Lappeenranta University of Technology
School of Engineering Science
Erasmus Mundus Master's Program in Pervasive Computing & Communications
for Sustainable Development (PERCCOM)

Sami Kabir

BRB BASED DEEP LEARNING APPROACH WITH APPLICATION IN SENSOR DATA STREAMS

Master's Thesis - 2019

Supervisors: Prof. Dr. M. Shahadat Hossain (University of Chittagong)
Prof. Dr. Karl Andersson (Luleå University of Technology)
Raihan Ul Islam, PhD Student (Luleå University of Technology)

Examiners: Prof. Dr. Eric Rondeau (University of Lorraine)
Prof. Dr. Jari Porras (LUT University)
Prof. Dr. Karl Andersson (Luleå University of Technology)

This thesis is prepared as part of an European Erasmus Mundus Programme PERCCOM - PERvasive Computing & COMmunications for sustainable development.

Co-funded by the
Erasmus+ Programme
of the European Union

This thesis has been accepted by partner institutions of the consortium (cf. UDL-DAJ, n°1524, 2012 PERCCOM agreement).

Successful defense of this thesis is obligatory for graduation with the following national diplomas:

- Master in Complex Systems Engineering (University of Lorraine)
- Master of Science in Technology (LUT University)
- Master of Science in Computer Science and Engineering, specialization in Pervasive Computing and Communications for Sustainable Development (Luleå University of Technology)

ABSTRACT

Lappeenranta University of Technology
School of Engineering Science
PERCCOM Master Program

Sami Kabir

BRB BASED DEEP LEARNING APPROACH WITH APPLICATION IN SENSOR DATA STREAMS

Master's Thesis – 2019

106 pages, 38 figures, 8 tables, and 2 appendices.

Examiners: Professor Dr. Eric Rondeau
Professor Dr. Jari Porras
Assoc. Prof. Dr. Karl Andersson

Keywords: BRBES; Deep Learning; integration; sensor data; predict.

Predicting events based on available data is an effective way to protect human lives. Issuing health alert based on prediction of environmental pollution, executing timely evacuation of people from vulnerable areas based on prediction of natural disasters are the application areas of sensor data stream where accurate and timely prediction is crucial to safeguard people and assets. Thus, prediction accuracy plays a significant role to take precautionary measures and minimize the extent of damage. Belief rule-based Expert System (BRBES) is a rule-driven approach to perform accurate prediction based on knowledge base and inference engine. It outperforms other such knowledge-driven approaches, such as, fuzzy logic, Bayesian probability theory in terms of dealing with uncertainties. On the other hand, Deep Learning is a data-driven approach which belongs to Artificial Intelligence (AI) domain. Deep Learning discovers hidden data pattern by performing analytics on huge amount of data. Thus, Deep Learning is also an effective way to predict events based on available data, such as, historical data and sensor data streams. Integration of Deep Learning with BRBES can improve prediction accuracy further as one can address the inefficiency of the other to bring down error gap. We have taken air pollution prediction as the application area of our proposed integrated approach. Our combined approach has shown higher accuracy than relying only on BRBES and only on Deep Learning.

ACKNOWLEDGEMENTS

This research work has been pursued at campus Skellefteå of Luleå University of Technology (LTU), Sweden under direct supervision of Professor Dr. Mohammad Shahadat Hossain, University of Chittagong, Bangladesh; Associate Profesor Dr. Karl Andersson, Luleå University of Technology (LTU), Sweden and Mr. Raihan Ul Islam, PhD student, LTU. It was funded by European Commission's Erasmus Mundus Joint Master Degree in Pervasive Computing and Communications for Sustainable Development (PERCCOM) program [58].

I would like to appreciate the immense support I received from my thesis supervisors in taking this research to a reasonable end. Their undeviating guidance enabled this thesis to be my unique work, but directed me to the proper route whenever it turned out to be necessary to them. I presented my work progress through the Skype meetings held after every two weeks. They regularly assessed my progress and gave me valuable feedback which led me to the right direction and complete the thesis. Whenever I went back to my home countryBangladesh during Summer and Christmas vacations, my supervisor Prof. Dr. Shahadat Hossain conducted one-to-one physical meeting with me to explain some of the core concepts pertaining to my thesis topic and check my latest thesis status. He also created opportunity for me to deliver a guest lecture concerning this research work at a local private university in Chattogram, Bangladesh. Such endeavors undoubtedly indicate his whole-hearted dedication and sincerity to this research.

I also strongly recognize the supportive role played by Prof. Dr. Eric Rondeau, Prof. Dr. Jari Porras, Associate Prof. Dr. Jean-Philippe Georges as well as my cohortmates in executing this Master's level thesis. Whenever I faced any difficulty with respect to my theoretical or coding part of my thesis, I always received all-out cooperation from all of them.

Finally, I convey my deep appreciation to my parents and my spouse for providing me with steadfast cooperation and persistent inspiration throughout my Master Degree studies and through the process of carrying out this research and communicating it through this thesis report. This manoeuvre could have become unattainable without their helping hands.

Skellefteå, August 30, 2019

Sami Kabir

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

AI	Artificial Intelligence
AdaGrad	Adaptive Gradient
ANN	Artificial Neural Network
AQI	Air Quality Index
BC	Backward Chaining
BRB	Belief Rule Base
BRBES	Belief Rule Base Expert System
CART	Classification and Regression Trees
CNN	Convolutional Neural Network
DCP	Dark Channel Prior
DCNF	Deep Convolutional Neural Fields
DE	Differential Evolution
DRNN	Deep Recurrent Neural Network
EPA	Environmental Protection Agency
ER	Evidential Reasoning
EU	European Union
FC	Forward Chaining
FOPC	First Order Predicate Calculus
VGG	Visual Geometry Group
ICT	Information and Communication Technology
IoT	Internet of Things
IQA	Image Quality Assessment
KNN	K-Nearest Neighbours
LSTM	Long Short-Term Memory
MLP	Multi-Layer Perceptron
MRI	Magnetic Resonance Imaging
MSE	Mean Square Error
NB	Naive Bayes
NLP	Natural Language Processing
PCA	Principal Component Analysis
PL	Propositional Logic
PM	Particulate Matter
PUE	Power Usage Effectiveness
RGB	Red Green Blue
RMSProp	Root Mean Square Propagation
RNN	Recurrent Neural Network
STDL	Spatiotemporal Deep Learning
SVM	Support Vector Machine
WSN	Wireless Sensor Network
ZCA	Zero Component Analysis

(All symbols and abbreviations are listed on this page in alphabetical order)

1 INTRODUCTION

This chapter demonstrates the context of this thesis. It starts with background and sustainability aspects of this work. Then it presents background as well as objectives and deliverables of this research. It also illustrates opportunities and major constraints of this thesis work. Finally, this chapter is concluded with an overall structure of the whole content concerning this research.

1.1 Background

Preventive steps always play crucial role to reduce the extent of damage significantly. Accuracy of the prediction is key to facilitate such preventive measures. Availability of data, such as, sensor data, historical data, are prerequisites to achieve this accuracy. In terms of time-series sensor data, there are many application areas of sensor data streams where prediction can let the policy-makers take precautionary steps to safeguard both people and assets. Performing systematic computational analysis, alternatively known as analytics, on such sensor data streams results in prediction. For example, air pollution level can be predicted by doing analytics over the sensor data of concentrations of major air pollutants, e.g., Particulate Matter (PM) with diameter less than 2.5 micro-meters ($PM_{2.5}$) and diameter less than 10 micro-meters (PM_{10}), CO, O_3, SO_2, NO_2. Outdoor air pollution causes around 3 million deaths every year [108]. Therefore, generating accurate air pollution prediction can improve people's health condition.

There are two categories of approaches to generate prediction. One is knowledge-driven approach and the other is data-driven approach [17]. Knowledge-driven approach represents an expert system which consists of two components: knowledge base and inference engine. This knowledge base, which illustrates rules and facts, is constituted by if-then rules, instead of typical procedural code. Inference engine makes reasoning over these rules against the known facts to infer predictive output. Thus, an expert system, which demonstrates a knowledge-based system, makes predictive analytics by reasoning over input data. Belief Rule Base Expert System (BRBES), fuzzy logic, MYCIN [12], PERFEX [4] are the examples of knowledge-driven approach. However, BRBES outperforms other knowledge-driven approaches as it can deal with different types of uncertainties, specially ignorance

[44]. An expert system is made up of two main elements – knowledge base and inference engine. Propositional Logic (PL) and First Order Predicate Calculus (FOPC) are applied to develop knowledge base while Forward Chaining (FC) and Backward Chaining (BC) are employed to develop inference mechanism. However, PL and FOPC represent assertive knowledge. Hence, they cannot handle uncertain knowledge [44]. BRBES overcomes this shortcoming by employing Evidential Reasoning (ER) as its inference engine as it has the capability to address uncertainties [112].

Data-driven approach learns autonomously from external data. It can compute prediction by discovering hidden pattern of the external data. It performs analytics on big amount of data, such as, sensor data, historical data to extract patterns and knowledge from data leading to actionable insight. Thus, this approach represents data mining which also works in the same way. It has no rule-base. Rather, it continuously trains itself up by learning from external data. Machine Learning and statistical methods are used by data mining to uncover hidden patterns in a large volume of data [79]. A machine learning approach comprises mathematical/statistical models, necessary for training data and to accomplish the task of prediction in an implicit way. Its application areas include email filtering, computer vision, network intruder detection and so on, where it is not feasible to build a specific rule base to perform the predictive analytics. There are three types of learning approaches of Machine Learning algorithms – Supervised learning, Unsupervised learning and Reinforcement learning. Supervised learning algorithms develop a mathematical model of a labeled data set. These are training data comprising both inputs and corresponding outputs. Testing data are used to compute the accuracy of the mathematical model. Finally, prediction is generated against new input data complying with the mathematical model. Examples of Supervised learning algorithms include Support Vector Machine (SVM), Classification and Regression Trees (CART), Naive Bayes (NB), K-Nearest Neighbours (KNN) [10]. Unsupervised learning algorithms take input data with no output label and discover hidden structure of those input data. This algorithm learns features from dataset which is not labeled, classified or categorized. It detects similarity in data and predicts with respect to existence of such similarity in new dataset. Apriori, K-means, Principal Component Analysis (PCA) are the examples of unsupervised learning algorithms [84]. Reinforcement learning focuses on how a software agent can take next course of action to maximize the reward. Such algorithms typically master optimal operation on trial and error basis. For example, a robot, through

reinforcement learning, learns to avoid collision by receiving negative feedback after facing obstacles [54]. On the other hand, Artificial Neural Network (ANN) is a framework for neuron based network to work together to process complex data inputs. ANN is an assemblage of nodes termed “artificial neurons”. Connections between every two neurons constitute “edges”. Each edge transmits signal from one neuron to the other. Several neurons constitute a layer. There are several layers between input and output layer, which are referred to as hidden layers. Neurons of the hidden layers apply various types of transformations through activation functions on their inputs. Signals traverse from first layer (input layer), through intermediate multiple hidden layers, to last layer (output layer).

However, Machine Learning cannot process raw data in their natural shape [65]. It doesn’t have feature extractor to convert these raw data, such as, pixels of an image, into an appropriate internal characterization or vector of features, such as an edge at a particular location of the image. Feature extractor preprocesses the raw data to uncover the characterizations necessary for detection or classification. Upon preprocessing by feature extractor, the representation learning data are fed to the Machine Learning for classification purpose. On the other hand, Deep learning, which is mainly based on the neural network architecture, addresses this lacking as it can directly accept raw data as input. It processes such raw data by representation-learning methods with multiple levels of representations. Fig. 1 demonstrates this distinction between Machine Learning and Deep Learning. Deep neural networks, deep belief networks, Recurrent Neural Networks, Convolutional Neural Networks (CNN) etc. are some of the deep learning architectures which are applied on various fields, such as, computer vision, speech recognition, machine translation, natural language processing, sequence prediction, bioinformatics, drug design, image analysis. The term “Deep” in “Deep Learning” refers to the large number of hidden layers through which data are transformed. Generally, number of hidden layers is more in deep learning than simple neural network [18]. There is no universal threshold of depth separating shallow learning from deep learning. However, Schmidhuber [88] has considered the number of hidden layers in a deep learning to be more than 10.

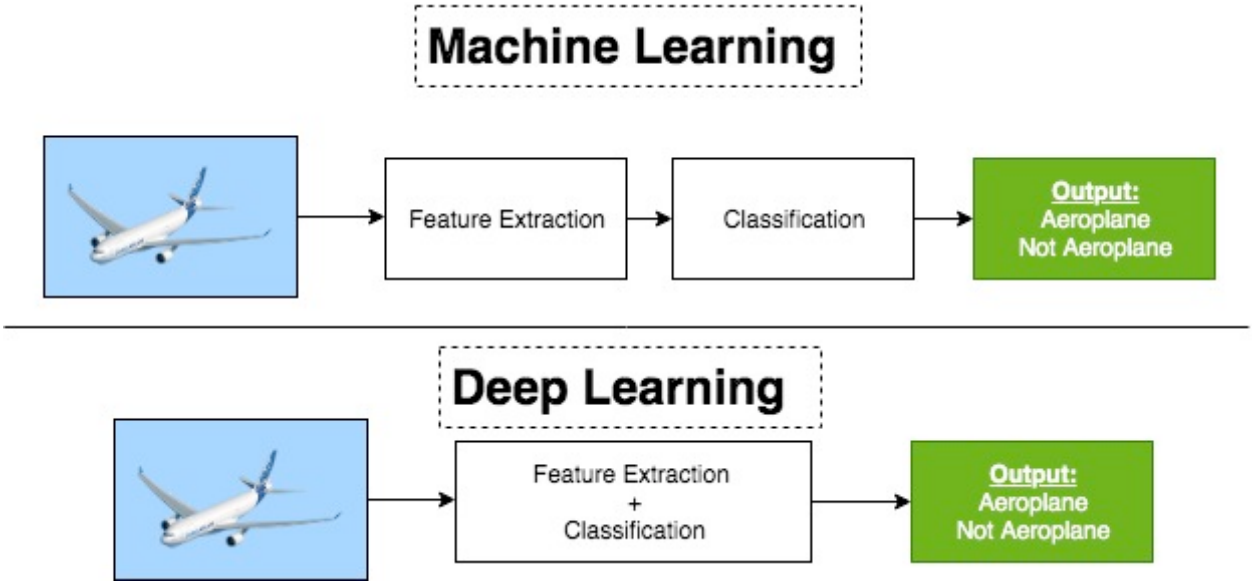

Fig. 1. Distinction between Machine Learning and Deep Learning.

BRBES, as an expert system, performs inference through its knowledge base. However, it does not learn autonomously from external data. On the other hand, deep learning extracts pattern from large volume of data. It does not have any knowledge base. Inspired by the efficacy of deep learning for predictive analytics, we propose to enhance the accuracy of BRBES further by integrating Deep Learning with it. Thus, the objective of the proposed integrated approach of BRBES and deep learning is to combine the strength of both the methods and develop a predictive model with improved accuracy.

Presently, air pollution is the fourth largest global human health concern [35]. It costs world economy around US$ 5 trillion annually [95]. Driven by this, we introduce air pollution prediction as the application area of sensor data streams to minimize its adverse impact on the earth. We have taken into account concentration of air pollutant $PM_{2.5}$ in the air to predict the air quality level in terms of Air Quality Index (AQI). $PM_{2.5}$ refers to atmospheric Particulate Matter (PM) whose diameter is less than or equal to 2.5 micrometers [102].

AQI is a piecewise linear function of six air pollutants: Ozone (O_3), $PM_{2.5}$, PM_{10}, Carbon Monoxide (CO), Sulphur Dioxide (SO_2) and Nitrogen Dioxide (NO_2). It is applied by the public bodies to convey to the citizens current air pollution level or estimated pollution level of near future. EPA of United States has developed the breakpoint table of AQI [29].

As we have taken into account $PM_{2.5}$ as our air pollutant in this research, the AQI table against this pollutant is shown in Table 1. To convert from $PM_{2.5}$ concentrations to AQI, the following equation is used.

$$I = \frac{I_{high} - I_{low}}{C_{high} - C_{low}}(C - C_{low}) + I_{low} \quad (1)$$

where I is the AQI, C is the concentration of $PM_{2.5}$, C_{low} is the concentration breakpoint which is <=C, C_{high} is the concentration breakpoint which is >=C, I_{low} is the index breakpoint with regard to C_{low} and I_{high} is the index breakpoint with regard to C_{high}. For calculating $PM_{2.5}$ AQI, 24-hr average concentration of $PM_{2.5}$ is required.

We adopt Design Science Research (DSR) as our research methodology to carry out this research. The purpose of this research work directly falls under the jurisdiction of ICT for sustainability. Air pollution prediction, as use case of our proposed predictive model, has been reviewed in the context of sustainability. It supports all three pillars of sustainability: people, planet and profit, as shown in Fig. 2.

Table 1. Breakpoint Table of AQI.

$PM_{2.5}$ ($\mu g/m^3$)	AQI	AQI	Colors
C_{low}- C_{high} (avg)	I_{low} - I_{high}	Category	
0.0-12.0 (24-hr)	0 – 50	Good	Green
12.1-35.4 (24-hr)	51 – 100	Moderate	Yellow
35.5-55.4 (24-hr)	101 – 150	Unhealthy for sensitive groups	Orange
55.5-150.4 (24-hr)	151 – 200	Unhealthy	Red
150.5-250.4 (24-hr)	201 – 300	Very Unhealthy	Purple
250.5-350.4 (24-hr)	301 – 400	Hazardous	Maroon
350.5-500.4 (24-hr)	401 – 500		

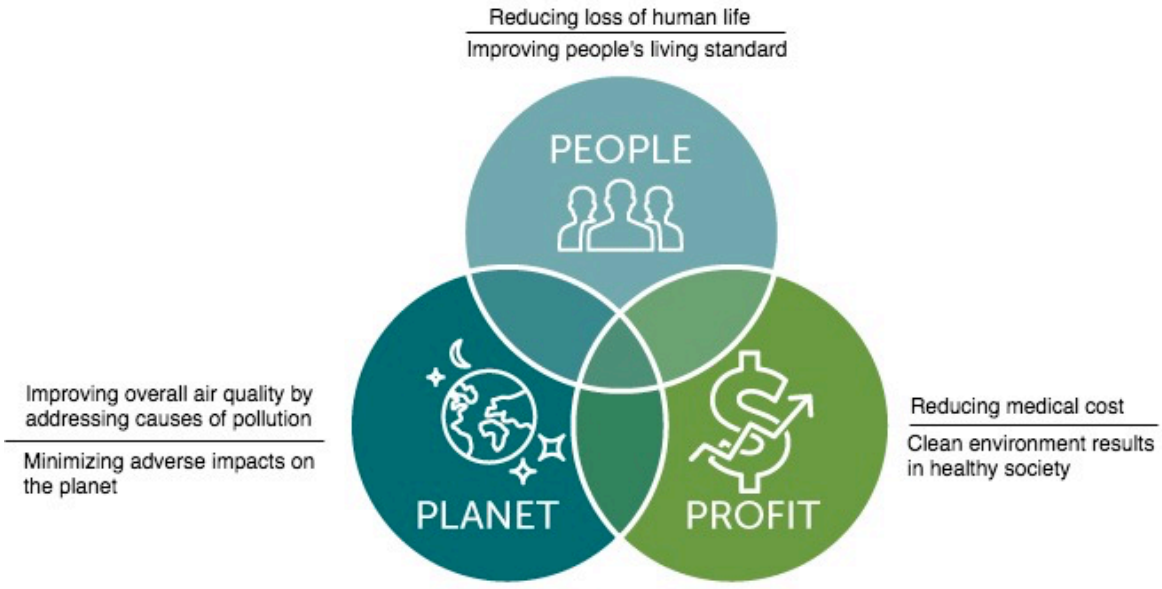

Fig. 2. Sustainability aspects of this work.

1.2 Aim, Research Objectives and Questions

The main aim of this research is to develop a BRB based Deep Learning approach with the capability of handling sensor data uncertainty as well as discovering hidden data pattern of sensor data to predict the level of air pollution with improved accuracy. It aims to develop a prediction model with higher accuracy to assist the environment regulatory authorities and policy-makers to assess different aspects of air pollution and take precautionary/preventive measures. To achieve the stated aim of this research, the following objectives have been identified.

1. Investigation of existing data-driven and knowledge-driven predictive approaches
2. Design and Implement BRBES to predict air pollution level
3. Application of Deep Learning to predict air pollution level
4. Integrating Deep Learning based analytics model with BRBES

The following research questions have been raised to realize these objectives.

1. What are the benefits of using BRBES to predict air pollution?

There are several knowledge-driven approaches in the existing literature for prediction purpose. Rationale of opting for BRBES over other approaches is the main concern of this question.

2. What are the advantages of using Deep Learning to predict air pollution?

In addition to BRBES, the rationale of applying Deep Learning to develop a predictive model is justified by answering to this question.

3. Why and how we should integrate Deep Learning with BRBES?

This question is intended to justify the reason of integration and how this integration can be achieved to increase accuracy of the predictive model.

1.3 Novel Contributions

The novelty of this thesis lies in the development of a mathematical model to combine Deep Learning with BRBES. The novel contributions concerning this research are the following.

1. A mathematical model to integrate BRB and Deep Learning to utilize strength of both the systems and increase prediction accuracy.
2. Development of BRBES to predict air pollution level based on $PM_{2.5}$ concentrations.

1.4 Scope and Delimitations

Predictive analytics model can be applied on various domains, such as, medical diagnosis, natural disaster prediction, customer segmentation, stock management, predictive maintenance and so on. However, this research is focused on air pollution prediction. As sustainability and green ICT top the list of PERCCOM agenda, this thesis also defines sustainability in the context of air pollution prediction. The novel approach of BRB based deep learning facilitates complete assessment of air quality of any area. The proposed integrated approach is capable of dealing with both structured and unstructured data to increase the effectiveness of the system. Thus, the approach provides fruitful insight to take preventive steps before the air quality downgrades further. However, using satellite images,

instead of ground images, to measure the concentrations of $PM_{2.5}$ can be a future scope of this thesis. Although the system is providing satisfactory assessment with 3024 ground images, feeding the system with more images from variety of sources can be considered to be a future work. The study has used image data and numerical sensor data. It would be interesting to combine geospatial data with our system to inform the citizens of the specific location of the polluted zone, in addition to AQI value.

1.5 Thesis Outline

Content outline of the upcoming chapters of this thesis is presented below.
Chapter 2 – Background and Literature Review

This chapter focuses on the sensor data streams, prediction analytics and introduces air pollution as application area of the proposed system. It also covers uncertainties associated with sensor data as well as how researcher are applying various methods for predicting air pollution.

Chapter 3 – Methodology and System Architecture

This chapter describes Design Science Research (DSR) methodology, definition, challenges as well as various methods and applications of deep learning. It also demonstrates our system architecture.

Chapter 4 – Integrated Approach of BRB and Deep Learning

This chapter presents model configuration of our integrated approach mathematically by focusing separately on deep representation part, BRB part and integrated part. It also shows how belief degrees for each of the six AQI categories are calculated with respect to the predicted AQI value.

Chapter 5 – Optimized BRB

This chapter covers conjunctive and disjunctive BRB as well as trained BRB by parameter, structure and joint optimization.

Chapter 6 – Results and Discussion

This chapter describes the dataset we have used and presents comparative results of our proposed model. It demonstrates performance comparison between conjunctive and disjunctive BRBES as well as trained and non-trained BRBES. It concludes with sustainability analysis and ICT ethics pertinent to our work.

Chapter 7 – Conclusion and Future Works

This chapter makes concluding remarks based on the evaluated results and observations. It also refers to few constraints and future scope to advance this research further.

1.6 Summary

We have introduced background, research objectives, novel contributions and delimitations of this thesis in this chapter. Next chapter will focus on predictive analytics on sensor data, use case of this thesis, sensor data uncertainties as well as literature review.

2 BACKGROUND AND LITERATURE REVIEW

This chapter focuses on the concept of sensor data streams and prediction analytics. It introduces air pollution prediction as the application area of sensor data streams. Moreover, it defines uncertainties of sensor data and explains various methods which are presently being used by researchers to predict air pollution.

2.1 Sensor Data Streams

Sensor is a device which detects events in environment, processes the data locally and transmits digital signal to electronics [2]. Data transmitted by sensor constitute sensor data. When these sensor data continuously grow over time, it becomes sensor data streams.

Recent advancement in Wireless Sensor Network (WSN) has turned sensor data streams into a promising area of research and development. Sensor devices connected with each other constitute a part of Internet of Things (IoT). IoT refers to a network of everything around us. Upcoming 4th Industrial Revolution has brought the concept of IoT under global focus. WSN and Radio Frequency Identification (RFID) are integral parts of IoT [37]. Reasoning is applied over the sensor data streams at the inference layer of IoT. This reasoning extracts pattern of sensor data to compute predictive output. Such reasoning for predictive output is known as predictive analytics [52].

2.2 Predictive Analytics

The process of applying computational techniques to unearth and communicate hidden patterns of data is called analytics [1]. Analytics is intended to gain insight into data by examining these and come up with prediction. This term drew significant attention in 2005 mainly due to the introduction of Google Analytics. As more and more data are being generated all over the world, there is a natural progression toward utilizing these to facilitate decisions, estimates and improve efficiency.

When analytics carries out the task of prediction, that is called predictive analytics. Predicting the future behavior is the main objective of predictive analytics, on the contrary to

business intelligence, which looks back into the past [1]. Predictive analytics is prospective while business intelligence is retrospective. Predictive Analytics concerns several related disciplines, such as, Artificial Intelligence (AI), machine learning, data mining, pattern recognition etc. It derives key characteristics of the model from the input data in an automated manner. Learning mechanism of algorithms for predictive modeling can be supervised, unsupervised or reinforcement. These predictive algorithms can detect new patterns and reveal new casual mechanism which affect final decision [89]. On the other hand, business intelligence evaluates how effective a past business model was.

Predicting the likelihood a client will buy an apartment, flood will happen, a customer will open an email, a transaction is fraudulent, a website will be overloaded during Christmas vacation are some of the real-life examples of predictive analytics. Similarly, sensor data streams are also an application area of predictive analytics for accomplishing the objective of predictive output.

2.3 Application area – Air Pollution prediction

Since this research is intended to apply predictive analytics on sensor data streams, we have taken prediction of air pollution as the application area of sensor data streams. Sensor readings of the air pollutants constitute sensor data. These sensor data, with the passage of time, become sensor data streams. We collect the sensor data streams of air pollutant and apply predictive analytics on these data streams with a view to predicting the level of air pollution of the affected area.

Outdoor air pollution is one of the top ten global health concerns which causes premature mortality [72]. Chronic respiratory diseases like bronchitis, asthma, reduced lung function, lung cancer, cardiopulmonary diseases lead to such premature deaths. In the European Union (EU), air pollution is considered to be the topmost environmental reason of premature deaths [27]. As per World Health Organization (WHO) report, outdoor air pollution caused 3.7 million deaths in 2012 [105]. Air pollution costs EU countries 23 billion Euro ever year including damage caused to crops and buildings [27][106]. Global premature mortality due to $PM_{2.5}$ concentrations in the air was around 3.5 million in the year 2010 [34]][53]. This mortality was highest in China with around 1.33 million, followed by India and Pakistan.

This mortality number was 173,000 across 28 member countries of the EU and 52,000 in the United States. Dhaka, the capital of Bangladesh, is now the second most polluted capital city of the world after New Delhi. Fig. 3 shows a polluted Dhaka street which is causing immense suffering to its citizens.

Such figures have prompted us to take air pollution prediction as our application area and also to take into account $PM_{2.5}$ as air pollutant.

2.4 Uncertainties associated with sensor data

Predictive analytics based on sensor nodes become unreliable due to incorrect and deceptive nature of sensor data. Sensor data may have missing data, duplicated data or inconsistent data because of resource constraints, such as, battery power, computational and memory capacities as well as communication bandwidth [46][49]. These issues result in inaccurate sensor data. Further, sensor deployment become unprotected in harsh environments, leading to malfunction. Such malfunction results in noisy, missing and redundant sensor data. Moreover, sensor nodes are susceptible to malicious attacks, such as, denial of service attacks, eavesdropping and black hole attacks.

Fig. 3. Polluted air in Dhaka city, Bangladesh [24].

The term ‘uncertainty’ means unpredictable outcome. Missing, duplicate or inconsistent sensor data create various categories of uncertainty, such as, ignorance, incompleteness, imprecision, vagueness and ambiguity. Resource constraints of sensor nodes cause some data to go missing, resulting in ambiguity and ignorance. The malfunction triggers sensor nodes to generate incomplete data. Inaccuracy due to malicious attacks causes vagueness in sensor data. Less precise data reading due to low-power battery of sensor nodes causes imprecision. Similarly, a camera sensor can start taking blurred images due to turbulent weather. Also, a camera’s captured images may become hazy during snowstorm if its glasses remain covered with snow or marked with water.

Presence of uncertainty with sensor data due to these factors cause anomaly in sensor data. This anomaly makes sensor data unreliable. If these anomalous data are not filtered before feeding to the expert system, output of the expert system will become inaccurate. Therefore, it is essential to address such anomalous sensor data with uncertainty handling capability in an integrated framework. Reliable results in terms of air pollution prediction can be achieved by this framework.

There are parametric (statistical) and nonparametric model-based anomaly detection methods. Parameter techniques analyze data using density distribution where less relevant data are considered anomalies. Multivariate Gaussian method is a statistical technique for detecting anomaly. However, this method is unable to handle uncertainty due to ignorance, randomness and fuzziness. On the other hand, nonparametric models refer to rule based techniques. These rule based techniques, such as, association rule, applies assertive knowledge which is evaluated either true or false. Therefore, this method is not capable of handling uncertainty due to ignorance, incompleteness or fuzziness. Fuzzy logic can deal with uncertainty due to fuzziness but cannot tackle ignorance and incompleteness. Hence, none of these approaches can deal with all sorts of uncertainty in a coherent framework.

Therefore, our objective is to develop a reliable system which can address all types of uncertainty pertinent to the sensor data of $PM_{2.5}$ in an integrated framework while predicting the level of air pollution in terms of AQI.

2.5 Methods for Air Pollution prediction

This subsection presents various methods employed by different research groups and organizations around the world for predicting the level of air pollution.

Dynamically pre-trained Deep Recurrent Neural Network (DRNN) has been deployed by Theang et al. [96] to predict time-series concentrations of $PM_{2.5}$ in the air of Japan. Weights of the networks of their proposed pre-training method gradually adjust themselves to approach a dynamically and sequentially growing outcome, leading to more accurate learning representation of the time-related input data. They have used environmental observation data produced by physical sensors for this purpose. Spatial consistency in the physical position of the selected sensors has been taken into account to increase the prediction accuracy of DRNN. In terms of sensor data, they have considered $PM_{2.5}$ concentrations, speed and direction of wind, temperature, illuminance, humidity and rain. They have also presented an efficient method to bring down the computational costs by discarding sensors which have insignificant contributions for better predictions. They have applied the elastic net method for filtering sensors based on sparsity. DRNN has shown better prediction accuracy compared with auto-encoder training method. However, DRNN has also not taken into account how to handle anomalous sensor data for prediction.

Spatiotemporal deep learning (STDL)-based air quality prediction system has been proposed by Li et al. [68] which takes into account space and time related interrelationship concerning measurement data of air pollutants. They have used stacked autoencoder (SAE) model as deep learning architecture to obtain intrinsic spatiotemporal features of air pollutants' data and trained SAE in a greedy layer-wise way. This learned representation has been applied to develop a regression model for predicting air quality. In comparison with the conventional time-series prediction model, this model can compute air quality prediction of all stations concurrently while maintaining temporal stability throughout the whole year. It predicts the level of $PM_{2.5}$ based on existing $PM_{2.5}$ concentration generated by Thermo Fisher detector/sensor. They have demonstrated superior performance of STDL over spatiotemporal artificial neural network (STANN), auto regression moving average (ARMA) and support vector regression (SVR) models. However, this model also does not consider uncertainty associated with sensor data.

Geographic Forecasting Models using Neural Networks (GFM_NN) method has been employed by Kurt et al. [63] to forecast the level of sulfur dioxide (SO_2), carbon monoxide (CO) and particulate matter (PM_{10}) of Besiktas district of Turkey 3 days in advance. They have used air pollutant monitoring data from Istanbul city's 10 air pollutant measurement stations. Daily meteorological forecasts and air pollutant concentration data were fed as input to feed-forward back-propagation neural network. They have proposed 3 different geographic models for forecasting purpose. First model uses values of air pollution indicator of a chosen adjacent district. Second model considers two adjacent districts. Third model takes into account the distance between the triangulating districts and the target district of which they have to estimate the level of air pollution. They have demonstrated that their proposed distance-based geographic models produce lower error than non-geographic plain models. Accuracy of their proposed three models mainly depends on selection of neighboring district(s) and number of district(s) chosen. Third model, which uses three districts, has outperformed other two models. Even though, none of their proposed models has considered uncertainty associated with sensor data of air pollutants.

Moreover, Li et al. [69] have proposed an image-based method to approximate and observe air pollution. Due to high price and limited coverage of sensors, they have come up with an efficient technique to evaluate haze level from single images. Given an input image, they have estimated the transmission matrix using haze removal algorithm. Dark Channel Prior (DCP) has been used for this purpose. Simultaneously, they have also estimated the depth map from pixels. Deep Convolutional Neural Fields (DCNF) has been applied for depth map estimation. DCNF infers from a learned Conditional Random Fields (CRF) over superpixels to estimate depth of an image. Objective function of CRF is an amalgamation of unary and pairwise potentials which include set of superpixels, set of neighborhood superpixel pairs, multi-layer CNN over the pixel values, single-layer neural network over a set of common quantifications, such as, color histogram as well as Local Binary Pattern (LBP) resemblance. They have combined transmission matrix and depth map using transformation functions. Then they have used pooling function to pile-up the matrix to a lone figure and estimate haze level of a photo. It has been demonstrated that combining transmission and depth results in higher accuracy for haze level estimation than applying these two factors separately. The gain becomes even more significant when the scenes and haze situations turn more complicated. Their proposed method has shown 89.05% accuracy on $PM_{2.5}$ dataset where only depth map

and only transmission matrix has produced 70.14% and 84.32% accuracy respectively. However, this work has not taken into account uncertainty associated with captured images by a camera sensor (as mentioned in Sect. 2.4).

Liu et al. [75] have deployed image analysis method to predict the level of $PM_{2.5}$ in the air. They have obtained several image features, such as, transmission, sky smoothness, image color, image entropy, whole image and local image contrast, time, geographical location, sun, weather condition of each outdoor image to estimate the particle pollution of the air. With respect to these features, they have built a regression model for predicting PM concentrations from images of Beijing, Shanghai and Phoenix over a period of one year. Support Vector Regression (SVR) has been applied to develop this regression model. Their results have demonstrated reasonable prediction of $PM_{2.5}$ with various features showing various levels of significance in the process. Simplicity and smart phone readiness of this model can promote air pollution awareness. However, this model also has not addressed the uncertainty associated with photos captured by a camera.

Zhan et al. [118] have proposed a standard haze image dataset which covers all sorts of haze images of the same place ranging from haze-free image to extremely haze image along with related weather and air quality information. The database also offers mean opinion score (MOS) for every image as subjective evaluation of haze severity. They have also offered an innovative no-reference image quality assessment (IQA) technique to evaluate the quality of haze images. They have analyzed factors which cause degradation of image quality for this purpose. Experimental results of IQA on this haze database have turned out to be consistent with subjective evaluation. They have demonstrated superior performance of their proposed method over spatial and spectral entropies (SSEQ) and Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE). Even though, they also have not considered image data uncertainty in their IQA method.

All of these air pollution prediction methods described above utilize sensor data in different ways. While some of these methods use deep learning on numerical sensor data, some frameworks directly apply computer vision techniques on images, without necessitating any numerical sensor data. Therefore, it can be stated that none of these works have dealt with both numerical sensor data as well as image sensor data simultaneously. Neither did these

approaches address various types of uncertainty concerning such sensor data. Hence, inspired by the success of multimodal learning (as explained in Sect. 4.1), this research lays emphasis on considering both numerical sensor data and image sensor data in an integrated manner to enhance prediction accuracy while handling associated uncertainty.

2.6 Summary

This chapter has explored sensor data streams, predictive analytics and introduced air pollution prediction as application area of sensor data streams. Moreover, this chapter has also clarified sensor data uncertainties and brought to notice existing methods to predict air pollution. The next chapter will highlight research methodology, the role of deep learning in air pollution prediction and our system architecture.

3 METHODOLOGY AND SYSTEM ARCHITECTURE

This chapter describes research methodology as well as defines deep learning along with its features, challenges and applications. It also explains several methods of deep learning and how air pollution can be predicted from images using deep learning architecture. The chapter is finished with demonstration of our system architecture.

3.1 Design Science Research (DSR)

Information Systems (IS) design research aims to improve artifact design knowledge [40]. DSR renders proper guidelines to assess and go through an artifact with regard to a research project [32]. Inspired by this, we have incorporated DSR methodology into this research for developing our proposed integrated model.

It facilitates the functional performance of an artifact for having in-depth analysis. Application areas of DSR include various algorithms, human-computer interfaces, process models, languages etc. Compliance with DSR approach has made it possible for us to develop the integrated model of BRB and Deep Learning in a planned and reliable manner. Fig. 4 shows the iterative steps of DSR flow.

First phase of DSR concerns identification of a research problem to motivate the researchers to come up with a solution. In our case, the problem is the erroneous prediction over sensor data in uncertain situation. Second phase aims to propose a solution against that research problem which, in our thesis, is the integrated approach of data-driven and knowledge-drive techniques to address the erroneous prediction issue. Third phase focuses on design and development of the proposed solution. In this thesis, we have developed a combined system of BRB and CNN using python programming language and keras neural network library. Fourth phase is with regard to the demonstration of developed solution. We have demonstrated our developed system by taking into account air pollution prediction as our use case. Evaluation is done in the fifth phase of DSR. We have compared the prediction accuracy of our system with other existing algorithms to prove higher accuracy of our solution. Final step of DSR is the communication of the whole research through a formal

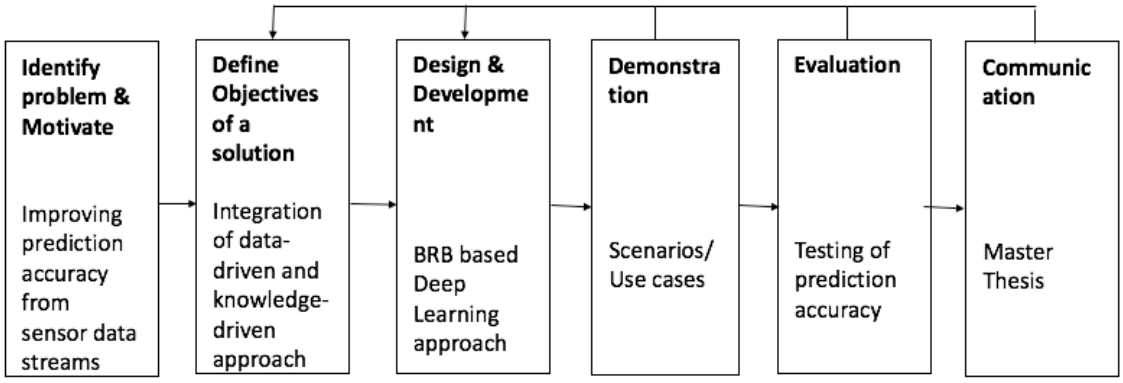

Fig. 4. Iterative steps of DSR.

publication. In our case, we are disseminating the research work in the form of a Master Thesis report. However, even after the communication phase, DSR allows to go back to second or third phase if any flaw of the proposed solution is detected or scope of further improvement is identified. Thus, DSR ensures consistency between theory and practice throughout the whole research making room for continuous improvement [104].

3.2 Definition and Features of Deep Learning

Artificial Intelligence (AI) is intended to exhibit machine intelligence by following human learning and reasoning as high as possible. “The Turing Test” of 1950, proposed by Alan Turing, was a satisfactory explanation of how a machine could imitate a human brain [20]. He talked about machine which could learn from experience and alter its own instructions. AI consists of several specific research sub-fields. Machine learning is a subset of AI domain which apply statistical techniques to empower machines to learn with experience. Deep learning is a subset of machine learning which refers to the computation carried out by neural network with multiple hidden layers.

Machine Learning, being a subset of AI, makes machines intelligent so that the machines can learn and work by themselves with minimum human intervention. Thus, machine learning makes machines mimic human intelligence. The phrase was coined by Arthur Samuel in 1959, expressing it as, “the ability to learn without being explicitly programmed” [57][79]. AI

without Machine Learning would result in writing millions of lines of codes with decision-trees and composite rule sets. So, rather than hard coding a software program with thousands of lines of codes, machine learning algorithm trains itself based on the input data and gradually adjusts itself with the hidden data pattern to perform accurate analytics.

For example, machine learning facilitates computer vision to detect an object in image or video. For instance, users tag pictures having a car in them versus those that do not. Then, the machine learning algorithm develops a model which can properly label a picture with car or not as well as a human. Once the accuracy level becomes reasonably satisfactory, the machine has "learned" the appearance of a car. Further, in health informatics, diagnosis and medical advices produce a rich database which can be processed by machine learning algorithms to predict proper treatments and advise patients accordingly.

Deep learning, being a subset of machine learning, is an effective approach to complement machine learning. This concept first came to light in 2006 [100]. At the beginning, it was known as hierarchical learning [78]. It mainly covers pattern-recognition related research fields. Clustering, reinforcement learning, Decision tree learning, Bayesian networks etc. are also some other approaches to this effect [57][59].

Deep learning is mainly inspired by the structural function of human brain, specifically, interconnection among the neurons of the brain. It is an application of neural network, where each layer consists of several neurons as shown in Fig. 5. Each neuron has an activation function which takes input and produces a new output. There is an input layer, output layer and multiple hidden layers in between. Each hidden layer learns a certain feature, such as, curves/edges in image recognition. Output of one layer is fed as input to next layer. It is this multiple layering that signifies the term 'deep' of deep learning. Depth is created by creating multiple hidden layers as opposed to a single layer. These extra layers enable deep learning algorithms to learn features more deeply and rigorously as shown in Fig. 6.

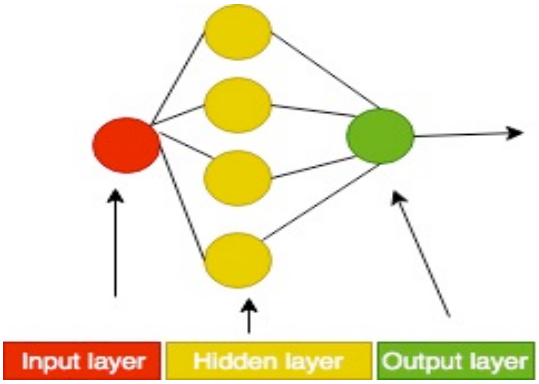

Fig. 5. Simple neural network with one hidden layer.

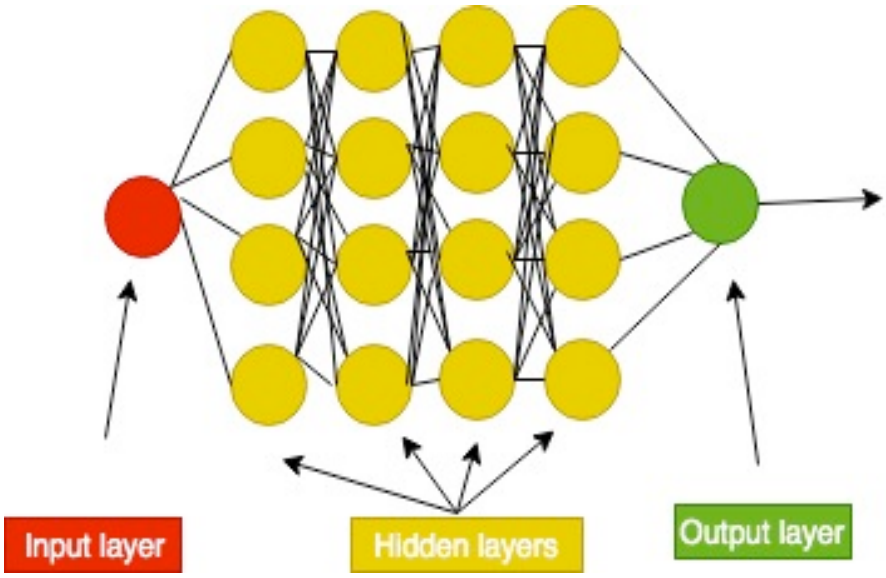

Fig. 6. Deep Learning neural network with multiple hidden layers.

Deep learning concerns two key parameters: nonlinear processing in multiple layers and learning in supervised or unsupervised way [8]. Nonlinear processing in multiple layers means that the present layer takes output of the previous layer as input and it goes on in this way. Hierarchy among the layers is determined based on the weight of the connection between every two layers. This connection weight reflects importance of concerned data. On the other hand, supervised and unsupervised learning depends on whether there is a labeled dataset or not.

There are two types of supervised learning: classification and regression [15]. In terms of classification, output variable is a category, such as, ‘male’ or ‘female’, ‘disease’ or ‘no disease’, ‘fraudulent’ or ‘authorized’. For example, a classification model will predict an email to be ‘spam’ or ‘not spam’. Decision tree, Random Forest, Logistic Regression, Naïve

Bayes etc. are classification models. If the output variable is real or continuous value, it will be regression. Predicting a person's age or weight, house price, how many copies of an album will be sold next week etc. are the examples of prediction. Linear regression is a regression algorithm.

Unsupervised learning is "learning without a teacher" [26]. It is also of two types: clustering and association [39]. Clustering is grouping a set of objects based on similar features. The objects of one group have different characteristics than the object of other groups. Association means to discover meaningful relation between variables in a large database. For example, if an online shopper has already purchased several products, the association recommends another similar product to that shopper based on the purchased products. Previous shoppers' preferences, product similarity etc. influence such recommendation of the association method.

There is another type of learning which is called reinforcement learning. An agent learns through trial-and-error interaction in a dynamic environment in this learning category [55]. An agent is programmed by reward and punishment with no specific instruction on how the task will be achieved. Chess game is an example of reinforcement learning. Fig. 7 illustrates all of these learning categories.

Deep learning algorithms, such as, deep neural networks, deep belief networks, recurrent neural networks etc. have been applied to various areas including image processing, machine translation, natural language processing, chat-bot development, social network filtering, health informatics, medicine design, chess game programs, where their output has turned out to be identical to, and in some cases, better than human produced output.

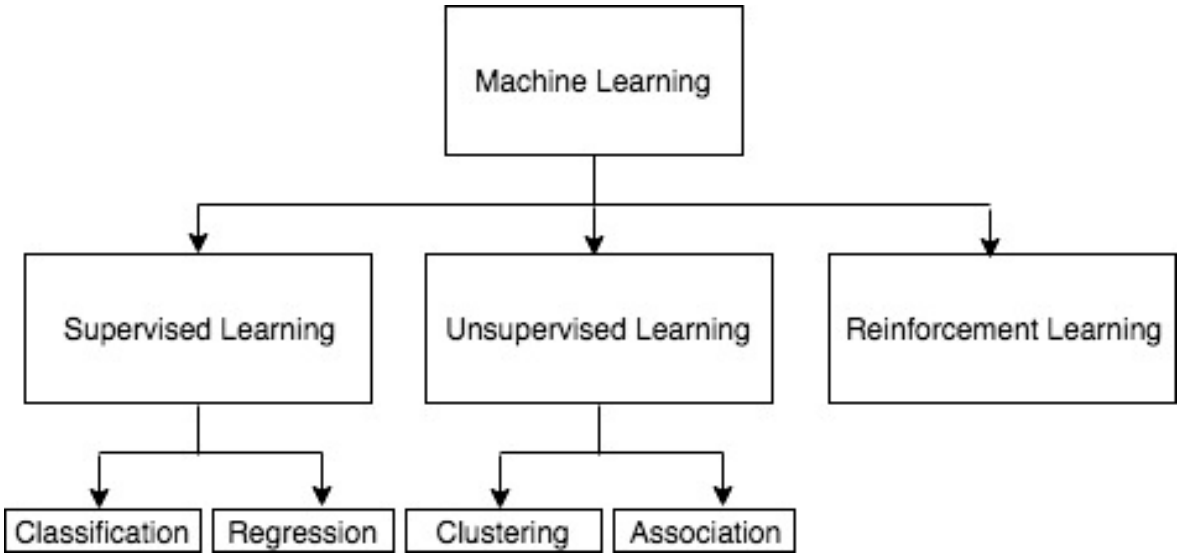

Fig. 7. Machine Learning Categories.

AI and IoT are related with each other as one complements the other. Machine learning and deep learning have been contributing to the advancement of AI significantly over the past few decades. Machine learning and deep learning necessitates huge amount of data to work, and these data are being produced by billions of sensors deployed as part of IoT. Thus, IoT strengthens AI to realize its objective.

3.3 Challenges of Deep Learning

Two major challenges concerning deep learning are overfitting and computation time.

Generally, a deep learning dataset is split into training data and testing data, where 80% of the whole dataset is used as training data and the remaining 20% as testing data [4]. Deep Learning is vulnerable to overfitting due to the extra hidden layers, which are responsible for modeling sparse dependencies of the training data. Various regularization methods including Ivakhnenko's unit pruning, weight decay and sparsity can be applied over training data to address overfitting. Moreover, dropout regularization randomly removes some of the hidden layers during training phase to reduce rare dependencies. Finally, training sets with small amount of data can be increased in size by augmenting data through different techniques, such as, cropping and rotating to minimize the overfitting risk. On the other hand, number of hidden layers, number of neurons per layer, activation function calculation, learning rate etc. necessitate high computation cost and time. This computation time can be optimized by

applying various methods, such as, batching, parallel processing, adopting multi-core architectures e.g., GPUs, Intel Xeon Phi etc., cloud computing, high-bandwidth communication network and so on.

3.4 Methods of Deep Learning

Deep Learning is based on the principle of neural network. There are different settings of neural network based on which there are different deep learning methods. There are separate application areas for each of these methods as well. Three major classes of deep learning are furnished below.

Multi-layer Perceptron (MLP)

Perceptron is a computer model which imitates the structure of biological brain to acknowledge objects. It is an algorithm for executing binary classification, i.e., it predicts whether an input object belongs to a certain class or not, such as, bus or not bus, aircraft or not aircraft etc. [116].

A perceptron performs linear classification which classifies input by separating two classes using a straight line. It predicts a single output based on multiple real-valued inputs through a non-linear activation function where output y is defined as

$$y = \varphi(w^T x + b) \tag{2}$$

where w refers to the weight vector, x symbolizes the input vector, b is the bias and φ is the non-linear activation function. Step, tanh, ReLU, sigmoid etc. are the examples of activation function.

However, single layer perceptron is not capable of making non-linear classification, which created room for multilayer perceptrons. MLP is a deep, artificial neural network which consists of more than one perceptron [3]. It has an input layer, an output layer and several hidden layers between input and output layer. Fig. 8 shows the MLP architecture. These hidden layers formulate the computational system of MLP. This neural network deals with

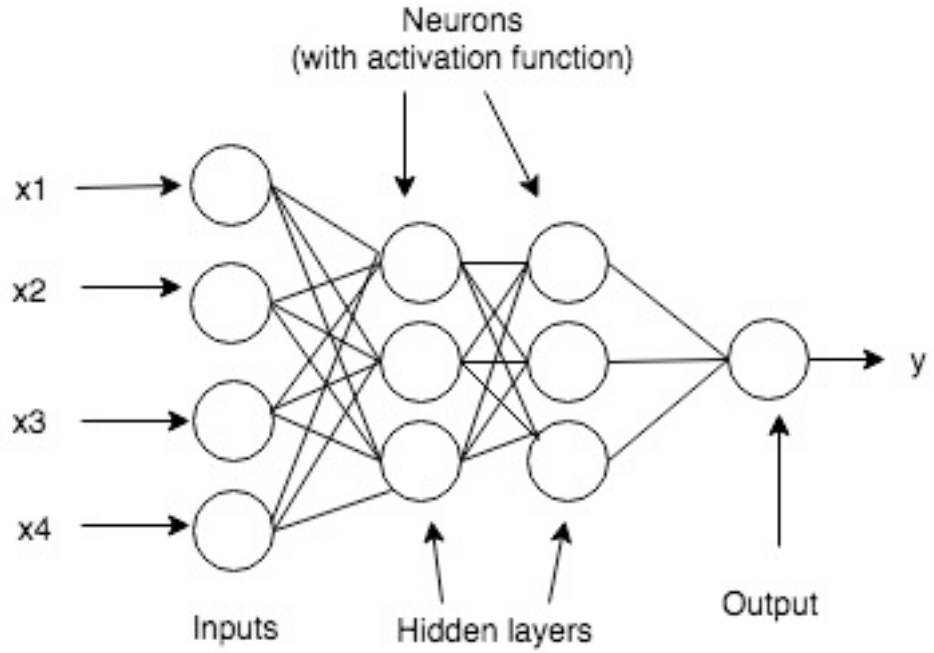

Fig. 8. Multilayer Perceptron (MLP).

supervised learning problems, where it trains itself by a set of training data consisting of input-output pairs and learns the correlation between those inputs and outputs to discover actionable insight. Such insight results in classification/regression prediction. Input/output length of MLP is fixed. Activation functions used by neurons of MLP include ReLU, sigmoid, step, tanh for modelling non-linear relationship between input and output.

Backpropagation is a technique which is used to adjust weights and bias of activation function with a view to minimizing the error gap [107]. The error is calculated based on the difference between the ground truth labels and the predicted output, which results in a gradient. This gradient is then reduced with gradient based optimization algorithm, such as, stochastic gradient descent. This process continues until the error cannot be minimized any further. This state is termed convergence. As MLP training is done by labeled dataset, it falls under supervised learning approach [85]. It is applied on tabular data (csv, spreadsheet).

Convolutional Neural Network (CNN)

This class of deep, feed-forward artificial neural networks is mainly intended to perform analytics on visual images. Convolutional network structure is based on the biological connectivity shape between neurons, which is similar to the structure of animal visual cortex [31][64]. Individual cortical neuron receives stimuli in its concerned receptive field and acts

upon it. The entire visual field is covered by the partially overlapped receptive fields of various neurons.

Apart from an input and an output layer, CNN consists of multiple hidden layers between input and output layer. Convolutional layers, pooling layers, fully connected layers and normalization layers are the CNN hidden layers [59][65]. CNN uses convolution and pooling functions as activation functions unlike other neural networks which use conventional activation functions.

Convolution function deals with two inputs – one is the input image matrix and the other is the kernel/filter matrix. This kernel matrix is then applied on the input image to produce an output image. Kernel matrix is multiplied by the input matrix to compute a modified signal [74]. A convolution function, as a dot product of input function, f and kernel function, g, is defined as

$$(f * g)(i) = \sum_{j=1}^{m} g(j).f(i - j + m/2) \quad (3)$$

where m is the total number of cells, i is the current cell and j refers to all the m cells successively. Each cell of the kernel matrix is multiplied by a certain cell of the input image matrix. Multiplication results of all the cells of kernel matrix with input matrix are then added. This summation result is the output of convolution function. This output is called convolved feature map. Size of this feature map is same as size of the kernel matrix. ReLU is applied to this feature map to bring non-linearity by setting negative pixels to zero and performing element-wise operation. Multiple convolution and ReLU layers are applied on original image to identify hidden features and patterns of the image in the form of a matrix. This matrix focuses on target features of an image, such as, detecting a curve or circle of an image, while discarding unnecessary segments of that image. This output is called rectified feature map [117].

However, feeding raw input images to the CNN is likely to result in classification/prediction performance with low accuracy [82]. It happens due to hazy edges of the objects of an image. Such hazy edges make it difficult for feature maps of CNN to detect features of the image. Therefore, applying various preprocessing techniques, such as, Mean Normalization,

Standardization, Zero Component Analysis (ZCA) on the raw input images prior to feeding the CNN improves the accuracy of CNN output [82]. Such preprocessing techniques make the edges of the objects of an image more prominent. Feature detection based on these prominent edges becomes easier by feature maps of CNN. This preprocessing is different from machine learning preprocessing which does not have any feature extractor. In machine learning, preprocessing is done to transform raw data into feature vector or higher level representation, which is then fed to the machine learning. On the other hand, preprocessing in CNN is carried out to make the edges of an image sharper, so that the feature maps can detect features more accurately. CNN can learn the features by itself, unlike the machine learning algorithms, where manual training is necessary to teach the features. This automated feature-learning characteristic, without necessitating any human intervention, makes CNN outperform other traditional machine learning algorithms.

Rectified feature map is fed as input to the pooling function which performs sample-based discretization by reducing dimensionality of this feature map. Output of the pooling function is called pooled feature map. Pooling is of two types- max pooling and min pooling. Max pooling picks up the maximum value from the selected sub-region. It chooses the brighter pixels of the image. Mathematically,

$$h_{i,j} = \max\{x_{i+k-1,j+l-1} \text{ where } 1<= k <= m \text{ and } 1 <= l <= m\} \quad (4)$$

where (i, j) refers to each matrix cell position and m refers to all the cells of concerned sub-region. Max pooling is useful when the background of the input image is dark and main focus is on the lighter pixels. This is particularly suitable for the separation of features which are very sparse [11]. We have applied max pooling over the air pollution images in this research due to sparsity of features of these images. Min pooling opts for the minimum value of the concerned sub-region. This pooling is appropriate when the background of the image is light and key focus is on the dark pixels. There is a fully connected layer of neurons after the last pooling layer of CNN. Output of the last pooling layer is fed as input to this fully connected layer. Neurons in this layer have full connections with all neurons of the previous layer. Fully connected layer learns non-linear functions of features and performs

classification/regression based on the features extracted by the previous layers. CNN architecture is illustrated in Fig. 9.

CNN's learning approach falls under supervised category. The labeled dataset is split into training and testing data. CNN learns image features based on the labeled images of training dataset. In this research, we have considered haze-relevant features of air pollution images to evaluate the perceptual hazy density and scene depth. Then CNN tests the accuracy of its prediction against the images of testing dataset. It does so by comparing its predictive output with the original labels of testing images. Like MLP, input/output length of CNN is also fixed.

However, CNN cannot deal with uncertainty, such as, hardware defect, ignorance, camera malfunctioning, blurred images, camera glasses full of scratch [71]. Hence, addressing such unexpected cases is key to upholding prediction accuracy.

Recurrent Neural Network (RNN)

Recurrent Neural Network (RNN) is a class of artificial neural network. A general neural network processes an input through a number of layers to produce an output, presuming that two successive inputs are not related with each other. However, this assumption is contradictory with a number of real-life cases, such as, predicting stock market price at a certain time or predicting next word of a sentence in a sequence, where prediction output is dependent on multiple previous observations [65]. RNN is termed recurrent because, it repeats the same task over every element of a sequence where previous computations influence every output. It retains state in an arbitrarily long context window to render

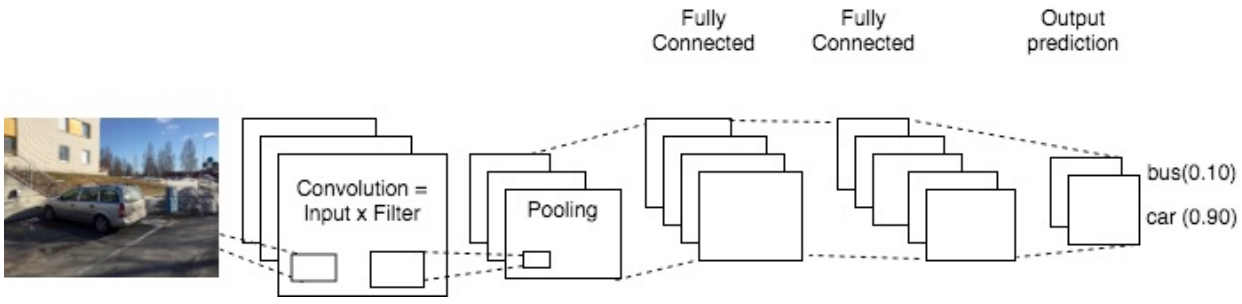

Fig. 9. Convolutional Neural Network (CNN).

information [73]. RNN stores information of all the calculations made so far in its own memory. Thus, it utilizes information over a long sequence of elements. However, practically, it can look back only a few steps [9]. This memory feature to store information has made RNN an appropriate algorithm for tasks, such as, unsegmented connected handwriting recognition, machine translation and speech recognition.

RNN represents two classes of networks with a similar structure. One is infinite impulse while the other is finite impulse [90]. Both of the classes exhibit temporal dynamic behavior [5]. An infinite impulse RNN is a directed cyclic graph that cannot be unrolled [65]. On the other hand, a finite impulse RNN refers to a directed acyclic graph which is possible to be unrolled and replaced with a feedforward neural network. Fig. 10(a) illustrates the infinite impulse RNN, where the cycle continues to loop infinitely with weight 'W' between each run. Unrolling the single-layer infinite impulse RNN into a multi-layer network creates finite impulse RNN, as shown in Fig. 10(b). For example, if the target is to deal with a sentence of 7 words, the finite impulse RNN will consist of 7 layers, with 1 layer for each word. That is why, input/output length of RNN is arbitrary. Such RNN's generated output is different at different time instances. For example, at time t, x_t is the taken input, which is then processed

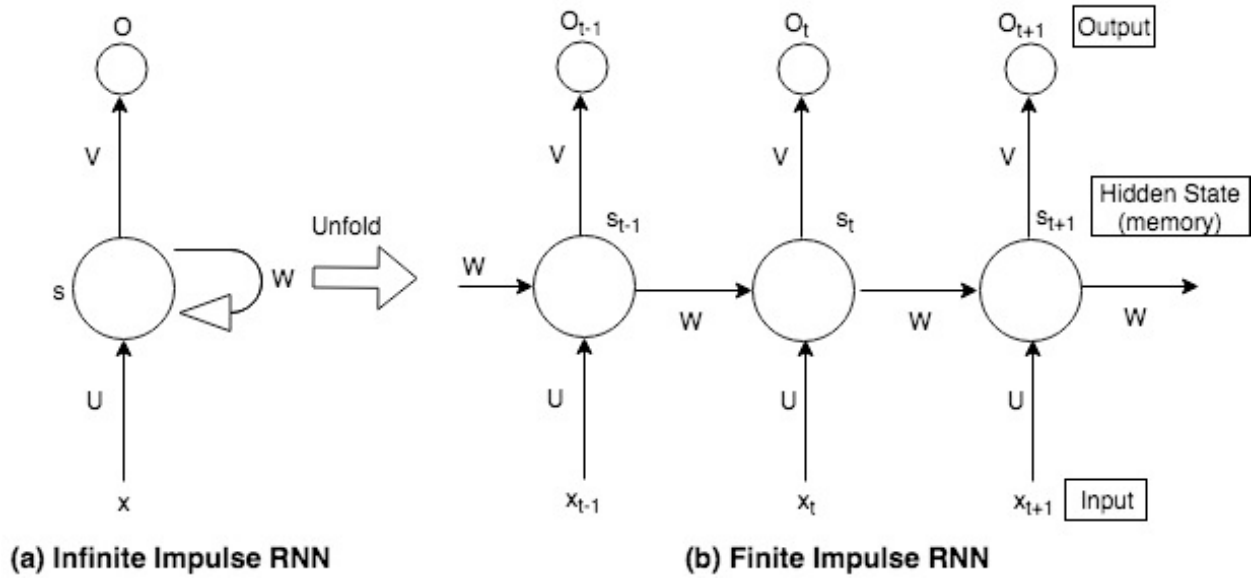

Fig. 10. Recurrent Neural Network (RNN).

through hidden state s_t leading to output o_t. This s_t, which functions like a memory of the network and computed using the previous hidden state and the current input, is defined as

$$s_t = f(Ux_t + Ws_{t-1}) \tag{5}$$

where f is the activation function, such as, softmax, ReLU, tanh and U is a variable concerning the input. Finally, o_t , computed by applying activation function on s_t , is defined as

$$o_t = \text{softmax}(Vs_t) \tag{6}$$

Unlike traditional deep neural network, RNN uses same value of all the parameters (U, V, W) through each step to repeat same task at each step with various inputs. It is a network of nodes resembling neurons, with each node connected with every other node with a directed one-way connection. Activation function of each node differs with time [65]. Each connection (synapse) has a real-valued weight (W) to accommodate the significance of concerned data inside the whole network.

Long Short-Term Memory (LSTM) units (or blocks) constitute building units of RNN layers [43] [87]. Enhancing the memory capacity of RNN is the main objective of LSTM. A RNN combined with LSTM units is sometimes known as LSTM network. LSTM unit consists of a cell, an input gate, an output gate and a forget gate [33]. The cell works as a memory to retain values over a reasonable time period. Each of the three gates is like a neuron, which computes activation of a weighted sum by using activation function. Thus, these gates control the flow of values which go through the LSTM connections, validating the term "gate". These gates communicate with the memory cell through a connection. CNN, combined with RNN, can generate caption for images where output of CNN is used as input of RNN [101][23]. Caption generation process is demonstrated in Fig. 11. RNN's learning approach is unsupervised [61]. It learns from data stored in the memory cells over a certain time period. However, Reinforcement Learning can improve the performance of RNN further by minimizing bias or excessive repetition [51]. Also, RNN is a generative model as it generates next predictive output of a sequence which is consistent with the earlier elements of that sequence [66].

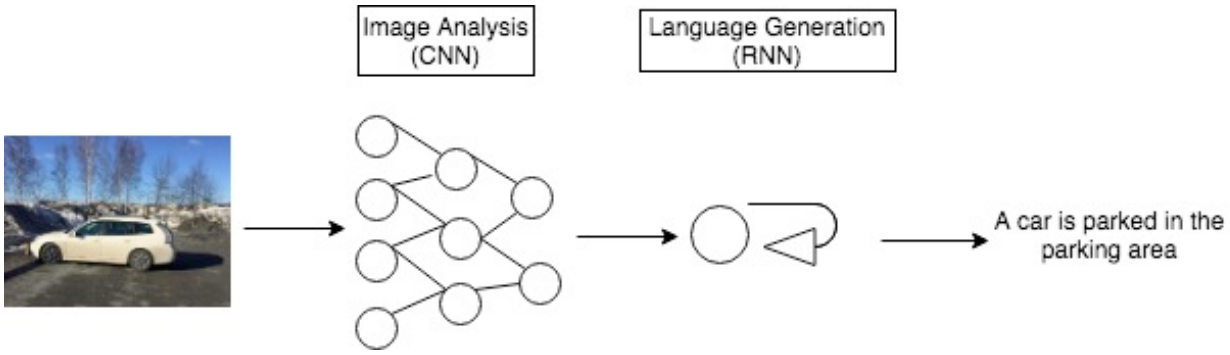

Fig. 11. Captions generated by RNN with CNN as extra input.

CNN and RNN focus on separate problem domain. CNN extracts spatial features and RNN extracts temporal features. This feature extraction characteristic of CNN reduces number of trainable parameters which is useful for processing high-dimensional images in computer vision, where value of one pixel depends on its neighboring pixels as well. Moreover, CNN is a deeper model than RNN. Such depth is quite effective to capture features of an image. Shallow depth of RNN hampers its performance accuracy for image processing. RNN is designed for sequence prediction. That is why, it has memory to retain information over a long sequence. On the other hand, CNN is specifically designed for image recognition as it can extract features of an image quite effectively through convolution operation. Therefore, we have applied CNN, instead of RNN, to recognize air pollution images in this research.

Comparative analysis of various characteristics of the aforementioned deep learning models are shown in Table 2.

Table 2. Comparative Analysis of Deep Learning Models.

Feature	MLP	CNN	RNN
Application area	Tabular data (csv, spreadsheet)	Image/Video data	Text/Speech data
Learning	Supervised	Supervised	Unsupervised or Reinforcement
Input/Output length	Fixed	Fixed	Arbitrary
Activation Functions	ReLU, sigmoid, step, tanh	convolution, pooling, ReLU, softmax	softmax, tanh, ReLU
Classification prediction	Yes	Yes	Yes
Regression prediction	Yes	Yes	Yes
Sequence prediction	No	No	Yes
Memory	No	No	Yes (LSTM)
Generative Model	No	No	Yes

It can be noted in the aforementioned table that same activation function ReLU is used in MLP, CNN and RNN. One advantage of ReLU over other activation functions is less probability of the gradient to be vanished. ReLU is defined as h = max (0, a) where a = Wx + b. The gradient has a constant value when a > 0. In contrast, gradient of sigmoids become smaller as the absolute value of x increases. The constant gradient of ReLUs lead to faster learning. Since ReLU is zero for all negative inputs, it is likely for a given input not to activate at all. Such sparsity results in concise model which has higher predictive power and less overfitting/noise. Moreover, ReLU is computationally efficient as it just picks the maximum value from max (0, a), unlike sigmoid like functions which perform expensive exponential operations. Such expensive operation makes network of sigmoid like functions dense and less efficient. Therefore, ReLU is a widely used activation function in deep learning models.

3.5 Applications of Deep Learning

There are numerous application areas of deep learning models ranging from healthcare to face recognition. Several significant applications of deep learning are described below.

Image Processing

CNN can recognize iris, with 99.35% accuracy, which is far better than traditional iris sensors [8]. Facial recognition is done by deep learning based digital image processing. Google, Facebook and Microsoft have deployed their own deep learning based face recognition models [98]. Sighthound Inc., for instance, can detect emotions, in addition to age and gender, by dint of their own deep CNN algorithm [22]. Processing outdoor images to predict air pollution level is also a prominent application area of deep learning based model.

Medicine

Deep Learning is used in healthcare domain as well. It can predict Alzheimer disease by processing Magnetic Resonance Imaging (MRI) image of a human brain [41][99]. Performance of Deep Learning methods are reliable in the area of Optical Coherence Tomography (OCT) as well. Noncoding RNA is a problem of biology where sophisticated computational techniques like deep learning has shown satisfactory performance [76]. Deep

Learning is applied on drug discovery and evaluation purpose to reduce risk and save resources. For instance, toxicity is a major reason of removal of a drug from production. This type of toxicity, such as, hepatotoxicity (liver toxicity) can be predicted using deep learning approach with raw chemical structure where no complex encoding process is necessary [111].

3.6 Predicting air pollution from images

In addition to sensor data, analyzing outdoor images is also an effective way to predict air pollution level. Combining two data sources, instead of relying on only one, results in improved prediction accuracy. Besides, sensor devices are costly with limited coverage area. On the other hand, outdoor images can be captured even by a smart phone anywhere anytime. Thus, image based air pollution prediction is cost-effective and handy with wider coverage area. CNN is a suitable image analytics algorithm to analyze outdoor images and predict air pollution level accordingly. Moreover, image presents a holistic view of the overall environment, which a mere numerical data of sensor cannot demonstrate. Such images, captured by camera sensor, also facilitates visual analysis of polluted sites by the policy-makers. This analysis leads the decision-makes to take critical decisions, such as, shutting down brick kilns, closing educational institutions, stopping the fossil fuel burning, diverting traffic and pedestrians to a less polluted path and so on. Fig. 12 highlights opportunities of such image based prediction.

By using geospatial data, city authorities can integrate the polluted sites with geographical map and display real-time pollution update through web service to the citizens to warn them in advance and take precautionary measures to minimize risk. Such digital dissemination of air pollution related information makes image-based prediction an appropriate approach for both city authority and citizens.

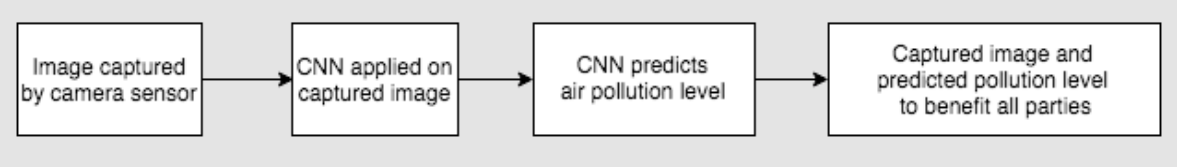

Fig. 12. Opportunities for image based air pollution prediction.

3.7 System Architecture

We apply a combined approach of BRB and Deep Learning on sensor data streams to predict air pollution. We use Deep Learning method CNN on outdoor images to predict the level of air pollutant $PM_{2.5}$. Initially, CNN is trained by different outdoor images of the same place with varying level of $PM_{2.5}$. Thus, it learns the representation of which image has what level of $PM_{2.5}$. Upon completion of the training, a new image of a certain time of the same place is fed to this trained CNN. Next, CNN performs analytics on this new image based on its training representation to predict the $PM_{2.5}$ level of the concerned place.

This CNN prediction output concerning the level of $PM_{2.5}$ goes to BRB as input. Further, numerical values of the level of $PM_{2.5}$ of the same place and same time instance as the outdoor image generated by the physical sensor devices are also fed to BRB as input. Thus, BRB has two input values with regard to $PM_{2.5}$ level both from CNN prediction and sensor reading. In this architecture, these two inputs constitute two antecedent attributes of BRB.

We predict the air pollution level in terms of AQI, which is a numerical scale with corresponding color code and is divided into several fixed ranges [86]. Public bodies use this AQI to disseminate outdoor air pollution level information to citizens. In addition to pollution level, this index warns people of potential health risks, which are critical for children, elderly people and people with respiratory diseases. We use the breakpoint table, developed by EPA based on six common air pollutants, to predict the AQI value against the level of $PM_{2.5}$ [29].

Based on the two antecedent attributes concerning two values of $PM_{2.5}$, BRB carries out its inference mechanism and predicts AQI value as single numerical crisp value (as demonstrated in Sect. 4.3). Moreover, rather than showing one single AQI category, BRB shows all the six AQI categories with concerned belief degrees to let a citizen gain a holistic assessment with regard to the overall pollution scenario of the outdoor environment (as demonstrated in Sect. 4.4). Fig. 13 illustrates our proposed system architecture.

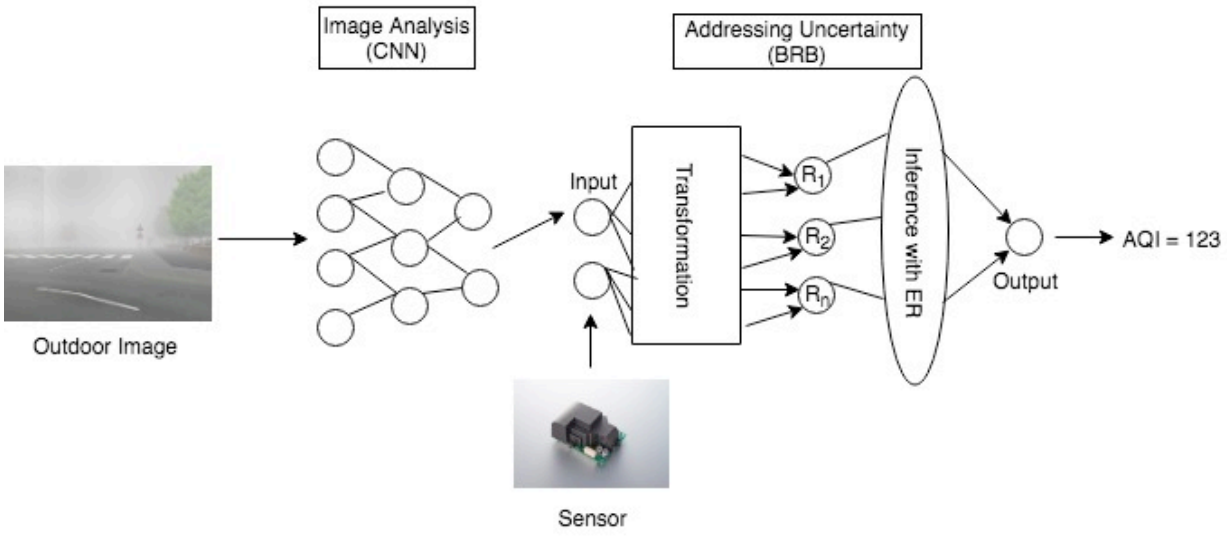

Fig. 13. System Architecture.

3.8 VGGNet

CNN has several architectures, such as, AlexNet, VGGNet, GoogLeNet [67]. We have used VGGNet in our architecture as it has uniform architecture to extract features from images.

GoogLeNet, developed by Google Inc., uses several smaller convolutions along with inception modules [94]. It has 22 convolutional layers with 4 million parameters. Its filter matrix is of size 1x1, 3x3 and 5x5. There are total 100 layers in this network including 5 pooling layers. It uses RMSprop, a variant of Stochastic Gradient Descent, to adapt learning rate of each of the parameters. It also applies batch normalization to increase learning rate. On the other hand, AlexNet consists of five convolutional layers, followed by max-pooling layers, three fully connected layers and 1000-way softmax to produce distribution over 1000 class labels [62]. It has 60 million parameters and 650,000 neurons with two parallel CNN lines trained on two GPUs. This architecture uses ReLU activation after each convolutional and fully-connected layer.

VGGNet is a CNN architecture proposed by Visual Geometry Group (VGG), University of Oxford, UK [91]. It uses (3x3) filters in the convolution layer. It has total 16 convolutional layers and 3 fully connected layers. It uses small 3x3 receptive fields (with stride 1) over the input image matrix throughout the whole net to perform convolution operation with the filter

and compute feature map. Applying small receptive fields several times is more advantageous than applying one large receptive field once. First, several non-linear rectification layers can be incorporated, instead of a single rectification layer. It makes decision function more discriminative. Second, small receptive fields decrease the number of parameters. VGGNet architecture has total 144 million parameters to learn image representation.

Depth of representation is beneficial for increasing classification accuracy. Deep networks with small filters outperform shallow networks with large filters. Though GoogLeNet is deeper (22 convolutional layers) than VGGNet (16 convolutional layers), GoogLeNet's network topology is more complex than VGGNet. GoogLeNet decreases spatial resolution of the feature maps more aggressively in the first layers to reduce the amount of computation. On the other hand, AlexNet is a shallow network (with only 5 convolutional layers) compared with VGGNet. Both GoogLeNet and AlexNet have been outperformed by VGGNet in terms of classification accuracy. VGGNet also stood first in localization and second in classification track in the ImageNet Challenge 2014. Inspired by this, we have adopted VGGNet as our CNN architecture. Presently, it is the most preferred choice in the community for image recognition as well.

3.9 Summary

This chapter has covered DSR, deep learning and system architecture. The chapter has also focused on the benefits of image-based analytics for performing air pollution prediction. Next chapter will demonstrate mathematical model to combine BRB and Deep Learning.

4 INTEGRATED APPROACH OF BRB AND DEEP LEARNING

This chapter mathematically demonstrates how the integration of deep learning with BRB has been achieved in this research. The chapter begins with the rationale of the integrated approach. Then it sheds light on neural network and its integration with BRB, enabling the prediction of AQI with higher accuracy. This is followed by the demonstration of assigning belief degree to the various categories of AQI against the predicted AQI value.

4.1 Rationale

Multimodal learning means to relate information from multiple sources [23][81]. Instead of relying on single modality, it learns representation over multiple modalities. For example, in terms of speech recognition, integrating audio and visual modalities results in higher performance than using only audio modality. In particular, visual modality provides information on place of articulation and lip movement, which is helpful to distinguish between speech with alike acoustics.

Inspired by the efficacy of multimodal learning, we propose to integrate two approaches: BRB and deep learning, to predict AQI with higher accuracy. As air pollutant, we take $PM_{2.5}$ and map its concentration to corresponding AQI value. However, for computing the concentrations of $PM_{2.5}$ in the air, instead of relying only on sensor reading (single modality), we take both sensor modality and image modality. Thus, we adopt multimodal learning in our proposal to improve prediction accuracy. We apply BRB to reduce uncertainty concerning the numerical sensor data of $PM_{2.5}$. Moreover, we apply deep learning to reduce noise of raw image data. Deep learning, with its multiple hidden layers, sequentially decreases and removes contaminations on such raw data and extracts target features from image for higher-level representation.

The following subsections explain our proposed integrated approach as well as distributed categorization of AQI.

4.2 Deep Representation

This part transforms the captured image into high-level representation using deep learning concept. As Deep Learning architecture, we apply CNN as it is specifically customized to perform analytics on visual images. Convolution operation, pooling operation and fully connected layer of CNN have already been described in Sect. 3.4. As CNN architecture, we have used a smaller version of VGGNet (as explained in Sect. 3.8) in this research for predicting $PM_{2.5}$ concentrations from the outdoor images. After fully connected layer, VGGNet uses softmax activation function to predict probability between 0 and 1 for each of the classes against the input image.

Dimension of the air pollution images we have used in this research is 640 x 480 pixels (length x width) with RGB color space. So, number of channels of the input images is 3, making the volume of input images 640 x 480 x 3. There are 5 convolution layers and 1 fully connected layer in our VGGNet. First convolution layer has 32 filters, second and third convolution layer has 64 filters while fourth and fifth convolution layer has 128 filters each. Our kernel matrix size is of 3 x 3. As pooling function, we have applied max pooling. Pooling layer after first convolution layer is of size 3x3. Each of the pooling layers after remaining four convolution layers is of size 2x2. We have used ReLU as our activation function (rationale of using ReLU has been explained in Sect. 3.4). When the input distribution of VGGNet changes, it causes covariate shift [42]. When this change happens on the input of internal node of VGGNet, it is called internal covariate shift [48]. We have applied batch normalization in our VGGNet to reduce this internal covariate shift and increase learning rate. This method does so by adding an extra step between the layers, where output of the previous layer is normalized.

Dropping out 20% of the input units has been found to be optimal [92]. Driven by this, we have added 20% dropout to our model by discarding nodes from the present layer to the next layer randomly after every pooling layer. This dropout process helps the network to reduce overfitting as no single node of a layer is single-handedly assigned to predict a certain class. Output of the last pooling layer has been flattened into a single feature vector which is used by the fully connected layer for final recognition. We have added one fully connected layer which extracts 1024 features. Excluding 50% of the hidden units has turned out to be the most efficient [92]. Hence, we have dropped 50% of the neurons after fully connected layer. Our

output layer consists of 3 nodes as it computes probability for three classes (Nominal Pollution, Mild Pollution and Severe Pollution) with softmax activation. Number of epochs defines the number of times that our VGGNet will work through the entire dataset of air pollution images. We have trained our network for 75 epochs in order to learn image patterns by incremental improvements through backpropagation. Our validation set accuracy stopped improving after 75^{th} epoch.

Batch size of our VGGNet is 32, which means that the network will process 32 samples in one iteration. We have set our batch size to be small as large batch size degrades overall quality of the model [56]. As we have 2419 training images in our air pollution dataset, there are total 2419/32 = 76 iterations per epoch, where 32 refers to the batch size. We have applied Adam optimization algorithm as optimizer to compile our VGGNet model. Adam is an extension to the Stochastic Gradient Descent (SGD). SGD keeps the learning rate static for all weight updates and no change is brought to it during training. On the other hand, learning rate computed by Adam for different parameters is individual and adaptive. More so, it has outperformed two other optimization methods: Adaptive Gradient (AdaGrad) and Root Mean Square Propagation (RMSProp), as it computes an exponential moving average of gradient and the squared gradient [60]. Learning rate refers to the updated amount of weights during training. Initial learning rate of our model has been set at 0.001, as this is the default value for Adam optimizer. We have used binary cross entropy as the loss function of our model as our goal is to consider every single output label as an independent Bernoulli distribution and hold each output node responsible individually. Table 3 demonstrates the architecture of our VGGNet model. We have used multi-label binarizer for multi-label image classification. For example, Fig. 14 shows the probability of each of the 3 classes against an outdoor image, where probability that the image belongs to severe pollution class is 99.99%, mild pollution class is 43.47% and nominal pollution class is 0.56%. We have mapped this pollution level to the $PM_{2.5}$ concentrations in the air. We then normalize all the three probabilities. Thus,

$$\left.\begin{aligned} &\text{PM\_Image\_High, PIH} = 0.9999/(0.9999 + 0.4347 + 0.0056) = 0.70 \\ &\text{PM\_Image\_Medium, PIM} = 0.4347/(0.9999 + 0.4347 + 0.0056) = 0.30 \\ &\text{PM\_Image\_Low, PIL} = 0.0056/(0.9999 + 0.4347 + 0.0056) = 0.00 \end{aligned}\right\} \quad (7)$$

Table 3. CNN Architecture.

Model Content	Details
Input image size	640 x 480 x 3
First Convolution Layer	32 filters of size 3x3, ReLU, input size 640 x 480 x 3
First Max Pooling Layer	Pooling Size 3x3
Second Convolution Layer	64 filters of size 3x3, ReLU
Second Max Pooling Layer	Pooling size 2x2
Third Convolution Layer	64 filters of size 3x3, ReLU
Third Max Pooling Layer	Pooling size 2x2
Fourth Convolution Layer	128 filters of size 3x3, ReLU
Fourth Max Pooling Layer	Pooling size 2x2
Fifth Convolution Layer	128 filters of size 3x3, ReLU
Fifth Max Pooling Layer	Pooling size 2x2
Fully Connected Layer	1024 nodes, ReLU
Dropout Layer	excludes 50% neurons randomly
Output Layer	3 nodes for 3 classes, SoftMax
Optimization Function	Adam optimization algorithm
Learning Rate	0.001
Loss Function	Binary Cross Entropy

Then we employ Algorithm 1 to calculate $PM_{2.5}$ concentrations from the input image. We have determined the value of regression coefficients of this algorithm in compliance with AQI breakpoint table. This algorithm calculates $PM_{2.5}$ concentrations from our example image to be 417.99 $\mu g/m^3$.

Algorithm 1: $PM_{2.5}$_calculations_from_image (PIH, PIM, PIL)

```
if ((PIH > PIM) and (PIH > PIL)) then
        PM_Image = (150.5 + 349.9* PIH) + (150.4 * PIM)/2
else if ((PIL > PIH) and (PIL > PIM)) then
        PM_Image = (35.4 * (1 – PIL)) + ((150.4 * PIM)/2)
else if ((PIM > PIH) and (PIM > PIL)) then
        if (PIH > PIL) then
                PM_Image = (35.5 + 114.9 * PIM) + ((500.4 * PIH)/2)
        else if (PIL > PIH) then
                PM_Image = (35.5 + 114.9 * PIM) + ((35.4 * PIL)/2)
return PM_Image
```

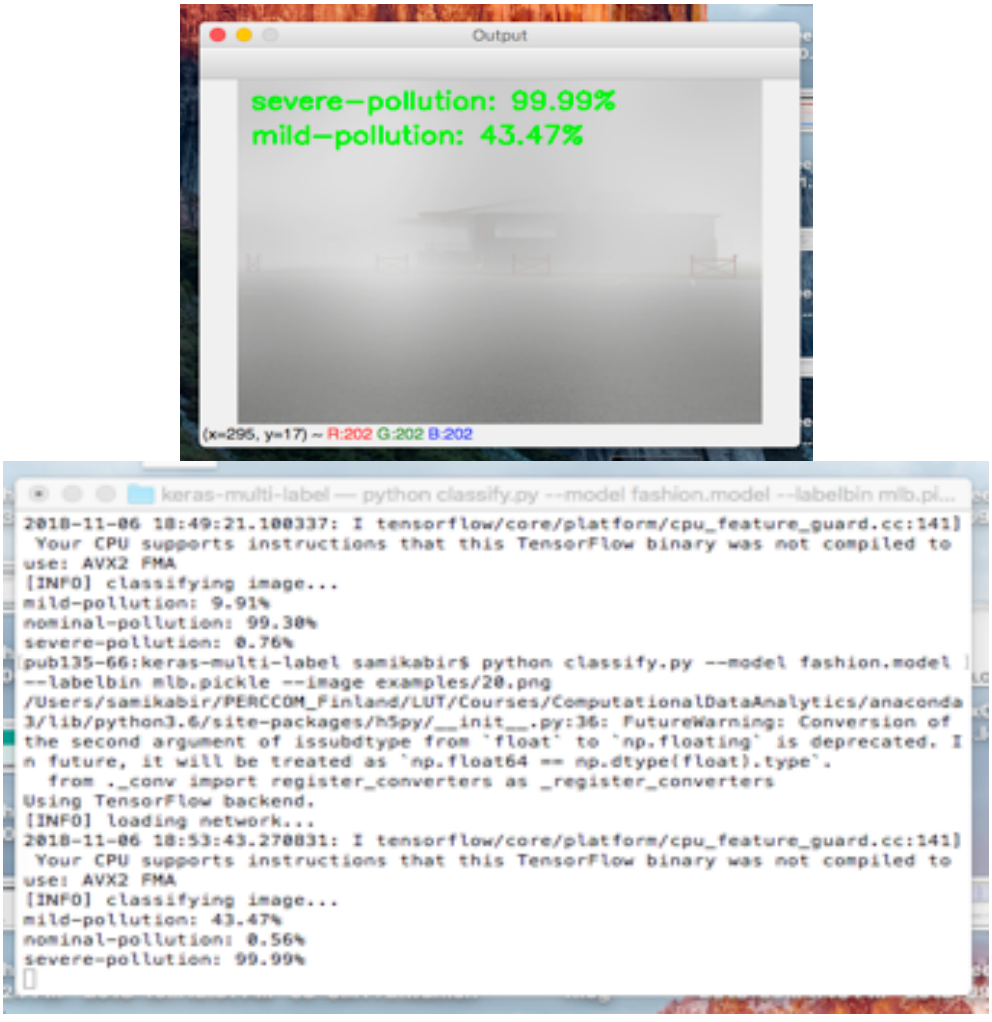

Fig. 14. VGGNet calculates probability for each class.

4.3 Integrating CNN with BRBES

This part covers the working methodology of BRBES as well as how CNN is integrated with BRBES. It clarifies how BRBES transforms input to infer predictive output. It also explains how the belief degree is updated if any of the sensor data goes missing.

BRBES consists of belief rules, which represent the detailed formation of conventional If-Then rules to accommodate uncertain knowledge concerning sensor data of $PM_{2.5}$. The reasoning approach of BRBES comprises four inference procedures – Input transformation, Rule activation weight calculation, Belief degree update and the Rule aggregation [113].

Domain Knowledge Representation

Belief degrees of the consequent attribute, rule weight, antecedent attribute weight, utility scores of the referential values of the antecedent and consequent attributes are some of the knowledge representation parameters which are reflected in the belief rules of BRB. These parameters can deal with the uncertain knowledge regarding the quantity of $PM_{2.5}$ measured by the sensor. A belief rule has two parts: antecedent part and consequent part. Antecedent part has one or more antecedent attributes with corresponding referential value, while consequent part consists of only one consequent attribute with separate belief degree for each of the referential values. There are two antecedent attributes in our antecedent part: $PM_{2.5}$ reading by sensor and $PM_{2.5}$ predictions by CNN. Each of these two antecedent attributes has three referential values: High, Medium and Low. An example belief rule can be defined as

R_k :

IF $PM_{2.5}$ (Sensor) is High AND $PM_{2.5}$ (CNN) is Medium
THEN AQI is (Hazardous, 0.60), (Unhealthy, 0.40), (Good, 0.00)

where R_k is the k^{th} rule of a rule base and {(Hazardous, 0.60), (Unhealthy, 0.40), (Good, 0.00)} are the referential values along with related belief degrees associated with the consequent attribute 'AQI'. As the sum of the belief degrees is 1 (0.60 + 0.40 + 0.00 = 1), this belief rule is complete.

The rule base of BRB can be formulated in four ways: 1) from expert knowledge; 2) using historical data; 3) from previous rule bases, if any; and 4) by creating random rules without having any prior knowledge [45]. In this research, the rule base has been developed based on the expert knowledge provided by breakpoint table of U.S. Environmental Protection Agency (EPA) [29]. Number of rules of BRBES is defined as

$$\text{number of rules} = R^A \tag{8}$$

where R is number of referential values of the antecedent attribute and A is the number of antecedent attributes. Thus, every possible combination of referential values of the antecedent attributes are taken into account to construct this rule base. As there are three referential values (High, Medium and Low) and two antecedent attributes ($PM_{2.5}$ measurements by sensor and CNN) in this research, there will be $3^2 = 9$ rules in our rule base, as shown in Table 4. Consequent attribute AQI has also three referential values (Hazardous, Unhealthy and Good). However, the belief degrees of the consequent attribute can be optimized further by training the BRBES.

Inference Procedures

The reasoning or inference mechanism of BRBES are demonstrated below.

Input transformation

The distribution of the sensor data of $PM_{2.5}$ into its referential values is called input transformation. “Low”, “Medium” and “High” referential values are assigned utility values

Table 4. Initial Rule Base.

Rule Id	Rule Weight	IF		THEN		
		$PM_{2.5}$ (Sensor)	$PM_{2.5}$ (CNN)	AQI		
				Hazardous	Unhealthy	Good
R1	1.0	H	H	1.00	0.00	0.00
R2	1.0	H	M	0.60	0.40	0.00
R3	1.0	H	L	0.60	0.20	0.20
R4	1.0	M	H	0.40	0.60	0.00
R5	1.0	M	M	0.00	1.00	0.00
R6	1.0	M	L	0.00	0.60	0.40
R7	1.0	L	H	0.20	0.20	0.60
R8	1.0	L	M	0.00	0.40	0.60
R9	1.0	L	L	0.00	0.00	1.00

as $h_{i1} = 0$, $h_{i2} = 35.5$ and $h_{i3} = 500.4$ respectively. We have set the utility value of these three referential values in line with $PM_{2.5}$ range of U.S. EPA breakpoint table. The input transformation procedure is defined as

IF $h_{i3} >= input >= h_{i2}$ THEN Medium = $(h_{i3} - input)/(h_{i3} – h_{i2})$
High = (1 - Medium), Low = 1 - Medium - High

IF $h_{i2} > input >= h_{i1}$ THEN Low = $(h_{i2} - input)/(h_{i2} – h_{i1})$
Medium = (1 - Low), High = 1 - Low - Medium

We have used labeled dataset of air pollution images (as described in Sect. 6.1) where each image's corresponding sensor reading of same time and same place is given. Corresponding sensor reading of $PM_{2.5}$ of the image we have used in (7), for predicting $PM_{2.5}$ with VGGNet, is 423 $\mu g/m^3$. This $PM_{2.5}$ sensor reading of 423 $\mu g/m^3$ is transformed into its referential values as follows:

Medium_Sensor, M1 = (500.4 - 423)/(500.4 – 35.5) = 0.17;
High_Sensor, H1 = (1-0.17) = 0.83 and
Low_Sensor, L1 = (1- 0.17 – 0.83) = 0.

On the other hand, transformation of input to another antecedent attribute, $PM_{2.5}$ predicted by CNN, into its referential values are obtained from (7):

High_CNN, H2 = 0.70;
Medium_CNN, M2 = 0.30 and
Low_CNN, L2 = 0

Rule activation weight calculation

Rule activation weight calculation necessitates the matching degree of the referential value at which the belief is matched [44]. Matching degree of k^{th} rule is defined as

$$\alpha_k = \prod_{i=1}^{T_k} (\alpha_i^k)^{\overline{\delta_{ki}}} \text{ and } \overline{\delta_{ki}} = \frac{\delta_{ki}}{\max_{i=1,\ldots,T_k}\{\delta_{ki}\}} \quad (9)$$

where T_k is the total number of antecedent attributes used in the k^{th} rule and δ_{ki} is the weight of i^{th} antecedent attribute. A rule is said to be activated once the referential values of the antecedent attributes of a rule are allotted the matching degrees [113]. Activation weight of k^{th} activated rule is defined as

$$\omega_k = \frac{\theta_k \alpha_k}{\sum_{j=1}^{L} \theta_j \alpha_j} = \frac{\theta_k \prod_{i=1}^{T_k} (\alpha_i^k)^{\overline{\delta_{ki}}}}{\sum_{j=1}^{L} \theta_j \prod_{i=1}^{T_k} (\alpha_i^k)^{\overline{\delta_{ki}}}} \quad (10)$$

where $\overline{\delta_{ki}}$ is the relative weight of i^{th} antecedent attribute in the k^{th} rule and θ_k is the rule weight of k^{th} rule. The value of θ_k stands between 0 and 1 (both inclusive).

Based on the input transformation of both the antecedent attributes, the activation weight of each of the 9 rules of our rule base has been calculated using (10). The activation weights of all the 9 rules are as follow.

Activation weight of rule 1 is: 0.58
Activation weight of rule 2 is: 0.25
Activation weight of rule 3 is: 0.00
Activation weight of rule 4 is: 0.12
Activation weight of rule 5 is: 0.05

Activation weight of rule 6 is: 0.00
Activation weight of rule 7 is: 0.00
Activation weight of rule 8 is: 0.00
Activation weight of rule 9 is: 0.00

Belief degree update

In some cases, input data of some of the antecedent attributes may remain unavailable. This is a case of uncertainty due to ignorance. For instance, AQI prediction is dependent on two antecedent attributes. However, suddenly, sensor data for one of the attributes may go missing. In such a situation, the initial belief degrees of the referential values of the consequent attribute have to be revised by a mathematical equation [113].

Rules aggregation

The rules aggregation of BRBES inference mechanism is executed by employing ER mechanism [12]. Either recursive or analytical ER approach is applied for this aggregation. However, analytical ER approach requires less computational complexity than recursive one [103]. Hence, we have applied analytical ER approach to calculate belief degree for each of the three referential values of the consequent attribute AQI. Eventual concluding result C(Y) with referential values of the consequent attribute O_j is defined as

$$C(Y) = \{(O_j , \beta_j), j = 1, \ldots, N\} \tag{11}$$

where β_j is the belief degree of O_j and N is the number of referential values of the consequent attribute. β_j can be computed by using analytical ER algorithm, defined as

$$\beta_j = \frac{\mu \times \left[\prod_{k=1}^{L}\left(\omega_k \beta_{jk} + 1 - \omega_k \sum_{j=1}^{N} \beta_{jk}\right) - \prod_{k=1}^{L}\left(1 - \omega_k \sum_{j=1}^{N} \beta_{jk}\right)\right]}{1 - \mu \times \left[\prod_{k=1}^{L} 1 - \omega_k\right]} \tag{12}$$

where L is the total number of rules and μ is defined as

$$\mu = \left[\sum_{j=1}^{N} \prod_{k=1}^{L} \left(\left(\omega_k \beta_{jk} + 1 - \omega_k \sum_{j=1}^{N} \beta_{jk} \right) \right) - \prod_{k=1}^{L} (1 - \omega_k \sum_{j=1}^{N} \beta_{jk}) \right]^{-1} \quad (13)$$

Using (12), calculated belief degree for each of the three referential values of the consequent attribute AQI is as follows.

Hazardous = 0.8762
Unhealthy = 0.1238
Good = 0.0

We then convert this multi-value output into one single numerical crisp value by employing the Algorithm 2. All the regression coefficients of this algorithm have been set in line with AQI breakpoint table.

Algorithm 2: AQI_calculations (Hazardous, Unhealthy, Good)

```
if ((Hazardous > Unhealthy) and (Hazardous > Good)) then
    AQI = (201 + 299*Hazardous) + ((200*Unhealthy)/2)
else if ((Good > Hazardous) and (Good > Unhealthy)) then
    AQI = (100*(1 - Good)) + ((200*Unhealthy)/2)
else if ((Unhealthy > Hazardous) and (Unhealthy > Good)) then
    if (Hazardous > Good) then
        AQI = (101 + 99*Unhealthy) + ((500*Hazardous)/2)
    else if (Good > Hazardous) then
        AQI = (101 + 99*Unhealthy) + ((100*Good)/2)
return AQI
```

As per the aforementioned algorithm, crisp value of AQI against the input values of two antecedent attributes (417.99 µg/m^3 and 423 µg/m^3) is: ((201 + 299*0.8762) + ((200*0.1238)/2)) = 475.3638. Fig. 15 shows the conceptual architecture of BRBES.

4.4 Distributed Categorization of AQI

The final part is the regression layer which distributes belief degrees over the six AQI categories against the predicted crisp value of AQI (as calculated in Sect. 4.3). In this thesis, a regression coefficient is applied to distribute this AQI over its 6 set categories with corresponding belief degrees. Belief degree of the c^{th} AQI category, calculated by the categorization function, is defined as

$$\ddot{y}_{ic} = \text{categorize(aqi\_predicted)} = r_c * b_c \quad (14)$$

where r_c represents a regression coefficient and b_c refers to belief degree of Hazardous, Unhealthy or Good (as calculated in Sect. 4.3). $\ddot{y}_i = [\ddot{y}_{i1}, \ddot{y}_{i2}, \ldots \ddot{y}_{i6}]$ refer to the predicted belief degrees of each of the 6 AQI categories.

We have developed Algorithm 3, in line with AQI breakpoint table, to calculate r_c.

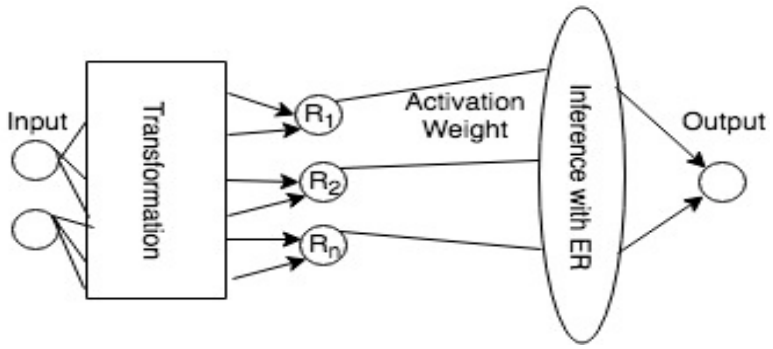

Fig. 15. Conceptual architecture of BRBES.

Algorithm 3: RegressionCoefficient(AQI)

```
1  if (AQI >= 301) then
2          r_c = (AQI - 301)/199
3  else if ((AQI >= 201) and (AQI <= 300)) then
4          r_c = (AQI - 201)/99
5  else if ((AQI >= 151) and (AQI <= 200)) then
6          r_c = (AQI - 151)/49
7  else if ((AQI >= 101) and (AQI <= 150)) then
8          r_c = (AQI - 101)/49
9  else if ((AQI >= 51) and (AQI <= 100)) then
10         r_c = (AQI - 51)/49
11 else if (AQI <= 50) then
12         r_c = AQI/49
13 return r_c
```

AQI, mentioned in this algorithm, refers to the predicted crisp value of AQI (as calculated in Sect. 4.3). Against the predicted crisp value of AQI, which has turned out to be 475.35, the aforementioned algorithm calculates r_c to be (475.3638 – 301)/199 = 0.8762.

Finally, we calculate belief degree for each of the 6 categories of AQI in the following way.

Belief Degree for 'Hazardous' category, $\ddot{y}_{i1}$ = (Hazardous) * r_c
Belief Degree for 'Very Unhealthy' category, $\ddot{y}_{i2}$ = (Hazardous) * (1 - r_c)
Belief Degree for 'Unhealthy' category, $\ddot{y}_{i3}$ = (Unhealthy) * r_c
Belief Degree for 'Unhealthy for sensitive groups' category, $\ddot{y}_{i4}$ = (Unhealthy) * (1 - r_c)
Belief Degree for 'Moderate' category, $\ddot{y}_{i5}$ = (Good) * r_c
Belief Degree for 'Good' category, $\ddot{y}_{i6}$ = (Good) * (1 - r_c)
Here, 'Hazardous', 'Unhealthy' and 'Good' on the right side of all the equations refer to the belief degree of the referential values of consequent attribute AQI (as calculated in Sect. 4.3), which are 0.8762, 0.1238 and 0.00 respectively.

Hence, we calculate belief degree for each of the 6 AQI categories in terms of our example case below.

Belief Degree for 'Hazardous' category, $\ddot{y}_{i1} = 0.8762 * 0.8762 = 0.77$
Belief Degree for 'Very Unhealthy' category, $\ddot{y}_{i2} = (0.8762) * (1 – 0.8762) = 0.11$
Belief Degree for 'Unhealthy' category, $\ddot{y}_{i3} = (0.1238) * 0.8762 = 0.11$
Belief Degree for 'Unhealthy for sensitive groups' category,
$\ddot{y}_{i4} =(0.1238) * (1 – 0.8762) = 0.01$
Belief Degree for 'Moderate' category, $\ddot{y}_{i5} = (0.00) * 0.8762 = 0.00$
Belief Degree for 'Good' category, $\ddot{y}_{i6} = (0.00) * (1 – 0.8762) = 0.00$

Then, the mean-square-error, E of our proposed integrated approach is calculated based on the difference between predicted AQI (crisp value) and actual AQI over m training data pairs as shown in (15).

$$E = \frac{1}{m}\sum_{i=0}^{m-1}(\text{aqi\_predicted-aqi\_actual})^2 \quad (15)$$

4.5 Summary

This chapter has mathematically clarified our proposed integrated approach. Next chapter will shed light on optimized version of BRB to increase efficiency and prediction accuracy further.

5 OPTIMIZED BRB

This chapter explains how to optimize BRB in terms of both efficiency and performance. It introduces the concept of conjunctive and disjunctive BRB to minimize memory requirement and computational cost. It also covers trained version of BRB with a view to enhancing the accuracy level of BRB.

5.1 BRB Expert System

The BRB Expert System explained in subsection 4.3 is of conjunctive type. Conjunctive assumption covers every possible combination of referential values of the antecedent attributes to construct rules in the rule base. Thus, it results in a large number of rules when there are too many referential values and/or antecedent attributes. Such a large rule base necessitates higher memory capacity and computational cost. Disjunctive BRB has been proposed by researchers to address this drawback without hampering prediction accuracy. Thus, disjunctive BRB enables us to predict AQI with less memory and less computation, while upholding prediction accuracy. Following two subsections explain both conjunctive and disjunctive BRB.

5.1.1 Conjunctive BRB

Conjunctive assumption covers every possible combination of referential values of the antecedent attributes [13]. This assumption makes the input space complete, ensuring the activation of corresponding rules regardless of whichever the input is.

Such assumption causes combinatorial explosion problem as the size of BRB grows with increasing number of antecedent attributes and/or referential values of these attributes. If there are n_m referential values of the m^{th} attributes, m = 1, … , M , the size of conjunctive BRB will be

$$size_{BRB} = \prod_{m=1}^{M} n_m \tag{16}$$

Fig. 16 shows the exponential rise of the size of conjunctive BRB with various number of attributes and referential values of the attributes. In terms of two attributes, with 2 to 6 referential values, there can be 4 to 36 rules. When there are three attributes, the number of rules range from 9 to 216.

Given 216 rules against 3 attributes and 6 referential values, it is impractical for an expert to formulate 216 rules at a time. Moreover, it is difficult to further interpret and optimize 216 rules effectively. This calls for a new assumption of BRB construction to bring down its size. Disjunctive BRB has been proposed by the researchers to address this call.

5.1.2 Disjunctive BRB

Disjunctive BRB is intended to address the combinatorial explosion problem of conjunctive BRB leading to less computational cost and memory requirement [16]. Disjunctive BRB is

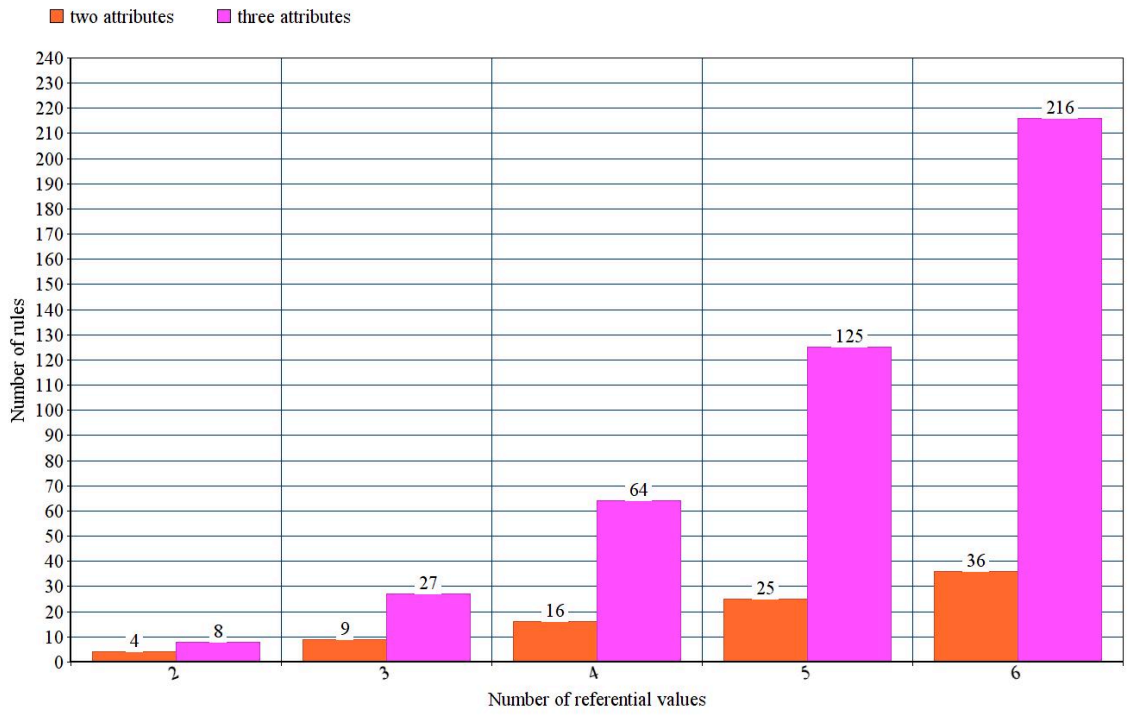

Fig. 16. Comparison of BRB size with different number of referential values and attributes.

more indicative for reasoning compared with conjunctive one. Rule activation weight calculation procedure is different in disjunctive assumption [14]. However, input matching degree calculation remains same as conjunctive one. All antecedent attributes have same number of referential values in disjunctive BRB [110]. Number of antecedent attributes is regardless to determine the size of disjunctive BRB. Number of rules, num_{rules} , of a disjunctive BRB is equal to the number of referential values of the antecedent attribute. It is defined as

$$num_{rules} = K = m_1 = m_2 = \ldots = m_M \tag{17}$$

where 1, … , M refers to the number of referential values.

If there are two attributes, A and B, with two referential values each, there will be four rules in conjunctive assumption as shown in Fig. 17(a) and two rules in disjunctive assumption as demonstrated in Fig. 17(b). For a certain input Y, all the four rules (A1B1, A1B2, A2B1, A2B2) are activated under conjunctive assumption while only two rules (A1B1, A2B2) are activated in the disjunctive assumption. Disjunctive assumption only takes the upper and lower limits of the referential values, which formulates two rules A1B1 and A2B2 at the boundaries of the input space. Thus, BRB size under disjunctive assumption is dependent only on the number of referential values of antecedent attributes. Fig. 18 compares BRB size under different assumptions.

The new rule activation weight calculation technique assumes the attributes to be disjunctive. Under disjunctive assumption, the activation weight of k^{th} rule, w_k is defined as

$$w_k = \frac{\theta_k \sum_{i=1}^{M} \alpha_i^k}{\sum_{l=1}^{L} \theta_l \sum_{i=1}^{M} \alpha_i^l} \tag{18}$$

where θ_k is the initial weight of the k^{th} rule, α_i^k is the matching degree of the input with the k^{th} rule, M is the total number of inputs and L is the total number of rules. w_k under conjunctive assumption is calculated as

$$w_k = \frac{\theta_k \sum_{i=1}^{M} \alpha_i^k}{\sum_{l=1}^{L} \theta_l \prod_{i=1}^{M} \alpha_i^l} \tag{19}$$

In conjunctive BRB, k^{th} rule will not be activated even if one input matching degree is 0. However, in disjunctive BRB, k^{th} rule will be activated even with only one non-zero matching degree.

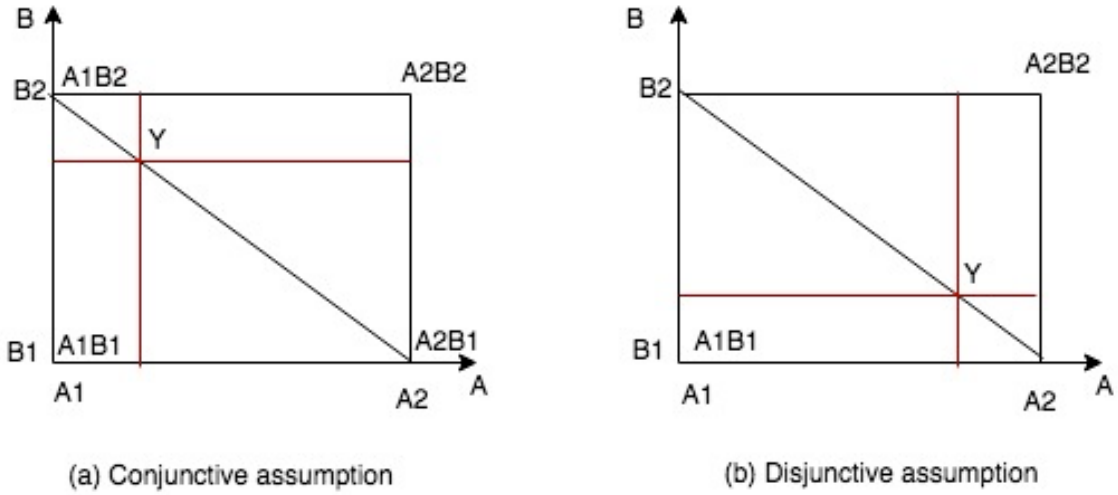

Fig. 17. Rule activation in two types of BRB.

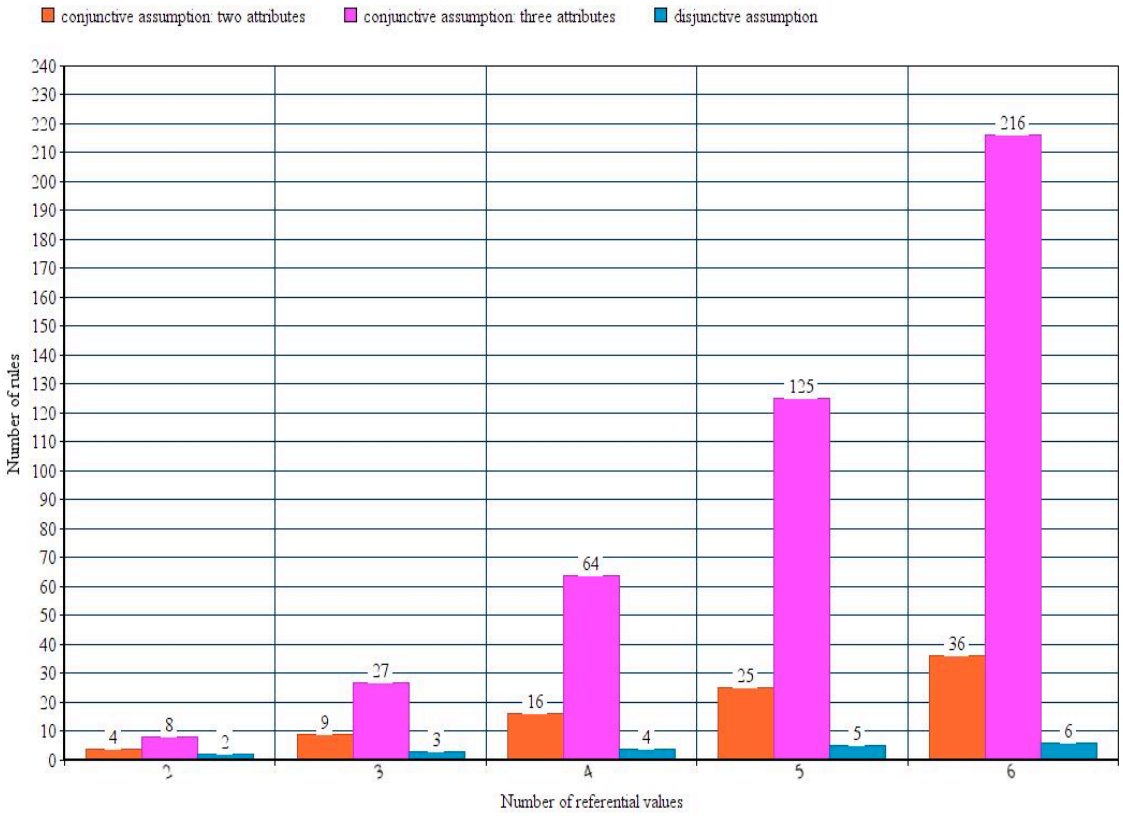

Fig. 18. Comparative sizes of BRB under different assumptions.

The rule base of our proposed BRB based Deep Learning approach for predicting air pollution under disjunctive assumption has been shown in Table 5.

5.2 Trained BRB

The performance of BRB can be optimized further by training up this algorithm. Such optimization will enable more accurate prediction of AQI. This optimization can be done both in terms of parameter and structure. Joint optimization is also executed by combining parameter and structure optimization at a time.

Table 5. Rule base under disjunctive assumption.

Rule Id	Rule Weight	IF		THEN		
		$PM_{2.5}$ (Sensor)	$PM_{2.5}$ (CNN)	AQI		
				Hazardous	Unhealthy	Good
R1	1.0	H	H	1.00	0.00	0.00
R2	1.0	M	M	0.00	1.00	0.00
R3	1.0	L	L	0.00	0.00	1.00

5.2.1 Parameter Optimization

Identification of proper values of the relevant parameters of BRB can optimize the prediction accuracy of BRB further. Such BRB parameters include attribute weight, rule weight, belief degrees in the consequent part of rule base, utility scores of the referential values of both antecedent and consequent attributes. Differential Evolution (DE), a black box, meta-heuristic algorithm to solve optimization problems, is used for parameter optimization of BRB [84]. Matlab also provides a command called 'fmincon', which discovers the lowest value of constrained nonlinear multivariable function. However, reliability of the search for finding the global minimum attained by DE is more than that of fmincon [97]. Here, reliability refers to the point with the lowest function value obtained during the search process is reasonably near the global minimum point. That is why, we apply DE for parameter optimization in this research and implement it with Python programming language.

Differential Evolution (DE)

DE was invented by R. Storn and K. Price in 1997 for minimizing continuous space functions leading to global optimization [93]. Its convergence rate is faster and more certain than many other global optimization approaches. It keeps setting different values of the set variables within a pre-defined range until a global minimum is reached. The sequential steps of DE for performing parameter optimization of BRB are shown in Algorithm 4.

Algorithm 4: DifferentialEvolution

1 Initial dataset divided into training and testing dataset.

2 BRB parameter initialization.

3 Rule activation weight calculation.

3.1 Crossover

if ((rand <= CR) **or** (j == sn)) **then**

$u_{ij} = v_{ij}$

else

$u_{ij} = x_{ij}$

where crossover operator, CR = 0.9 and sn $\in$ [1,2,…,n] represents a random integer which is generated against every single new individual.

3.2 Mutation

The i^{th} individual in new generation is defined as

$v_i = x_{r1} + F * (x_{r2} - x_{r3})$

where x_{r1}, x_{r2} and x_{r3} are three random individuals, r1 != r2 != r3 and F = 0.5 refers to the mutation operator.

4 ER approach integrates activated rules.

5 Selection

The i^{th} individual u_i^t moves into the new generation if the value of its fitness function is higher, as defined by

if ($f(u_i^t) <= f(x_i^t)$) **then**

$x_i^{t+1} = u_i^t$

else

$x_i^{t+1} = x_i^t$

where f(.) represents the fitness function, which in this thesis is MSE/RMSE/MAE/RRSE/R-Squared/RAE.

6 Check on Stop Criterion (no. of generations)

if not met **then**

go to step 3

if met **then**

select individual having the lowest MSE as final solution

7 Develop the new BRB classification algorithm with obtained solution.

8 Validate efficiency with testing dataset.

BRB adaptive DE

As DE is intended to fine-tune the BRB parameters to optimize its accuracy further, the performance of DE itself depends on the control parameters. These parameters need to be

adjusted to address the uncertainty or noise of the optimization problem, which in this research, is the AQI prediction. Further, a balance between exploration and exploitation of the search space has to be ensured to obtain appropriate values of these control parameters. Driven by this, BRB adaptive DE (BRBaDE) has been proposed which optimizes the control parameters of DE by taking into account the uncertainties handled by BRB while balancing exploration and exploitation [50].

BRBaDE changes the values of two control parameters of DE, Crossover Factor (CR) and Mutation Factor (F) after every iteration. It uses BRBES to predict either higher values of CR and F to trigger exploration or lower values of CR and F to ensure exploitation, leading to optimal solution in fewer iterations. If the current population is far from convergence, BRBaDE makes exploration of the search space to find global optima, which in turn finds a population close to optimal solution. On the other hand, if the current population is close to optimal solution, BRBaDE makes exploitation of the search space to find local optima.

BRBaDE increases the accuracy compared with conventional DE. However, it takes more computational time and cost as it predicts new CR and F after every iteration based on the result of previous iteration. Even though, the more time and cost are worth the value as it deals with local and global optima.

5.2.2 Structure Optimization

In addition to parameter optimization, structure optimization is done by restructuring the BRB for improving its performance further. Structure Optimization based on Heuristic Strategy (SOHS) algorithm executes this structure optimization by changing the number of referential values of each antecedent attribute [84]. SOHS evaluates the performance of BRB with different number of referential values of each antecedent attribute and finally chooses the one which works the best.

Initially, SOHS chooses any two BRBs randomly. Then it picks up the BRB with higher error between the two. If the selected BRB's antecedent attribute has lower or equal number of referential values than that of the other BRB, SOHS opts the structure of the chosen BRB. Then it randomly adds one or few more referential values to each of the antecedent attributes

of this BRB. On the other hand, if the other BRB has less number of referential values of the antecedent attributes than the chosen BRB, SOHS prunes some of the referential values of the antecedent attributes randomly from the chosen BRB and accepts the remaining ones.

The newly structured BRB will have half or double number of referential values of each antecedent attribute with respect to two designated BRBs. To uphold the diversity of BRB, the addition or pruning of the referential values of the antecedent attributes is implemented in a random way. Fig. 19 shows the flowchart of SOHS.

5.2.3 Joint Optimization

Parameter optimization and structure optimization is combined together with a view to achieving maximum performance. Such combined optimization is called Joint Optimization [114]. This has been achieved by the Joint Optimization on Parameter and Structure (JOPS) algorithm [115]. First, DE is run on an initial structure of BRB to obtain the optimized parameters. Set of BRBs is updated based on these optimized parameters. Then the SOHS algorithm is run on the updated set of BRBs to obtain a new structure for each BRB. Again DE is run on the newly structured BRBs until the stop criterion is met. Finally, the BRB with the lowest error among the available structures of BRBs is chosen as the output of JOPS. Fig. 20 shows the flowchart of JOPS.

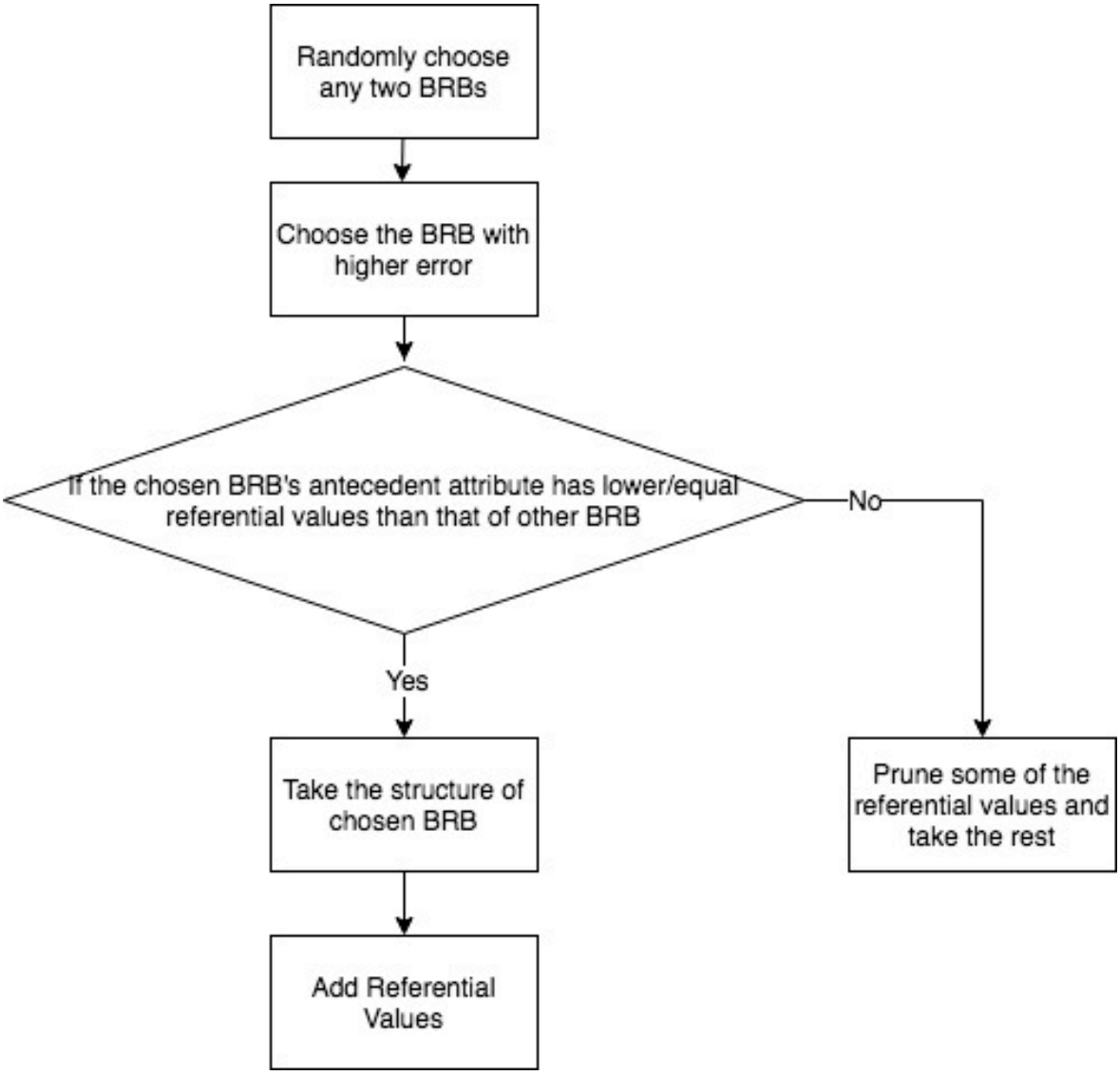

Fig. 19. The flowchart of SOHS.

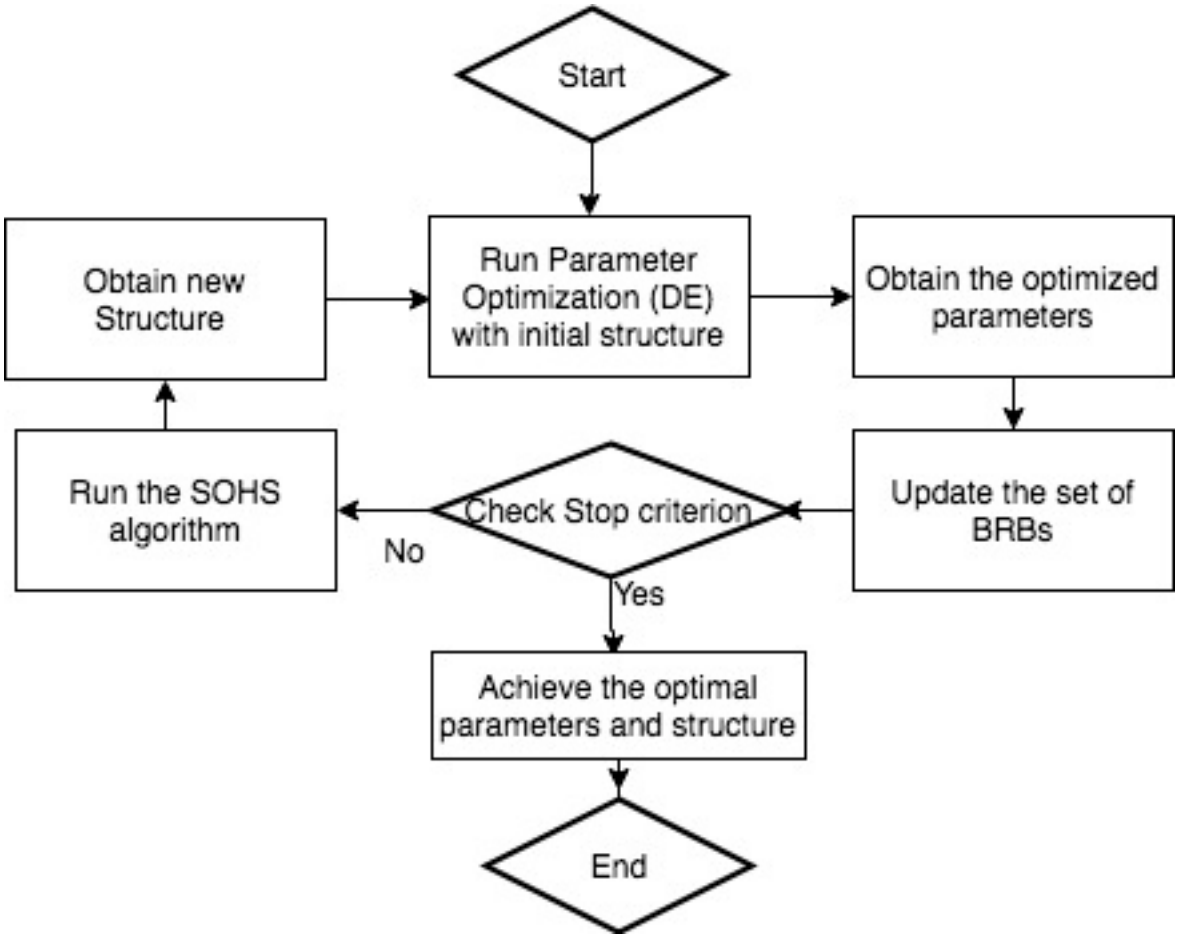

Fig. 20. The flowchart of JOPS.

5.3 Summary

This chapter has focused on optimization of BRB. Next chapter will demonstrate our dataset, comparative results, discussion, sustainability and ICT ethics.

6 Results and Discussion

This chapter explains air pollution dataset we have used and covers comparative analysis of our proposed approach with other relevant approaches. It also demonstrates the performance of conjunctive and disjunctive BRB in terms of numerical figures. Moreover, the comparative accuracy of BRB in terms of trained and non-trained versions have also been illustrated in this chapter. The performance metrics used in this chapter for showing comparative analysis among various systems are: Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Relative Absolute Error (RAE), Root Relative Squared Error (RRSE) and R-Squared (Co-efficient of determination). Finally, we conclude this chapter with the rationale of this research with respect to sustainability and ICT ethics.

6.1 Air Pollution Dataset

We have investigated various air pollution datasets to apply our proposed integrated approach and predict pollution level. Some of them were unlabeled datasets, which contained images of various levels of air pollution with no numerical label of $PM_{2.5}$ concentrations. Such unlabeled dataset is not appropriate with respect to our proposed approach as we need both image and $PM_{2.5}$ concentrations to train our developed algorithm. Some of the datasets we found were labeled. They contained numerical level of carbon dioxide (CO_2) and Lead against the air pollution images. However, such air pollutants are not taken into account to calculate AQI in the AQI breakpoint table developed by U.S. EPA.

Then we found a labeled dataset which contains air pollution images along with numerical label concerning $PM_{2.5}$ concentrations against every image. As $PM_{2.5}$ is considered to calculate AQI in the AQI breakpoint table, we have used this labeled dataset provided by Li et al. [70]. This dataset contains 3024 synthetic images of various levels of outdoor air pollution as well as corresponding sensor reading of $PM_{2.5}$ of the same place and same time. These 3024 images have been divided into two parts: training part with 2419 images and testing part with 605 images. We have trained up our CNN with 2419 training images and tested its prediction accuracy with the remaining 605 test images. These 2419 training images have been split into 3 categories: High Pollution, Medium Pollution and Low Pollution, where High refers to $PM_{2.5}$ reading of 150.5 $\mu g/m^3$ and above, Medium refers to $PM_{2.5}$ value from 35.5 to 150.4 $\mu g/m^3$

and Low refers to $PM_{2.5}$ of 35.4 µg/m^3 and lower. Fig. 21 shows the training images of our dataset of all the three categories (High, Medium and Low pollution).

6.2 Comparative Analysis

We have used Python 3.6.4 and Keras neural network library to implement our proposed BRB based Deep Learning approach. Both BRB and CNN have been implemented with python programming language. OpenCV, an image and video processing library, has been used to pre-process the outdoor images for prediction purpose. Functions of Keras library has been applied to implement VGGNet layers. We ran 100 epochs to train and test this VGGNet model. As VGGNet uses softmax as its activation function, it generates multi-label prediction against an input image showing class-wise probability for each of the three pollution levels. Our achieved testing accuracy with VGGNet over the testing dataset of outdoor images was around 83.64%. We have used file I/O to feed the multi-label prediction of VGGNet to the python script of BRB.

$PM_{2.5}$ predicted by VGGNet from input image and sensor readings of $PM_{2.5}$ constitute two values for two antecedent attributes of BRB. Based on these two values, BRB predicts the AQI as well as category-wise belief degree for each of the six AQI categories. Table 6 illustrates higher accuracy of our proposed integrated approach than other approaches when sensor generates wrong data. For instance, if sensor miscomputes $PM_{2.5}$ concentrations to be 126 µg/m^3, where accurate reading is 447 µg/m^3, BRB predicts AQI to be 208.34 and our proposed approach predicts AQI to be 263.96. Hence, the AQI predicted by our proposed

Fig. 21. Air Pollution images with low, medium and high levels.

Table 6. In case sensor gives wrong reading.

$PM_{2.5}$ (μg/m^3)		AQI		
sensor data	predicted by CNN	predicted by only BRB (only sensor data are considered)	predicted by integrated approach (BRB and CNN)	Actual value
447	440.64	496.70	477.32	464
126 (wrong reading, accurate is 447)	440.64	208.34	263.96	
4	2.50	9.17	14.02	17
243 (wrong reading, accurate is 4)	2.50	259.48	159.53	

approach (263.96) is closer to the actual AQI (464) than the one predicted by BRB (208.34). Table 7 also shows the better performance of our proposed approach than other approaches when CNN predicts inaccurate $PM_{2.5}$ from outdoor images. For instance, if CNN miscomputes $PM_{2.5}$ concentrations to be 463.44 μg/m^3, when the proper reading is 13.42 μg/m^3, our proposed approach predicts AQI to be 228.48, where CNN predicts AQI to be 475. Thus, our proposed approach's predicted AQI (228.48) is closer to the actual AQI (76) than the one predicted by CNN (475). We attribute higher accuracy of our proposed approach to the adoption of multimodal learning, which uses data from two sources (both sensor reading and image), rather than relying on only one source (either sensor or image).

Table 7. In case CNN generates inaccurate prediction.

$PM_{2.5}$ (μg/m^3)		AQI		
sensor data	predicted by CNN	predicted by only CNN (only image data are considered)	predicted by integrated approach (BRB and CNN)	Actual value
453	454.72	497.47	482.53	468
453	16.79 (wrong reading, accurate is 454.72)	60	396.28	
24	13.42	53	92.27	76
24	463.44 (wrong reading, accurate is 13.42)	475	228.48	

6.3 Conjunctive versus Disjunctive BRBES

We have implemented disjunctive BRB in python as well. Disjunctive BRB has lower number of rules than conjunctive counterpart, thus necessitating less computational cost. However, disjunctive BRB has shown higher MSE compared with conjunctive BRB over the testing dataset as shown in Fig. 22. This higher error of disjunctive BRB is due to less number of rules in the rule base, resulting in less amount of reasoning than its conjunctive counterpart. Same pattern also goes to RMSE as illustrated in Fig. 23. From these figures, it is evident that the accuracy of conjunctive BRB is compromised when we shift to disjunctive BRB to bring down computational cost and memory requirement. We are testing the performance of disjunctive BRB to evaluate whether it is possible to predict AQI with same or even higher accuracy compared with its conjunctive counterpart.

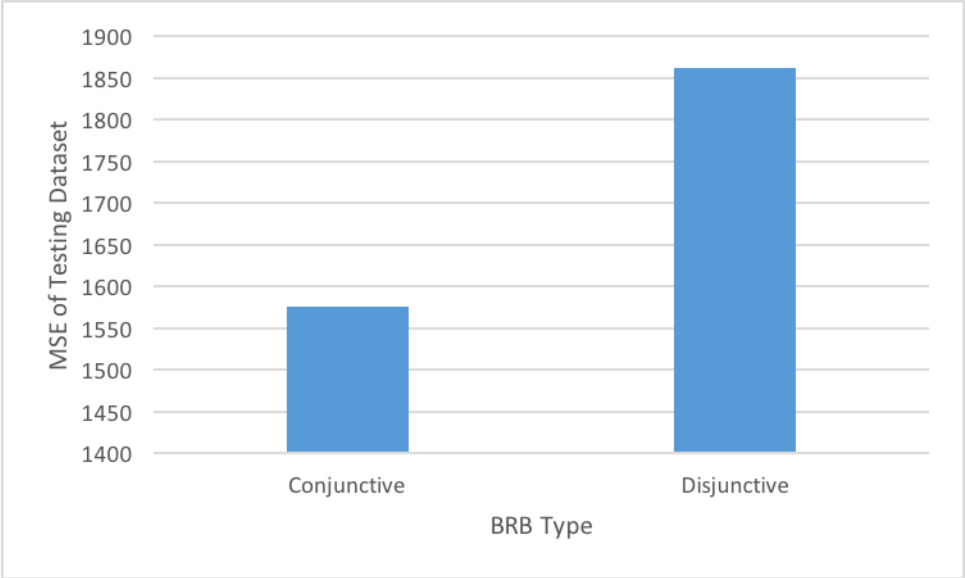

Fig. 22. MSE of Conjunctive and Disjunctive BRB.

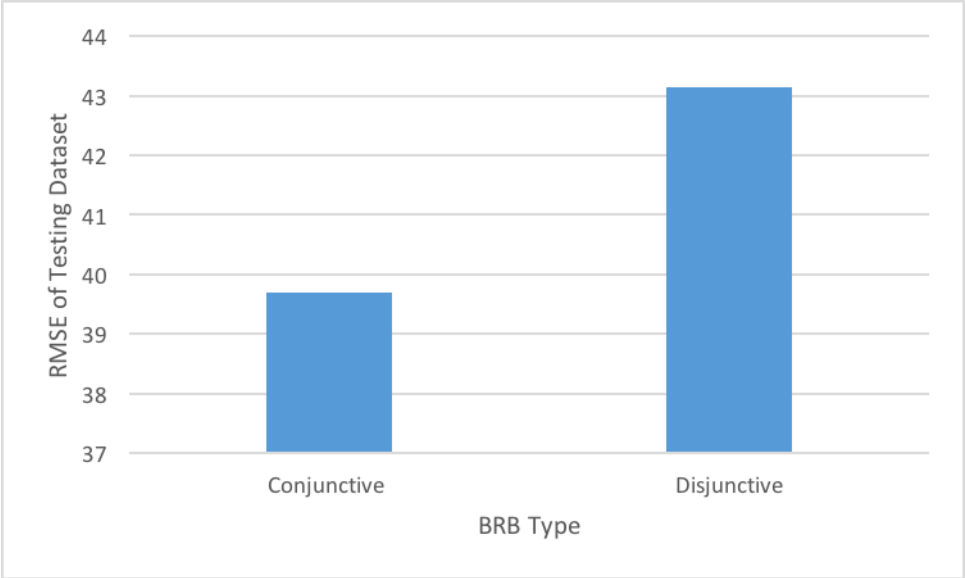

Fig. 23. RMSE of Conjunctive and Disjunctive BRB.

6.4 Trained versus Non-trained BRBES

We have run parameter optimization over both conjunctive and disjunctive BRB using DE algorithm. These parameters are initial rule weights, antecedent attribute weights, consequent belief degrees and utility scores of the referential values of the antecedent and consequent attributes. In the DE-optimized version, disjunctive BRB has turned out to be more accurate than its conjunctive counterpart. DE optimizes 41 parameters in conjunctive assumption and 17 parameters in disjunctive assumption. DE handles less than half number of parameters in disjunctive BRB than conjunctive one, which has contributed to lower error of disjunctive BRB. Fig. 24 shows comparative MSE on training dataset of both types of BRB over 1000 iterations of DE. Fig. 25 illustrates the values of performance metrics MAE, RAE, RRSE and R-squared over testing dataset with respect to DE-trained disjunctive BRB. MAE, RAE and RRSE have turned out to be 0.28, 0.17 and 0.19. The lower the values of MAE, RAE and RRSE are, the better. In terms of R-squared, the higher the value, the better the regression model accommodates variation. R-squared is always between 0 and 1, where 0 represents a model that does not explain any variation and 1 represents a model that represents all the variations. R-squared of our DE-trained disjunctive BRB is 0.96, which represents small difference between predicted AQI and actual AQI.

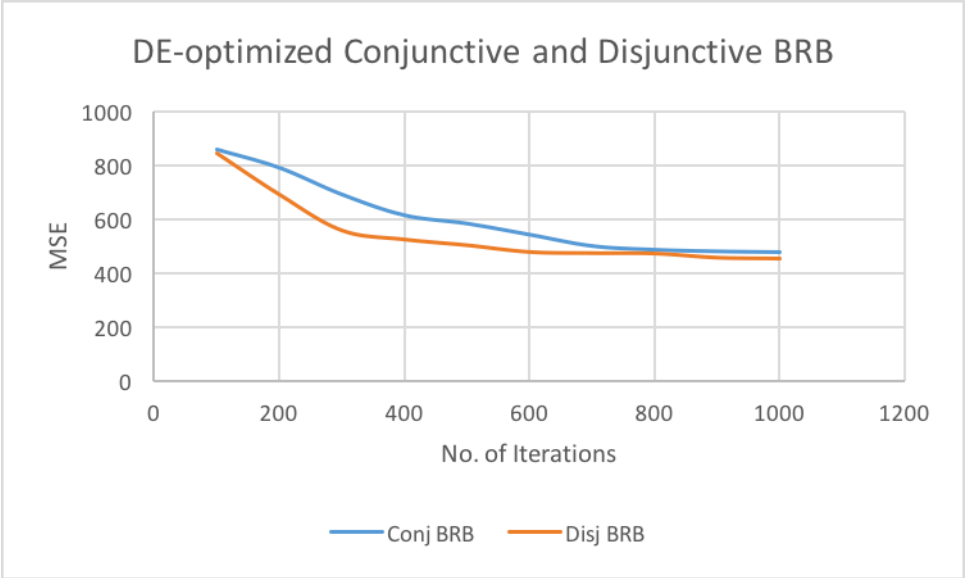

Fig. 24. MSE of DE-optimized Conjunctive and Disjunctive BRB.

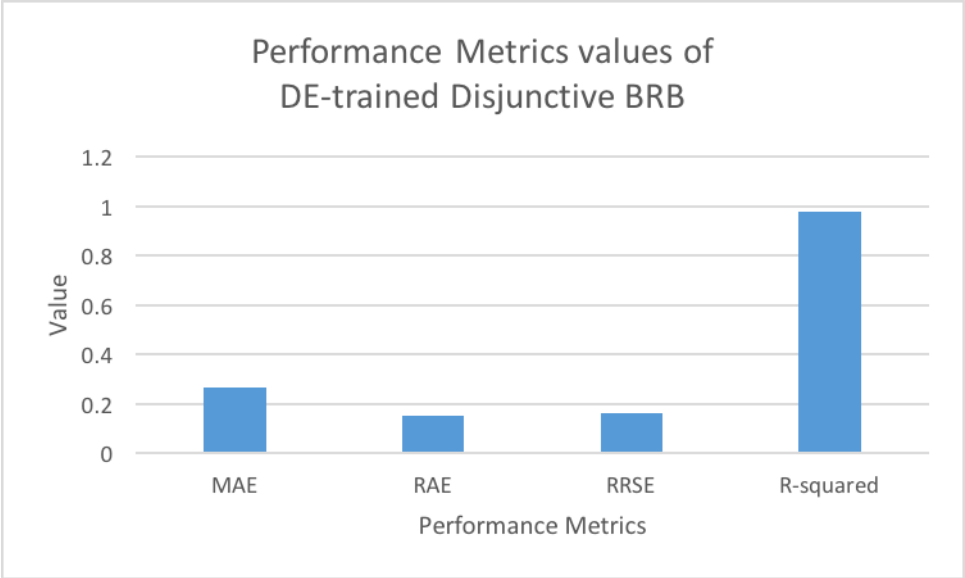

Fig. 25. Performance Metrics of DE-optimized Disjunctive BRB.

Afterwards, we have trained both conjunctive and disjunctive BRB separately in terms of joint optimization using the JOPS algorithm [115]. In terms of structure optimization part, we tested the MSE of both conjunctive and disjunctive BRB with varying number of referential values of the antecedent attributes. For conjunctive BRB, optimum number of referential values of the antecedent attributes with the lowest MSE turned out to be 3, as shown in Fig. 26. As there are 3 referential values of the consequent attribute, equal number of referential values of both antecedent and consequent attributes has resulted in this optimum

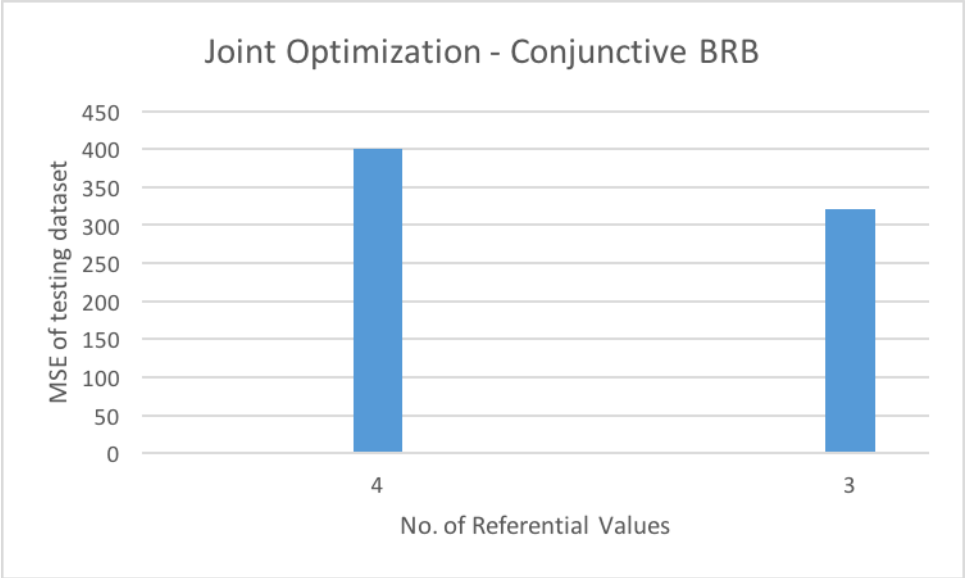

Fig. 26. MSE at various referential values of Joint Optimized Conjunctive BRB.

performance. On the other hand, optimum number of referential values of antecedent attributes for disjunctive BRB has turned out to be 4, as illustrated in Fig. 27. If there are 4 referential values at the antecedent part, there will be total 4 rules in disjunctive assumption. Thus, it increases reasoning level compared with 3 or lower number of referential values. However, if the number of referential values is 5 or more, it causes imbalance between antecedent and

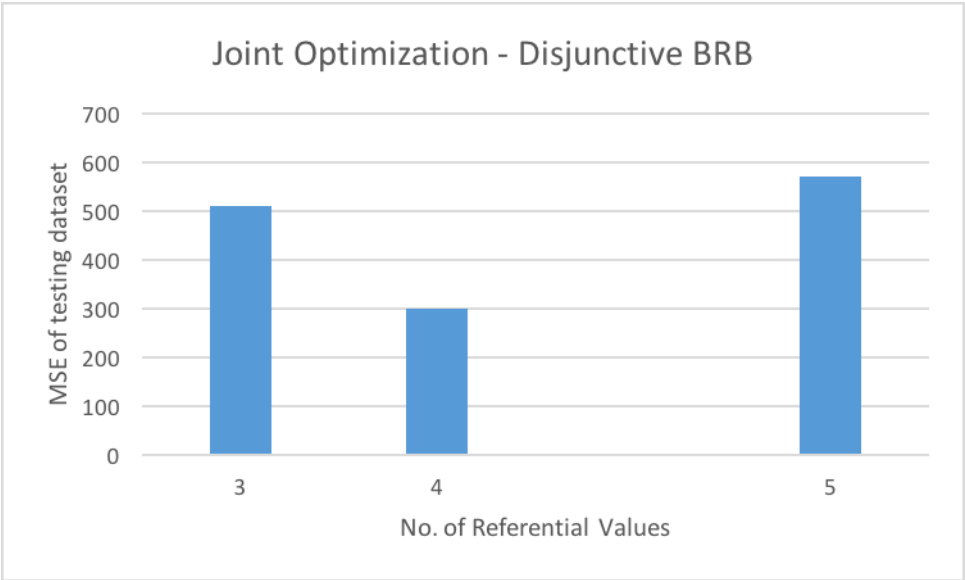

Fig. 27. MSE at various referential values of Joint Optimized Disjunctive BRB.

consequent part, as there are only 3 referential values in the consequent part. Hence, optimum number of referential values of the antecedent attributes is 4 in the disjunctive assumption.

Finally, jointly optimized disjunctive BRB has proved to be more accurate than its trained conjunctive counterpart as shown in Fig. 28. During joint optimization, DE deals with 152 parameters in conjunctive BRB, where this number is only 32 in disjunctive assumption. Number of parameters DE fine-tunes in disjunctive BRB is far less than that of conjunctive one. Hence, DE adjusts values of less number of parameters in disjunctive BRB to achieve the global minimum in terms of error, resulting in higher accuracy than its conjunctive counterpart.

We have also applied BRB adaptive DE (BRBaDE) over both conjunctive and disjunctive BRB separately. BRBaDE alters the values of mutation factor and crossover factor after every iteration based on the result of previous iteration. BRBaDE has minimized MSE of both types of BRBs. Fig. 29 shows the comparative MSE of conjunctive BRB with DE and with BRBaDE. Fig. 30 shows the comparative MSE of disjunctive BRB with DE and with BRBaDE. At the end, conjunctive BRB has shown lower MSE than disjunctive one with respect to BRBaDE, as demonstrated over 1000 iterations of BRBaDE in Fig. 31.

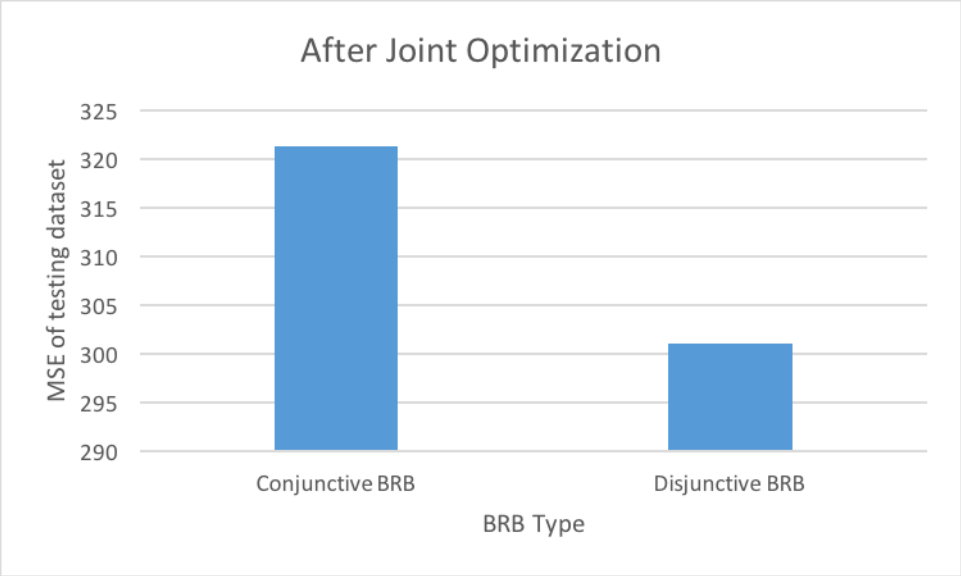

Fig. 28. MSE of conjunctive and disjunctive BRB after joint optimization.

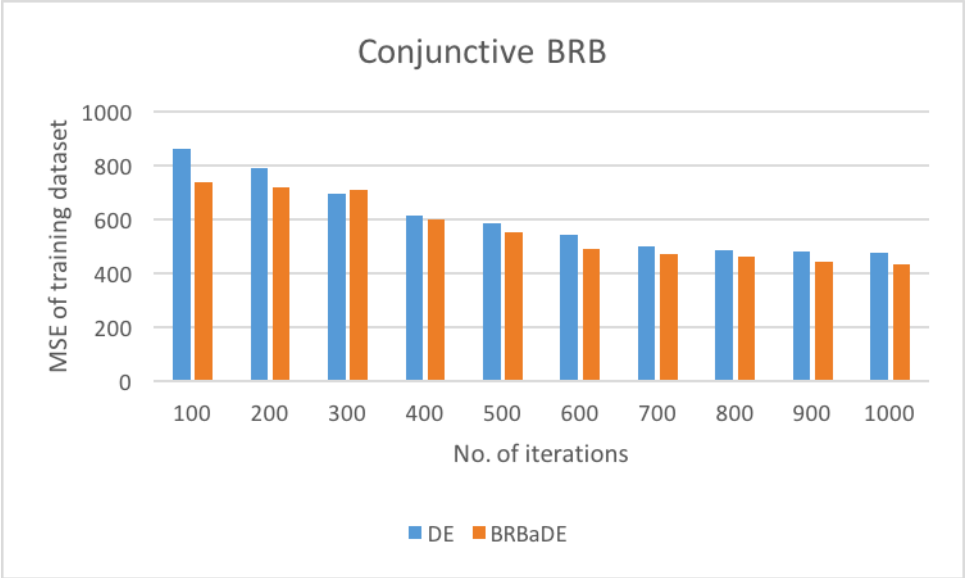

Fig. 29. MSE of conjunctive BRB with DE and with BRBaDE.

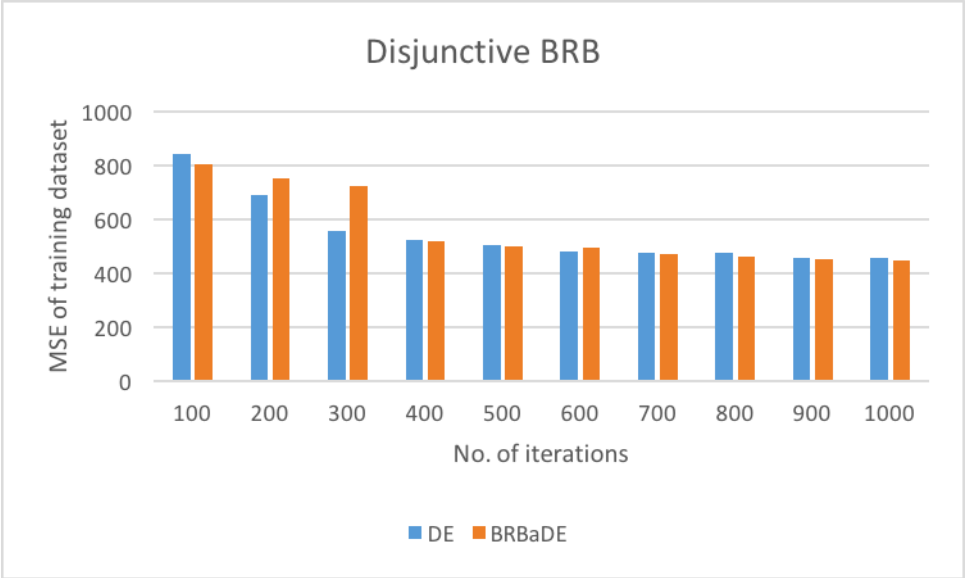

Fig. 30. MSE of disjunctive BRB with DE and with BRBaDE.

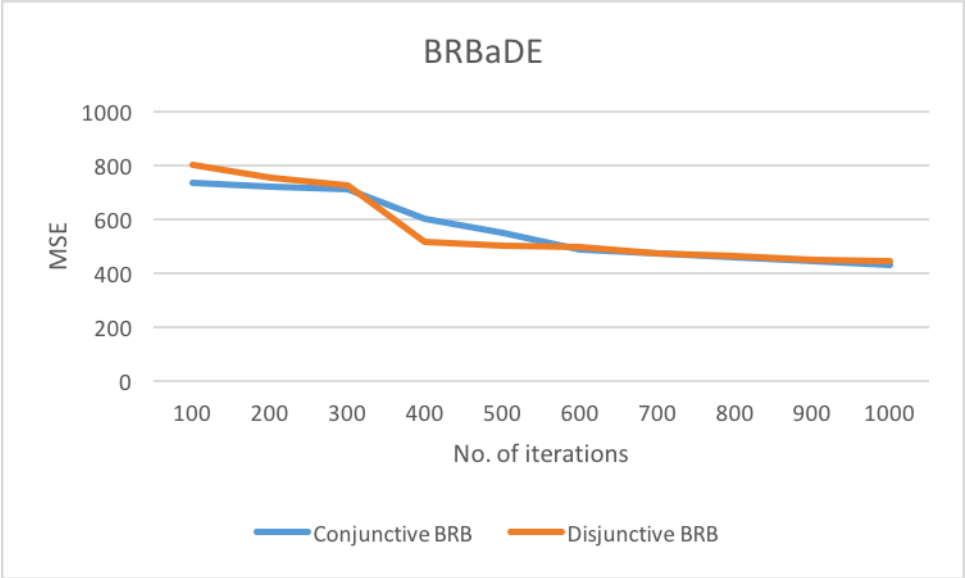

Fig. 31. MSE of conjunctive and disjunctive BRB with BRBaDE.

Moreover, we have replaced DE with BRBaDE in the joint optimization part to evaluate how it performs with respect to DE. We have run structure optimization with SOHS and parameter optimization with BRBaDE in this revised joint optimization phase both on conjunctive and disjunctive BRB. Fig. 32 illustrates the conjunctive BRB performance in this BRBaDE based joint optimization where three referential values of the antecedent attributes have shown the lowest MSE. Fig. 33 demonstrates the same in terms of disjunctive BRB where four referential values of the antecedent attributes have shown the best result.

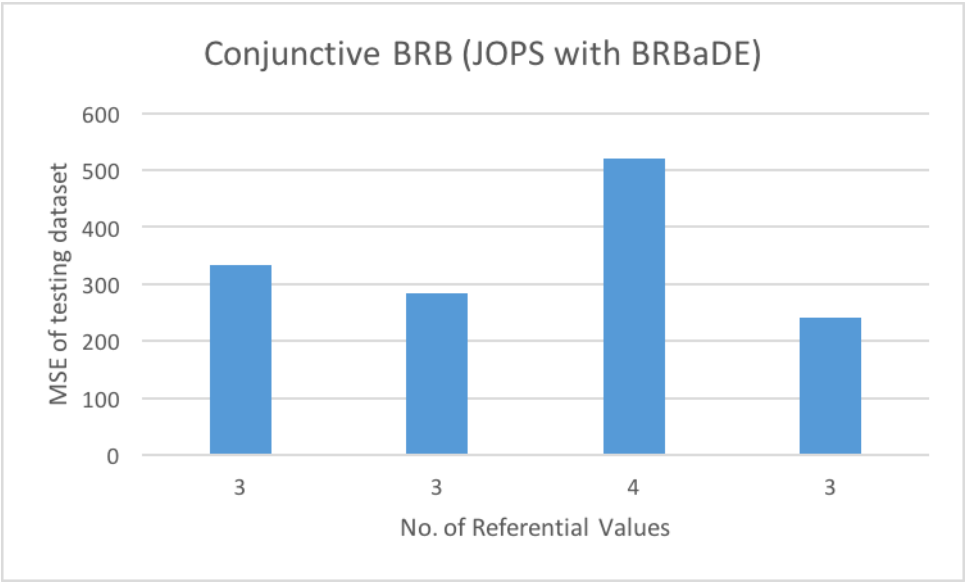

Fig. 32. MSE of conjunctive BRB at different referential values.

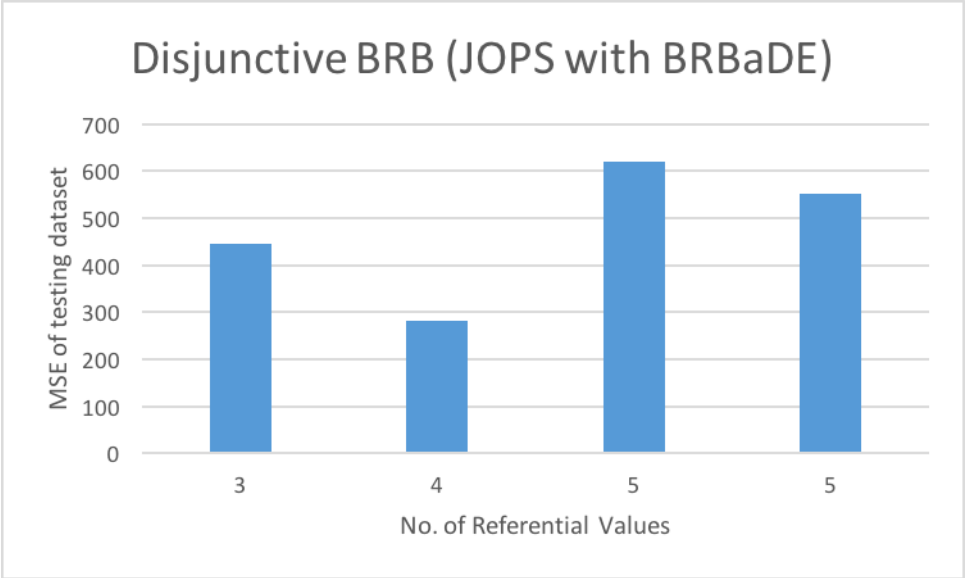

Fig. 33. MSE of disjunctive BRB at different referential values.

Fig. 34 shows comparative MSE of conjunctive and disjunctive BRB when JOPS is applied with BRBaDE. It is evident from the figure that conjunctive BRB offers higher accuracy than disjunctive one after 1000 iterations of BRBaDE. Conjunctive BRB has more than double parameters than disjunctive BRB. Hence, BRBaDE can fine-tune more parameters in conjunctive assumption. Therefore, with BRBaDE, conjunctive BRB outperforms its disjunctive counterpart.

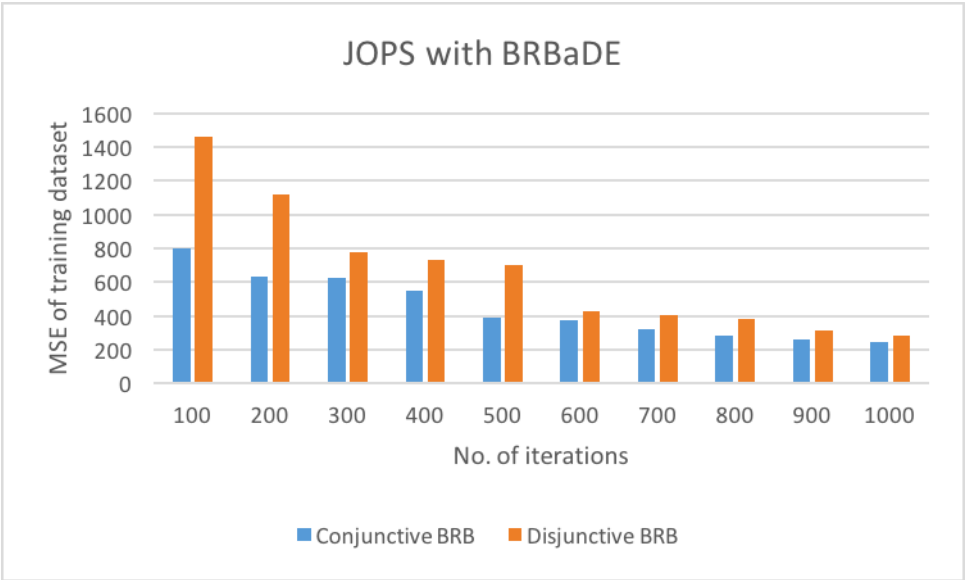

Fig. 34. Comparative MSE of conjunctive and disjunctive BRB.

BRBES outperforms various other machine learning tools, such as, Decision Tree, Random Forest, Linear Regression and ANN in terms of classification/regression accuracy [80]. Hence, we compare our integrated approach (with joint optimized disjunctive BRB) with only BRB (conjunctive, non-trained) and only CNN. Table 8 shows the sensitivity, specificity and Area Under Curve (AUC) of each of these 3 approaches. It is clearly evident from this table that our integrated approach outperforms both only BRB and only CNN. Such higher performance of our proposed approach is attributed to multimodal learning. Receiver Operating Characteristic (ROC) curve is widely employed to compare and asses the performance of predictive models. It is widely used for assembling and choosing classifiers through visualization of their performance [30]. Fig. 35 shows ROC of 3 methods. AUC is linked with ROC. Greater the value of AUC, higher the reliability and accuracy of the concerned predictive model is. Table 8 demonstrates that our proposed approach has higher AUC than both only BRB and only CNN. As shown in Table 8, our integrated approach performs better in comparison with only BRB and only CNN in terms of sensitivity and specificity as well. Thus, our proposed model is reliable enough to predict AQI with reasonable accuracy.

Table 8. Comparison of Reliability among three models.

Model	Sensitivity (%)	Specificity (%)	AUC
BRB based CNN (Joint Optimized with Disjunctive assumption)	94.07	95.61	0.936
BRB	92.34	93.61	0.905
CNN	89.73	90.74	0.893

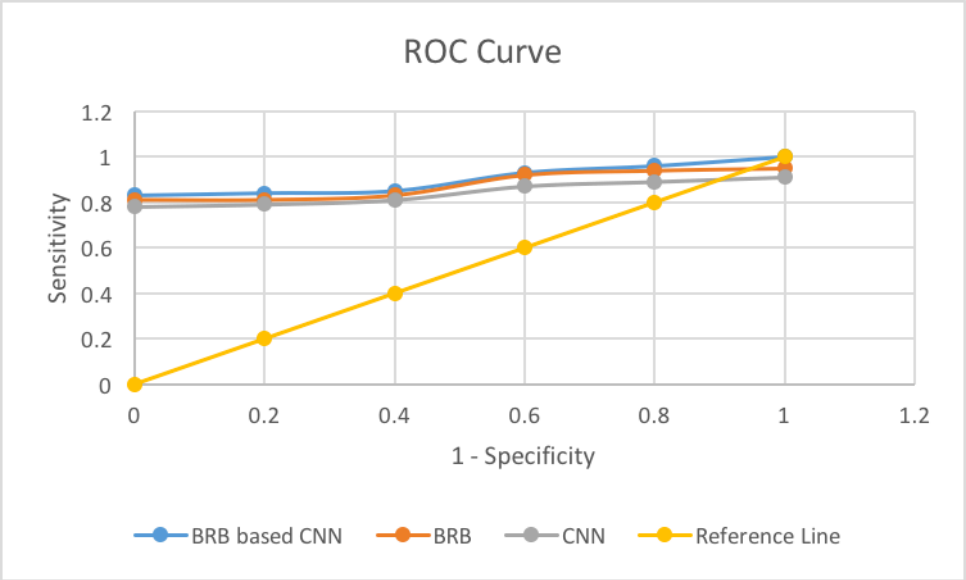

Fig. 35. Comparison of results using ROC curves.

Fig. 36 shows AQI predicted by various methods against sensor reading of 447 $\mu g/m^3$ and corresponding image. In addition to only BRB and only CNN, we have predicted AQI with our proposed integrated approach. We have employed our integrated approach with non-trained BRB, trained disjunctive BRB and trained disjunctive BRB with BRBaDE. AQI predicted by our integrated approach (with trained disjunctive BRB using BRBaDE) has turned out to be the closest to the ground truth. Fig. 37 shows MSE of testing dataset by same five methods as in Fig. 36. This figure also demonstrates that our integrated approach (with trained disjunctive BRB using BRBaDE) has the lowest MSE among all the methods. Thus, we prove that our novel mathematical model combining BRB and CNN has increased accuracy of our predictive approach than only BRB and only CNN. Moreover, we justify the adoption of trained version of BRB, instead of relying on non-trained BRB, as trained BRB has significantly brought down the MSE level of non-trained BRB.

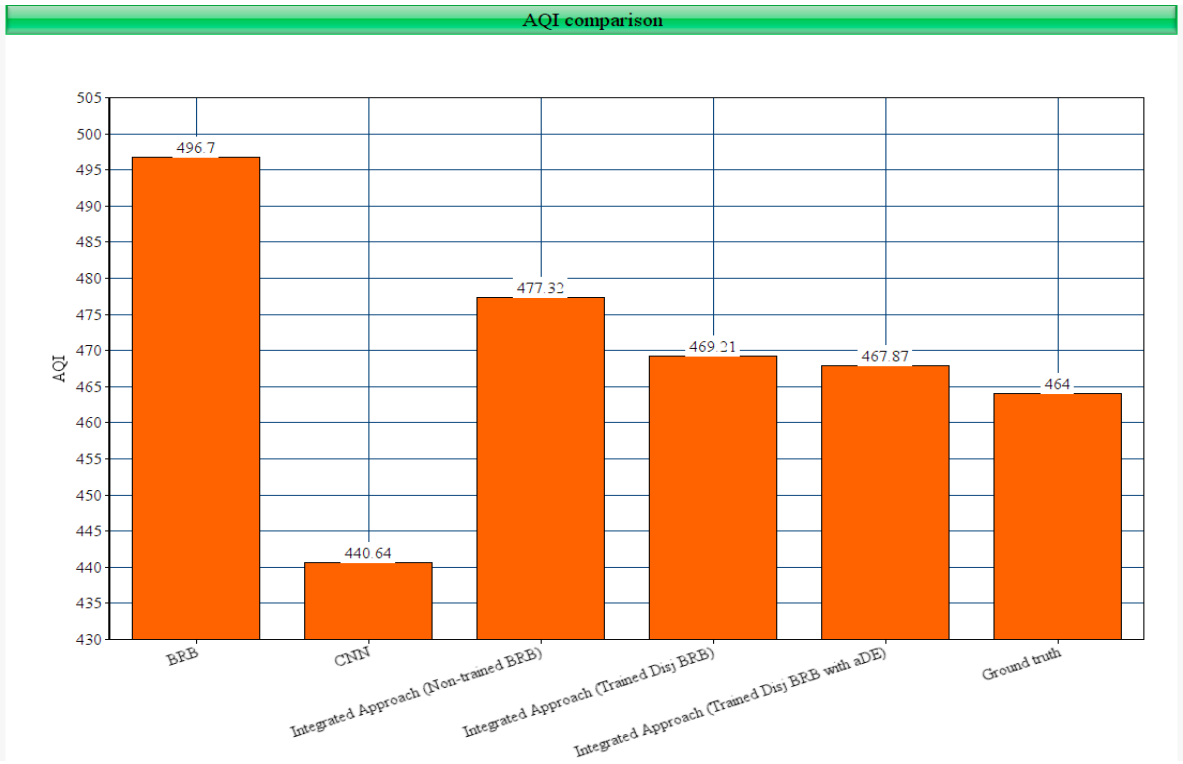

Fig. 36. AQI prediction by different methods.

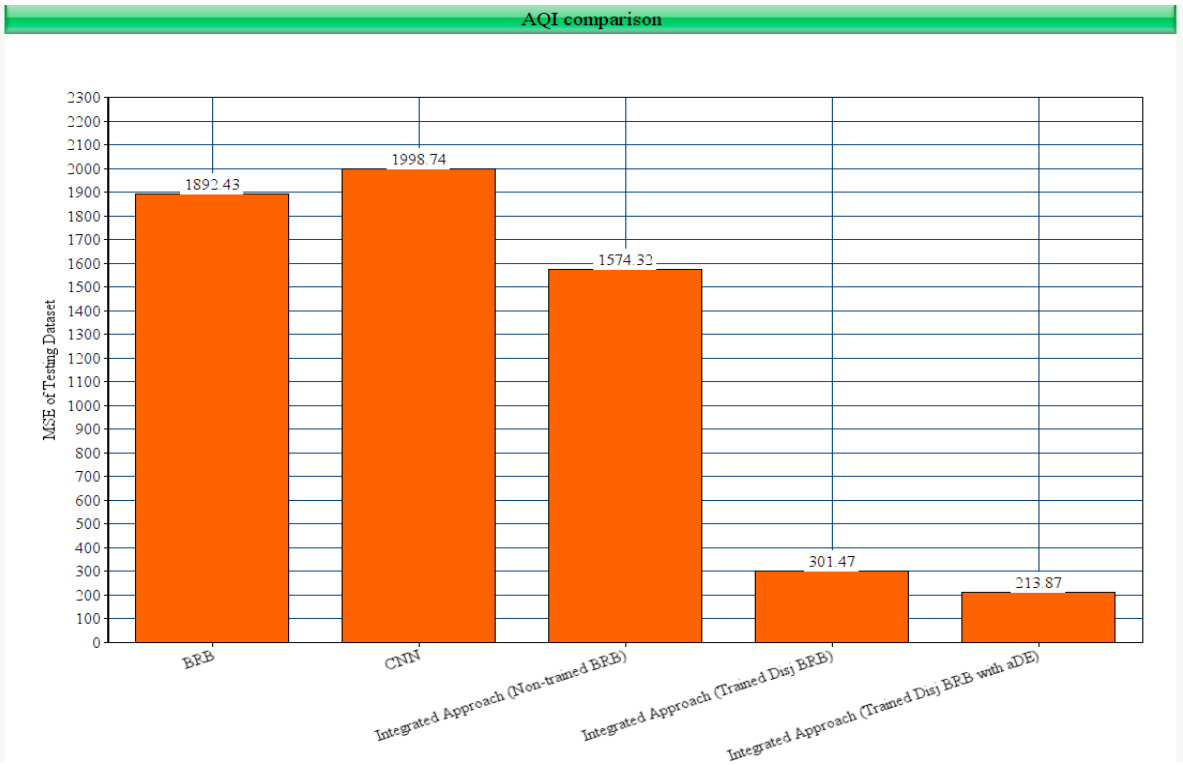

Fig. 37. Testing dataset MSE of various methods.

6.5 Sustainability Aspects

Sustainability refers to the development which addresses the need of the present without compromising the need of the future generations to meet their demands [28]. The term "Sustainable Development" was first coined in October 1987 by "Our Common Future", also known as Brundtland Report [47]. It is a framework to ensure balanced change and avoid exploitation of natural resources. Moreover, sustainable development is becoming increasingly significant day by day as the world is moving towards Sustainable Development Goals (SDG) set by the United Nations (UN) to be achieved by 2030 [21].

Our proposed predictive model enables the concerned authorities to warn people of air pollution and advise them to skip the polluted route. Thus, our model facilitates healthy lifestyle of citizens. Hence, it directly contributes to SDG Goal 3: Good health and well-being. Citizens' sound health also contributes to enhance the overall sustainability of cities and communities. Thus, our thesis plays a supportive role to achieve SDG Goal 11: Sustainable cities and communities. Moreover, municipality can come to know the location of polluted area from the physical position of the deployed sensors. This location awareness lets the municipality take appropriate actions to address the causes of air pollution. Such actions include shutting down brick kilns, enforcing a ban on burning fossil fuel, bringing down vehicular emission etc. In this way, our proposed approach supports SDG Goal 13: Climate action.

There are three major pillars of sustainability: social, economic and environmental [38]. These are also denoted as people, profit and planet. This research has delivered on all three aspects of sustainability with respect to air pollution prediction.

Social Pillar: Social development and public welfare are the key features of this pillar.
Economic Pillar: This pillar covers the economic and monetary benefits.
Environmental Pillar: This pillar puts emphasis on conservation of natural resources and impact of physical development on environment, such as, carbon footprint from burning fossil fuel.

As this thesis is in partial fulfillment of the Erasmus Mundus Master PERCCOM degree, it puts emphasis on sustainability in the context of ICT. Sustainability of a software determines how it will function under varying conditions. A software developed today is no longer isolated from the society. Rather, it is part of the socio-technical system within which it is deployed. Hence, a software system's potential chains-of-effects on sustainability plays a significant role in designing a software [25]. The software architecture needs to take into account its long-term impact in terms of sustainability. However, only long-term consequence cannot address all aspects of sustainability [83]. In addition to long-term cumulative impacts, a software system needs to address its short-term effects as well [7].

We have developed Sustainability Awareness Diagram (SusAD) [25], in line with Becker's model [6], to visualize sustainability impacts of our proposed AQI prediction system as a software service, as shown in Fig. 38. It has five interrelated dimensions: individual,

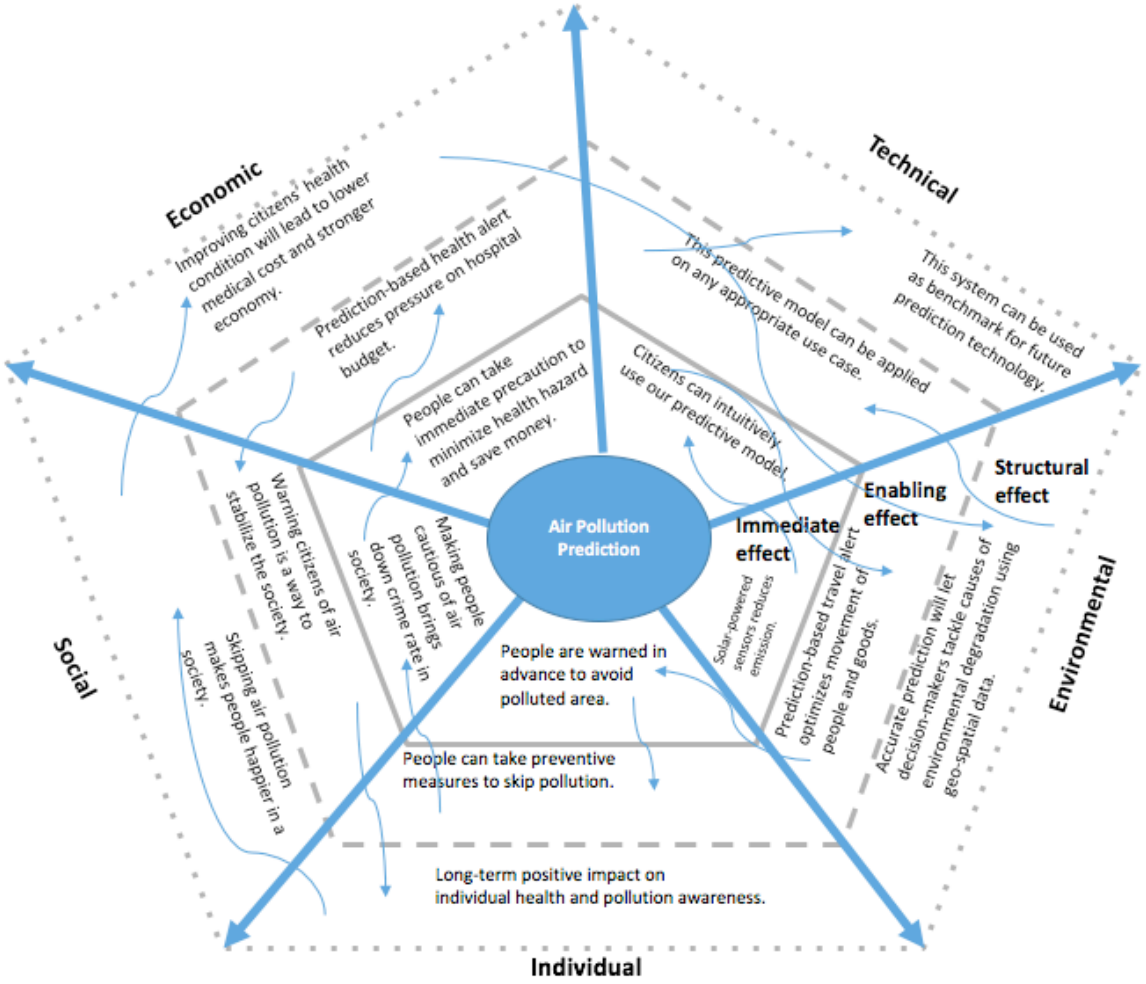

Fig. 38. SusAD diagram for AQI prediction system.

environmental, economic, technical and social. Individual dimension concerns freedom of expression and social rights of inhabitants. The environmental dimension is intended to combat climate change and minimize natural resource consumption. The economic dimension focuses on monetary issues, such as, liquidity of the currency and capital growth. Technical dimension ensures proper preservation, flexibility and progression of software system with the passage of time. Finally, social dimension is with regard to the relationship among individuals in a group or society.

Each of these five dimensions are categorized into three segments: immediate, enabling and structural [109]. Immediate impact is caused by the direct function of the system while the opportunities it creates over a long period of time is the enabling effect. Structural effect refers to the transformation triggered by the system on a large scale, that can be observed in a macro level.

Our developed system informs people of AQI prediction to let them have a holistic assessment of air quality. Thus, it creates immediate impact on citizens to take precautionary measures. We propose to run our sensors using solar power to bring down the reliance on fossil fuel. Thus, we reduce hardware-related emission, which has been shown as immediate effect of environmental dimension. Output of our predictive model is legible enough for the common people to understand and act. This is the immediate effect of the technical dimension of our system. As people are getting aware of air quality of their surrounding over a significant amount of time, it triggers enabling positive effect on their health and lifestyle. In addition to air pollution prediction, our predictive model can be applied on any suitable application area of sensor data streams, which makes enabling impact of our software system from technical dimension. Further, the structural effect of our system facilitates the development of a resilient and healthy community by minimizing the adverse impact of air pollution on public health. Our predictive model can act as a benchmark for developing future prediction algorithms, which constitutes a strong structural effect of our software system from technical perspective.

Now, we explain the chains-of-effects (as indicated by curved arrows in Fig. 38) of our predictive model on sustainability. We propose to deploy solar-powered sensor devices to reduce hardware-related emission. Prediction computed based on data produced by these green sensors are user-friendly enough for the citizens to understand and act. Prediction based travel

alert optimizes vehicular movement and warns people to take preventive measures to skip pollution. Making people cautious of air pollution brings down crime rate in society and air pollution related diseases of people. Thus, our proposed prediction model reduces medical expenditure, which in turn, stabilizes a society and leaves a long-term positive impact on public health. Therefore, people enjoy sound health and higher economic solvency. Geo-spatial data produced by sensor devices enable authorities to address causes of air pollution on a long-term basis. In addition to air pollution, our prediction model can be applied on any appropriate use case. Moreover, our system can be used as benchmark to develop future prediction technology.

6.6 ICT Ethics

Ethics is a branch of philosophy which concerns moral principles and distinguishes between right and wrong. Information and Communication Technology (ICT) refers to the digital technologies. Ethical issues in the context of ICT is called ICT ethics.

ICT ethics analyses the essence and social footprint of digital technology and formulates policies to ensure ethical and lawful use of such technology [77]. It considers both personal and social strategies to govern the principled use of digital technology. ICT ethics is intended to address policy gap regarding how digital technology will be deployed. It stresses certain values, such as, privacy, transparency, intellectual property, freedom of speech, responsibility etc. [19]. It is expected to carry out watchful observation to avert the risk of detrimental use of ICT [36].

This research is upholding ICT ethics as our work does not hamper the privacy of any individual. We are not dealing with personal data of the people. Neighboring people are made aware of the existence of a physical camera sensor which is capturing images every now and then. Physical deployment of both camera and $PM_{2.5}$ sensors are subject to concerned authority's approval and awareness of local people. Sensor data we are collecting are used for prediction purpose only with a view to promoting healthy lifestyle of citizens and appropriate policy-making decisions. It is neither handed over to any 3rd party nor available in a public repository. Confidentiality of the sensor data is maintained fully.

6.7 Summary

This chapter has presented the comparative analysis of our BRB based Deep Learning approach with only BRB and only CNN approach where our integrated approach has shown higher accuracy than separate application of only BRB and only CNN. It has also shown the performance of conjunctive and disjunctive BRB in both trained and non-trained state using six different performance metrics. The chapter has been wrapped up with sustainability aspects and ICT ethics of our research work. Next chapter will conclude our thesis and give direction for future works.

7. Conclusion and Future Works

This chapter concludes our thesis by highlighting our objective and what we have done to realize our objective. Moreover, this chapter sheds light on various limitations of our thesis and our future plan to address these constraints.

7.1 Conclusion

Since incidents of air pollution, wildfire, flood, cyclone, drought, human diseases etc. are going up day by day, it has become very crucial to predict such events well before their actual occurrence. Early prediction facilitates precautionary measures to save human lives and assets as well as bring down the level of casualties. Hence, accuracy of such predictive models is always vital to achieve the desired objective. In this thesis, we continued the line of research on developing a predictive model with improved accuracy. We took air pollution prediction as use case of our proposed predictive model.

We clarified the concept of Artificial Intelligence, Machine Learning and Deep Learning as well as the distinction among them. We explained knowledge-driven and data-driven approach, predictive analytics, sensor data streams as well as uncertainties associated with such sensor data. We investigated various drawbacks of the existing models for predicting air pollution. Some of these models use only neural network for prediction purpose, while other models apply only image based prediction. However, none of these models dealt with all types of uncertainties associated with sensor data. We developed a novel mathematical model to combine strength of BRB and Deep Learning to improve prediction accuracy, rather than being reliant on one single approach. We addressed all sorts of sensor data uncertainties with BRB while utilizing the strength of applying Deep Learning to learn image features. By taking advantage of the benefits of multimodal learning, our proposed BRB based Deep Learning approach outperformed other approaches in terms of prediction accuracy. As Deep Learning method, we clearly justified the adoption of VGGNet over other CNN architectures. We employed DSR as our research methodology to conduct this research. We used python language to implement our model. Interaction among different components of our integrated model, ranging from input to final output, has been demonstrated through system architecture. We complied with the global standard by predicting the air quality level in terms of globally

used AQI. Also, we made distributed categorization of AQI, instead of showing one single AQI category. We optimized our model further by incorporating disjunctive BRB as well as trained BRB into it (as demonstrated in Sect. 6.3 and 6.4). We gave a detailed overview of the air pollution dataset we have used in this research. We performed comparative analysis of our model with performance metrics MSE, RMSE, MAE, RAE, RRSE and R-Squared as well as ROC curve. We also examined sustainability aspects of our research and clarified how this research is compliant with ethical standard of ICT.

In closing, our approach lets the citizens be aware of likely polluted environment and the authorities take preventive steps. In addition to air pollution, our proposed model can be applied on various other appropriate use cases, such as, floor risk prediction, wildfire prediction, breast cancer prediction and so on. Hence, this model contributes significantly to make the planet more sustainable for future generations. In short, this thesis is an exhibition of power of accuracy in developing predictive model for achieving sustainable development.

7.2 Future Works

Although our system's prediction accuracy is satisfactory, the air pollution dataset we have used is not large enough. In future we hope to evaluate our system performance while dealing with petabytes or yottabytes of data, consisting of hundreds of thousands of outdoor images over a long period of time with corresponding label concerning $PM_{2.5}$ concentrations. We plan to deploy Hadoop ecosystem to accommodate such big data and apply our proposed model on it. That means our designed integrated model should learn deeper representation from large volume of images. Therefore, it can achieve significant gain in terms of prediction accuracy. Further, we have used outdoor ground images in this thesis to predict $PM_{2.5}$ level. Instead of relying on ground images, predicting $PM_{2.5}$ directly from satellite images is also part of our future works. We intend to incorporate geospatial data into our model to inform the users of specific location of the polluted area precisely. In addition to $PM_{2.5}$, we expect to use sensor data of other major air pollutants, such as, PM_{10}, SO_2, NO_2 and CO in future to make our predictive output more representative. We have used training data to train our proposed algorithm and tested its prediction accuracy with testing data. We plan to include real-time validation in our work in future to assess the reliability of our model on real-time basis. At the

same time, we argue that the research community needs to focus on unsupervised approaches to make such predictive models more dynamic.

REFERENCES

[1] Abbott, D. (2014). *Applied predictive analytics: Principles and techniques for the professional data analyst*. John Wiley & Sons.

[2] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y. and Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer networks*, 38(4), 393-422.

[3] Atlas, L., Cole, R., Muthusamy, Y., Lippman, A., Connor, J., Park, D., El-Sharkaw, M. and Marks II, R.J. (1990). A Performance Comparison of Trained Multilayer Perceptrons and Trained Classification Trees. *Proceedings Of The IEEE*, Vol. 78, No. 10, October 1990.

[4] Ball, N. M., Brunner, R. J., Myers, A. D. and Tcheng, D. (2006). Robust machine learning applied to astronomical data sets. I: star-galaxy classification of the Sloan Digital Sky Survey DR3 using decision trees. *The Astrophysical Journal*, 650(1), 497.

[5] Barbounis, T.G., Theocharis, J.B., Alexiadis, M.C. and Dokopoulos, P.S. (2006). Long-Term Wind Speed and Power Forecasting Using Local Recurrent Neural Network Models. *IEEE Transactions On Energy Conversion*, Vol. 21, No. 1, March 2006.

[6] Becker, C., Betz, S., Chitchyan, R., Duboc, L., Easterbrook, S.M., Penzenstadler, B., Seyff, N. and Venters, C.C. (2016). Requirements: The key to sustainability. *IEEE Software*, 33(1), pp.56-65.

[7] Becker, C., Chitchyan, R., Duboc, L., Easterbrook, S., Penzenstadler, B., Seyff, N. and Venters, C.C. (2015). Sustainability design and software: The karlskrona manifesto. In: *Proceedings of the 37th International Conference on Software Engineering*. Volume 2 (pp. 467-476). IEEE Press.

[8] Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and trends in Machine Learning 2*, 1-127.

[9] Bengio, Y., Simard, P. and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Networks* 5, 157–166 (1994).

[10] Bhavsar, H. and Ganatra, A. (2012). A Comparative Study of Training Algorithms for Supervised Machine Learning. *International Journal of Soft Computing and Engineering (IJSCE)*, ISSN: 2231-2307, Volume-2, Issue-4, September 2012.

[11] Boureau, Y.L., Ponce, J. and LeCun, Y. (2010). A theoretical analysis of feature pooling in visual recognition. In: *Proceedings of the 27th international conference on machine learning* (ICML 10) (pp. 111-118).

[12] Buchanan, B. and Shortliffe, E. (1984). Rule-based expert systems: the MYCIN experiments of the stanford heuristic programming project. *Addison-Wesley*, Reading, Massachusetts.

[13] Chang, L., Ma, X., Wang, L. and Ling, X. (2016). Comparative Analysis on the Conjunctive and Disjunctive Assumptions for the Belief Rule Base. In: *2016 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)* (pp. 153-156). IEEE.

[14] Chang, L., Zhou, Z., You, Y., Yang, L. and Zhou, Z. (2016). Belief rule based expert system for classification problems with new rule activation and weight calculation procedures. *Information Sciences*, 336, 75-91.

[15] Chapelle, O., Scholkopf, B. and Zien, A. (2009). *Semi-supervised learning* (chapelle, o. et al., eds.; 2006) [book reviews]. IEEE Transactions on Neural Networks, 20(3), 542-542.

[16] Chang, L., Ma, X., Wang, L. and Ling, X. (2016). Comparative Analysis on the Conjunctive and Disjunctive Assumptions for the Belief Rule Base. In: *2016 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)* (pp. 153-156). IEEE.

[17] Chen, L., Nugent, C.D. and Wang, H. (2011). A knowledge-driven approach to activity recognition in smart homes. *IEEE Transactions on Knowledge and Data Engineering*, 24(6), pp.961-974.

[18] Cireşan, D.C., Meier, U., Gambardella, L.M. and Schmidhuber, J. (2010). Deep, big, simple neural nets for handwritten digit recognition. *Neural computation*, 22(12), pp.3207-3220.

[19] Collste, G. (2008). Global ICT-ethics: the case of privacy. *Journal of Information, Communication and Ethics in Society*, 6(1), 76-87.

[20] Copeland, B.J. (2001). The Turing Test. *Minds and Machines*. 10: 519–539, 2000, Kluwer Academic Publishers.

[21] Costanza, R., Fioramonti, L. and Kubiszewski, I. (2016). The UN Sustainable Development Goals and the dynamics of well-being. *Frontiers in Ecology and the Environment*, 14(2), 59-59.

[22] Dehghan, A., Ortiz, E.G., Shu, G. and Masood, S.Z. (2017). DAGER: Deep Age, Gender and Emotion Recognition Using Convolutional Neural Networks. *Computer Vision Lab*, Sighthound Inc., Winter Park, FL.

[23] Deng, Y., Ren, Z., Kong, Y., Bao, F. and Dai, Q. (2017). A Hierarchical Fused Fuzzy Deep Neural Network for Data Classification. *IEEE Transactions On Fuzzy Systems*, Vol. 25, No. 4.

[24] www.thedailystar.net, (2018). *Dhaka air ranked world's 3rd most polluted: WHO*. [online] Available at: https://www.thedailystar.net/city/air-pollution-in-bangladesh-dhaka-air-ranked-world-3rd-most-polluted-who-1570399 . [Accessed 27 April. 2019].

[25] Duboc L. et al. (2019). Do we really know what we are building? Raising awareness of potential Sustainability Effects of Software Systems in Requirements Engineering. In: Accepted to *International Requirements Engineering Conference (RE'19)*, Sept. 23-27, South Korea.

[26] Dundar, A., Jin, J. and Culurciello, E. (2016). CONVOLUTIONAL CLUSTERING FOR UNSUPERVISED LEARNING. *Workshop track - ICLR 2016*.

[27] www. ec.europa.eu, (2014). *EC (European Commission) Environment*. [online] Available at: http://ec.europa.eu/environment/air/index_en.htm. [Accessed 27 April. 2019].

[28] Emas, R. (2015). The concept of sustainable development: Definition and defining principles. *Florida International University*.

[29] EPA, (2006). *Guidelines for Reporting of Daily Air Quality - Air Quality Index (AQI)*.

[30] Fawcett T. (2004). ROC graphs: Notes and practical considerations for researchers. *Machine learning,* 31(1), pp.1-38.

[31] Fukushima, K., (1980)., "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position". Biological cybernetics, 36(4), 193-202.

[32] Gacenga, F., Cater-Steel, A., Toleman, M. and Tan, W.G. (2012). A proposal and evaluation of a design method in design science research. *Electronic Journal of Business Research Methods*, 10(2), pp.89-100.

[33] Gers, F.A., Schmidhuber, J. and Cummins, F. (1999). Learning to Forget: Continual Prediction with LSTM. *Technical Report IDSIA-01-99*, Lugano, Switzerland.

[34] Giannadaki, D., Lelieveld, J. and Pozzer, A. (2016). Implementing the US air quality standard for $PM_{2.5}$ worldwide can prevent millions of premature deaths per year. *Environmental Health*, 15(1), 88.

[35] www.eesi.org, (2016). *Globally, Air Pollution is fourth largest killer, causing 6.5 million deaths annually*. [online] Available at: https://www.eesi.org/articles/view/globally-air-pollution-is-fourth-largest-killer-causing-6.5-million-deaths- . [Accessed 12 Jan. 2019].

[36] Górniak-Kocikowska, K. (2007). From computer ethics to the ethics of global ICT society. *Library Hi Tech*, 25(1), 47-57.

[37] Gubbi, J., Buyya, R., Marusic, S. and Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7), 1645-1660.

[38] Hansmann, R., Mieg, H.A. and Frischknecht, P. (2012). Principal sustainability components: empirical analysis of synergies between the three pillars of sustainability. *International Journal of Sustainable Development & World Ecology*, 19(5), pp.451-459.

[39] Harrington, P. (2012). Machine learning in action (Vol. 5). *Greenwich: Manning*.

[40] Hevner, A. and Chatterjee, S. (2010). Design science research in information systems. *In Design research in information systems* (pp. 9-22). Springer US.

[41] Herbst MD., Garcia EV., Cooke CD., Ezquerra NF., Folks RD. and DePuey EG. (1992). Myocardial ischemia detection by expert system interpretation of thallium-201 tomograms. In: *Reiber JHC, van der Wall EE (eds) Cardiovascular nuclear medicine and MRI*. Kluger Academic Publishers, Dordrecht, pp 77–88.

[42] Hidetoshi S. (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90 (2): 227-244.

[43] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[44] Hossain, M. S., Rahaman, S., Mustafa, R. and Andersson, K. (2018). A belief rule-based expert system to assess suspicion of acute coronary syndrome (ACS) under uncertainty. *Soft Computing*, 22(22), 7571-7586.

[45] Hossain, M. S., Rahaman, S., Kor, A. L., Andersson, K. and Pattinson, C. (2017). A belief rule based expert system for datacenter pue prediction under uncertainty. *IEEE Transactions on Sustainable Computing*, 2(2), 140-153.

[46] Hossain, M.S., Zander, P.O., Kamal, M.S. and Chowdhury, L. (2015). Belief rule based expert systems for evaluation of e-government: a case study. *Expert Systems*, 32(5), pp.563-577.

[47] Imperatives, S. (1987). Report of the World Commission on Environment and Development: Our Common Future.

[48] Ioffe, S, and Christian S. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*(2015).

[49] Islam, R.U., Hossain, M.S. and Andersson, K. (2018). A novel anomaly detection algorithm for sensor data under uncertainty. *Soft Computing*, 22(5), pp.1623-1639.

[50] Islam, R.U., Hossain, M.S. and Andersson, K. Belief Rule Based Adaptive Differential Evolution Algorithm under Uncertainty. [Submitted for review].

[51] Jaques, N., Gu, S., Turner, R. E. and Eck, D. (2017). TUNING RECURRENT NEURAL NETWORKS WITH REINFORCEMENT LEARNING. *Workshop track* - ICLR 2017.

[52] Jennex, M. E. (2017). Big data, the internet of things, and the revised knowledge pyramid. *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, 48(4), 69-79.

[53] Jonkman, S.N. (2005). Global Perspectives on Loss of Human Life Caused by Floods. *Natural Hazards* (2005) 34: 151–175.

[54] Kaelbling, L.P., Littman, M.L. and Moore, A.W. (1996). Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research* 4 (1996) 237-285.

[55] Kaelbling, L. P., Littman, M. L. and Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of artificial intelligence research*, 4, 237-285.

[56] Keskar, N.S., Mudigere, D., Nocedal, J., Smelyanskiy, M. and Tang, P.T.P., (2016). On large batch training for deep learning: Generalization gap and sharp minima. arXiv preprint arXiv:1609.04836.

[57] Koza, J. R. SURVEY OF GENETIC ALGORITHMS AND GENETIC PROGRAMMING. *Stanford University*, Stanford, California, USA.

[58] Kor, A.L., Rondeau, E., Andersson, K., Porras, J. and Georges, J.P. (2019). Education in Green ICT and Control of Smart Systems: A First Hand Experience from the International PERCCOM Masters Programme. In: *12th Symposium on Advances in Control Education* (ACE 2019), 7-9 July 2019, Philadelphia, USA.

[59] Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *arXiv preprint* arXiv: 1408.5882.

[60] Kingma, D.P. and Ba, J., (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[61] Klapper-Rybicka, M., Schraudolph, N. N. and Schmidhuber, J. (2001). Unsupervised Learning in Recurrent Neural Networks. In: *Proc. Intl. Conf. Artificial Neural Networks (ICANN)-2001*, Vienna, Austria.

[62] Krizhevsky, A., Sutskever, I. and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *In Advances in neural information processing systems* (pp. 1097-1105).

[63] Kurt, A., Oktay, and Betül, A. (2010). Forecasting air pollutant indicator levels with geographic models 3 days in advance using neural networks. *Expert Systems with Applications* 37 (2010) 7986–7992.

[64] Lawrence, S., Giles, C.L., Tsoi, A.C. and Back, A.D. (1997). Face Recognition: A Convolutional Neural-Network Approach. *IEEE Transactions On Neural Networks*, Vol. 8, No. 1, JANUARY 1997.

[65] LeCun Y, Bengio Y. and Hinton, G. (2015). Deep learning. *NATURE* Vol 521 doi:10.1038/nature14539.

[66] Lev, G., Sadeh, G., Klein, B. and Wolf, L. (2016). Rnn fisher vectors for action recognition and image annotation. In: *European Conference on Computer Vision* (pp. 833-850). Springer, Cham.

[67] Long, J., Shelhamer, E. and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3431-3440).

[68] Li, X., Peng L., Hu Y., Shao, J. and Chi, T. (2016). Deep learning architecture for air quality predictions. *Environ Sci Pollut Res* (2016) 23:22408–22417. DOI 10.1007/s11356-016-7812-9.

[69] Li, Y., Huang, J. and Luo, J. (2015). Using User Generated Online Photos to Estimate and Monitor Air Pollution in Major Cities. In: *ICIMCS-2015*, Zhangjiajie, Hunan, China.

[70] Li, Y., Huang, J. and Luo, J. (2015). Using user generated online photos to estimate and monitor air pollution in major cities. In: *Proceedings of the 7th International Conference on Internet Multimedia Computing and Service* (p. 79). ACM.

[71] Li, Y., Wang, H., Ding, B. and Che, H. (2017). Learning from internet: handling uncertainty in robotic environment modeling. In: *Proceedings of the 9th Asia-Pacific Symposium on Internetware* (p. 7). ACM.

[72] Lim, S. S., Vos, T., Flaxman, A.D., Danaei, G. and Shibuya, K. (2010). A comparative risk assessment of burden of disease and injury attributable to 67 risk factors and risk factor clusters in 21 regions, 1990–2010: a systematic analysis for the Global Burden of Disease Study 2010. *Lancet*. 2012;380:2224–60.

[73] Lipton, Z. C., Berkowitz, J. and Elkan, C. (2015). A critical review of recurrent neural networks for sequence learning. *arXiv* preprint arXiv:1506.00019.

[74] Liu, B., Wang, M., Foroosh, H., Tappen, M. and Penksy, M. (2015). Sparse Convolutional Neural Networks. *CVPR2015*, Computer Vision Foundation.

[75] Liu, C., Tsow, F., Zou, Y. and Tao, N. (2016). Particle Pollution Estimation Based on Image Analysis. *PLoS ONE* 11(2):e0145955. https://doi.org/10.1371/journal.pone.0145955

[76] Mamoshina, P., Vieira, A., Putin, E. and Zhavoronkov, A. (2016). Applications of deep learning in biomedicine. *Molecular pharmaceutics*, 13(5), 1445-1454.

[77] Moor, J. H. (1985). What is computer ethics? *Metaphilosophy*, 16(4), 266-275.

[78] Mosavi, A. and Varkonyi-Koczy, A. R. (2017). Integration of Machine Learning and Optimization for Robot Learning. *Advances in Intelligent Systems and Computing* 519, 349-355.

[79] Munoz, A. Machine Learning and Optimization. *Courant Institute of Mathematical Sciences*, New York, NY.

[80] Munrat, AA 2018, *A Belief Rule Based Flood Risk Assessment Expert System Using Real Time Sensor Data Streaming,* Lulea University of Technology, Lulea.

[81] Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H. and Ng, A. Y. (2011). Multimodal deep learning. In: *Proc. 28th Int. Conf. Mach. Learn*., 2011, pp. 689–696.

[82] Pal, K.K. and Sudeep, K.S. (2016). Preprocessing for image classification by convolutional neural networks. In: *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)* (pp. 1778-1781). IEEE.

[83] Perdan, S. (2004). Introduction to sustainable development. In Sustainable development in practice: case studies for engineers and scientists. *Wiley*, West Sussex, England.

[84] Riaz, M. M. and Ghafoor, A. (2012). PRINCIPLE COMPONENT ANALYSIS AND FUZZY LOGIC BASED THROUGH WALL IMAGE ENHANCEMENT. *Progress In Electromagnetics Research*, Vol. 127, 461–478, 2012.

[85] Riedmiller, M.A. (1994). Advanced supervised learning in multi-layer perceptrons — From backpropagation to adaptive learning algorithms. DOI:10.1016/0920-5489(94)90017-5.

[86] Saad S. M., Shakaff, A. Y. M, Saad, A. R. M, Yusof, A. M., Andrew, A., Zakaria, M. A. and Adom, A. H. (2017). Development of indoor environmental index: Air quality index and thermal comfort index. *AIP Conference Proceedings*. 1808. 020043. 10.1063/1.4975276.

[87] Sak, H., Senior, A. and Beaufays, F. (2014). Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In: *Fifteenth annual conference of the international speech communication association*.

[88] Schmidhuber, J. (2015). Deep Learning in Neural Networks: An Overview. *Neural Networks*. 61: 85–117. arXiv:1404.7828free to read. doi:10.1016/j.neunet.2014.09.003.

[89] Shmueli, G. and Koppius, O. R. (2011). Predictive analytics in information systems research. *MIS quarterly*, 553-572.

[90] Siegelmann, H. T., Horne, B. G. and Giles, C. L. (1997). Computational capabilities of recurrent NARX neural networks. *IEEE Transactions on Systems, Man, and Cybernetics*, Part B (Cybernetics), 27(2), 208-215.

[91] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint* arXiv:1409.1556.

[92] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R., (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research,* 15(1), pp.1929-1958.

[93] Storn, R. and Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4), 341-359.

[94] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... and Rabinovich, A. (2015). Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).

[95] www.worldbank.org, (2016). *The Cost of Air Pollution - Strengthening the Economic Case for Action*. [online] Available at: http://documents.worldbank.org/curated/en/781521473177013155/pdf/108141-REVISED-Cost-of PollutionWebCORRECTEDfile.pdf . [Accessed 21 Feb. 2018].

[96] Theang, Ong B., Sugiura, K. and Zettsu, K. (2016). Dynamically pre-trained deep recurrent neural networks using environmental monitoring data for predicting $PM_{2.5}$. *Neural Comput & Applic* 27:1553–1566.

[97] Tvrdık J. (2006). Competitive differential evolution and genetic algorithm in GA-DS toolbox. *Tech. Comput.* Prague, Praha, Humusoft, 1(2), pp.99-106.

[98] Vaezipour, A., Mosavi, A. and Seigerroth, U. (2013). Machine learning integrated optimization for decision making. In: *26th European Conference on Operational Research*, Rome, Italy.

[99] Vaezipour, A, Mosavi, U. and Seigerroth, A. (2013). Visual analytics and informed decisions in health and life sciences. In: *International CAE Conference*, Verona, Italy.

[100] Vargas, R., Mosavi, A. and Ruiz, L. (2017). DEEP LEARNING: A REVIEW. *Advances in Intelligent Systems and Computing*.

[101] Vinyals, O., Toshev, A., Bengio, S. and Erhan, D. (2015). Show and Tell: A Neural Image Caption Generator. *CVPR2015*, Computer Vision Foundation.

[102] Wang, C., Tu, Y., Yu, Z. and Lu, R. (2015). $PM_{2.5}$ and Cardiovascular Diseases in the Elderly: An Overview. *Int. J. Environ. Res. Public Health 2015*, 12, 8187-8197.

[103] Wang Y-M, Yang J-B. and Xu D-L .(2006). Environmental impact assessment using the evidential reasoning approach. *Eur J Oper Res* 174(3):1885–1913.

[104] Weber, S. (2010). Design Science Research: Paradigm or Approach? In: *AMCIS* (p. 214).

[105] www.who.int, (2014). *7 million premature deaths annually linked to air pollution*. [online] Available at: http://www.who.int/mediacentre/news/ releases/2014/air-pollution/en/. [Accessed 27 Apr. 2019].

[106] WHO Regional Office for Europe, OECD. (2015). Economic cost of the health impact of air pollution in Europe: Clean air, health and wealth. Copenhagen, Denmark.

[107] Widrow, B., Lehr, M. A., (1990), "30 years of adaptive neural networks: perceptron, madaline, and backpropagation". Proceedings of the IEEE, 78(9), 1415-1442.

[108] World Health Organization. (2016). Ambient air pollution: A global assessment of exposure and burden of disease.

[109] Wu, L., Subramanian, N., Abdulrahman, M.D., Liu, C. and Pawar, K.S. (2017). Short-term versus long-term benefits: Balanced sustainability framework and research propositions. *Sustainable Production and Consumption*, 11, pp.18-30.

[110] Xiong, Q., Chen, G., Mao, Z., Liao, T. and Chang, L. (2017). Computational requirments analysis on the conjunctive and disjunctive assumptions for The Belief Rule Base. In: *2017 International Conference on Machine Learning and Cybernetics (ICMLC)* (Vol. 1, pp. 236-240). IEEE.

[111] Xu, Y., Dai, Z., Chen, F., Gao, S., Pei, J. and Lai, L. (2015). Deep Learning for Drug-Induced Liver Injury. *J. Chem. Inf. Model*. 2015, 55, 2085− 2093.

[112] Yang J-B, Singh MG. (1994). An evidential reasoning approach for multiple-attribute decision making with uncertainty. *IEEE TransSyst Man Cybern* 24(1):1–18.

[113] Yang, JB., Liu, J., Wang, J., Sii H-S. and Wang, H-W. (2006). Belief rule-base inference methodology using the evidential reasoning approach-RIMER. *IEEE Trans Syst Man Cybern* Part A Syst Hum 36(2):266–285.

[114] Yang, L. H., Wang, Y. M., Liu, J. and Martínez, L. (2018). A joint optimization method on parameter and structure for belief-rule-based systems. *Knowledge-Based Systems*, 142, 220-240.

[115] Yang, L. H., Wang, Y. M., Liu, J. and Martínez, L. (2018). "A joint optimization method on parameter and structure for belief-rule-based systems". Knowledge-Based Systems, 142, 220-240.

[116] Yuan, Y., Feldhamer, S., Gafni, A., Fyfe, F. and Ludwin, D. (2002). The development and evaluation of a fuzzy logic expert system for renal transplantation assignment: is this a useful tool? *Eur J Oper Res142(1)*:152–173.

[117] Zeiler, M.D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In: *European conference on computer vision* (pp. 818-833). springer, Cham.

[118] Zhan, Y., Zhang, R., Wu, Q. and Wu, Y. (2016). A new Haze image database with detailed air quality information and a novel no–reference image quality assessment method for haze images. In: *ICASSP-2016*, Shanghai, China.

APPENDIX 1. Software Repositories and Files

1. Implementation of VGGNet architecture of CNN: https://github.com/samikabir/CNN
2. Implementation of conjunctive BRB based CNN approach: https://github.com/samikabir/CNN_ConjBRB
3. Implementation of disjunctive BRB based CNN approach: https://github.com/samikabir/CNN_DisjBRB
4. Implementation of DE-optimized conjunctive BRB based CNN approach: https://github.com/samikabir/CNN_ConjBRB_DE
5. Implementation of DE-optimized disjunctive BRB based CNN approach: https://github.com/samikabir/CNN_DisjBRB_DE
6. Implementation of joint optimized conjunctive BRB based CNN approach: https://github.com/samikabir/JointOptimizedConjBRB
7. Implementation of joint optimized disjunctive BRB based CNN approach: https://github.com/samikabir/JointOptimizedDisjBRB
8. BRBaDE with conjunctive BRB: https://github.com/samikabir/AdaptiveDE-ConjBRB
9. BRBaDE with disjunctive BRB: https://github.com/samikabir/AdaptiveDE-DisjBRB

.

APPENDIX 2. Installation Dependencies

We have used Python 3.6.4 in the implementation part of this thesis. Further, keras neural network library, openCV library, numpy, matplotlib and other necessary libraries have to be installed in the machine to run the code successfully.