Korean J. Chem. Eng., 35(6), 1231-1239 (2018)
DOI: 10.1007/s11814-018-0028-6

pISSN: 0256-1115
eISSN: 1975-7220

Random forest classifier for real-time chemical leak source tracking using fence-monitoring sensors

Hyunseung Kim*, Addis Lulu Gebreselassie*, Seungkyu Dan, and Dongil Shin*,†**

*Department of Chemical Engineering, Myongji University, Yongin, Gyeonggi-do 17058, Korea
**Korean Gas Corporation, Ansan, Gyeonggi-do 15328, Korea
(*Received 4 August 2017 • accepted 9 February 2018*)

Abstract−Fast and reliable diagnosis of chemical leak and leak location(s) can save lives and reduce the damage from chemical accidents by enabling quick response. This paper presents a method that uses random forest (RF) classifier to track the location of chemical leak in real-time. A set of big data of leak accidents, which is needed to learn the RF classifier, is extracted by performing massive CFD simulations on a real chemical plant. The RF model is designed with optimal parameters and descriptors through parameter effect experiment. Feature ranking is also implemented to eliminate unnecessary attributes from the dataset. Using the pre-processed data, the optimal RF model achieved a test accuracy of 86.9% for the classification problem of predicting the leak location among 40-potential leak sources in the plant. Furthermore, when analyzing prediction errors by visualizing the classification boundary of RF model, most of the prediction errors are confirmed to be misclassification of adjacent leak locations. Considering the high prediction accuracy of the RF model, the RF-based leak source tracking model is expected to be effectively applied to industrial leak accidents.

Keywords: Chemical Leak Accident, Source Tracking, Inverse Problem, Random Forest, Artificial Intelligence

INTRODUCTION

The 1984 Bhopal accident in India, called the worst chemical accident in human history, was caused by the release of about 42 tons of methyl isocyanate from storage tanks [1]. The leak accident killed 3,800 people at the time of the accident [2], and made 30,000-40,000 people permanently disabled [3]. In fact, two weeks before the Bhopal disaster, an LPG tank rupture occurred in San Juanico killed more than 500 people and injured more than 5,000 people [4]. Although many chemical safety agencies, regulations, and researches have been released since then [5], accidents involving casualties have been occurring continuously: Texas, 1989, polyethylene leakage, 23 killed; Longford, 1998, hydrocarbon leakage, 2 killed; Gumi, 2012, hydrofluoric acid leakage, 5 killed.

Since the possibility of a chemical leak cannot be completely eliminated, techniques that can mitigate the damage of leak accidents are needed. In particular, the techniques for initial response to chemical leak are very important because leak accidents that are not responded to quickly and appropriately can lead to additional accidents such as fire explosions. If there is a technique that can track the location of chemical leak as soon as possible, the field safety personnel can respond quickly and appropriately and reduce the damage.

Thereby, reliable models to track leak source location(s) have been actively researched for many decades. However, because of the number of unknown variables involved and arbitrary variations in wind speed and direction, reliable leak source tracking model development is still a challenging problem. The common approach to solve leak source tracking problems is the inverse vector method. Ishida et al. [6] proposed a method using movable sensors to update the measured leak concentration in different positions and create a vector data to find the leak point. It's a method of deriving the leak source tracking vector by combining the inverse wind vector along with the concentration field vector of the leaked chemical. Pisano and Lawrence [7], and Zhen and Chen [8] used gradient descent and optimization to find a vector to the source location using measured leak concentration from movable sensor. While these methods have shown some success, they have required high cost and long tracking time because the use of mobile sensors is essential. Also, in a complex terrain structure, the sensors can read outraged concentration which can change the direction of the inverse vector in a misleading direction. To avoid these problems, machine learning algorithm has been implemented to solve leak source tracking problem in this paper.

Along with the development of artificial intelligence (AI) applications, several researchers are applying machine learning algorithms to come up with reliable leak source tracking models. In this paper, we present leak source tracking model applying the random forest algorithm. We present the chemical leak source tracking problem as a classification problem and apply the random forest algorithm, an effective tool in prediction or general purpose of classification.

LEAK SOURCE TRACKING MODEL USING FENCE MONITORING SENSOR DATA

The source tracking model of chemical leak that developed in

†To whom correspondence should be addressed.
E-mail: dongil@mju.ac.kr

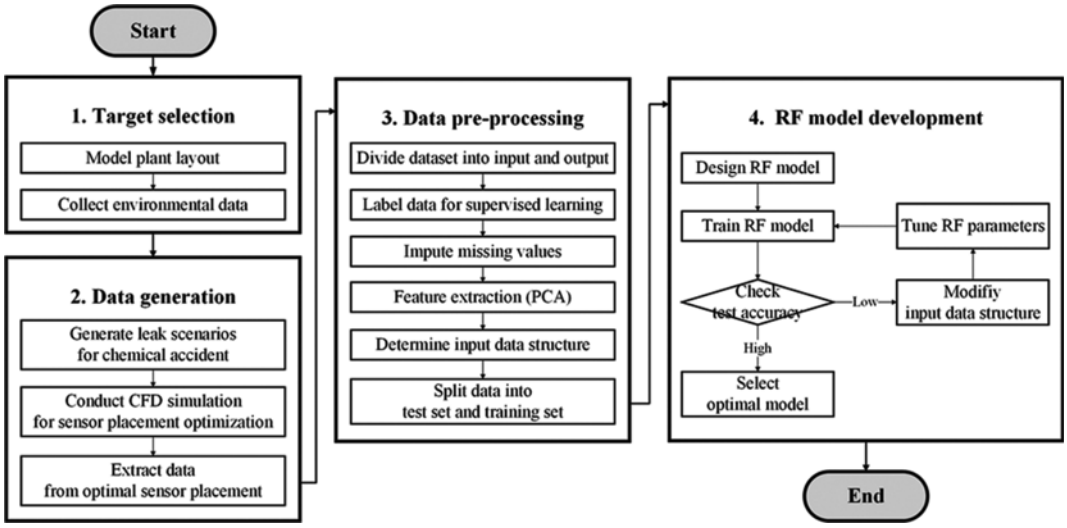

Fig. 1. Flow chart of the CFD data generation and proposed RF model.

this study is to be able to predict leak location by learning fence monitoring data in RF classifiers. The flow chart to develop the model is shown in Fig. 1. The proposed method (step 3 to 4 on Fig. 1) was developed based on the CFD simulation (step 1 to 2 on Fig. 1). The result of the CFD simulation, which was conducted by Cho [9], is used to learn the RF classifier. The data generation process based on the CFD simulation is described in Section 3, and the RF model development process is described in Section 4.

CFD SIMULATION OF CHEMICAL LEAK SCENARIOS

In any kind of machine learning based model, a reliable and huge dataset is important for the quality of the prediction or classification. Since, the objective of our RF based model is to predict a chemical leak location based on chemical leak concentration data, wind direction, wind speed, sensors physical coordinates, etc. Theoretically, the best way to generate these data is to perform an actual leak scenario on a real plant and collect the information from pre-installed fence monitoring sensors. However, considering the accident risk of the experiment and the cost related to it, CFD simulation was used to generate the data. A CFD simulation, which targeted D chemical plant in Korea Y industrial complex, was performed.

COMSOL Multiphysics 5.0 was used in 2D-horizontal plane CFD simulation. 40-Storage tanks were selected as potential leak candidates and were set to leak at circumference of each tank; spatial features such as release height, orientation and direction were not considered. 16-wind directions were applied to each of 40-storage tanks to generate accident scenarios. In other words, a total of 640 leak scenarios were generated. In all scenarios, toluene in the gaseous state was set to be leaked, and the leak rate was set at 3.69 kg/m^3/s (293 K, 1 atm). 0 to 750-seconds real time simulation were done for each scenario and the leak occurred at 100 seconds. The internal area of the chemical plant is 57,820 m^2 and it is filled with 23,603 elements for finite element method (FEM) solving; this means that one element per 1.5 m is arranged on average. With dual Intel Xeon CPUs with 2.6 GHz, a total of 12 cores, it took about 30 days to run 640 scenarios.

The concentration of toluene was recorded from the 11 sensors. After the concentration data was collected, since we were going to apply supervised learning, the dataset was labeled from 0 to 40/41.

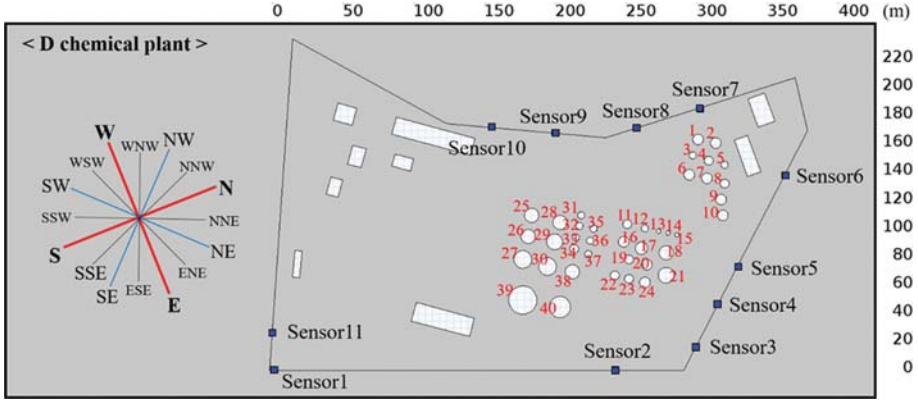

Fig. 2. Locations of labeled leak sources and optimal placement of sensors on D chemical plant.

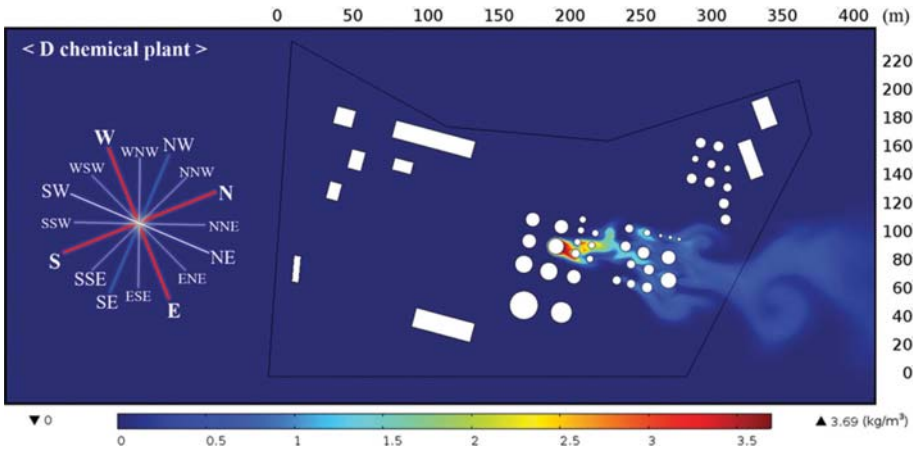

Fig. 3. CFD simulation: Concentration field of leaked toluene for source 29, wind direction NNE.

The label 0 represents non-release (no leak) case and the label 1 to 40 represents the 40 possible leak source locations in the chemical plant.

APPLICATION AND DESIGN OF RANDOM FOREST CLASSIFIER MODEL

1. Basics on Random Forest

Random forest (RF) classification is an ensemble learning method for classification and other tasks. It is operated by constructing a multitude of decision trees using training data and outputting the results. RF can be applied to two kinds of problems: classification and mean prediction such as regression [10].

The logic behind the RF algorithm is by maximum voting from each tree. Better prediction can be achieved than a single tree decision. The method takes a random sample of the data and recognizes a key set of attributes to grow decision trees.

Using misclassification or out of bag (OOB) error rate, we can determine how often the classifier gives false predictions. The misclassification error rate can be determined using confusion matrix. The built decision trees have their OOB error rate. OOB error rate measures how accurate the built decision tree predicts. The trees with lower OOB error rate (high prediction accuracy) are collected to form the forest. The prediction from each tree is averaged to get a prediction with high accuracy.

Once the forest is trained, it can be used to make predictions for new unlabeled data points. But to make these predictions as accurate as possible, the classifier parameters must be tuned. These RF parameters in scikit-learn library [11], Python include the following:

1-1. Number of Estimators

The total number of trees which will be built before taking the maximum voting of averages of predictions. Theoretically, except slowing down the processing speed of the computer where the RF model is running and for some case, the higher the number of estimators, the better for the prediction accuracy.

1-2. Bagging

Bagging (Bootstrap+aggregating) is a random sampling of the dataset with replacement. For a standard training set X of size n, bagging generates m new training sets X_i, each of size n_i, where $n_i<n$ by sampling from X uniformly and with replacement. Since its sampling with replacement, some observation data could be repeated in each X_i. By applying K iterations of bagging it creates total K number of trees.

1-3. Attribute Bagging

After the creation of K number of trees, the algorithm applies attribute bagging (random subspace creation), which is selecting the best feature for each K number of trees. Among extracted random subspaces, it applies attribute bagging and trains the decision tree with the variable from any new node with the least misclassification error.

1-4. Maximum Feature

The number of features to consider when looking for the best split.

In Python, the two best RF libraries are the open-source scikit-learn and the closed-source wise RF. Both libraries are fast and reliable with a package of more than 90% of what is needed for statistical and machine learning tasks. wiseRF has slightly fast computation time than scikit-learn. These libraries use only CPU. The GPU usage RF library is called CudaTree, which is 2-6 times faster than scikit-learn [12]. Even though CudaTree is faster than scikit-learn, it cannot handle too large datasets. Thereby, the open-source scikit-lean library has been used.

2. Leak Source Tracking Random Forest Model Design

This study presents four random forest models to solve the leak source tracking problem. To find the maximum achievable accuracy, each model was designed with either different input structure or different RF parameter. In all models, the implemented general flow of the RF code is shown in Fig. 1 and Fig. 4.

2-1. Model 1

The data used for this model consists of 13 features (attributes):

- Feature 1: Wind velocity determines how fast the leak is dispersing and how close the leak location is to the sensors. When wind velocity is used with wind direction in RF model, it creates a reliable decision to determine the leak point. Thus, wind velocity has been taken as one feature.
- Feature 2: Information of the wind direction at the time of the leak plays a big role in having an initial guess of where the leak

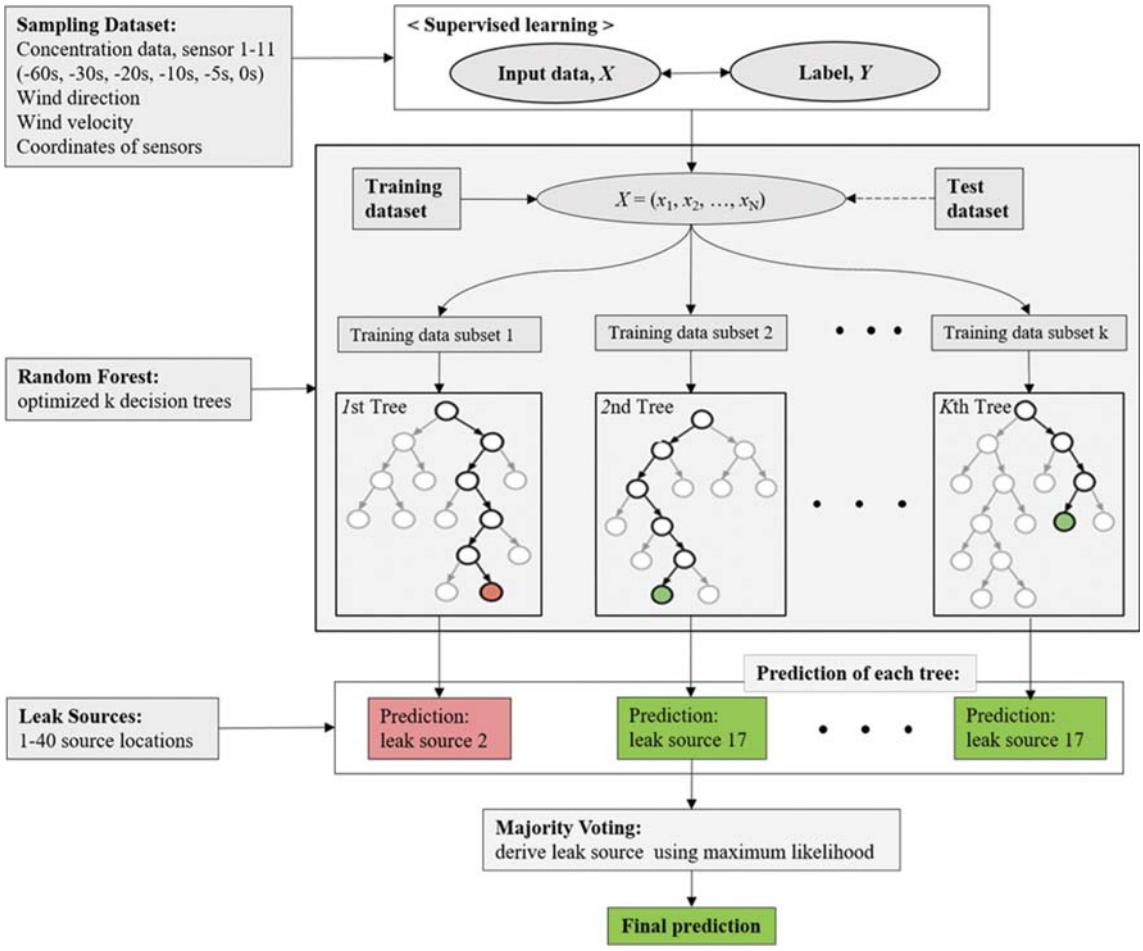

Fig. 4. Schematic diagram of Random Forest classification method for predicting leak source locations.

starts. By taking the inverse wind direction vector, one can assume to reach the leak point. This holds true only when there is no physical object between the leak source location and sensor location. The physical object between leak source location and sensor location diverges the direction of the leak concentration vector to different directions. Thus, the wind direction has been taken as one feature.

- Feature 3 to 13: When the chemical leak occurs, 11 sensors which are optimally placed on the fence of the plant detect the leak. Leak detection amount and the time to detect the leak is different from sensor to sensor. Even though uniform sensors are placed on the fence, the wind direction, the initial leak source location and X, Y coordinates of the sensors relative to the leak point are the factors for varying concentration reading. Thus, the concentration data from the 11 sensors have been taken as 11 different features.

If the maximum concentration of 11 sensors is above ERPG-2 of toluene, the data is labeled with the digit of the storage tanks shown in Fig. 1. The data was split to 7-to-3 ratio for training and test dataset. On this model, 50 trees were used in the forest with gini criterion (data split method). The number of trees and other parameters will be tuned on the next models. After training the model, when we tested and observed the model prediction, 41.96% accuracy was achieved.

2-2. Model 2

On this model, similar data structure and RF parameters were used as of Model 1. The only difference was that one more class was added on the label. Instead of using only 40 classes, an additional class, which was a no-leak-detection data, was included. The

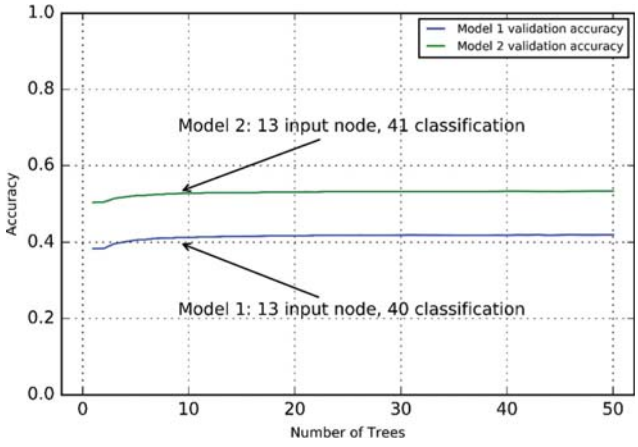

Fig. 5. Accuracy comparison of Model 1 and Model 2.

Table 1. Effect of Class 0 on the total prediction accuracy

Top 3 prediction accuracy			Least 3 prediction accuracy		
Class 0	Class 5	Class 6	Class 32	Class 31	Class 34
99.04%	64.24%	59.90%	9.35%	10.44%	26.43%
Average of prediction accuracy of Class 1 to Class 40: 41.35%					

no-leak-detection data was labeled as 0.

Because of the modification, the prediction accuracy of Model 2 increased to 53.25% as shown in Fig. 5. Even though the total accuracy of the model increased, it is hard to say that this is a good model. When we analyze the accuracy per individual class, the accuracy of 0 labeled class was 99.04%, where the average prediction accuracy of the other classed was 41.35%. Table 1 shows the effect of Class 0 on the total prediction accuracy. Class 0 is easy to predict on the RF model since it is quite logical to describe a decision where most scenarios are classified into the no-leak-detection category.

2-3. Model 3

In this model, class 0 was excluded based on the result observation of model 2 (it created unbalanced dataset representation). In addition, it was quite important to add other features since considering the 40-classification problem, 13 features were not good enough to get higher prediction accuracy. This model consists of a total of 90 features including the features stated for the previous models. The additional features are the following;

- Features 14 to 68: The 11-sensor's concentration measurement should be inputted in the RF model as periodic concentration detecting pattern. This is because the wind field is deformed over time by various structures in the chemical plant. Thus, past-time concentration measurement data was used to clearly define and represent the leak plume distribution. The 14 to 68 features are the representation of the measured concentration at −5 sec, −10 sec, −20 sec, −30 sec, and −60 sec.
- Features 69 to 90: Assuming the X, Y coordinates of the position of the sensor on the fence might create further decisions on the individual decision trees that help to classify the leak points more accurately, X, Y coordinates were added as an additional feature.

When the result of model 3 was analyzed, as shown in Fig. 6, the prediction accuracy increased to 85.99%. The additional features clearly improved the prediction accuracy. Especially, defining the data structure based on past-time concentration measurement played a key role in the accuracy improvement.

2-4. Model 4

On models 1 to 3, only the input data structure was modified. In this model, the RF parameters discussed in section 4.1 were tuned and the best RF parameters were selected by experimenting the parameter effects on the model. The parameter effect experiment was effective in enhancing the prediction accuracy. Using feature ranking, unnecessary features were also eliminated from the data set.

RESULTS AND DISCUSSION

Model 1, which was designed with 13 features and 40 labeled data (40-class), gave a prediction accuracy of 41.96%. To improve the accuracy of model 1, model 2 was designed by adding one more class, which is the no leak release class. Although model 2 gave a higher prediction accuracy over model 1 (53.25% prediction accuracy), after analyzing each leak source location predictions of model 2, it was noticed that the accuracy improvement only occurred due to the additional class added. The additional class created unbalanced data representation, and it made that the prediction accuracy cannot represent the whole model (leak locations).

Model 3, which was designed with 90 features and 40-class, gave a prediction accuracy of 85.99%. This improvement of prediction accuracy was achieved by adding important features like past time leak data to the model, which can fully represent the chemical leak accident scenario.

Model 4 gave the maximum accuracy when the optimal tuned parameters are being used: 400 trees, 20% maximum-feature, and 1 minimum-sample-leaf. The data set collected from CFD simulation was also filtered using feature ranking to eliminate unnecessary attributes from the dataset. As a result, model 4 predicts the 40-classification problem with 86.85% accuracy. In section 5.1, the effects of parameter tuning on the prediction accuracy of model 4

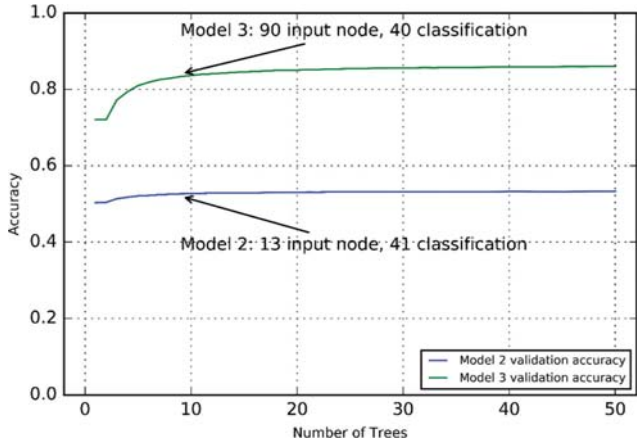

Fig. 6. Accuracy comparison of Model 2 and Model 3.

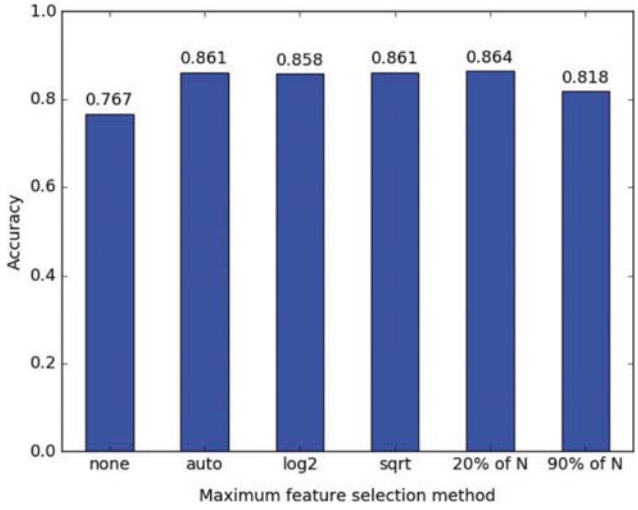

Fig. 7. Effect of maximum feature on the test accuracy.

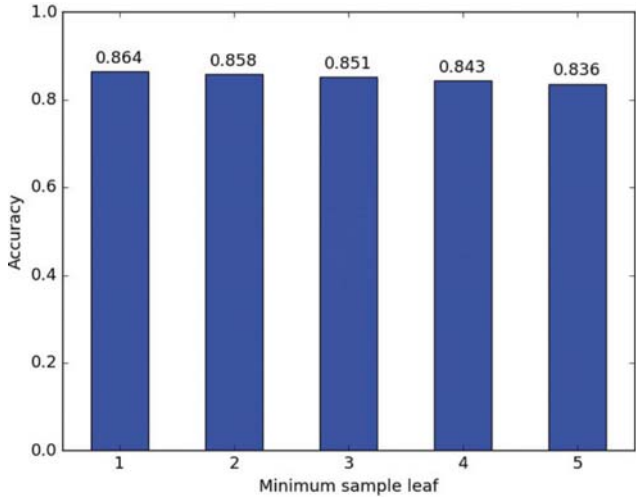

Fig. 8. Effect of minimum sample leaf on the test accuracy.

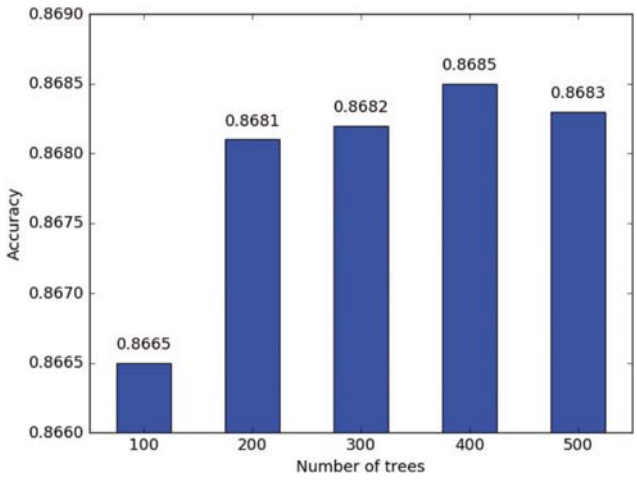

Fig. 10. Model 4: Maximum 86.85% accuracy at 400 trees in the forest.

are discussed.

1. Optimal Parameter and Descriptor

1-1. Maximum Feature

On scikit-learn library, Python, there are four options of maximum feature selection: 'auto', 'None', 'log2', and 'sqrt'. These options determine how many features are to be used for each bootstrap sample to grow each tree. For instance, if 'sqrt'/'auto' is selected, the RF classifier takes $\sqrt{N}$ features to build the individual trees, where N is total number of features. The user can also define the percentage of the total number of features to be used on the attribute bagging stage. In the case of model 4, 20% of N showed the highest prediction accuracy.

1-2. Minimum Sample Leaf

Depending on the data and the prediction/classification model, how far to split the data (on each decision tree) has a huge impact on the final prediction/classification accuracy.

1-3. Feature Ranking

Even though feature ranking is not a parameter to tune, it is important to screen the features based on their importance in the model. Using a feature which has no relevance to the model may reduce the prediction accuracy of the model. Among the 90 features used in model 3, features 69 to 90 did not have any relevance in the final prediction. These features are the X, Y coordinates of the position of the sensor on the fence. Consequently, features 69 to 90 were eliminated from the data set for model 4.

1-4. Number of Trees (Number of Estimators)

More trees in the forest is good for the model only if it is not overfitting the data. As shown in Fig. 10, after the number of trees reached 400, the prediction accuracy started to drop; this case showed that after 400 trees in the forest, the model is overfitting the data. Except for grid search and engineering judgment, there is no general guide to choose the number of trees in the forest.

2. Misclassification Error Analysis

The source of the 13.15% total prediction error rate of the RF model was analyzed to describe its practical applicability. As shown in Fig. 11, the least leak source prediction occurred on sources 31

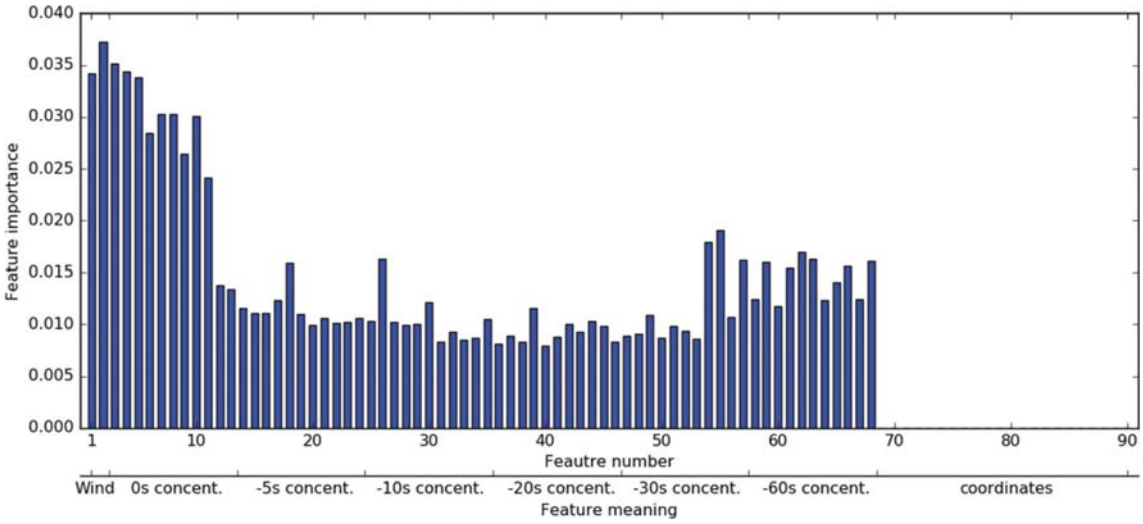

Fig. 9. Feature ranking.

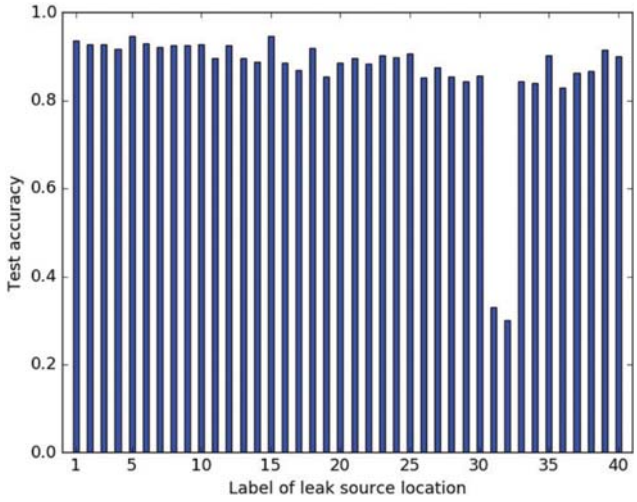

Fig. 11. Prediction accuracy per leak source location.

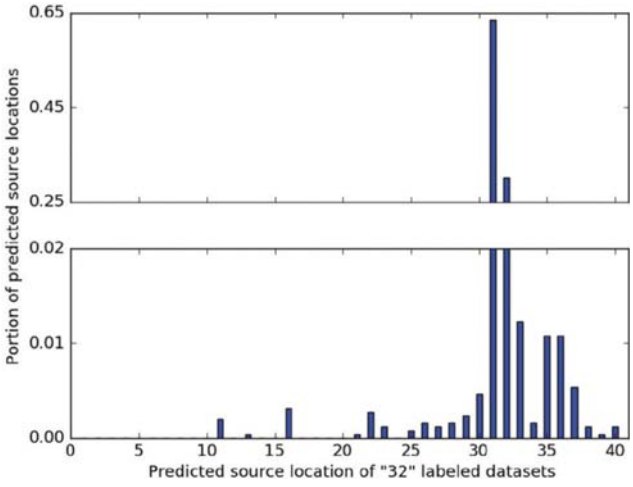

Fig. 12. Sources of leak source '32' prediction error.

to 36. These locations are physically very small and complex relative to the other leak source locations, which makes the RF model predict wrong leak sources around those areas.

Even though the prediction accuracy was low on the small and complex leak source locations, as shown in Fig. 12, most of the prediction errors arose from predicting the leak source locations right next to the true leak location. If 0 penalty is set to consider the mis-predicted leak sources which are physically next to the true leak location, the total prediction accuracy can get close to 100%.

The misclassification error can be better visualized by plotting the decision surfaces learned by the RF model. However, if all the 40-classes are used to show the classification boundaries, it can be ambiguous to understand the clear decision boundaries as shown

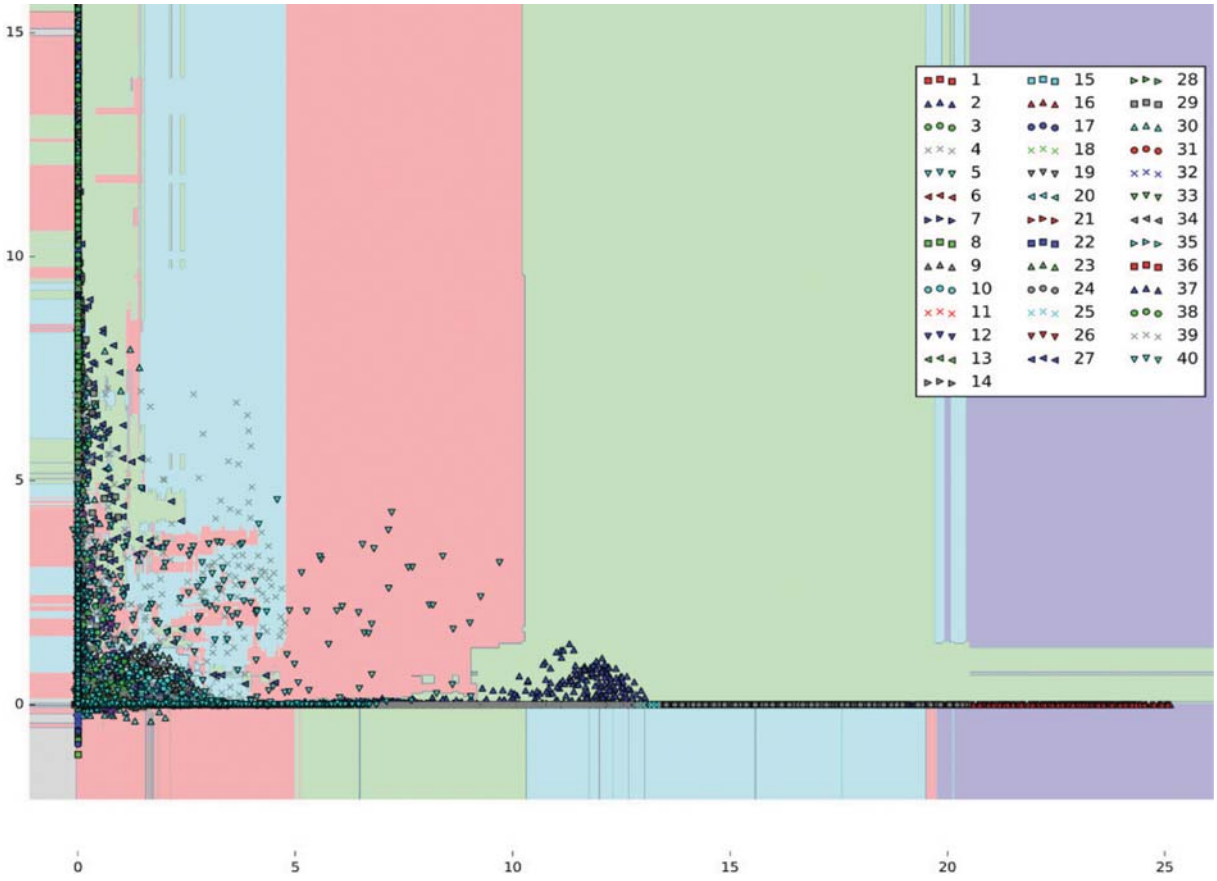

Fig. 13. 40-Class classification boundaries on feature subset of the leak simulation dataset of the RF model.

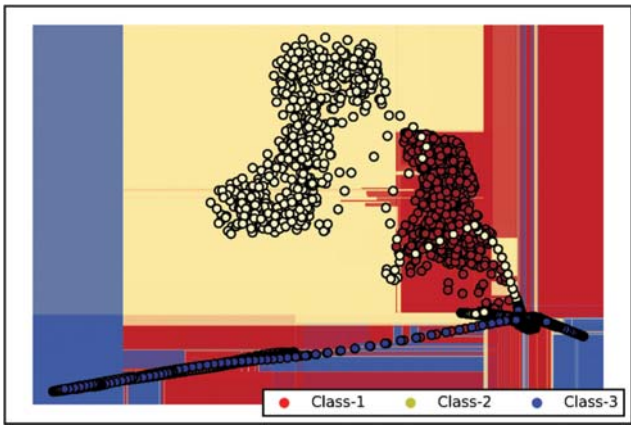

Fig. 14. (Best predicted 3-class) classification boundary on feature subset of the leak simulation dataset of the RF model.

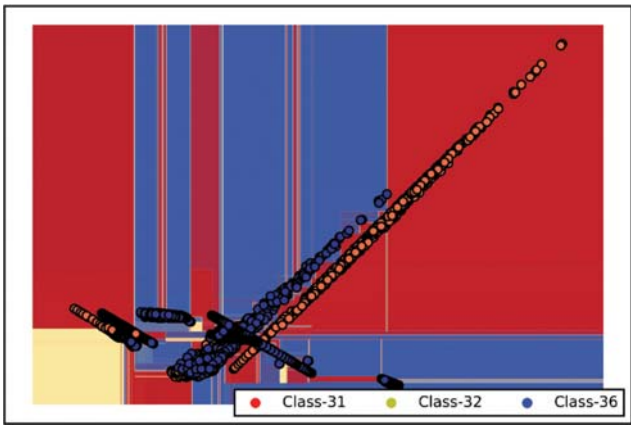

Fig. 15. (Least predicted 3-class) classification boundary on feature subset of the leak simulation dataset of the RF model.

in Fig. 13. For clarity purpose first, we reduced the 40-classification problem into two sets of 3-classification problems. The first 3 class represents those locations in which the RF model can easily learn and classify; these leak locations are physically large and far from each other. Fig. 14 shows the decision surfaces of the classification of class 1, class 2, and class 3. We can see that the decision boundaries are clear and the data points are separated unambiguously since the leak locations are easily differentiable by the RF model. The data points which can be seen within unmatched background color are the misclassified points.

The second set of the 3-classification problem represents the leak locations which are physically small and very compact to each other; on these areas, the RF model showed the lowest prediction accuracy. Fig. 15 shows the decision surfaces of the classification of class 31, class 32, and class 36. We can observe that the decision boundaries are very compact and overlapping data points are quite large.

CONCLUSIONS

This study presented an innovative application of RF on solving the challenging leak-source tracking problem. The RF model handled the huge variance and noise in the dataset very well and since it is ensembles of trees, high prediction accuracy was achieved. Feature ranking was also implemented to eliminate unnecessary attributes from the dataset. Features which are irrelevant to the model make the model learn very slowly and may even reduce its prediction accuracy; after analyzing the feature ranking, the X, Y coordinates were eliminated from the dataset due to their 0 relevance to the model.

Using the CFD simulation data as an input to train the RF model to predict the 40-classification problem, 86.9% prediction accuracy was achieved when the RF model was tuned to 400 number of trees, 20% maximum-feature, and 1 minimum-sample-leaf. The 13.15% misclassification error was confirmed as no failures but nearmisses due to mis-prediction of nearby leak locations, which are physically close and present in a compact manner. In real chemical leak accidents, our proposed model can be used quite effectively by setting a physical boundary range around the predicted leak locations.

Application of RF as a method of diagnosing unknown leak-source locations has advantages over inverse vector method. The proposed method uses fixed fence-monitoring sensors to detect the chemical leak. This application can reduce the cost and calculation difficulty and errors that come with inverse vector method and movable sensors to detect leak source location: following the inverse wind direction to find the leak source location is not always reliable; complex geometric structures can divert the wind direction causing the leak to be detected on the sensor which is not aligned with the inverse wind direction vector. Thus, the proposed method would be easily applied for tracking real chemical leak with low cost and high reliability.

ACKNOWLEDGEMENTS

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 Korean Government. And this work was also supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2017R1E1A2A01079660).

REFERENCES

1. T. R. Chouhan, *J. Loss Prevention in the Process Industries*, **18**, 205 (2004).
2. E. Broughton, *Environ. Health*, **4**, 1 (2005).
3. I. Eckerman, *The Bhopal saga: Causes and consequences of the world's largest industrial disaster*, Universities Press, Hyderabad, India (2005).
4. G. Arturson, *Burns*, **13**, 87 (1987).
5. J. Park, Y. Lee, Y. Yoon, S. Kim and I. Moon, *Korean J. Chem. Eng.*, **28**, 2110 (2011).
6. H. Ishida, K. Yoshikawa and T. Moriizumi, Three-dimensional gas-plume tracking using gas sensors and ultrasonic anemometer, *Proceedings of IEEE Sensors 2004*, Vienna, Austria (2004).
7. W. J. Pisano and D. A. Lawrence, Data Dependant Motion Planning for UAV Plume Localization, *Proceedings of the AIAA Guidance, Navigation and Control Conference*, Hilton Head, SC, U.S.A. (2007).
8. X. Zheng and Z. Chen, *J. Loss Prevention in the Process Industries*,

24, 293 (2011).
9. J. Cho, *Placement Optimization and Reliability Analysis of Stationary and Mobile Sensors for Chemical Plant Fence Monitoring*, M.S. Thesis, Myongji University, Yongin, Korea (2017).
10. L. Breiman, *Machine Learning*, **45**(5), 5 (2001).
11. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, *J. Machine Learning Res.*, **12**, 2825 (2011).
12. A. Rubinsteyn, *Training Random Forests in Python using the GPU*, http://blog.explainmydata.com/2013/10/training-random-forests-in-python-using.html, Accessed on 15 Jul. 2017 (2013).