-TOWN. The trained DNN model is then leveraged to predict the near and/or exact locations of pipe leakages in L-TOWN using the real-world measured data from the reported years of 2018 and 2019.

Methodology:

Genetic Algorithm in WNTR

We first calibrate a numerical model in WNTR capable of simulating the 2018 measured pressure profiles from all 33 installed pressure sensors, as shown in Figure 1a earlier, to the best possible extent under no-leak conditions (i.e. ideal scenario) for the water distribution network in L-TOWN. To do so, a genetic algorithm (GA), as illustrated in Figure 2, is adhered in WNTR by calibrating 4 key parameters, namely: (a) pipe diameter coefficient, α_1; (b) pipe roughness, α_2; (c) residential seasonality coefficient, α_3; and (d) commercial seasonality coefficient, α_4. The objective function of the GA method is to minimize the average mean squared error (MSE) between the simulated and measured pressure data from all 33 pressure sensors across all timestamps (see Equation 1) which vary at every 5-minute interval.

$$MSE = \frac{\sum_{t=1}^{T}\left(\sum_{i=1}^{N}\left(P_{s,i} - P_{m,i}\right)^2\right)}{T \cdot N} - (1)$$

where t is the time counter which varies at every 5-minute interval, i the pressure node index, $P_{s,i}$ the simulated pressure value at pressure sensor i, and $P_{m,i}$ the measured pressure value at pressure sensor i.

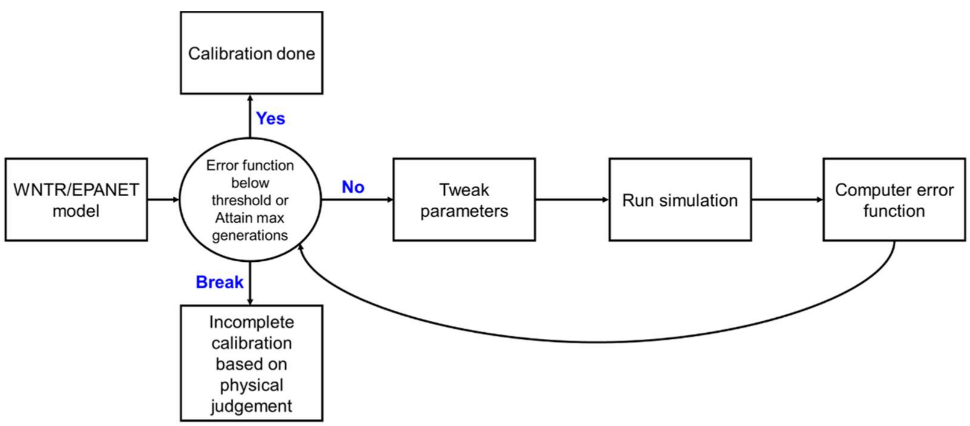

Figure 2: Framework of implemented genetic algorithm (GA) in WNTR to calibrate pipe diameter (α_1), pipe roughness (α_2), residential seasonality (α_3) and commercial seasonality (α_4) coefficients based on 2018 measured pressure data from the 33 installed pressure sensors in the water distribution network of L-TOWN

By conducting several iterations using the proposed GA method from Figure 2, the lowest possible average MSE value, as compared between the simulated and 2018 measured pressure values for all 33 pressure sensors, approximates to 0.138 (average RMSE of 0.371) based on the calibrated values of: (a) α_1 = 0.947; (b) α_2 = 1.03; (c) α_3 = 1.10; and (d) α_4 = 1.07. These calibrated coefficients are then leveraged with a built-in numerical model in WNTR to simulate random leaks (arbitrary 10% of the calibrated pipe diameter for leak size) at the available pipes of the water distribution network in L-TOWN using a random generator. For each leak simulation in the random pipe selected, the following time boundary conditions are adopted.

$$T_{L,start} = 0 \leq T_{L,start}\ (days) < 365 - (2a)$$
$$T_{L,max} = 365 - T_{L,start} - (2b)$$
$$T_{L,sim} = 0 \leq T_{L,sim}\ (days) \leq T_{L,max} - (2c)$$

where $T_{L,start}$ is the random starting time of the leak condition at the random pipe selected as measured in days, $T_{L,max}$ the maximum allowable time duration of the leak condition at the same pipe before its repair as measured in days, and $T_{L,sim}$ the random time duration of the leak condition at the same pipe before its repair as measured in days.

To simulate the leaks at random pipes across the network, the calibrated model first randomly selects the nodes available connected to defined pipe IDs. Upon targeting the random nodes selected, the leak condition is then simulated at the respective connected pipes using the time boundary conditions from Equations (2a – 2c) using a leak percent of 10% of the corresponding calibrated pipe diameter size. Each simulation run will be performed for the period of 365 days. For extensiveness, the following quantities of the anomaly nodes are simulated: (a) 50 to 100; (b) 100 to 200; (c) 200 to 300; (d) 300 to 400; and (e) 400 to 500. In each scenario, a total of 200 simulation runs has been performed which generally incurred an average computational time of 6 hours due to multiple iterations involved to handle the relatively vast number of anomalies occurring at the different node locations. 100 simulation sets are then randomly picked from each scenario to build a common pool of simulation data which is subsequently used for training and validating a 1D feedforward DNN model for classification analysis.

Deep learning model development

Figure 3 depicts the simplified design of the 1D feedforward DNN model adopted in this research study. The input layer to the DNN model comprises of 33 neurons of which each neuron represents the simulated pressure value from each of the 33 pressure sensors from each simulation run. The output layer (i.e. final hidden layer 6) of the same model is then built with 905 neurons to represent all available 905 pipes within the water distribution network of L-TOWN. Note that each output neuron is associated with a 1D vector representation of shape (1,2), hence representing the binary class of no leak or leak condition at the specific pipe. Softmax activation function is attached to all neurons of the output layer for computing the probabilities outputs from all neurons at the end of each epoch run. The configuration of the multiple hidden layers of the proposed DNN model is summarized in Table 1, together with the hyperparameters and cost-function used to perform the model training and validation steps. It is worth noting that only the simulation data derived from the calibrated numerical model in WNTR is used to train and validate the DNN model, followed by independent model testing with the available 2018 measured data provided.

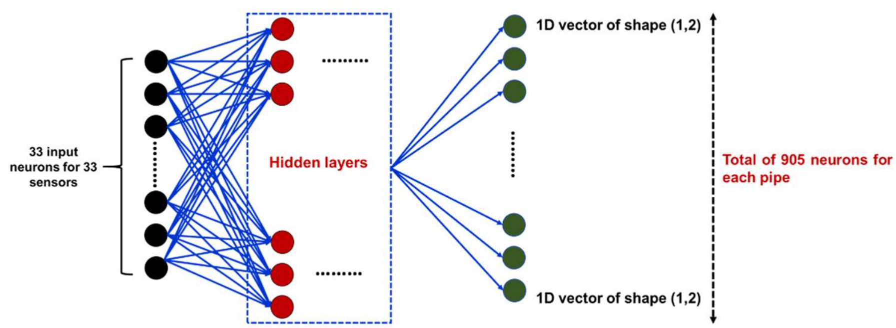

Figure 3: Simplified representation of proposed DNN model which undergoes model training and validation with simulation data derived from calibrated model in WNTR

Table 1: Quantitative description of the design configuration of DNN model adopted in this study

Component of DNN model	**Quantitative description**
Hidden layer 1	66 neurons + Rectified Linear Unit (ReLU) activation function
Hidden layer 2	132 neurons + Rectified Linear Unit (ReLU) activation function
Hidden layer 3	264 neurons + Rectified Linear Unit (ReLU) activation function
Hidden layer 4	528 neurons + Rectified Linear Unit (ReLU) activation function
Hidden layer 5	1810 neurons + Rectified Linear Unit (ReLU) activation function
Cost-function	Training step: Cross-entropy loss; Validation step: Accuracy Score
Other hyperparameters	Batch Size: 2190, 3284; Epochs: 5, 10; Learning Rate: 0.0001; Optimizer: Adam

Results:

Table 2 summarizes the final cross-entropy loss values and accuracy scores for the DNN model's training and validation steps for the multiple pools of simulation data by adhering to different sets of hyperparameters as shown. Again, we note that each simulation run is performed for the entire period of 365 days. The derived results show that the cross-entropy loss reaches a stagnating value of 0.693 at an early-stage due to the large volume of training data used and the final accuracy score gradually drops with an increasing number anomaly nodes simulated which suggests that effective pre-detections of the anomalies may become more difficult. Finally, independent testing of the trained DNN model is performed using the available 2018 data and the best results derived are summarized in Tables 3a and 3b. Selection of the best results is based on two key criteria, namely: (a) detecting the exact or proximity of the actual leaking pipe; and (b) detection time with enough lead-time before the actual repair time of the leaking pipe.

Table 2: Summary of final cross-entropy loss values and accuracy scores for the DNN model's training and validation steps using multiple pools of simulation data with different sets of hyperparameters

Combination	Batch size	Number of Epochs	Cross-entropy loss (training step)	Accuracy score (validation step)
1*	2190	5	0.693	0.922
1	3284	5	0.693	0.922
1	2190	10	0.693	0.922
1	3284	10	0.693	0.922
2*	2190	5	0.693	0.905
2	3284	5	0.693	0.905
2	2190	10	0.693	0.905
2	3284	10	0.693	0.905
3*	2190	5	0.693	0.890
3	3284	5	0.693	0.890
3	2190	10	0.693	0.890
3	3284	10	0.693	0.890

**** Combination 1: Combined pool of 50 to 100, 100 to 200 and 200 to 300 anomaly nodes; each group has 200 simulation runs***

***** Combination 2: Combined pool of 50 to 100, 100 to 200, 200 to 300 and 300 to 400 anomaly nodes; each group has 200 simulation runs***

****** Combination 3: Combined pool of 50 to 100, 100 to 200, 200 to 300, 300 to 400 and 400 to 500 anomaly nodes; each group has 200 simulation runs***

Table 3a: Prediction results from the simulation data pool of ***Combination 1*** *using the hyperparameters of Batch size of 3285 and Number of Epochs of 5*

Actual leaking pipe no.	Actual repair time	Closest predicted leaking pipe no.	Predicted leaking time
p461	2018-04-02 11:40	p447	2018-03-31 00:15:00
p232	2018-02-10 09:20	p461	2018-02-01 04:50:00
p673	2018-03-23 10:25	p139	2018-03-01 03:50:00
p628	2018-05-29 21:20	p647	2018-05-03 04:35:00
p538	2018-06-02 06:05	p708	2018-02-03 01:05:00
p866	2018-06-12 03:00	p867	2018-06-02 09:25:00
p31	2018-08-12 17:30	p866	2018-05-01 03:50:00
p183	2018-09-01 17:10	p461	2018-08-31 15:35:00
p158	2018-10-23 13:35	p226	2018-10-01 17:10:00
p369	2018-11-08 20:25	p866	2018-06-20 01:45:00

*Table 3b: Prediction results from the simulation data pool of **Combination 3** using the hyperparameters of Batch size of 3285 and Number of Epochs of 5*

Actual leaking pipe no.	Actual repair time	Closest predicted leaking pipe no.	Predicted leaking time
p461	2018-04-02 11:40	p448	2018-03-31 00:15:00
p232	2018-02-10 09:20	p461	2018-02-01 04:50:00
p673	2018-03-23 10:25	p139	2018-03-01 03:50:00
p628	2018-05-29 21:20	p647	2018-05-03 04:35:00
p538	2018-06-02 06:05	p708	2018-02-03 01:05:00
p866	2018-06-12 03:00	p867	2018-06-02 09:25:00
p31	2018-08-12 17:30	p866	2018-06-20 01:45:00
p183	2018-09-01 17:10	p357	2018-08-06 18:30:00
p158	2018-10-23 13:35	p226 p647	2018-10-01 17:10:00 2018-10-21 08:05:00
p369	2018-11-08 20:25	p866	2018-06-20 01:45:00

Keywords: *Genetic Algorithm; Deep Learning; Neural Networks; Leakages Pre-Detections; Pressure-Demand Drinking Water Distribution Networks*

SUMMARY

Effective spatiotemporal pre-detections of pipe leakages in urbanized water distribution networks continue to be a challenging engineering problem today. Pre-detection criteria includes: (a) the ability to localize the exact leaking pipes or surrounding pipes of proximity to the actual leaking pipes; and (b) enough lead-time for detection timing prior to the actual repair times of the respective leaking pipes. This research study combines the numerical capabilities of genetic algorithm and deep learning to develop the required predictive tool which can achieve the above-outlined pre-detection criteria. Specifically, the genetic algorithm is responsible for calibrating the key physical parameters associated with the known water distribution network in L-TOWN using the available measured data from 2018. The calibrated numerical model is then leveraged to generate a series of simulation data relating to random anomalies simulated at the different node locations associated to their respective pipe IDs, hence simulating the leakage events at the associated pipes. Different pools of simulation data are then used to train and validate a personalized 1D feedforward deep neural network (DNN) model which maps the calibrated numerical pressure values at 33 known pressure sensor locations to an output layer for determining the specific pipe ID having the highest probability of leakage at the given time-step. By varying the hyperparameters of batch-size and number of epoch runs, the trained DNN model can pre-detect the pipes close to the actual leaking pipes before their respective repair times from the available 2018 measured data as part of the model testing phase. Finally, the trained DNN model is then adhered to perform the predictions on the 2019 measured data for pre-determining the possible leaking pipes within the water distribution network of L-TOWN at specific timings.