



Source localization for hazardous material release in an outdoor chemical plant via a combination of LSTM-RNN and CFD simulation



Hyunseung Kim^a, Myeongnam Park^b, Chang Won Kim^b, Dongil Shin^{b,*}

^a School of Chemical and Biological Engineering, Seoul National University, Seoul 08826, Korea

^b Department of Chemical Engineering, Myongji University, Yongin, Gyeonggi-do 17058, Korea

ARTICLE INFO

Article history:

Received 19 January 2019

Revised 7 March 2019

Accepted 7 March 2019

Available online 8 March 2019

Keywords:

Source tracking

Hazardous chemical release

Process safety

Neural network

Artificial intelligence

inverse problem

ABSTRACT

Chemical leak accidents not properly handled at the early stage can spread to major industrial disasters escalating through fire and explosion. Therefore, it is very important to develop a method that enables prompt and systematic response by identifying the location of leakage source quickly and accurately and informing on-site personnel of the probable location(s). In this study, a model that predicts the suspicious leak location(s) in real-time, using sensor data, is proposed. Feed-forward neural network and recurrent neural network with long short-term memory that learned the data gathered from the installed sensors are proposed to predict the Top-5 points in the order of highest likelihood. In order to train and verify the neural networks, the sensor data generated from computational fluid dynamics simulations for a real chemical plant are used. The model learns the inverse problem solving for accident scenarios and predicts the leak point with very high accuracy.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

In the leak accidents of hazardous chemical, the initial response to leakage is very important. This makes it possible to prevent the spread of the second and third accident such as fire and explosion by identifying, locating, and blocking the leak source at the early stage of accidents. In other words, systematic early response to leak accidents is a key factor in mitigating the spread of the damage.

We propose an integrated emergency response system for leak accidents that predicts the location of the leak source(s) quickly and accurately using the data gathered from the sensor network. Fig. 1 shows the scheme of the integrated emergency response system. When a chemical leakage occurs at a chemical plant, the system informs on-site response personnel of the predicted location of leak source in real-time. It enables a prompt and systematic response to the accident.

This study focuses on developing a model that predicts the location of probable leak source via Deep-learning algorithm. The proposed neural network based source localization model is illustrated in Fig. 2. The model predicts the Top-5 locations in order of highest likelihood. Computational Fluid Dynamics (CFD) simulations generate data for training and verifying the neural net-

works. The concentration data of leaked chemical are extracted at the points of the sensors located on the plant outer fence from the CFD simulations and those are used as imaginary sensor data. The neural networks learn the sensor data to predict the location of leak source by recognizing each different pattern of the sensor data for different leak sources. Recurrent Neural Network with Long Short-Term Memory (LSTM-RNN) and Feed-forward Neural Network (FNN) are trained by supervised learning. After learning, the prediction performance of LSTM-RNN and FNN are quantitatively analyzed to derive the best structure.

The rest of this paper is organized as follows. Section 2 introduces related works on source localization problems. The theoretical backgrounds of artificial neural networks are stated in Section 3. The problem statement and modeling process of a prediction model for locating leak source based on a Deep-learning algorithm are presented in Section 4. Section 5 shows the performance comparison between the two types of neural network structure; LSTM-RNN, FNN. Conclusions from model development are summarized in the final section.

2. Related works

The problem of source localization has been a challenge in a wide range of areas including truffle search, unmanned mine removal, underground pipe rupture detection, noise source identification, radar detection, human search & rescue, chemical

* Corresponding author.

E-mail address: dongil@mju.ac.kr (D. Shin).

Nomenclature

$a_i^{(j)}$	the value of i -th neuron in the j -th layer without activation function [-]
$b^{(i)}$	the bias of i -th layer [-]
$C(\cdot)$	cost function [-]
C_t	The state of LSTM cell at time t [-]
\tilde{C}_t	the cell activation vector of LSTM cell [-]
h_t	the calculated output vector of LSTM cell at time t [-]
i_{lj}	the i -th neuron of j -th layer [-]
i_t	the input gate of LSTM cell [-]
$f(\cdot)$	activation function [-]
f_t	the input gate of LSTM cell [-]
L_i	the i -th layer [-]
o_t	the output gate of LSTM cell [-]
$W_{ij}^{(k)}$	the weight of the connection between i -th neuron in the k -th layer and j -th neuron in the $(k+1)$ -th layer [-]
$X^{(1)}$	the input data [-]
$X^{(i)}$	the input value of i -th layer [-]
Y	the one-hot matrix (0 or 1) [-]
\hat{Y}_i	the calculated likelihood of belonging to i -th class from the output layer [-]

leak material tracking, etc. These are divided into odor/chemical source localization and acoustic source localization depending on what is the first principle governing the phenomenon. However, they share basic methodologies for locating the source. In particular, the methodologies for odor/chemical source localization are divided into three types according to approach: behavior based method, tracking vector-based method, and analytical model-based method.

The behavior-based method is a planning method that designs all actions for every situation that the mobile robot(s) can encounter while tracing the leak source. When the mobile robot(s) encounters each situation, it responds reflexively by properly selecting and combining the pre-set actions. Most of these methods set the two broad conditions. The one is the case that the concentration of the substance sensed at the current point is higher than that of the previous point and the other is the opposite case. If the concentration of the current point is higher than the previous point, move

a little bit. Since the mobile robot is already placed in a plume of the chemical. In the opposite case, the robot moves bigger to find the location of the plume. At this time, a method of moving in a random or predetermined angle (Holland and Melhuish, 1996; Russell, 2004a, 2004b); moving in a zigzag so as not to leave the plume (Li et al., 2001); spirally surge rotating and moving in the upwind direction (Hayes et al., 2002; Ferri et al., 2009) have been used. These methods have shown some success in tracking the leak source. However, when performing the pre-set actions while tracing the leak source, it cannot avoid unnecessary actions or repeated unnecessary voyages deviating from the shortest path. For this reason, even in simple terrain, it took a long time for tracking; it took 218 to 379 s to track an 11-meter plume formed by leaks at one end of an empty 11 × 3 m rectangular space without obstacles (Lochmatter et al., 2008).

The second method is a tracking vector-based method. The method generates a tracking vector to the leak source and moves the mobile robot along the direction. Marques et al. (2003) proposed a method to set the tracking vector as the function of the average wind speed, the instantaneous wind speed, and the upwind vector, and applying the proportional constant to each variable. The proportional constants are applied differently according to the four steps of leakage source tracking step: Out-of-plume, Far-from-source, Close-to-the-source, In-the-source. Ishida et al. (2004) proposed a method that derives a tracking vector directed to a leak source by measuring concentration gradient vector using a mobile robot equipped with three triangular sensors and synthesizing it with an upwind vector. Tracking vector-based method can reduce the unnecessary movement of the behavior-based method because it moves the mobile robot along the direct direction to the leak source by using gradient and upwind vector, not depending predefined moving angle and moving width. However, if a complex tracking vector is generated by eddy formed in complex terrain, it may take a long time for tracking or mistakenly point to a location where the flows are rapidly changed due to the terrain effects (Cho et al., 2018).

The last method is a data analytics-based method. Li et al. (2011) and Neumann et al. (2013) used a method to derive a source that has statistically maximum possibility by applying Particle Filter (PF) to the measured data. Singh et al. (2004) and Li et al. (2006) was not a study of chemical plant leaks but tried to solve the source localization problem of groundwater pollution source via Artificial Neural Network (ANN). They solved the classification problem that which is the leak source among the multiple

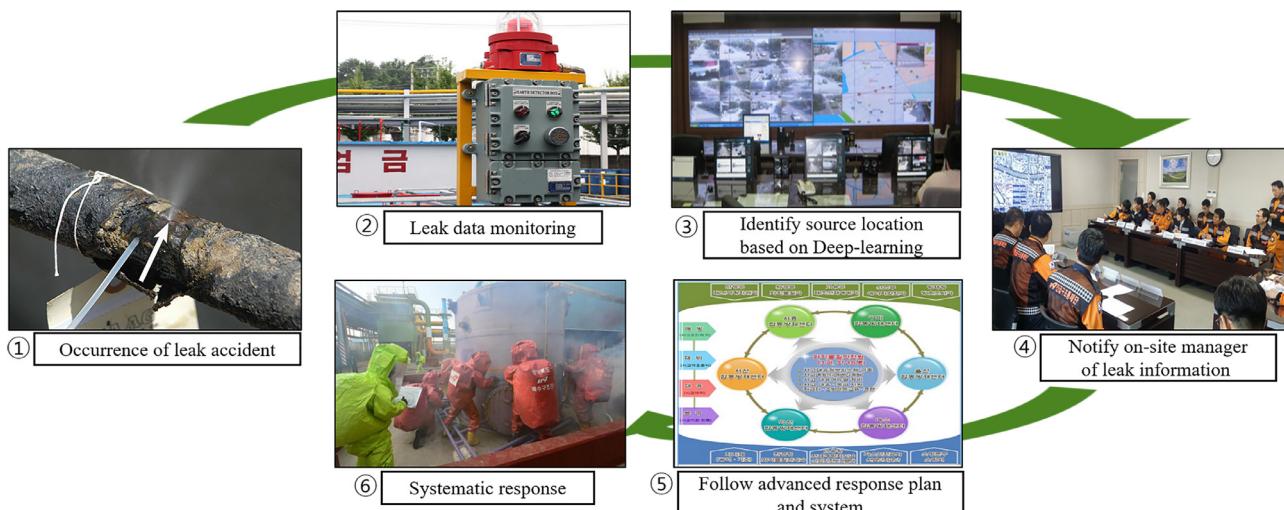


Fig. 1. Schematic diagram of the integrated emergency response system.

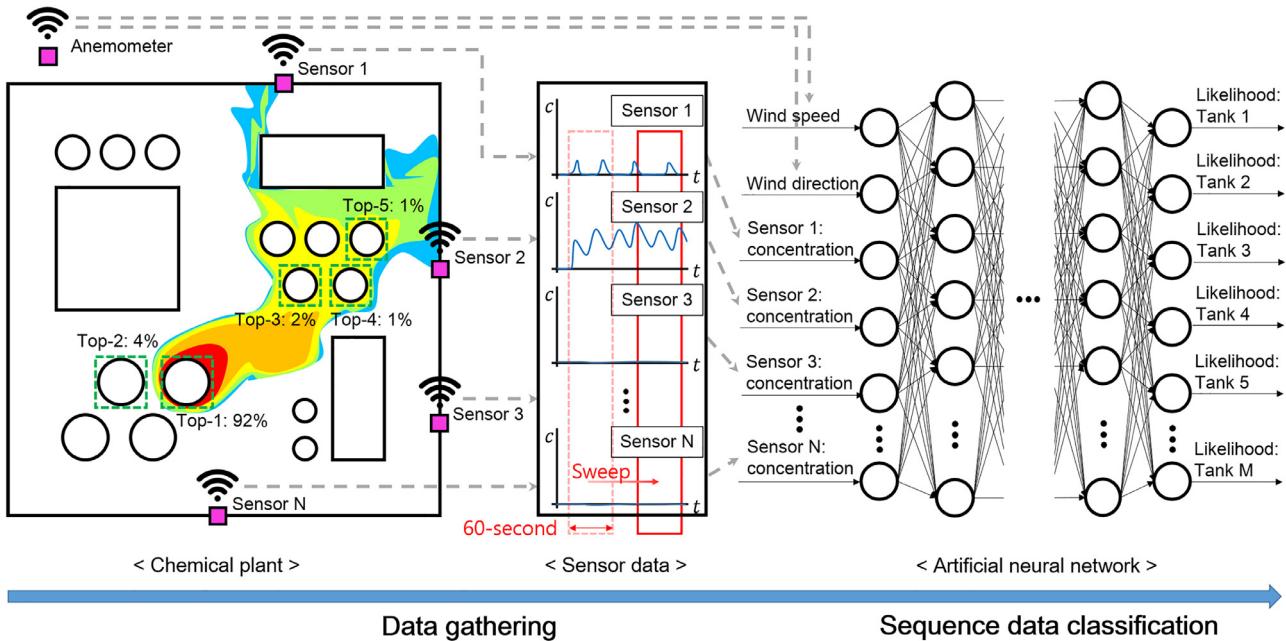


Fig. 2. Schematic diagram of a leak source localization model.

leak candidates using the sensed concentration data in the underground sensor grid. Wang et al. (2015) used the integration of neural networks and PHAST to estimate hazardous gas dispersion. PHAST is well-known software for dispersion simulation and the results can be utilized for decision making, however, some input data needed for PHAST simulation (source information) are not available to gather in the midst of events. Hence, the neural network plays a role of an alternative bypass model that estimates dispersion without source information. This method can estimate source location on a straight line, however, it is not proper to apply it in chemical plants which have complex terrain; PHAST cannot reflect terrain effects. Cho et al. (2018) and Kim et al. (2018) showed a combination of machine-learning and CFD simulation to solve the leak source localization problem of a chemical plant which has complex terrain. These studies try to train the machine-learning models using virtual sensor data which are generated by CFD simulations. These methods use a pre-train approach for the predictive machine. Hence, the long calculation time for accurate CFD simulations is not necessary at the prediction stage.

In the field of process safety, numerous studies have been conducted to solve challenging problems or improve the existing systems by using big-data combined with artificial intelligence beyond human cognitive capability. Methods of classification and prediction using neural networks and Markov analysis (Venkatasubramanian and Chan, 1989; Shu and Zhao, 2014; Zhang and Zhao, 2017) have been conducted to develop models that can assist decision making in abnormal detections, risk assessments, safety device specifications, etc. This paper also aims to improve the performance of the existing source localization models by applying the deep-learning algorithm to the problem of leak source track which is one of the challenging inverse problems in the process safety field.

3. Theoretical background

3.1. Feed-forward neural network

An artificial neural network is a learning machine that imitates a biological neural network of a human. Multiple neurons consti-

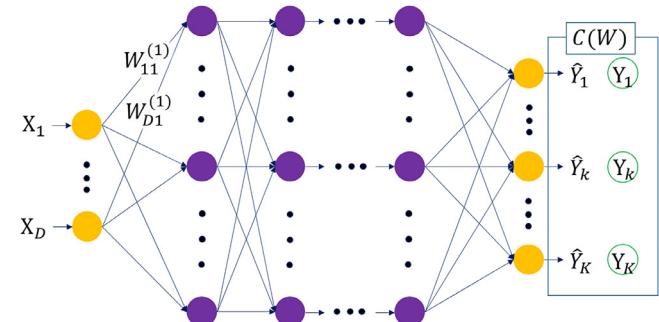


Fig. 3. Diagram of a feed-forward neural network.

tute one layer and propagate data to the next layer. Feed-forward Neural Network (FNN) is a kind of artificial neural network that propagates data only in feed-forward direction. Fig. 3 shows the generalized structure of FNN. The first layer is called an input layer and the last layer is called an output layer. There are hidden layers between the input layer and the output layer. Neurons between each layer have respective weighted connections and the weights are calculated as a sum of linear products of weights. Namely, if a set of D dimensional input data $X^{(1)} = [X_1^{(1)}, \dots, X_D^{(1)}]^T$ is input to the input layer, the value ($a_j^{(k+1)}$) derived at the j -th neuron in the $(k+1)$ -th layer is calculated as Eq. (1):

$$a_j^{(k+1)} = \sum_i^I W_{ij}^{(k)} X_i^{(k)} + b^{(k)} \quad (1)$$

for $i = 1, \dots, I^{(k)}, j = 1, \dots, J^{(k+1)}, k = 1, \dots, K - 1$

where $W_{ij}^{(k)}$ is the weight of the connection between i -th neuron in the k -th layer and j -th neuron in the $(k+1)$ -th layer, $X_j^{(k)}$ is the input value of a j -th neuron in the k -th layer, $b^{(k)}$ is bias term of the k -th layer, $I^{(k)}$ is the number of neurons in k -th layer, the $J^{(k+1)}$ is the number of neurons in $k+1$ -th layer, and the K is the number of layers. Then activation function ($f(\cdot)$) is applied to $a_j^{(k+1)}$ (See

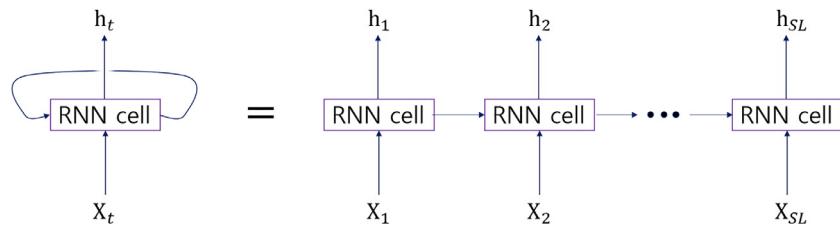


Fig. 4. Data propagation in recurrent neural network (adjusted from Olah, 2015).

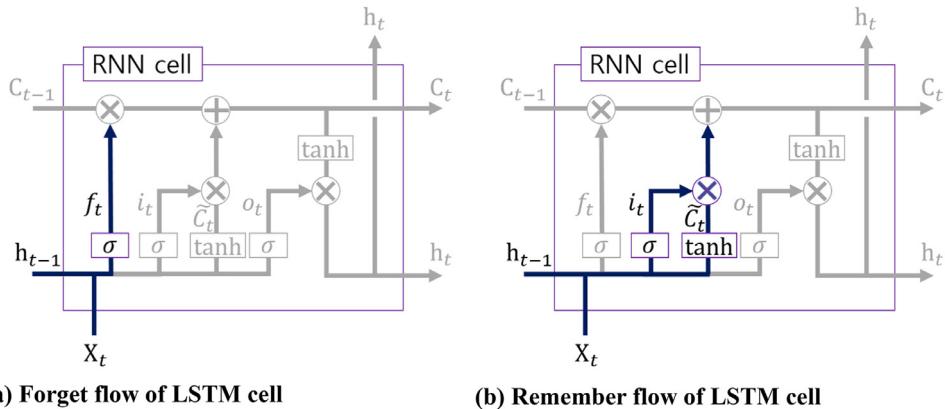


Fig. 5. Diagram of long short-term memory cell (adapted from Olah, 2015). (a) Forget flow of LSTM cell. (b) Remember flow of LSTM cell.



Fig. 6. The satellite photograph of “D*” chemical plant in Yeosu Industrial Complex, South Korea.

Eq. (2)).

$$X_j^{(k+1)} = f(a_j^{(k+1)}) \text{ for } k = 1, \dots, K-1 \quad (2)$$

Rectified Linear Unit (ReLU) which is widely used among the various types of activation function is used in the part of fully connected layers of FNNs and LSTM-RNN. ReLU is shown in Eq. (3):

$$\text{ReLU}(a) = \max(0, a) \quad (3)$$

All the neurons without neurons in the output layer are calculated in the same manner of $X_j^{(k+1)}$ however, only the output values of neurons in the output layer are converted to the likelihoods $\hat{Y} = [\hat{Y}_1, \dots, \hat{Y}_N]$ of belonging to N different classes by using softmax function (See Eq. (4)). The number of neurons in the output

layer must be equal to the number of N classes.

$$\hat{Y}_n = \text{softmax}(a_j^{(K)}) = \frac{\exp(a_j^{(K)})}{\sum_{j=1}^J \exp(a_j^{(K)})} \quad (4)$$

for $0 \leq \hat{Y}_n \leq 1$ and $\sum_{n=1}^N \hat{Y}_n = 1$

All data set should be one-hot labeled by $Y = [Y_1, \dots, Y_N]$ indicating the class to which the actual X_n belongs; herein, one-hot encoding is a labeling method that sets an element of Y to ‘1’ which is the correct class element. The other elements have 0 value. Since labeled tank digits have no physical meaning, one-hot encoding which can remove the meaning of labeled digits is appropriate. The training is conducted by modifying weights ($W_{ij}^{(k)}$) in the direction of minimizing the value of cost function $C(W)$ (Bishop, 2006; Goodfellow et al., 2016; Niu et al., 2017). The cross entropy function (Eq. (5)) is used as the cost function.

$$C(W) = Y_n \ln \hat{Y}_n + (1 - Y_n) \ln (1 - \hat{Y}_n) \quad (5)$$

3.2. Recurrent neural network and long short-term memory

As shown in Fig. 4, Recurrent Neural Network (RNN) has a unique structure that the past processed-output data is input to the network again. This structure allows for better classification of sequence data by memorizing past data. Of course, it is possible to let the general FNN learn the past data by inputting a predetermined sequence at a time. It is also possible to make the 1-dimensional kernel of the Convolutional Neural Network (CNN) find the features of the sequence data. However, RNN is more effective to extract the features of sequence data by re-inputting worthy output data that is output through processing to satisfy the objective function (Goodfellow et al., 2016).

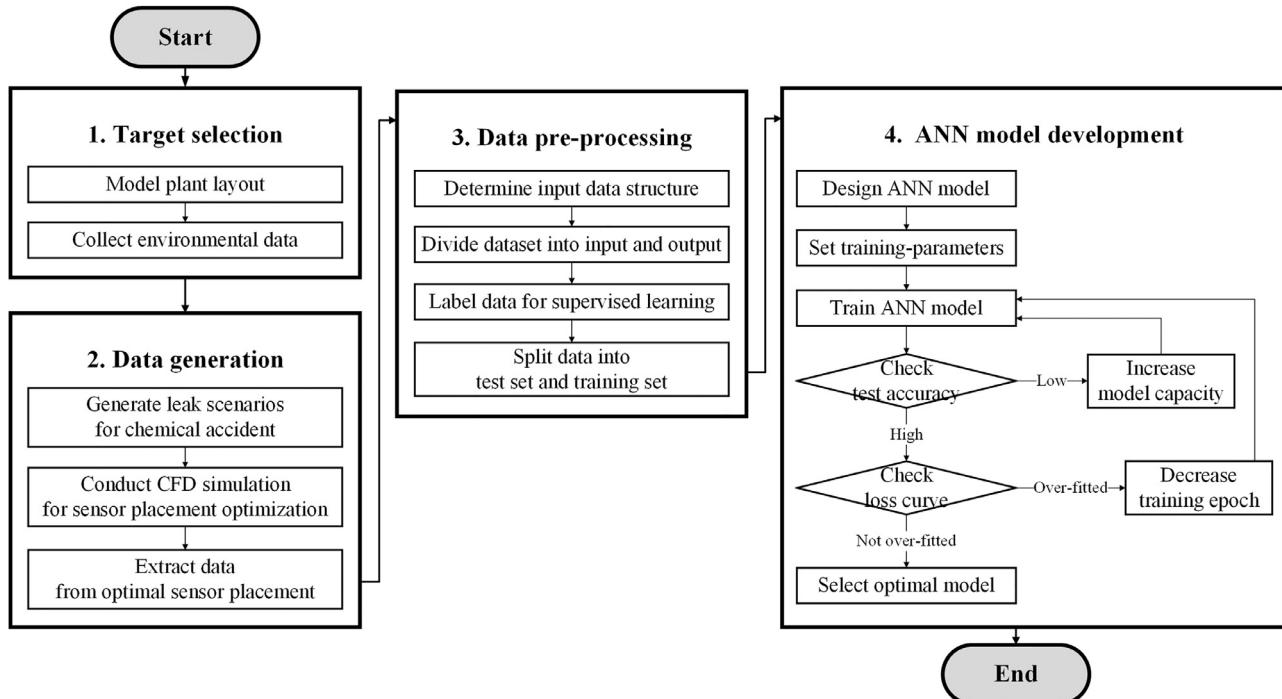


Fig. 7. Flowchart for developing leak source tracking system based on Deep-learning.

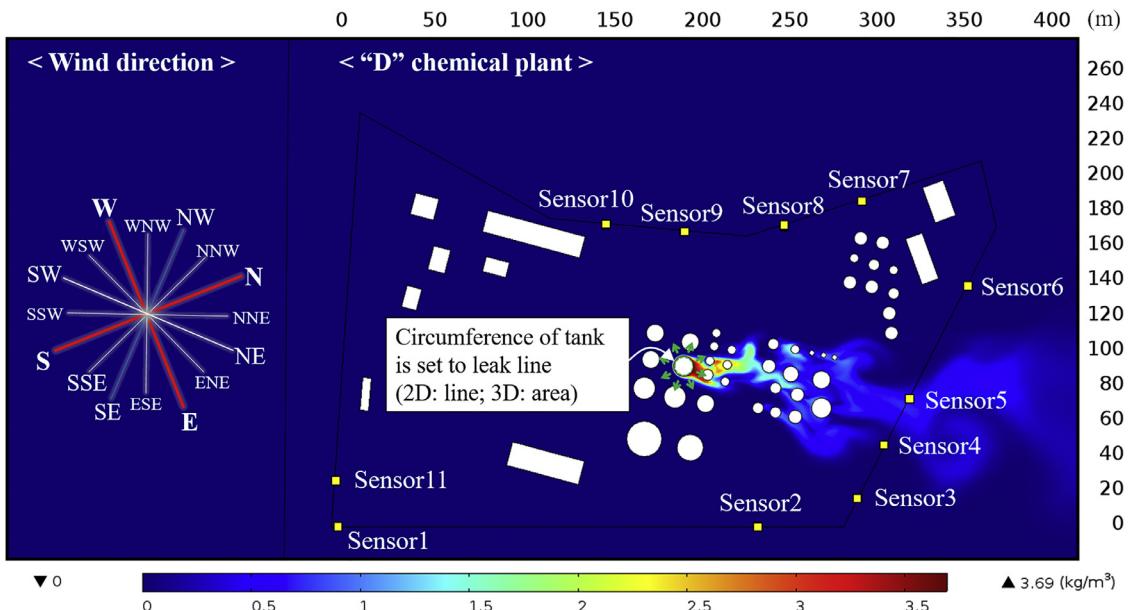


Fig. 8. An example of chemical leak CFD simulation: storage tank 29, NNE wind.

$$\begin{aligned}
 f_t &= \text{sigmoid}(W_f h_{t-1} + W_f X_t + b_f) \\
 i_t &= \text{sigmoid}(W_i h_{t-1} + W_i X_t + b_i) \\
 \tilde{C}_t &= \tanh(W_C h_{t-1} + W_C X_t + b_C) \\
 o_t &= \text{sigmoid}(W_o h_{t-1} + W_o X_t + b_o) \\
 C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \\
 h_t &= o_t \odot \tanh(C_t)
 \end{aligned} \tag{6}$$

As shown in Fig. 5 and Eq. (6), the RNN structure using Long Short-Term Memory (LSTM) finds out which data of the past data

is good to remember ($i_t \odot \tilde{C}_t$) and forget (f_t) to make an optimal decision. These features make LSTM-RNN suitable for long sequence data learning.

4. Modeling for leak source tracking networks

4.1. Problem statement

The purpose of this study is to develop a source localization model that can predict the suspected location using outer

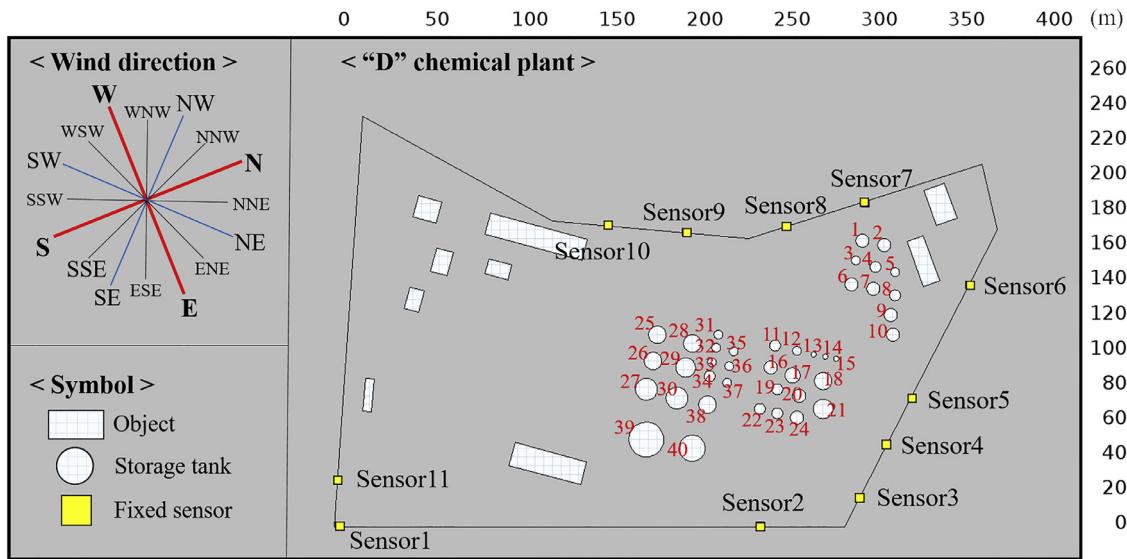


Fig. 9. Schematic diagram of the target.

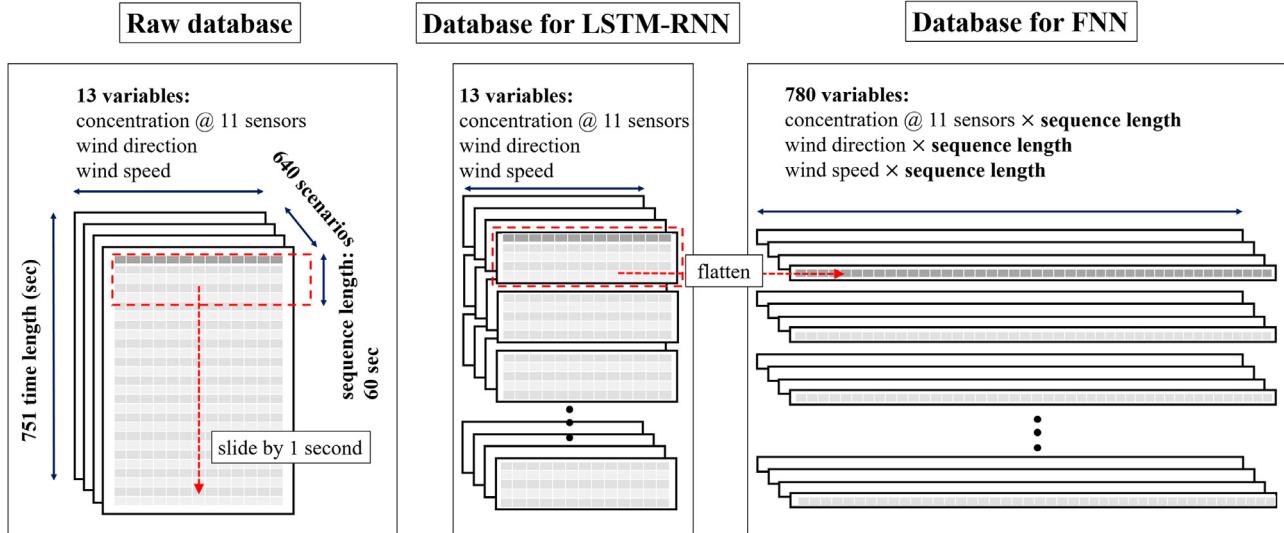


Fig. 10. Schematic diagram of reshaping raw database for LSTM-RNN and FNN.

fence monitoring data. Fig. 6 shows the target of our model: “D*” chemical plant in Yeosu, Korea. The data of concentration gathered from the points of 11 sensors, the wind direction, and wind speed obtained from the Korea Meteorological Agency are input to artificial neural networks and the network classifies it into possible leakage source candidates through maximum likelihood method.

The artificial neural network model should be able to predict the points where the actual leak points are likely to be located. Furthermore, this should be done in real-time. From the modeling point of view, it can be seen as a problem of inputting into the neural network with 13 time-series variable values (wind direction, wind speed and concentration data collected from 11 sensors) as input data and classifying it into 41 classes in real-time; leakage is not detected (class 0), leakage occurs in the leak candidates (class 1–40).

4.2. Flowchart for model development

The model is developed based on the flowchart shown in Fig. 7. The process of model development consists of two stages: database generation (step 1–3), the experiment of network structure (step 4).

It would be great if we could obtain sensor data from experiments on real chemical plants to train and test the network. However, in terms of cost and risk, it is practically impossible to obtain leak accident sensor data for real chemical plants. Therefore, we tried to perform CFD simulations and extract the data from the points where the sensors are located and build a database to train and test the networks. In this study, CFD simulation data and optimal sensors layout used for generating database were obtained from the collaborative research with Cho (2017).

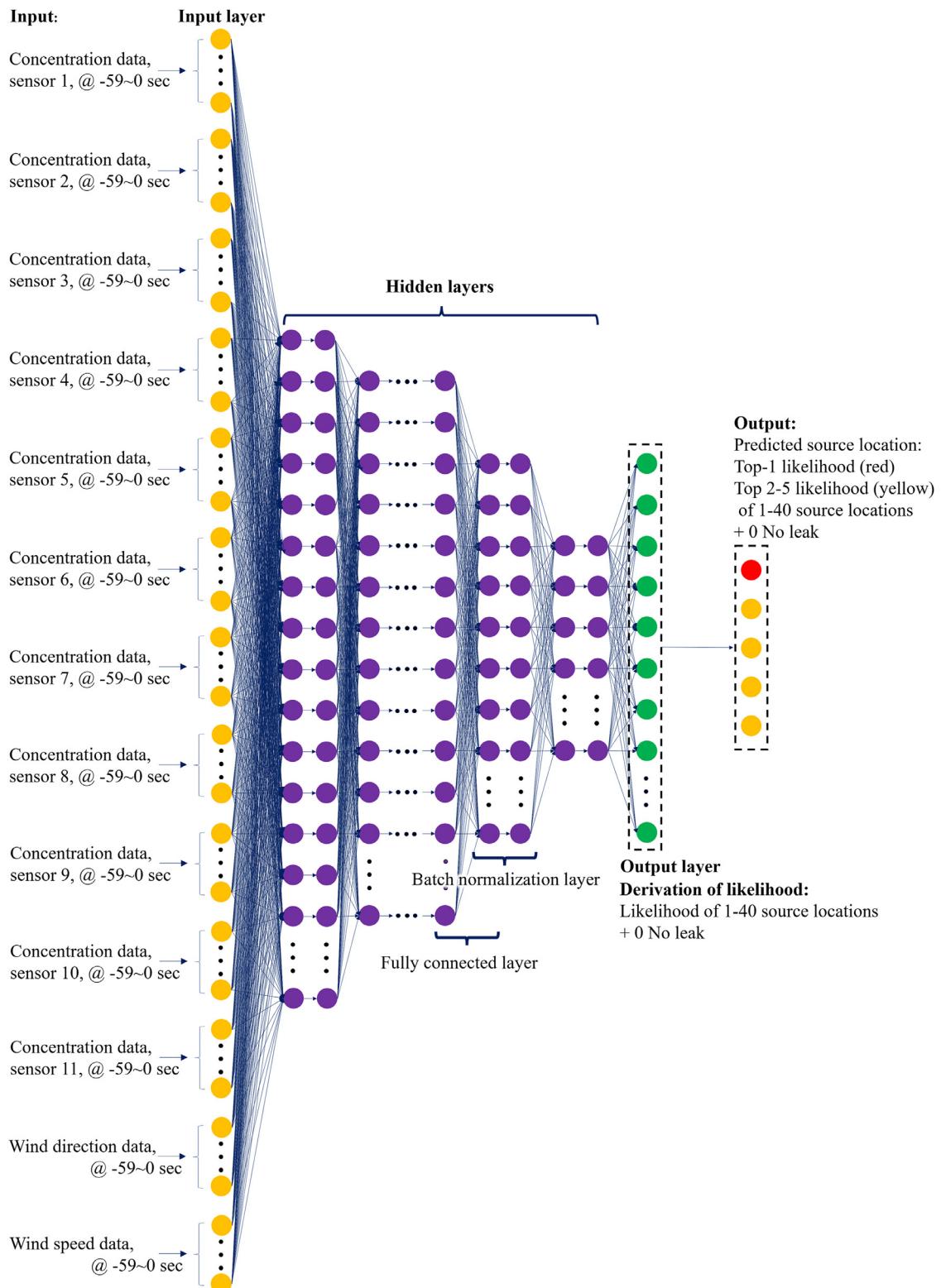


Fig. 11. Schematic diagram of FNN.

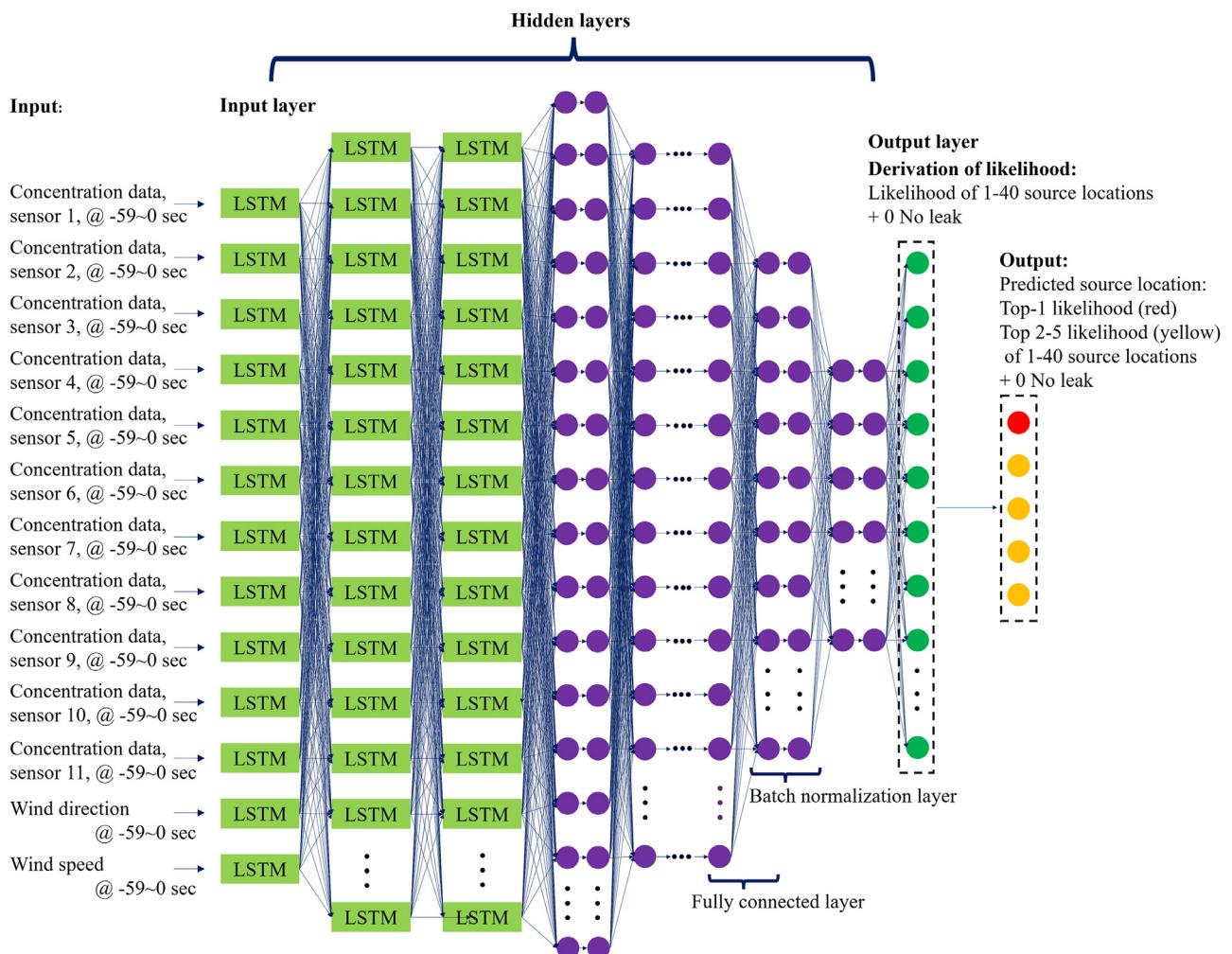


Fig. 12. Schematic diagram LSTM-RNN.

4.3. Database generation

4.3.1. Select target

Dispersion of leaking material in an outdoor chemical plant is greatly influenced by the wind field, and the wind field is very different depending on the terrain. For this reason, the chemical plant as a target of the CFD simulation must be selected first. This study selected the “D*” chemical plant located in Yeosu, Korea as the target.

4.3.2. Select scenarios of chemical leak accidents

In order to conduct CFD simulations, it is necessary to generate leak accident scenarios that can occur at the targeted chemical plant. In this study, 40 storage tanks located at the “D*” chemical plant are set as potential leak candidates, and they are labeled from storage tank 1 to 40. Even with the same storage tanks, the dispersion pattern of the leaking material varies depending on the wind fields. Therefore, 16 wind directions and those annual maximum wind speeds apply to each storage tank as wind fields: N(6.48 m/s), NNE(5.98 m/s), NE(8.80 m/s), ENE(9.11 m/s), E(8.86 m/s), ESE(4.53 m/s), SE(5.58 m/s), SSE(2.79 m/s), S(3.13 m/s), SSW(5.32 m/s), SW(8.14 m/s), WSW(7.70 m/s), W(7.23 m/s), WNW(8.29 m/s), NW(9.36 m/s), and NNW(9.64 m/s); actually, the annual average wind speeds are more proper than the annual maximum wind speeds. However, in order to reuse the data of

Cho (2017), the annual maximum wind speeds are selected instead of the annual average wind speeds. In conclusion, 640 types of leak accident scenarios which could occur in the storage tanks of the “D*” chemical plant are selected; 40 storage tanks × 16 wind fields = 640 types of accident scenarios.

4.3.3. Conduct CFD simulation for selected scenarios

Cho (2017) conducted 2D real-time dynamic simulations in the 0–750 s range for 640 leak accident scenarios. All leak accident scenarios were equally set to leak toluene at the emission rate of 40 mol/m²/s around the circumference of each storage tank (2D: leak line; 3D: leak area, See, Fig. 8); although leaks tend to occur at certain defect points rather than leaks at circumference line, setting a line as a leak source can reduce the simulation costs by reducing the number of scenarios. This is a reasonable simplification, as the difference in dispersion by the two methods would be small. Hence, the dispersion of the vapor formed from released liquid around tank which losses original momentum is strongly governed by wind fields.

The mass balance and the momentum balance were used to calculate the numerical solution for a total of 34,906 elements by Finite Element Method (FEM) method using COMSOL Multiphysics 5.0.

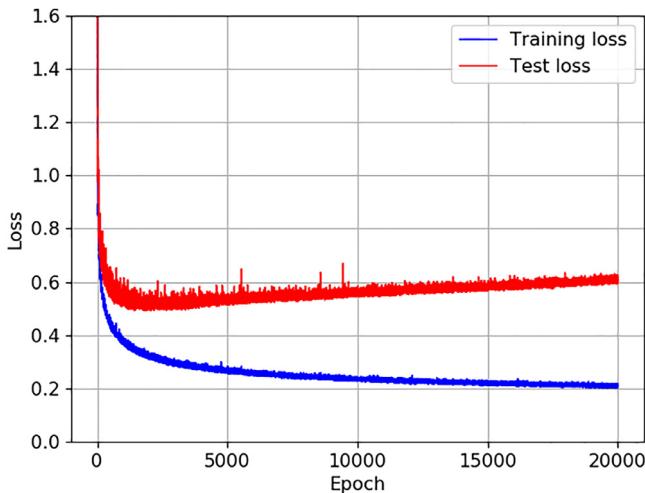


Fig. 13. Loss graph of FNN-100.

4.3.4. Optimize sensor placement

Cho (2017) derived the optimal sensor placement that can cover all 640 types of accident scenarios with a minimum number of sensors when installing sensors on plant outer fence. For all accident scenarios, the sensors must enable to detect the concentration above ERPG-2. Here, ERPG-2 refers to the maximum concentration in the air where almost everyone is exposed for an hour causing irreversible symptoms or serious health effects.

The optimal placement of the sensors is obtained through the following procedure. First, cut the fence into a linear shape and then make a grid of 0.5 m apart. For all scenarios, repeated experiments will find the case that sensor network can detect the concentration above ERPG-2. The logic used to derive the location of optimal sensor placement is as follows:

```

flatten the fence line as a straight line and generate the grid
number of sensors =1 (initial guess) and arrange it at the far left
repeat {
    if the maximum concentration of sensor locations for 640 scenarios < ERPG-2
        then move 'the last sensor' to the next grid point
            if 'the last sensor' reaches the final point and there is 'the former
                sensor' still on the left
                    then now, 'the former sensor' is called 'the last sensor' and
                        move it to the next grid
                    else add new one sensor and rearrange the sensors from
                        the left with a grid interval
                    else derive the sensor locations
}

```

The positions of the sensors arranged in this way are shown in Fig. 9.

4.3.5. Extract data from CFD simulations and generate database

After inputting the coordinates of the optimal sensor placement position in the CFD simulation results, the data are extracted from the coordinates. Wind direction, wind speed, and concentration data for 0–750 s for each scenario are collected at 11 sensor locations and stored as the Comma Separate Value (CSV) format. The obtained data are processed into time-series data having a sequence length of 60 s (shown in Fig. 10); 692 sequence datasets for each scenario. Each dataset consists of 780 elements: 13 elements (wind direction, wind speed and concentration data of 11 sensors) \times 60 s = 780 elements. The sequence length of 60 s does not have a mathematical meaning, however, it is equal to the sequence length that Cho et al. (2018) used to compare the results. The generated data are used as input data of the artificial neural network.

For supervised learning, input data should be labeled with the corresponding potential leak point number. In this study, if the concentration of ERPG-2 or higher is detected more than 10 times for 60 s, the data is labeled as the tank number. Actual sensors of the chemical plant can deliver false values to the model, and for this reason, the leakage point is labeled based on the above-mentioned criteria; more than 10 positive signals out of 660

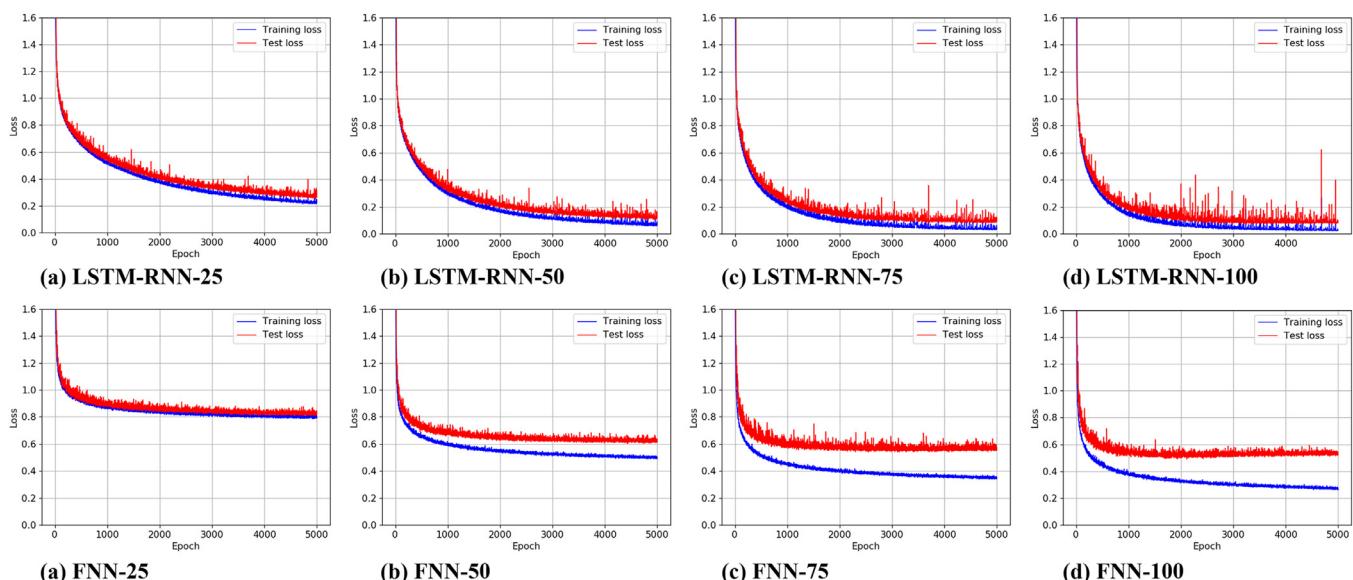


Fig. 14. Cross entropy loss graph of the learning process.

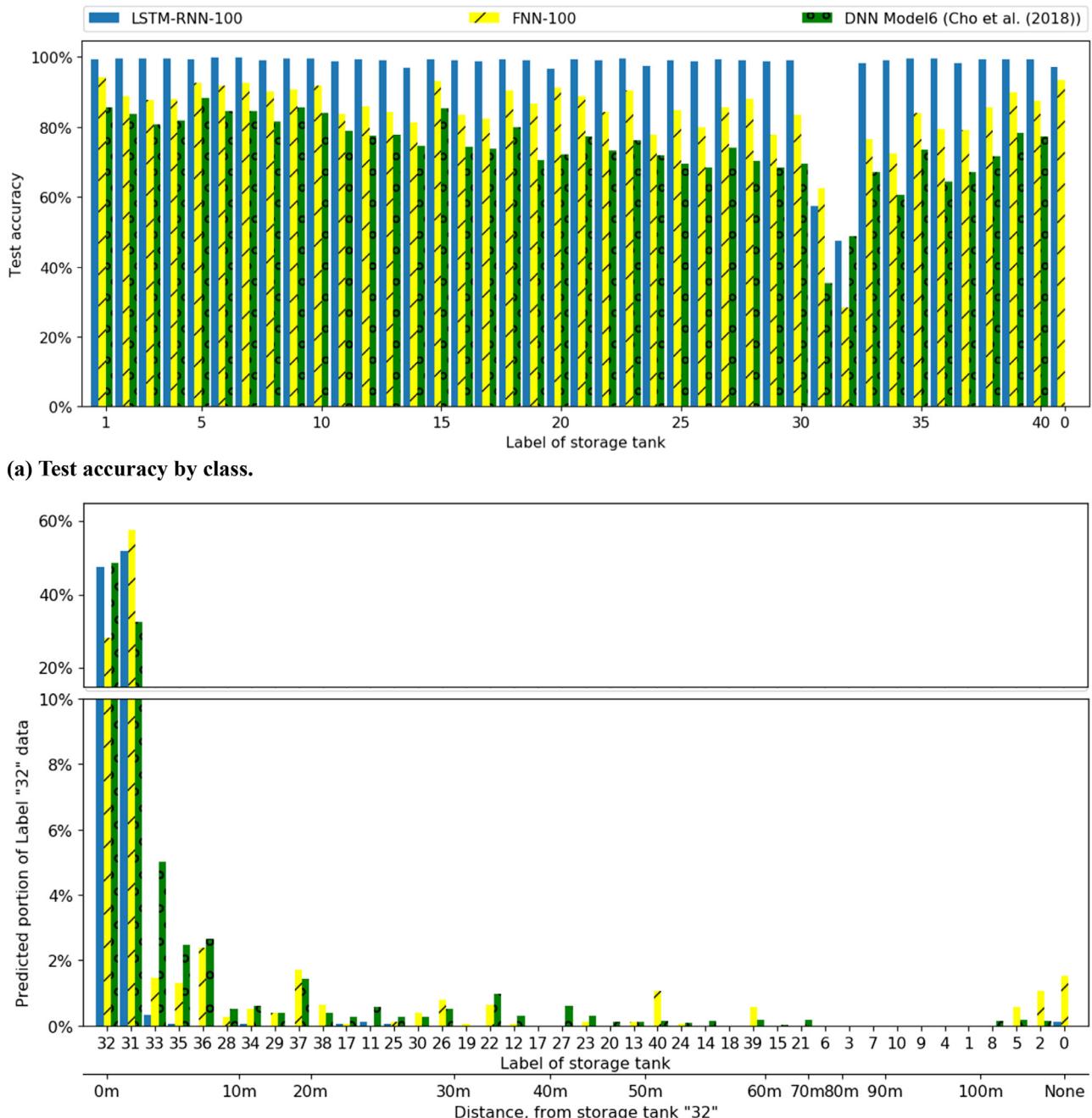


Fig. 15. Comparison of model prediction results. (a) Test accuracy by class. (b) Predicted result of Label “32” data.

detection opportunities ($11 \text{ sensors} \times 60 \text{ s} = 660$ detection opportunities). For data that does not meet the criteria, it is labeled as label 0, which means no leakage or leakage below detection criteria. The others are labeled with the corresponding number of the storage tank.

When the data are labeled according to the above criteria, the number of data belonging to each label became imbalanced. Training neural networks with imbalanced data may result in weight shifts in the direction of increasing the classification performance for a specific label. For this reason, stratified random sampling is used. As a result, the generated balanced database have 247,435 each dataset. Then the database is randomly separated by a ratio

of 7 to 3: 175,015 number of training data and 72,420 number of test data.

4.4. Modeling of network structures

Cho et al. (2018) proposed a method of using deep-FNNs to solve the source tracking problem of a chemical plant leak accident. The schematic diagram of deep-FNN is shown in Fig. 11. This study showed the possibility of using a neural network for this problem, however, there was room for improvement in the model to get better performance. The authors tried to construct input data in two forms. One was a form that uses only the current time

point data, and the other was a form of six-time step sequence data for considering past point data: sequence data of -60 s, -30 s, -20 s, -10 s, -5 s, 0 s. In the study, learning data in the form of sequence data showed a better prediction. Kim et al. (2018) solved the same problem using the Random Forest classifiers and found that the physical coordinates of the sensors do not affect classification. For this reason, the form of input data used in our study is a form of sequence data constituted of wind direction, wind speed, and concentration data, without the coordinates of sensors. This study also uses wind speed, direction, and concentration as inputs; the leak characteristics such as leak amount and rate are not considered since the data are not directly gathered in the midst of the event.

The two types of network structures are used: FNN and LSTM-RNN. Cho et al. (2018) selected specific time points without any data analysis base. By contrast, the LSTM-RNN structure is expected to automatically select the time point of data extraction time to obtain optimal classification performance by selecting the weights. For this reason, we quantitatively diagnosed how much the data propagation method affects the classification of leak source locations if network learns the same amount of information on the network by selecting FNN with the same capacity as the LSTM-RNN model as the control group.

Fig. 12 shows the schematic diagram of LSTM-RNN. In LSTM cell, hyperbolic tangent or sigmoid function is used as an activation function that has a low gradient value. Due to its structural characteristics, the gradient vanishing problem occurs if the LSTM cells are deeply stacked up. The multi-layered LSTM showed good performance when stacking up to three LSTM layers (i.e. Graves et al., 2013; Hammerla et al., 2016; Rawal and Miikkulainen, 2016). For this reason, the LSTM-RNN models used in this experiment are designed to be stacked up to three LSTM layers. The data propagation type of RNN that extracts features from the association between sequence data is similar to CNN kernel. For this reason, three fully-connected layers are attached to the back of the LSTM as a classifier part similar to CNN. Therefore, the six hidden layer LSTM-RNNs are used in the experiment, and FNNs in the control group are also designed to have six hidden layers. As mentioned in Section 4.3, LSTM-RNNs receive sequence data of wind direction, wind speed, and concentration data measured by 11 sensors with a sequence length of 60 s as each input dataset. For this reason, 13 neurons are placed on the input layer, so that the values of each variable in the database could be connected one by one. In the case of FNNs, 780 neurons are placed in the input layer. It means that the FNNs receive the same amount of input data of LSTM-RNN at once: 13 elements (wind direction, wind speed and concentration data of 11 sensor points) \times 60 each (60 s) = 780 elements. In the output layer, 41 neurons are arranged to derive each likelihood belonging to 41 labels (label 0 to 40).

Batch normalization (Ioffe and Szegedy, 2015), one of the powerful regularization technic, is applied to all fully-connected layers to improve learning speed and to prevent overfitting. Adam optimizer (Kingma and Ba, 2014) trains the networks.

5. Experiments

5.1. Experimental and control set-up

In order to analyze the performance changes according to the data propagation structure, we quantitatively compare the prediction accuracy of LSTM-RNNs and FNNs that have the same capacity. LSTM-RNNs are experimental group and FNNs are a control group. LSTM-RNNs and FNNs that have four different counterpart sets that have the same capacity levels. The four levels of capacity are chosen by doubling, tripling and quadrupling the number of parameters that make up each model, based on 25,000 parameters. In

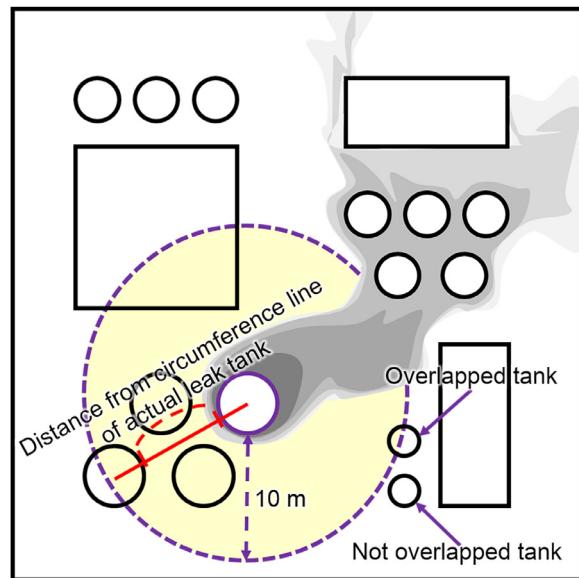


Fig. 16. Distance from circumference line of actual leak tank.

the first layer of the classifier part of the LSTM-RNN and FNN, the number of neurons is chosen equal to a size of one-thousandth of the total number of parameters used in each model for the regularity. The detailed structures of the models used in the experiments are listed in Table 1.

When determining the number of training epochs, the changes in the loss have to be observed during the learning process. In general cases, the training loss consistently goes to zero as learning progresses. However, the test loss decreases to a certain point and then increases again. The reason for this is known to be the effect of overfitting. Learning means modifying the weights to reduce training loss. Therefore, the training loss decreases infinitely while the test loss decreases to a certain amount of learning and the loss value increases due to the overfitting problem.

Overfitting occurs more strongly as the number of parameters used in curve fitting increases. For this reason, we observed the loss curve in the process of learning FNN-100 up to 20,000 epochs in order to understand the rough trend to set the training epochs; FNN-100 is a model with the largest number of parameters. As shown in Fig. 13, the test loss reached the minimum when the learning progressed to 1800 epochs, and then the increase of loss was observed. Considering this, the amount of learning about 5000 epochs is set for experiments.

5.2. Experimental results

Table 1 shows that the test accuracy of LSTM-RNN is higher than that of FNN at all counterparts. If the same number of parameters is used, the test accuracy of LSTM-RNNs is 20.45% higher than the average test accuracy of FNNs. Even FNN-100 that use four times more parameters shows lower accuracy than LSTM-RNN-25. The LSTM-RNN-100 model shows the highest prediction performance in the experimental procedure. It shows 97.08% of test accuracy and 0.04% of Top-5 error. The Top-5 error means the error rate in which the actual leak source location does not exist in the group of Top-5 candidates that have the highest likelihood. The Top-5 errors of LSTM-RNNs are also lower at all capacity levels. At the counterpart models, the Top-5 errors of LSTM-RNNs are as low as 6.98% of Top-5 errors of FNNs. The Top-5 error of 0.04% shown by LSTM-RNN-100 means that the model finds the target with a probability of 99.96% if it outputs the Top-5 locations.

Fig. 14 shows the loss curve of LSTM-RNNs and FNNs. The important point is that the gap between the training accuracy and

Table 1
Experimental results.

	Network structure (where L means LSTM layer)	Number of parameters	Epoch	Accuracy (training/test)	Top-5 error (training/test)
LSTM-RNN-100	13-84L-62L-41L-100-60-28-41	99,973	5000	0.9880 / 0.9708	0.0000 / 0.0004
			4000	0.9866 / 0.9700	0.0000 / 0.0005
			3000	0.9837 / 0.9678	0.0000 / 0.0005
			2000	0.9736 / 0.9588	0.0005 / 0.0014
			1000	0.9446 / 0.9304	0.0037 / 0.0046
			100	0.7638 / 0.7523	0.0372 / 0.0377
LSTM-RNN-75	13-75L-48L-41L-75-48-28-41	74,929	5000	0.9850 / 0.9676	0.0000 / 0.0005
			4000	0.9827 / 0.9656	0.0000 / 0.0007
			3000	0.9770 / 0.9603	0.0002 / 0.0012
			2000	0.9634 / 0.9475	0.0012 / 0.0024
			1000	0.9251 / 0.9108	0.0059 / 0.0070
			100	0.7378 / 0.7267	0.0448 / 0.0465
LSTM-RNN-50	13-50L-44L-41L-50-35-28-41	49,932	5000	0.9742 / 0.9578	0.0006 / 0.0014
			4000	0.9681 / 0.9518	0.0012 / 0.0021
			3000	0.9578 / 0.9430	0.0025 / 0.0033
			2000	0.9376 / 0.9243	0.0049 / 0.0056
			1000	0.8886 / 0.8787	0.0110 / 0.0114
			100	0.7070 / 0.7267	0.0573 / 0.0580
LSTM-RNN-25	13-25L-27L-41L-25-30-28-41	24,993	5000	0.9168 / 0.9054	0.0071 / 0.0078
			4000	0.9055 / 0.8933	0.0087 / 0.0094
			3000	0.8892 / 0.8797	0.0112 / 0.0115
			2000	0.8597 / 0.8483	0.0157 / 0.0159
			1000	0.8084 / 0.7985	0.0265 / 0.0270
			100	0.6662 / 0.6637	0.0732 / 0.0727
FNN-100	780-100-82-65-50-38-28-41	100,022	5000	0.8947 / 0.8412	0.0090 / 0.0128
			4000	0.8899 / 0.8383	0.0098 / 0.0130
			3000	0.8842 / 0.8341	0.0105 / 0.0138
			2000	0.8744 / 0.8287	0.0117 / 0.0148
			1000	0.8555 / 0.8127	0.0148 / 0.0172
			100	0.7583 / 0.7266	0.0363 / 0.0439
FNN-75	780-75-67-57-47-38-28-41	74,998	5000	0.8702 / 0.8242	0.0129 / 0.0165
			4000	0.8676 / 0.8217	0.0134 / 0.0170
			3000	0.8614 / 0.8176	0.0144 / 0.0138
			2000	0.8521 / 0.8108	0.0117 / 0.0148
			1000	0.8337 / 0.8004	0.0148 / 0.0172
			100	0.7385 / 0.7113	0.0363 / 0.0439
FNN-50	780-50-48-45-42-37-28-41	49,979	5000	0.8150 / 0.7868	0.0205 / 0.0237
			4000	0.8113 / 0.7835	0.0211 / 0.0243
			3000	0.8069 / 0.7788	0.0225 / 0.0253
			2000	0.7979 / 0.7737	0.0245 / 0.0273
			1000	0.7789 / 0.7597	0.0287 / 0.0314
			100	0.7000 / 0.6828	0.0527 / 0.0568
FNN-25	780-25-28-28-28-28-28-41	25,020	5000	0.7115 / 0.7113	0.0486 / 0.0499
			4000	0.7091 / 0.7107	0.0498 / 0.0499
			3000	0.7054 / 0.7035	0.0506 / 0.0515
			2000	0.6993 / 0.6989	0.0526 / 0.0534
			1000	0.6871 / 0.6857	0.0566 / 0.0568
			100	0.6182 / 0.6100	0.0806 / 0.0847

the test accuracy of the LSTM-RNNs is much less than that of the FNNs. This means that LSTM-RNNs can show similar prediction performance for data that they have never experienced before. Namely, the recurrent data propagation structure of LSTM-RNN play a key role in extracting general features of collected time-series sensor data to predict the source location. The model with the highest test accuracy among the models used in the experiment is LSTM-RNN-100 and its test accuracy is 97.08%. Since the amount of the used input data is different from that of Cho et al. (2018), it is difficult to make a clear comparison. Nevertheless, it is clear that LSTM-RNN shows a considerably improved performance; Cho et al. (2018) achieved the 75.43% test accuracy obtained by training FNN with 25 hidden layers up to 100,000 epochs.

Fig. 15(a) shows the prediction accuracy of test data by class. The test accuracy of class 32 is lowest in both LSTM-RNN-100 and FNN-100. Fig. 15(b) shows the results of prediction when the label 32 data which is the most likely to be misclassified is input

to neural networks. The size of the bar represents the portion of each class how many times the model derived the locations as the most suspected leak point from the label 32 data. The x-axis of the graph lists label of the storage tanks in order of closest distance from storage tank “32”. Fig. 16 shows the meaning of the distance.

The reason that the predictive model mispredicts the wrong point as a leak point is that the data showed a similar pattern to the data of the false leak point. Intuitively, it can be judged that the data of near leak points are likely to have a similar pattern. For this reason, even if the prediction is wrong, the frequency of predicting the point near the actual leak point as the leak point is high. Fig. 17 shows the confusion matrix of LSTM-RNN-100; the cells that have red dot are overlapped tanks in 10 m radial region from actual leak point. What is meaningful here is that, unlike the other models, LSTM-RNN-100 showed a 99.44% chance of bringing tanks overlapping within 10 m radial region from the actual leak point (circumference line of actual leak tank) to the wrong leak points.

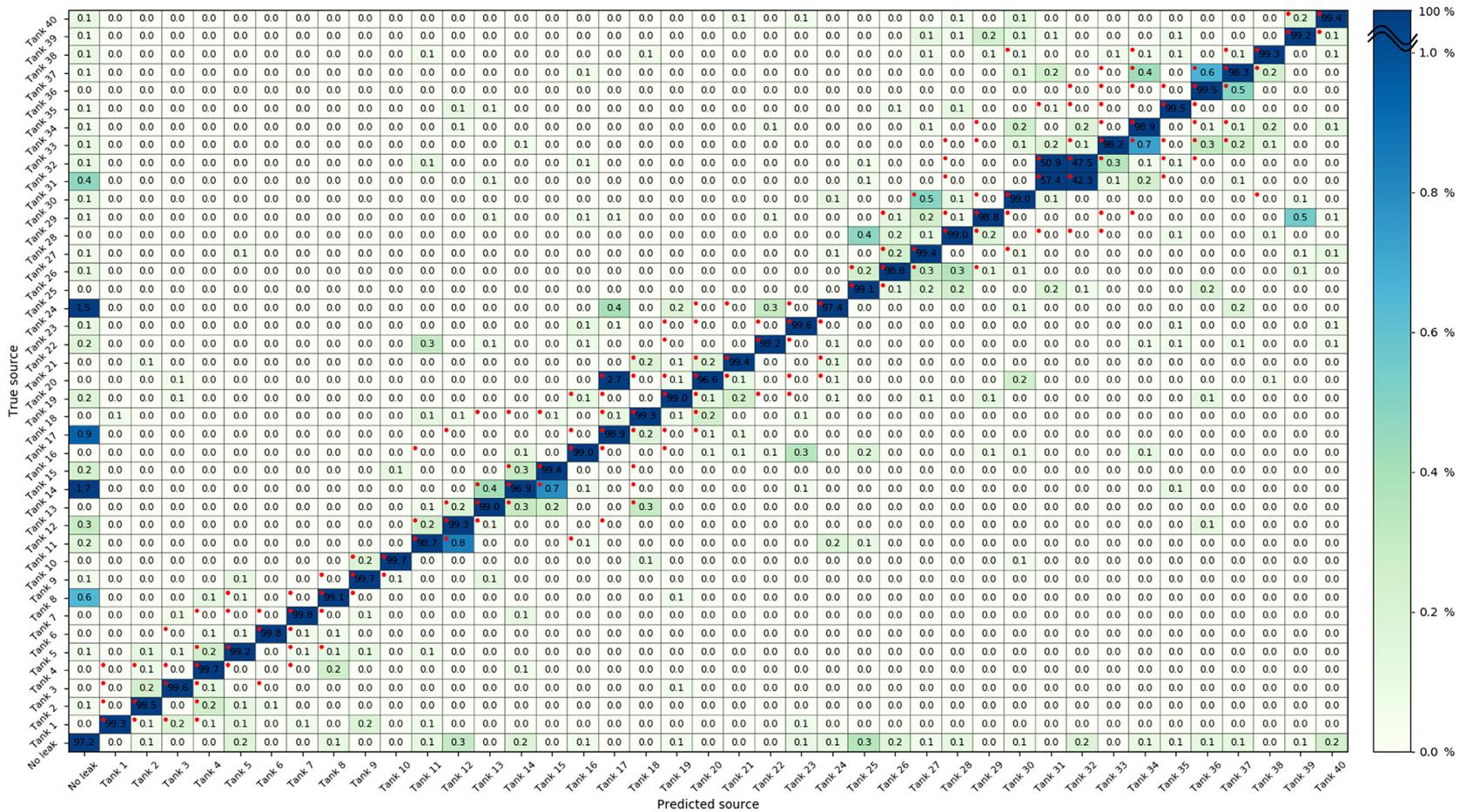


Fig. 17. Confusion matrix of LSTM-RNN-100.

6. Conclusions

In this study, to develop a source localization system for chemical leaks, FFNs and LSTM-RNNs have been trained and evaluated. As a result, it was concluded that LSTM-RNN is the most suitable model to solve the problem of leak source tracking for a real chemical plant. The proposed model predicted the leak source location in real-time with very high accuracy using the data gathered from only fixed type sensors. The proposed model overcomes the limitations of existing odor/chemical source localization models: the burden of using the expensive mobile sensor(s), the possibility of wrong point localization due to the formation of messy tracking vector, and long tracking time for trace.

40-potential leak sources were selected for the “D*” chemical plant located in Korea, and CFD simulations were carried out applying 16 wind fields per each source. Then, LSTM-RNNs were learned by the CFD simulation data. When the model was verified via the test data that never experienced, the best model predicts the leak source location with a high accuracy of 97.08%. In addition, when Top-5 points of suspected leak locations are presented, the probability of the actual leak point to be included in the five leak candidates reached 99.96%. This means that LSTM-RNNs show an improvement over 20.45% of the FFNs' accuracy on average.

Furthermore, in the worst case, even if the prediction does not diagnose the real cause of the leak, it was confirmed that a point within a radius of 10 m from the actual leak point is derived as a prediction point. This is meaningful for practical applications to emergency response because the 10 m distance is short enough for on-site safety personnel to find the actual leak point by investigation even if the predicted location is not on the leak point. Considering all these strengths, further systematization and extended applications of the proposed system are expected for deployment.

Acknowledgment

This research was supported by a grant [17IFIP-B087592-04] from Haptic-based Plant Operator Safety Training R&D Program funded by the Ministry of Land, Infrastructure and Transport of the Korean Government.

References

- Bishop, C.M., 2006. Pattern Recognition and Machine Learning, first ed Springer, New York.
- Cho, J., 2017. Placement Optimization and Reliability Analysis of Stationary and Mobile Sensors for Chemical Plant Fence Monitoring. Myongji University M.S. Thesis.
- Cho, J., Kim, H., Gebreselassie, A.L., Shin, D., 2018. Deep neural network and random forest classifier for source tracking of chemical leaks using fence monitoring data. *J. Loss Prev. Process Ind.* 56, 548–558. <https://doi.org/10.1016/j.jlp.2018.01.011>.
- Ferri, G., Caselli, E., Mattoli, V., Mondini, A., Mazzolai, B., Dario, P., 2009. SPIRAL: a novel biologically-inspired algorithm for gas/odor source localization in an indoor environment with no strong airflow. *Rob. Autom. Syst.* 57 (4), 393–402. <https://doi.org/10.1016/j.robot.2008.07.004>.
- Graves, A., Mohamed, A.R., Hinton, G., 2013. Speech recognition with deep recurrent neural networks. In: Proceedings of ICASSP, pp. 6645–6649. <https://doi.org/10.1109/ICASSP.2013.6638947>.
- Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning, first ed MIT Press, London.
- Hammerla, N.Y., Halloran, S., Ploetz, T., 2016. Deep, convolutional, and recurrent models for human activity recognition using wearables. *arXiv:1604.08880*.
- Hayes, A.T., Martinoli, A., Goodman, R.M., 2002. Distributed odor source localization. *IEEE Sens. J.* 2, 260–271. <https://doi.org/10.1109/JSEN.2002.800682>.
- Holland, O., Melhuish, C., 1996. Some adaptive movements of animats with single symmetrical sensors. In: *Proceedings of Animals to Animats*, 4, pp. 55–64.
- Ioffe, S., Szegedy, C., 2015. Batch Normalization: aAccelerating deep network training by reducing internal covariate shift, *arXiv Prepr. arXiv:1412.1441*.
- Ishida, H., Yoshikawa, K., Moriizumi, T., 2004. Three-dimensional gas-plume tracking using gas sensors and ultrasonic anemometer. In: *Proceedings of IEEE*, pp. 1175–1178. <https://doi.org/10.1109/ICSENS.2004.1426387>.
- Kim, H., Gebreselassie, A.L., Dan, S., Shin, D., 2018. Random forest classifier for real-time chemical leak source tracking using fence-monitoring sensors. *Korean J. Chem. Eng.* 35, 1231–1239. <https://doi.org/10.1007/s11814-018-0028-6>.
- Kingma, D., Ba, J., 2014. Adam: aA method for stochastic optimization, *arXiv Prepr. arXiv:1412.6980*.
- Li, J.G., Meng, Q.H., Wang, Y., Zeng, M., 2011. Odor source localization using a mobile robot in outdoor airflow environments with a particle filter algorithm. *Auton. Robots* 30, 281–292. <https://doi.org/10.1007/s10514-011-9219-2>.
- Li, W., Farrell, J.A., Carde, R.T., 2001. Tracking of fluid-adveced odor plumes: strategies inspired by insect orientation to pheromone. *Adapt. Behav.* 9, 143–170. <https://doi.org/10.1177/10597123010093003>.
- Li, Z., Rizzo, D.M., Hayden, N., 2006. Utilizing artificial neural networks to backtrack source location. In: *Proceedings of 3rd International Congress on Environmental Modeling and Software*, pp. 1–6.
- Lochmatter, T., Raemy, X., Matthey, L., Indra, S., Martinoli, A., 2008. A comparison of casting and spiraling algorithms for odor source localization in laminar flow. In: *IEEE International Conference on Robotics and Automation*, pp. 1138–1143. <https://doi.org/10.1109/ROBOT.2008.4543357>.
- Marques, L., Almeida, N., de Almeida, A.T., 2003. Olfactory sensory system for odour-plume tracking and localization. In: *Proceedings of IEEE*, 1, pp. 418–423. <https://doi.org/10.1109/ICSENS.2003.1278971>.
- Neumann, P.P., Bennetts, V.H., Lilienthal, A.J., Bartholmai, M., Schiller, J.H., 2013. Gas source localization with a micro-drone using bio-inspired and particle filter-based algorithms. *Adv. Robot.* 27, 725–738. <https://doi.org/10.1080/01691864.2013.779052>.
- Niu, H., Reeves, E., Gerstoft, P., 2017. Source localization in an ocean waveguide using supervised machine learning. *J. Acoust. Soc. Am.* 142, 1176–1188. <https://doi.org/1176-1188> <https://doi.org/10.1121/1.5000165>.
- Olah, C. Understanding LSTM networks (accessed 10 November 2017).
- Rawal, A., Miikkulainen, R., 2016. Evolving deep LSTM-based memory networks using an information maximization objective. In: *Proceedings of 2016 on Genetic and Evolutionary Computation Conference*, pp. 501–508. <https://doi.org/10.1145/2908812.2909841>.
- Russel, R.A., 2004a. Robotic location of underground chemical sources. *Robotica* 22, 109–115. <https://doi.org/10.1017/S026357470300540X>.
- Russel, R.A., 2004b. Locating underground chemical sources by tracking chemical gradients in 3 dimensions. In: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1, pp. 325–330. <https://doi.org/10.1109/IROS.2004.1389372>.
- Shu, Y., Zhao, J., 2014. A simplified Markov-based approach for safety integrity level verification. *J. Loss Prev. Process Ind.* 29, 262–266. <https://doi.org/10.1016/j.jlp.2014.03.013>.
- Singh, R.M., Datta, B., Jain, A., 2004. Identification of Unknown Groundwater Pollution Sources using Artificial Neural Networks. *J. Water Resour. Plan. Manage.* 130 (6), 506–514. [https://doi.org/10.1061/\(ASCE\)0733-9496\(2004\)130:6\(506\)](https://doi.org/10.1061/(ASCE)0733-9496(2004)130:6(506)).
- Venkatasubramanian, V., Chan, K., 1989. A neural network methodology for process fault diagnosis. *AIChE J.* 35, 1993–2002. <https://doi.org/10.1002/aic.690351210>.
- Wang, B., Chen, B., Zhao, J., 2015. The real-time estimation of hazardous gas dispersion by the integration of gas detectors, neural network and gas dispersion models. *J. Hazard. Mater.* 300, 433–442. <https://doi.org/10.1016/j.jhazmat.2015.07.028>.
- Zhang, Z., Zhao, J., 2017. A deep belief network based fault diagnosis model for complex chemical processes. *Comput. Chem. Eng.* 107, 395–407. <https://doi.org/10.1016/j.compchemeng.2017.02.041>.