A Project Report

On

LEAK DETECTION USING MACHINE LEARNING

BY

P V SRI HARSHA

2019A2PS1521H

Under the supervision of

Prof. A Vasan

SUBMITTED IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS OF

CE F377: DESIGN ORIENTED PROJECT

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI (RAJASTHAN)

HYDERABAD CAMPUS

(May 2022)

ACKNOWLEDGEMENT

I would like to thank Prof. A Vasan for providing this opportunity to work under his guidance. The value and time brought in was immeasurable and was of great help. I am indebted for all his help and guidance throughout the project.

Birla Institute of Technology and Science-Pilani,

Hyderabad Campus

Certificate

This is to certify that the project report entitled "**LEAK DETECTION USING MACHINE LEARNING**"

submitted by Mr. P V Sri Harsha (ID No. 2019A2PS1521H) in partial fulfillment of the requirements of the course CE F377, Design Project Course, embodies the work done by him under my supervision and guidance.

Date: 25/4/2022

(Prof. A Vasan)

BITS- Pilani, Hyderabad Campus

1 ABSTRACT

Water is a mandatory and daily requirement for life, health and economic development all around the world. Since the population is steadily increasing all around the world people expect 24-hour water supply to their homes and offices. And since water is so valuable to everyone, it is critical that it be easily available and of the highest quality. While designing the pipelines or the water distribution system of a city, we need to keep in mind the expected population of the city in the coming years, available water bodies near the city, and many other crucial factors. All these factors help in deciding the water supply system of that city, which include the pipe diameters, lengths, etc.

The water supply system loses a substantial volume of water. Water leaks have been a big issue in many parts of the world. Water loss due to leakages in the supply network reaches 40% of the water in some nations' supply systems. Water leakage reduction is a key priority for many nations throughout the world since it reduces the amount of money and energy spent on generating and pumping water, as well as ensuring consumer satisfaction through better system dependability. There are many ways of detecting water leaks which require costly equipment and time. This may not be possible all the time. Machine Learning or Deep Learning techniques use the available data to predict the outcomes. This works well with Leak Detection because we can have sensors in the pipelines and collect the data from it. This data can then be used to predict the leaks in the pipeline network. This is very cost efficient and time efficient way than manual procedures.

This report tries to find out efficient ways of detecting the leaks in the water distribution systems using Machine Learning and Deep Learning techniques.

2 CONTENTS

1 Abstract 4
3 Introduction 6
4 Problem Definition: 6
4.1 Problem Scope: 6
4.2 Literature Review 6
4.2.1 Vision Utilization 6
4.2.2 Acoustic Emission 6
4.2.3 IR Thermography 7
4.2.4 Ground Penetrating Radar 7
5 About the dataset: 7
6 Algorithms and Metrics 8
6.1 Decision Trees 8
6.2 Random Forest 8
6.3 Convolutional Neural Network (CNN) 8
6.4 RandomSearchCV 9
6.5 PCA (Principal Component Analysis) 9
6.6 True Positive Rate (TPR) 9
6.7 False Positive Rate (FPR) 9
6.8 AUC-ROC Curve 9
6.9 Accuracy 9
7 Design Description: 10
7.1 Reshaping 10
7.2 Scaling 10
7.3 PCA 10
8 Evaluation 10
8.1 Overview 10
8.2 Prototype 11
8.3 Intuition for the CNN model: 11
8.4 Testing and Results 12
8.5 Leak Localization: 14
9 Conclusions 15
10 References 16

3 INTRODUCTION

Water is an important source of life on our world; it is essential for agriculture, manufacturing, electricity production, and keeping humans healthy. Approximately one billion people throughout the world do not have access to safe drinking water. Underground pipes are used to transport urban water on a regular basis. Water transmission pipes lose 20% to 30% of the water that passes through them on a regular basis, with losses exceeding 50% in older systems, particularly those that have suffered from inadequate maintenance. Water loss in transmission pipelines can be caused by a variety of factors, including leaks, metering problems, public use (such as firefighting), and theft.

Leakage can be considered as the amount of potable water lost from a supply source in transmission and distribution. High pressure causes the increasing of leak, and the loss of consuming water and the number of the accidents in the network. On the other hand, low pressure in the network causes the inability of complete supplying water or result in providing unsuitable water. On 29th May 2017, a sudden pipeline burst occurred in the WDS of Ukraine, which gathered attention across the world. The pipeline burst due to high pressure and left a 6 m hole into the ground. The burst also caused damage to the road, cars and houses, thus also causing economic losses. Similar incidence of pipeline bursts due to high operating pressure was also reported in Melbourne, Australia in the year 2012. Because of these leakages, extra water has to be pumped out, which increases energy consumption, consequently, causing economic losses

4 PROBLEM DEFINITION

4.1 PROBLEM SCOPE

The problem statement for this project is to correctly classify a scenario as Leaky or Non-Leaky. This requires using classification algorithms. Many popular classification algorithms are Logistic Regression, K-Means Classifier, Decision Tree models and many other Neural Network classifiers. Any classifier that can be used in this project needs to be able to find whether the given Scenario has a leak and also be able to find the correct value of the leak node.

4.2 LITERATURE REVIEW

Leak losses and damages demanded creative tactics and ways to reduce their negative impact and respond as quickly and smoothly as possible. As a result, a number of academics have focused their efforts on the development of a wide range of leak detection and leak locating approaches. Some of the well-known methods of leak detection are:

4.2.1 Vision Utilization

The most basic and perhaps ineffective passive leak detection approach is to look for any signs of ponding at the ground surface or unusual plant growth that might indicate a leaky pipe.

4.2.2 Acoustic Emission

The propagation of elastic waves emitted from an active source is used in the acoustic emission technique. The acoustics emission approach is a good contender because departing liquid generates an acoustic signal as it travels through a leak. When a leak develops, an acoustic signal travels through the pipeline and is picked up by the acoustic sensors positioned along the pipeline; if the

received signal by one sensor is stronger and has a greater magnitude, the breach may be readily located.

4.2.3 IR Thermography

Infrared thermography, on the other hand, depends on the detection of IR radiation over spatial surfaces and energy transfer theory to detect temperature abnormalities in a variety of applications. Because water leaks generate temperature differences around the leak, an infrared camera may detect leaks by collecting the thermal profile of the surface above the pipeline.

4.2.4 Ground Penetrating Radar

The concept of operation for a ground penetrating radar is to obtain a picture of the pipe underneath, where the electromagnetic fingerprints of leak zones present themselves. A simple interpretation of the photos can then be used to determine the source of a leak. The type of soil in which the pipes are buried, however, has an impact on this detecting approach. A study found that reflections under the leak zones are weaker than the surrounding soil medium in most homogeneous soil, and that void phenomena may not be visible in most inhomogeneous soil.

There are several other techniques that are used for the leak detection as well. All these techniques are useful but also have their own limitations. Many of them need highly calibrated instruments, machinery etc. This can be avoided if we used Machine Learning for detecting the leaks. Machine Learning (or any other type of learning) requires good amount of data so that the model can learn about the patterns in the data. There are many benchmark networks that are used for leak detection and the prominent ones are Hanoi, Net1 etc. The models generated are tested on these benchmark datasets so, that the best model can be established without any bias related to the data.

5 ABOUT THE DATASET:

The data used in this report are from the open source LeakDB software

https://github.com/KIOS-Research/LeakDB

Time-series data of the Hanoi Network was used in this project. It consists of the Pressure, Flow and Nodal Demands of each of the components of the pipe network system. It has 1000 Scenarios where each scenario corresponds to a particular demand pattern. The Hanoi Network is as seen in Fig. 1 Hanoi Network. The Leaks are again classified into incipient and abrupt.

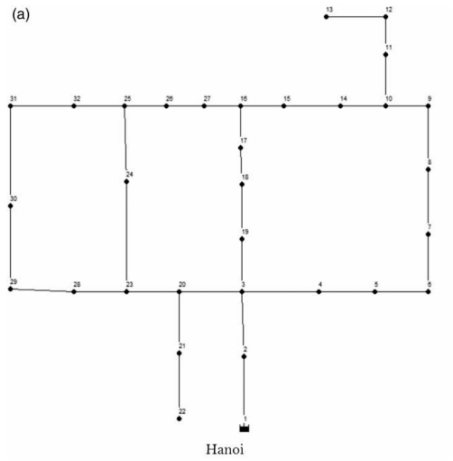

Fig. 1 Hanoi Network

The statistics about the dataset can be found below:

Leaky and Non-Leaky Scenarios: 763, 237
Abrupt and Incipient Leaks: 506, 514

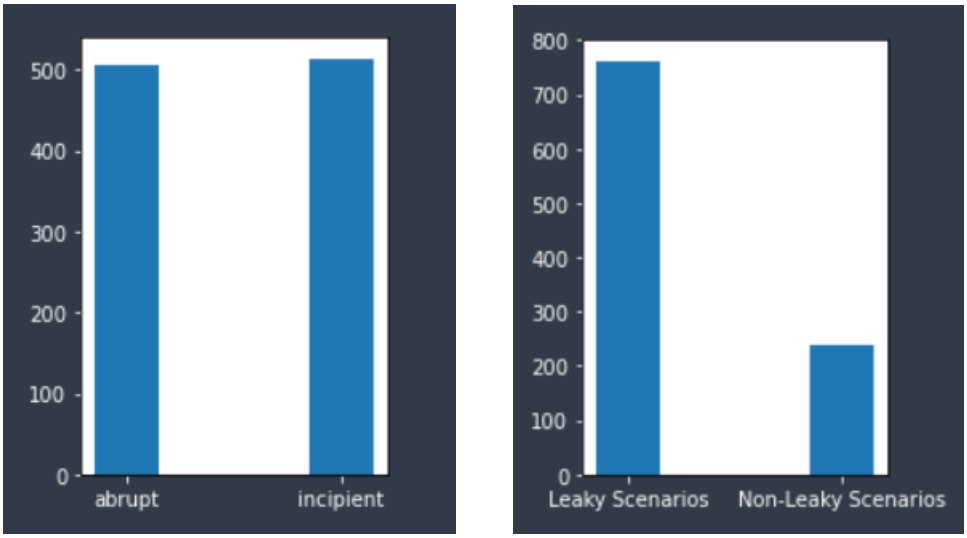

Figure 2 a) No. of Leaky and Non-Leaky Scenarios
b) No. of incipient and abrupt Scenarios

6 ALGORITHMS AND METRICS

The algorithms and metrics that are used in this paper are described below.

6.1 DECISION TREES

Decision trees are a prominent technique in machine learning and are often used in operations research, particularly in decision analysis, to assist determine the best method for achieving a goal. Decision Trees can be used for classification as well as regression. A decision tree is a decision support tool that uses a tree-like model of decisions and their possible consequences. Each node of the tree consists of a condition, if it is satisfied then the datapoint goes into one the branches of the node. If the condition fails then the datapoint goes into another branch of the node. This way the complete data is classified.

6.2 RANDOM FOREST

A random forest classifier, as the name suggests is a collection of decision trees, an ensemble model. Random forest is a supervised machine learning algorithm that is commonly used to solve classification and regression issues. It creates decision trees from several samples, using the majority vote for classification and the average for regression. One of the most distinctive characteristics of the Random Forest Algorithm is that it can handle data sets with both continuous and categorical variables, as in regression and classification. When it comes to categorization difficulties, it outperforms the competition.

6.3 CONVOLUTIONAL NEURAL NETWORK (CNN)

Convolutional neural networks (CNNs) are a type of deep learning technology that has grown popular in computer vision and is drawing interest from a wide range of fields. A convolutional neural network is made up of numerous layers, such as convolution layers, pooling layers, and fully connected layers, and it uses a backpropagation algorithm to learn spatial hierarchies of data automatically and adaptively.

6.4 RANDOMSEARCHCV

The models in this paper were tuned according to RandomSeachCV. RandomSearchCV randomly selects different combination of the values of the hyperparameters from the given range of values and finds the best set of hyperparameters. In RandomSearchCV, along with random search cross validation is done on the train data.

6.5 PCA (PRINCIPAL COMPONENT ANALYSIS)

Principal Component Analysis or PCA is a dimensionality reduction technique. This technique is generally used when there are too many features in the given data. All the features may or may not contribute to the prediction. Hence, PCA helps to summarize the information content in large data by means of a smaller set that can be more easily visualized and analyzed. PCA also ensures that the variance in the data is maximized even after reducing in dimensionality

6.6 TRUE POSITIVE RATE (TPR)

True Positive Rate is the proportion of positives that are classified as positives by the algorithm. Mathematically it is given by,

$$TPR = \frac{No.of\ True\ Positives}{No.of\ True\ Positives + No.of\ False\ Negatives}$$

6.7 FALSE POSITIVE RATE (FPR)

False Positive Rate is the proportion of true negatives that are misclassified as positives by the algorithm

$$FPR = \frac{No.of\ False\ Positives}{No.of\ False\ Positives + No.of\ True\ Negatives}$$

6.8 AUC-ROC CURVE

Receiver Operator Characteristic Curve (ROC) is one the most used evaluation metrics for binary classification. This is a plot that is drawn between the Specificity (FPR) and Sensitivity (TPR). This metric gives us the information about how well the model is doing while classifying positive and negative classes. We can find the area under the curve and this gives the score of the ROC curve. An ideal classifier will have the AUC value as 1 since it perfectly classifies all the points.

6.9 ACCURACY

One criterion for assessing classification models is accuracy. Informally, accuracy refers to the percentage of correct predictions made by our model. The following is the formal definition of accuracy:

$$Accuracy = \frac{No.\ of\ correct\ predictions}{Total\ number\ of\ predictions}$$

The following formula may be used to calculate accuracy in terms of positives and negatives for binary classification:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Where TP - True Positives TN - True Negatives,

FP - False Positives FN - False Negatives.

7 DESIGN DESCRIPTION:

The following pre-processing steps were applied to the each of the Scenarios in order make them ready for the classification algorithm.

7.1 RESHAPING

The given data is in the form of a 2D array with rows and columns. This data is then reshaped into a single row. This process is also called Flattening the array. All the classification algorithms take one-dimensional array as input and this is precisely why this needs to be done.

7.2 SCALING

The data is then scaled down in order to make it easy for the model to understand the patterns in the data. This also helps to remove the magnitude differences in the data. For example, if one feature is of the order 10 and the other features are of the order 1000 then the model gives the later more importance since it has larger values. This doesn't give the desired results. MinMaxScaler was used for scaling.

7.3 PCA

At last, PCA is applied to reduce the dimensionality of the data. After scaling the length of the input array is around 6 lakhs. This is a lot of features to deal with. There may be many useless features which may not contribute to the final prediction. So, the feature space is reduced from 6 lakhs to 1000, while still maintaining maximum variance in the data. This greatly reduced the running time of the algorithm and the memory usage.

After the preprocessing is done, the data is then passed onto the Machine Learning models (Decision Trees and Random Forest), and Deep Learning model (CNN).

8 EVALUATION

8.1 OVERVIEW

Initially the problem was to classify a given scenario as Leaky and Non-leaky. Since a given scenario contains Pressures and Flows. The required values were extracted from the csv files and read into a DataFrame. The DataFrame was then flattened into a 1-dimensional array so that it could be passed as an argument to the machine learning model. Similarly, the leaks were also imported which were the label for the machine learning model. Then the data was split into train and test sets with

80:20 split. The model was trained on the train set and the performance was tested on the test set. After the model has been fitted to the train set the required metrics TPR, FPR, Accuracy, ROC-AUC were found out and the confusion matrix is plotted.

```
[array([3337.2, 3186. , 1245.6, ...,  -86.4,  108. ,  273.6]),
 array([3232.8, 3092.4, 1209.6, ...,  -54. ,   75.6,  223.2]),
 array([3330. , 3175.2, 1198.8, ...,  -46.8,   64.8,  226.8]),
 array([3315.6, 3160.8, 1342.8, ...,  -36. ,   57.6,  212.4]),
 array([3355.2, 3200.4, 1303.2, ...,  -14.4,   36. ,  169.2]),
 array([3330. , 3182.4, 1303.2, ...,  -93.6,  118.8,  284.4]),
 array([3409.2, 3261.6, 1364.4, ...,  -43.2,   68.4,  262.8]),
 array([3394.8, 3222. , 1288.8, ...,  -50.4,   68.4,  201.6]),
 array([3366. , 3204. , 1263.6, ...,  -50.4,   72. ,  241.2]),
 array([3312. , 3142.8, 1270.8, ...,  -43.2,   68.4,  226.8]),
 array([3225.6, 3085.2, 1213.2, ...,  -82.8,  108. ,  244.8]),
 array([3204. , 3045.6, 1216.8, ...,  -39.6,   57.6,  201.6]),
 array([3387.6, 3243.6, 1260. , ...,  -39.6,   61.2,  237.6]),
 array([3344.4, 3178.8, 1245.6, ...,  -72. ,   97.2,  248.4]),
```

The length of each array is 595680
(48 x 34 x 365)

8.2 PROTOTYPE

The following models were used for prediction using only the first 300 Scenarios, using only the Flows data, to get a base line accuracy for the data. This exercise was done to make sure that the models that are used are viable for the given data and the approach is valid. The models that were used are Decision Tree Classifier, a simple ML model and Random Forest Classifier, an ensemble model. The results are given in Table 1 | Using 300 Scenarios.

Table 1 | Using 300 Scenarios

	Accuracy	
	Train	**Test**
Decision Tree	1.0	0.7833
Random Forest	0.8375	0.833

8.3 INTUITION FOR THE CNN MODEL:

CNNs are generally used for image data which contains pixels. Pixels are nothing but Greyscale values between 0 and 1. In the LeakDB data, we can find that the data is similar to image data. Consider the plots of the data from a few Scenarios. We can immediately observe that there are some patterns in the data. This feature of the data is exploited by the CNN model and classification is done based on these patterns.

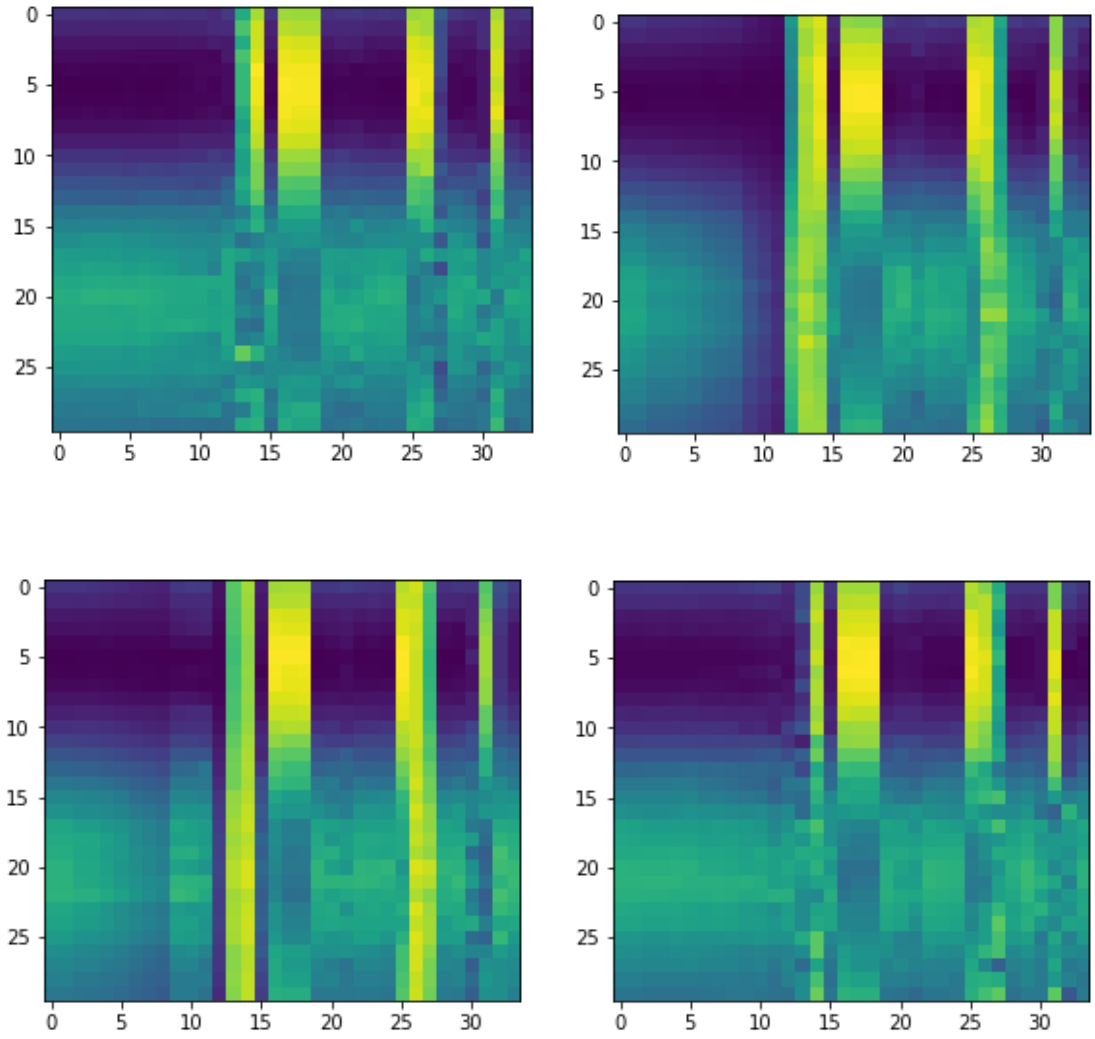

Figure 3: Plotting the Scenarios

8.4 TESTING AND RESULTS

Now that, all the models to be used are decided and ready. All the 1000 Scenarios were taken and a train-test split of 0.2 was used. Using all the 1000 Scenarios predictions were made to find out whether the given Scenario is Leaky or Non-Leaky. The above-mentioned models were trained with 800 Scenarios and tested with 200 Scenarios (with random split). The models were trained and tested on different parts of the data. Separate models were trained and tested using the Flow data alone and then using the Pressure data alone. The results are documents in *Table 2.* The ROC curves of the models are shown in Figure 5

Then both the pressure and flow values of random 500 Scenarios were taken together as the input for the model and again using the train-test ratio of 0.2, the models were trained. Here the dimensionality was reduced to 1200 features in order to maintain maximum explained variance. *Table 3.*

Table 2 | Results using 1000 Scenarios

	Model	With Hyperparameter Tuning				Kammoun et al. (Best values)		
		TPR	FPR	AUC	ACC	TPR	FPR	ACC
Flows	Decision Tree	89.63	0.340	0.835	0.84	0.375	0.87	0.613
	Random Forest	94.11	0.212	0.960	90.5			
	CNN	9671	0.145	0.9706	94			
Pressures	Decision Tree	0.8687	0.375	0.6764	0.82	0.406	0.022	0.731
	Random Forest	0.8765	0.315	0.9017	0.84			
	CNN	0.9194	0.333	0.9234	0.855			

Table 3 Results using both Pressure and Flow

Model	With Hyperparameter Tuning			
	TPR	FPR	AUC	ACC
Decision Tree	0.962	0.149	0.837	0.837
Random Forest	0.933	0.388	0.9329	0.86
CNN	0.987	0.26	0.9634	0.93

8.5 LEAK LOCALIZATION:

It is not just sufficient to find out whether there is a leak in the given Scenario or not, it is also important to know where is leak is present. It helps is to speed up the repair process and prevent water losses. This is a very difficult problem to answer as there are 32 nodes in the network and anyone them could be a leak node. This is typically called as a multi-label problem as multiple predictions are needed for a given input and the number of predictions is not the same for every input. This leads to many complications while fitting the model with the training data.

Here we are trying to predict a 32-length vector which has binary values, 1 specifying that there is a leak present in that particular node and 0 specifying that there is no leak in it. Since there are a maximum of only 3 leaks for a given Scenario, this vector is mostly sparse. This proved to be a big challenge as it was unable to perform well on the test set. The reasons for this maybe:

- The data taken is very less and hence the model may not be able to generalize the patterns in the data
- Maybe too many features are present in the data and hence we are not able to get the desired output.

Decision Tree: accuracy on the test set: 0.04

Random Forest: accuracy on the test set: 0.16

CNN Model: accuracy on the test set of 0.0799

Layer (type)	Output Shape	Param #
conv0 (Conv1D)	(None, 499, 16)	80
max_pooling1d (MaxPooling1D)	(None, 167, 16)	0
conv1 (Conv1D)	(None, 82, 16)	1040
dropout (Dropout)	(None, 82, 16)	0
flatten (Flatten)	(None, 1312)	0
dense (Dense)	(None, 32)	42016
dense_1 (Dense)	(None, 16)	528
dense_2 (Dense)	(None, 1)	17

Figure 4 | CNN Model Structure

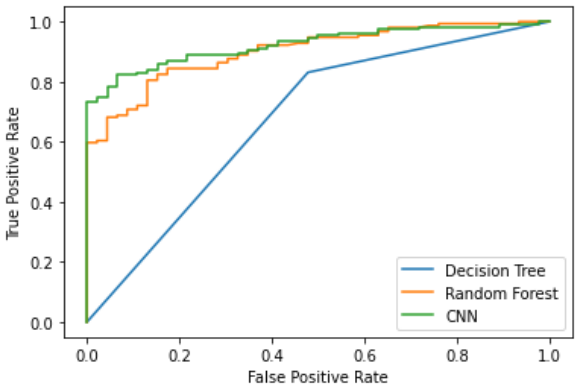

Figure 5 | ROC for Pressure data

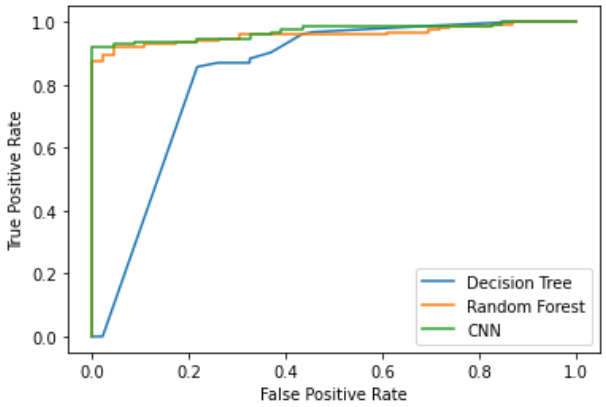

Figure 6 | ROC for Flow data

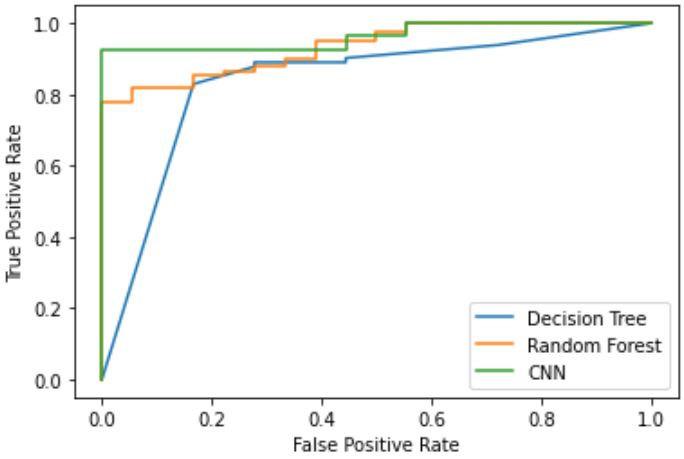

Figure 7 | ROC for combined data

9 CONCLUSIONS

From the above results we can see that it was really easy to predict whether a given Scenario is Leaky or not, but it was really difficult to predict which node had the leak. The CNN model outperformed the other two models. The Random Forest model was giving a better result than the simple Decision Tree model. The reason for this might be that the Decision Tree, being a simple classification algorithm was unable to generalize the trends of the data, while the Random Forest, an ensemble model and CNN, a neural network model, were able to find the patterns in the data and hence performed much better. The optimization used for the hyperparameter was done by RandomSearchCV which takes in the ranges for the values of the hyperparameters and gives out the best values of the hyperparameters from the given range of values. This reduces the burden of manually tuning the hyperparameters and finding out the best values.

It can be evidently seen that the models performed well with increase of the data from 300 Scenarios to 1000 Scenarios. Among the Flow data and the Pressure data the Flow data seems to be giving a better accuracy. This may be because of large fluctuations in the Pressures data. The combined Pressure and Flow data also gave good results and the results might improve even further with the use of all 1000 Scenarios.

10 REFERENCES

Stelios G. Vrachimis , Marios S. Kyriakou , Demetrios G. Eliades , and Marios M. Polycarpou, "LeakDB: A benchmark dataset for leakage diagnosis in water distribution networks" in " 1st International WDSA / CCWI 2018 Joint Conference"

Maryam Kammoun , Amina Kammouna and Mohamed Abida, " Experiments based comparative evaluations of machine learning techniques for leak detection in water distribution systems", "Water Supply Vol 00 No 0, 1 doi: 10.2166/ws.2021.248"

T. K. CHAN, CHENG SIONG CHIN , AND XIONGHU ZHONG2 , "Review of Current Technologies and Proposed Intelligent Methodologies for Water Distributed Network Leakage Detection"

Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

https://machinelearningmastery.com/hyperparameter-optimization-with-random-search-and-grid-search/, Hyperparameter Optimization With Random Search and Grid Search