UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

AI BASED FAULT DIAGNOSIS IN EMERGING CELLULAR NETWORKS

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE IN ELECTRICAL AND COMPUTER ENGINEERING

BY

SHRUTI BOTHE

Norman, Oklahoma

2019

AI BASED FAULT DIAGNOSIS IN EMERGING CELLULAR NETWORKS

A THESIS APPROVED FOR THE

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY

Dr. Ali Imran, Chair

Dr. Gregory MacDonald

Dr. Samuel Cheng

Acknowledgments

I would like to express my sincere gratitude to my advisor, Dr. Ali Imran, who has led me to the field of Data Science and advanced technologies in the cellular network. His continuous support during my master's study and research is impeccable. His expertise, guidance, support and patience added considerably to my graduate research experience. I am very fortunate and blessed to have had an opportunity to carry out my very first research work under his guidance.

His enthusiasm and in-depth understanding for research has motivated me to pursue career opportunities in this field. I could not have imagined having a better advisor and mentor for my master's study.

I would also like to thank the other committee members, Dr. Gregory MacDonald for his valuable comments on my work and directing me towards key-concepts during my very first classes of Data Analytics and Dr. Samuel Cheng for his insightful discussions and suggestions.

I am very grateful to my fellow colleagues in the 'AI4SON Center' for their support and for always maintaining a very healthy research conducive work environment. They are my colleagues, who became friends and are now are like family.

I want to dedicate this research work to my family who always encouraged me throughout my master's program. I am extremely grateful for their love, prayers,

moral support and sacrifices in educating and preparing me for my future. I would have not made it this far without them.

Table of Contents

List of Figures

List of Tables

Abstract

Network fault diagnosis aims at detecting and avoiding network problems, and thus increasing a network's availability, keeping downtimes to a minimum, and improving network performance. Mobile cellular network operators spend nearly a quarter of their revenue on network management and maintenance. A significant portion of that budget, is spent on resolving faults diagnosed in the system that degrade or disrupt cellular services. Historically, the operations to detect, diagnose and resolve issues were carried out by human experts. However, with growing cell density, diversifying cell types and increased complexity, this approach is becoming less and less viable, both technically and financially. To cope with this problem, research on self-healing solutions has gained significant momentum in recent years. One of the most desirable features of the self-healing paradigm is automated fault diagnosis.

While several fault detection and diagnosis machine learning models have been proposed recently, these schemes have one common tenancy. They still rely on human expert contribution for fault diagnosis and prediction in one way or another. In this thesis, an AI-based fault diagnosis solution that offers a key step forward towards a completely automated self-healing system without requiring human expert input is proposed. The proposed solution leverages Convolutional Neural Network and Random Forests classifier based deep learning model which uses RSRP map images of faults generated. Comparison of the performance of the proposed solution against state-of-the-art solutions in literature that mostly use Naïve Bayes models, while considering seven different commonly occurring fault

types is performed. The comparison uncovers several advantages of the proposed approach. Results show that Random Forests classifier achieves high classification accuracy as compared to Convolutional Neural Networks and Naïve Bayes even with relatively small training data.

CHAPTER 1

Introduction

1.1 Fault Diagnosis Overview

The last decade has witnessed unprecedented growth in the cellular network usage thereby making system operations more complex. Billions of dollars are spent in the United States alone by cellular operators to detect and diagnose faults [1]. With the genesis of 5G ultra dense network, fault management will be a primary challenge. It is not difficult to foresee that with such an amalgam of technologies and operational complexity required to configure, operate, optimize and maintain the networks the biggest challenges in the emerging cellular networks [2] would be to automate the fault diagnosis process.

Typically, a single fault in the network system always produces a large amount of alarm information. Operators then have to depend on domain experts to diagnose the exact root cause and devise a solution. The drawback of this is that it is very time consuming and is prone to human errors given the unfathomable complexity of the system. From a network operator's perspective, achieving accurate and timely diagnosis of the cause of the faults is critical for both improving subscriber perceived experience and maintaining network reliability.

Network applications necessitate that network fault issues be resolved in a very short time. The complexity of fault diagnosis has led to emergence of current solutions that combine expert input and an automated system. Although research is on-going, most of the current fault diagnosis systems were built on make-shift and incoherent basis. This involved simply transferring the knowledge of the human expert into a system that has the capability to reflect actions of a human expert when solving problems in a particular domain. But, by taking into consideration future emerging cellular networks, their complexity and shrinking profit margins; minimizing human interaction for fault diagnosis for Self-Organizing Networks (SON) would unquestionably be desirable [1] [2].

This means that automation for reducing costs, handling complexity and maximizing resources efficiency will not only become a necessity, but future cellular networks will depend on it [1]. The main task of an automated fault diagnosis tool is to identify the cause of problem. The fault management process can be divided into three stages: fault detection, fault diagnosis and testing. This thesis focuses only on the fault diagnosis process which is of significant importance since the speed and the accuracy of the fault management process is heavily dependent on it.

1.2 Previous Work and Motivation

Several recent studies have explored fault diagnosis in cellular networks. Authors in [3] have proposed the use of Bayesian networks to diagnose faults. In [4] authors focused on self-healing, which included all the functionalities that targeted automating troubleshooting in the Radio Access Network (RAN) by using 'if-then' rules. Similarly, in [5] automated diagnosis in troubleshooting (TS) for Universal

Mobile Telecommunications System (UMTS) networks using a Bayesian network (BN) approach is proposed. Alarms are modeled as discrete random variables with two states: OFF or ON. KPIs are modeled as discrete random variables with two, three, or more states. The authors in [6] propose a system for automated diagnosis based on a Naïve Bayesian classifier which is then applied to the identification of the fault cause in GSM/GPRS, 3G or multi-systems networks.

In [7] authors presented a self-healing framework developed for 3GPP Long Term Evolution (LTE) networks to provide a platform where the detection and compensation of cell outages are evaluated in realistic environments. Authors in [8] have compared thresholds obtained with three different algorithms such as knowledge-based method (TEXP), Entropy Minimization Discretization (EMD) and Selective Entropy Minimization Discretization. In [9] authors present an automatic unified detection and diagnosis framework to identify root causes of faults using unsupervised clustering. Five classification algorithms, namely Chi-squared automatic interaction detection (CHAID), quick unbiased efficient statistical tree (QUEST), Bayesian network, support vector machine (SVM) and classification and regression tree (CRT) were used.

An integrated detection and diagnosis framework is presented in [10] where the model can identify anomalies and find the most probable root cause of not only severe problems but even smaller degradations as well. Another self-healing approach is proposed in [11], where an automatic framework for detection and diagnosis is described. This system is built based on expert knowledge, which is derived from the reports of previous fault cases. In [12], a few improvements to this this framework are discussed. The main contribution of this work is to include more sophisticated profiling and detection capabilities.

1.3 Research Objectives

While the fault diagnosis problem has been investigated in several recent studies as discussed in Section 1.2, investigation of a pure machine learning model that eliminates the need for a domain expert contribution for fault diagnosis has not yet been researched on. Hence, the focus of this study is to identify several machine learning models that do the job of fault diagnosis without human expert input while leveraging a data set that does not have pre-defined fault types. The research objective is to incorporate a realistic machine learning model that uses seven commonly occurring fault types in the cellular network for fault diagnosis and then present several new insights based on the analysis.

1.4 Contributions

Although all the above-mentioned works contribute on similar lines in terms of automating the fault diagnosis process, the novel contributions of this thesis are outlined below:

1. To the best of the authors' knowledge, none of the existing studies have used Convolutional Neural Networks (CNN) and Random Forests models as proposed in this thesis.
2. Existing studies use small pre-existing data sets limiting the number and types of faults that can be diagnosed. To overcome this limitation, in this study for the first time, a synthetic data set has been generated. The entire process of generating the dataset has been automated in Atoll RF Design Tool [13] by writing VBScripts.

3. Most existing studies still rely on input/domain knowledge from human expert in one form or other. In the proposed solution, no input from human experts is needed. The models are purely trained on Reference Signal Received Power (RSRP) map images which are labeled as faults. This thesis presents a pure deep learning model where no additional external information provided by experts is needed.
4. The final contribution of this thesis is a comparison between Naïve Bayes solution that is the most widely investigated approach in literature for fault diagnosis, and our proposed Convolutional Neural Network and Random Forests based solutions. Results show that both Convolutional Neural Network and Random Forests outperform the prevailing Naïve Bayes fault diagnosis solutions and Random Forests even outperforms the Convolutional Neural Network (CNN).

1.5 Articles Currently Under Review for Publication

1. **S. Bothe**, U. Masood, H. Farooq and A. Imran, "AI based fault diagnosis in emerging cellular networks", in IEEE GLOBECOM 2019 (submitted in May 2019)

1.6 Organization

The rest of this thesis is structured as follows: Data generation and pre-processing methodology is presented in Chapter II and Chapter III respectively. Chapter IV presents the framework of the fault diagnosis model which includes

CNN, Naïve Bayes and Random Forests models used in this thesis. Performance analysis and numerical results are presented in Chapter V. Chapter VI concludes the thesis and future work is presented in Chapter VII.

CHAPTER 2

Data Generation

The dataset used in this thesis consists of RSRP map images generated in the Atoll RF design tool. Atoll is a multi-technology wireless network design and optimization platform that supports initial design to densification and optimization of the network. The area under consideration is the city of Brussels spanning over 800 km^2.

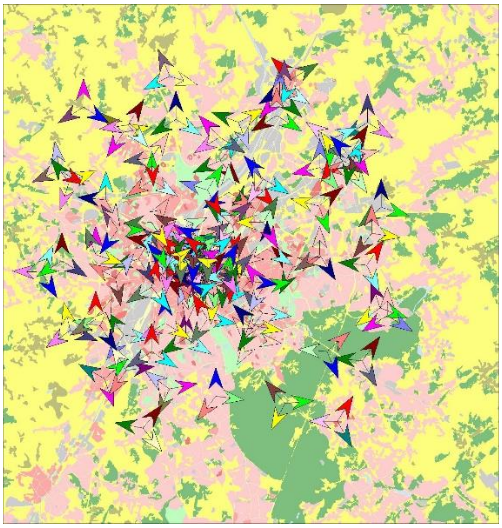


Fig. 2.1: Spatial representation of the Base-Stations

As seen in Fig 2.1, 291 transmitters have been positioned around the city over 15 clutter classes which have been included in this framework and each base-station (BS) has 3 transmitters connected to it. For our analysis, 120 transmitters i.e. 40 BS have been considered. Also, to minimize the boundary effect, all transmitters are kept functional. Table I summarizes the network default settings before any fault is induced. In this data generation technique, 7 faults have been generated one a time. These are generated in the BS in the area of interest. RSRP map images were created for each instance when an error is triggered and are later fed into the aforementioned models. Fig 3 represents the RSRP map images for 'transmitter off' and 'site outage' faults in one of the BS as encircled.

System Parameters	Values
Cellular Layout	120 Macrocell sites
Sectors	3 sectors per BS
Simulation Area	800 km^2
Path Loss Model	Ray-tracing
Land Cover (Clutter) Types	15 different classes
BS Transmit Power	43dBm
Cell Individual Offset	0 dBm
Antenna Tilt	0 deg
Antenna Gain	18.3dBm
Carrier Frequency	2100 MHz
Geographic Information	Ground Heights + Building Heights + Land Use Map

Table 1: Network Scenario Default Settings

Seven most commonly occurring faults in real networks were simulated in Atoll to generate the synthetic data as described below:

1. Cell Outage (C.O): This error was created by turning each transmitter off one at a time.

2. Site Outage (S.O): A BS consisting of 3 transmitters was turned off at one time.

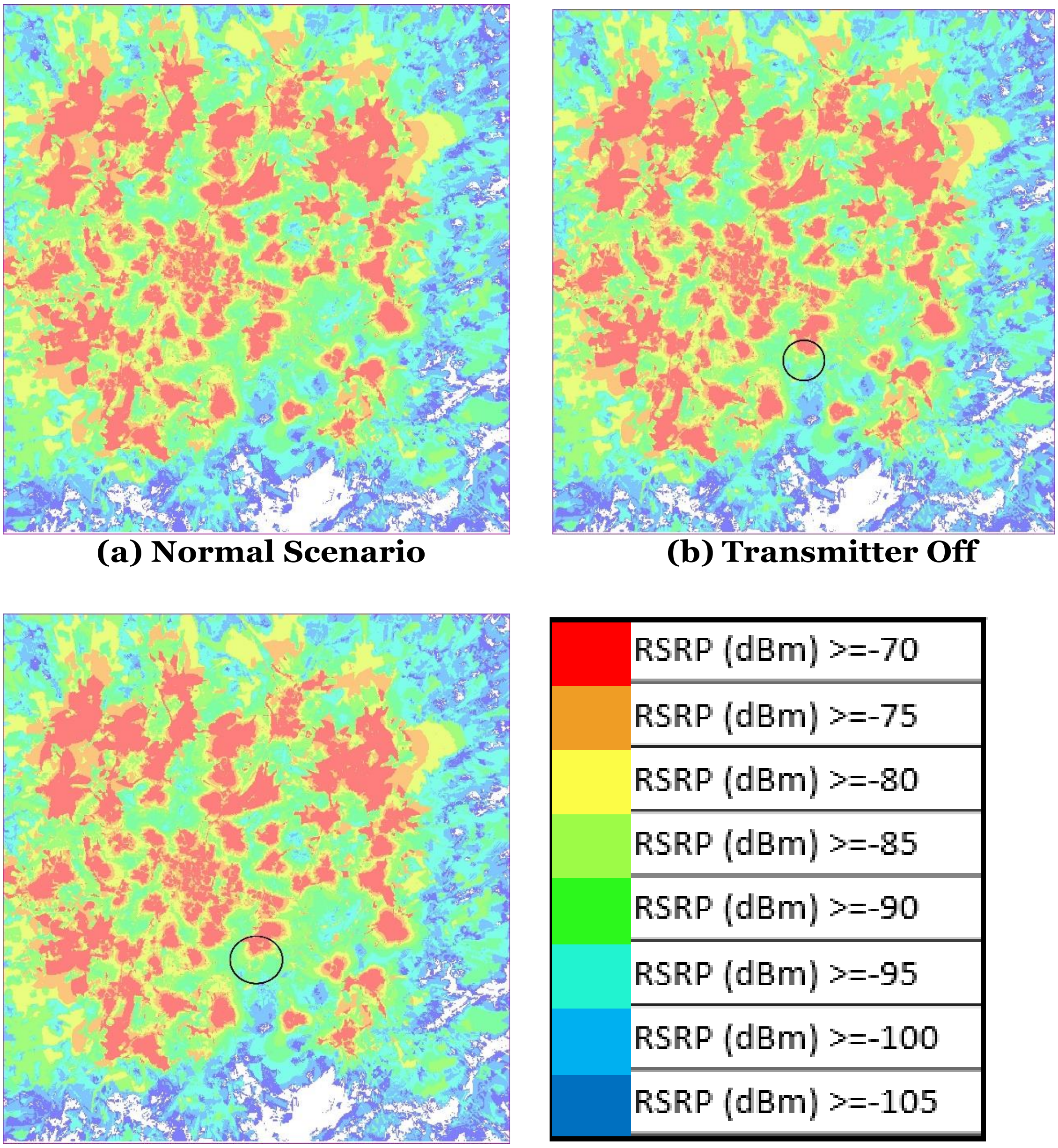
3. Transmission Power (TxP): An error in the transmission power of the BS was induced. The default value of 43 dBm was varied between 25 dBm to 35 dBm with a step size of one to trigger the error.

4. CIO Positive (CIO+): To account for handover parameter error, the CIO was increased to 10 dB from its default value of 0 dB.

5. CIO Negative (CIO-): A second error in CIO was created by reducing the CIO value from its default value of 0 dB to negative 10 dB.

6. Antenna Uptilt (AU): The tilt angle of the antenna was increased from 0 degrees to 25 degrees.

7. Antenna Downtilt (AD): Antenna tilt was decreased from its original value of 0 degrees to negative 25 degrees.



(a) Normal Scenario

(b) Transmitter Off

(c) Site Outage

Fig. 2.2: Comparison of different faults generated in the network

CHAPTER 3

Data Pre-Processing

This dataset has 1961 colored images of size 553 x 578 divided into 8 classes namely Normal Scenario, C.O., S.O., TxP, CIO+, CIO-, AU and AD. Among these images, 1372 have been segregated for training and remaining 589 are used for testing. The advantage of using RSRP map images over raw data for fault diagnosis, as most prior studies do, is that there is no need for identification of smart input features or labeling required from experts. No preprocessing of the data is done before feeding it to the CNN.

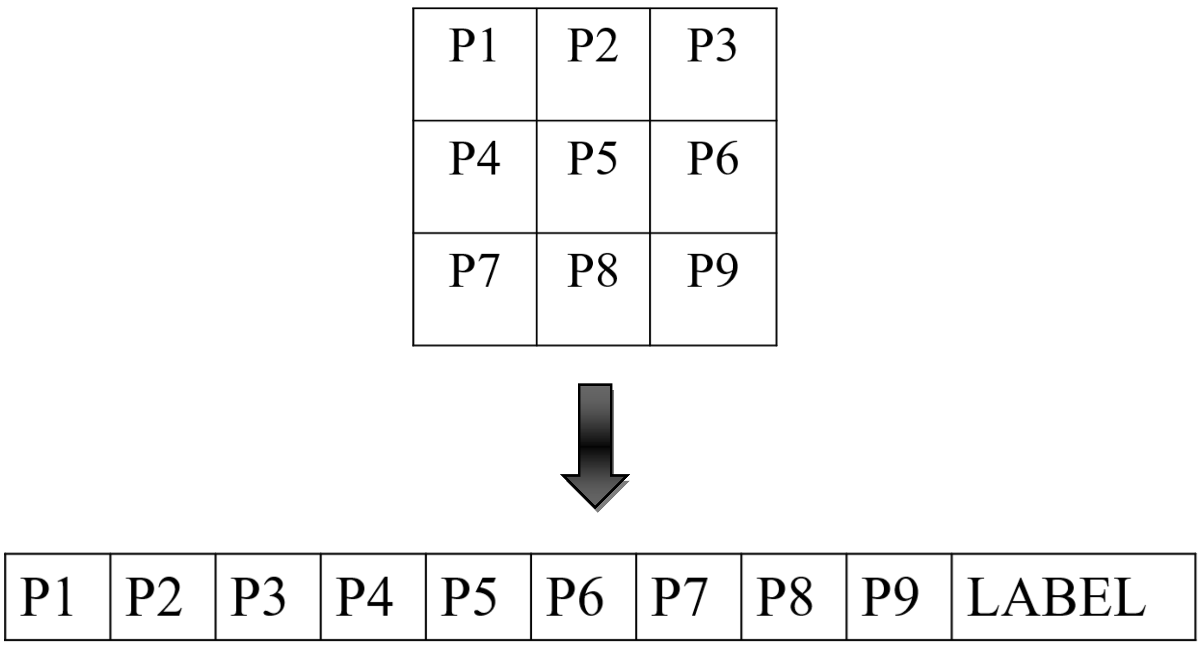


Fig. 3.1: Conversion of an image to a 1-D vector

Since images are simply RSRP maps, in real network such maps can be generated using MDT data and then fed to the proposed diagnosis framework.

However, since Naïve Bayes and Random Forests do not take images as inputs, 1-D matrices corresponding to the RSRP map images are generated. The associated label for each class are induced in the vector.

A digital Grey-scale image is represented by a pixel matrix. Each pixel of such an image is presented by one integer from the set {0,...,255}. The numeric values in pixel presentation uniformly change from zero (black pixels) to 255 (white pixels). RGB images are represented with three Greyscale images matrices (one for each red, green and blue color).

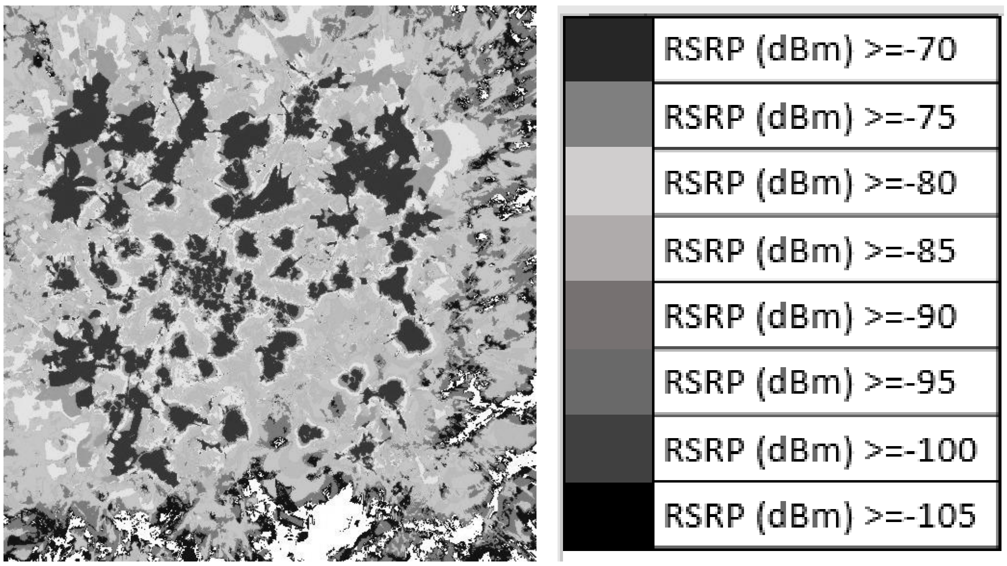


Fig. 3.2: Grey-scale RSRP map used as input to CNN model

All the RSRP map images are also converted to Grey-scale to be fed in these models to compare the networks' performance between Grey-scale and RGB input data. A representation of this is shown in fig 3.2.

CHAPTER 4

Framework for Fault Diagnosis

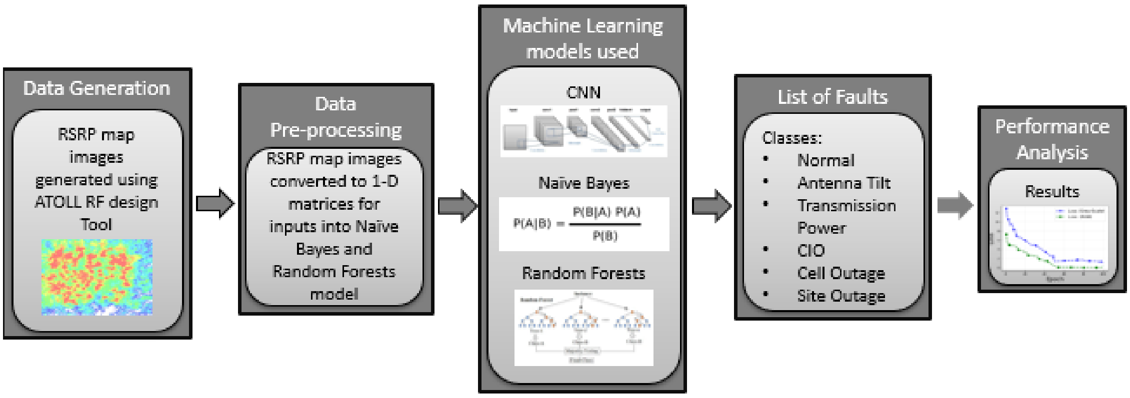


Fig. 4.1: Framework for a Fault Diagnosis Model

Fig 4.1 summarizes the framework for fault diagnosis model. In this section, the different machine learning models implemented in this thesis are discussed.

4.1 Convolutional Neural Network

CNNs also known as 'ConvNet', are Artificial Neural Networks (ANN) that have been most commonly used for analyzing images. Apart from image analysis they can also be used for other data analysis or classification problems as well. This pattern detection is what makes CNNs so useful for images analysis. What differentiates them from the standard multi-layer perceptron or MLP is that they have hidden layers called convolutional layers in addition to non-convolutional layers. The CNN was first introduced by Yann LeCun et al. for gradient-based learning applied to document recognition on the MINST dataset [14]. Since then it has been applied in various research such as sentence classification [15], face recognition [16], traffic signal recognition [17] and many more.

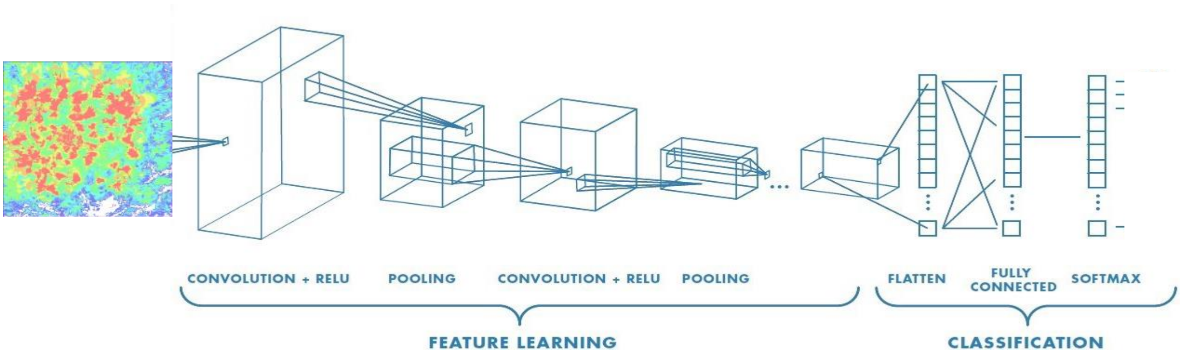


Fig. 4.2: Representation of a Convolutional Neural Network Model

To start, the CNN receives an input feature map: a three-dimensional matrix where the size of the first two dimensions corresponds to the length and width of the images in pixels. The size of the third dimension corresponds to the number of channels of an image. The CNN comprises a stack of modules, each of which performs three operations: convolution, transformation and pooling. To our model, RGB and Grey-scale RSRP map image have been provided. The basic architecture includes a convolutional layer consisting of predefined number of filters, called the feature map, which learns the features from the input image. During a convolution, the filters effectively slide over the input feature map grid horizontally and vertically, one pixel at a time, extracting each corresponding tile. For each filter-tile pair, CNN performs element-wise multiplication of the filter matrix and the tile matrix, and then sums all the elements of the resulting matrix to get a single value.

Each of these resulting values are the outputs in the convolved feature matrix. During training, the CNN 'learns' the optimal values for the filter matrices that enable it to extract meaningful features from the input feature map. As the number of filters applied to the input increases, so does the number of features the CNN can extract. However, the trade-off is that filters compose the majority of resources expended by the CNN, so training time also increases as more filters are added. Following each convolution operation, the CNN applies a Rectified Linear Unit (ReLU) transformation to the convolved feature, in order to introduce nonlinearity into the model. The equation for ReLU can be written as follows:

$$f(x) = \max(0, x) \quad \text{..................... (1)}$$

where x is the input to the neuron. ReLU can allow the model to account for non-linearities and interactions. Neurons are only locally connected by filters, followed by a pooling layer of fixed size 2x2.

For this model max-pooling is implemented. This max- pooling layer down samples the convolved feature, reducing the number of dimensions of the feature map, while still preserving the most critical feature information and to control over fitting. In a fully connected layer, all the neurons consider every activation in the previous layer. Each layer learns its weights and biases using gradient descent in small mini-batches of training samples. In this model, the batch size to train is 32 and the number of output classes is 8. Learning rate is set to 0.001 and the maximum number of epochs is 100. Since the neurons in a fully connected layer have connections to all activations in the previous layer, their activations can be computed with a matrix multiplication followed by a bias offset. In the convolutional layer the neurons are connected only to a local region in the input, and only those many the neurons in a convolutional volume share parameter. However, the neurons in both layers still compute dot products, so their functional form is identical. The last fully-connected layer holds the output.

4.2 Naïve Bayes Classifier Model

Naïve Bayes classifiers are a collection of classification algorithms based on Bayes Theorem with strong (naïve) independence assumptions between the features. It is a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other. It is a classifier that is able to predict, given an observation of an input, a probability distribution over a set of classes, rather than only giving the output for the most likely class that the observation should belong to. Naïve Bayes is particularly useful for large data sets. Bayes' theorem can be defined as follows:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Where, A and B are events and P(B) $\neq 0$

- P(A/B) is a conditional probability defining the likelihood of event A occurring given that B is true.
- P(B/A) is the likelihood of event B occurring given that A is true.
- P(A) and P(B) are the probabilities of observing A and B independently of each other; also known as the marginal probability.

To read the data-set consisting of images, a vector in which each row is a 1-D array associated to an image is created. Next, an array which contains the associated label for each class, i.e. all the fault labels used in this thesis as described above were added to the vector. The Naïve Bayes classifier combines this model with a decision rule: a function which maps an observation to an appropriate action. The corresponding classifier, a Bayes classifier, is the function that assigns a class label. The Bayes classifier used in this thesis is Gaussian Naïve Bayes classifier and can be expressed as:

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(- \frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right)$$

where x_i denotes the feature vector and y is the class variable. σ_y and μ_y are estimated using maximum likelihood. The class with the highest probability is considered as the most likely class. This is also known as Maximum A Posteriori (MAP).

The MAP for a hypothesis is:

$$MAP(H) = \max(P(H|E)) \quad (4)$$

$$= \max((P(E|H) * P(H))/P(E)) \quad (5)$$

$$= \max(P(E|H) * P(H)) \quad (6)$$

P(E) is evidence probability, and it normalizes the result. The advantage of using Naïve Bayes is that it is fast to predict class of test data set. It also performs well in multi-class prediction. When assumption of independence holds, a Naïve Bayes classifier performs better compared to other models like logistic regression and less training data is needed. But the downside of using Naïve Bayes is that if categorical variables have a category (in test data set), which were not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as Zero Frequency. To take into account Zero frequency issues, a few techniques that are often used are m-estimates and Laplace estimates. The parameter m is also known as pseudo count (virtual examples) and is used for additive smoothing. It prevents the probabilities from being 0. For m=1 it is called Laplace smoothing.

4.3 Random Forests

A third model in the proposed framework is the Random Forests classifier. A decision tree is a flowchart-like tree structure where an internal node represents feature, the branch represents a decision rule and each leaf node represents the outcome. Decision tree can handle categorical and numerical data. The decision tree constructs a set of rules which are used to predict a class to classify complex situations. Decision tree can become much more powerful when used as ensembles. These ensembles create state of the art machine learning algorithms that can outperform neural networks in some cases [18]. The most popular ensemble technique is Random Forests.

Although decision tree have their own advantages viz: easy interpretation and straightforward visualization, perform well on large datasets and are extremely fast, this model does not take into account the global optimum. However, choosing the best result at a given step does not ensure that leaf node would lead to the optimal decision. Also, decision tree are prone to over fitting, especially when a tree is particularly deep. Ideally, minimum of both, error due to bias and error due to variance are desired. Because of the above-mentioned disadvantages, Random Forests can be used to mitigate these problems well. A Random Forest is simply a collection of decision trees whose results are aggregated into one final result. Their ability to limit over fitting without substantially increasing error due to bias is why they are such powerful models.

One-way Random Forests reduce variance is by training on different samples of the data. It uses Boostrap resampling method to extract multiple samples from original samples and construct sub datasets, and then uses the sub dataset to form the base decision tree and train it. The bootstrap method is a statistical technique for estimating quantities about a population by averaging estimates from multiple small data samples. A second way is by using a random subset of features. This means, for a set of 8 features, Random Forests will only use a certain number of those features in each model.

Thus, in each tree a minimum number of random features can be utilized. If many trees in forest are used, eventually most or all of the features would have been included. Random Forests are a strong modeling technique and much more robust than a single decision tree. Random Forests also offer a good feature selection indicator. Random Forests use Gini importance or mean decrease in impurity (MDI) to calculate the importance of each feature. The Gini coefficient measures the inequality among values of a frequency distribution. A Gini coefficient of zero expresses perfect equality.

A Gini coefficient of 1 or 100% expresses maximal inequality among values. However, a value greater than one may occur if some data points represent negative contribution to the total. For larger groups, values close to or above 1 are very unlikely in practice. The Random Forests classifiers parameters need to be set, consist of number of trees, number of features, impurity function and stop criteria. Generally, the default values have been used for this. The maximum number of iterations chosen is 100. The depth of tree is limited to 5 levels. The overall accuracy and Kappa Coefficient obtained by RGB images are 77.82% and 0.63, respectively. For the Grey-scale images, these values are 87.16%, and 0.79 respectively.

CHAPTER 5

Performance Analysis

In this section, the results of the proposed methods have been presented. All the models explained earlier perform better for Grey-scale images than RGB images. As seen in Fig 5.1, a visible difference in accuracy can be noted. This difference in accuracy can be attributed to the increase in the number of kernels.

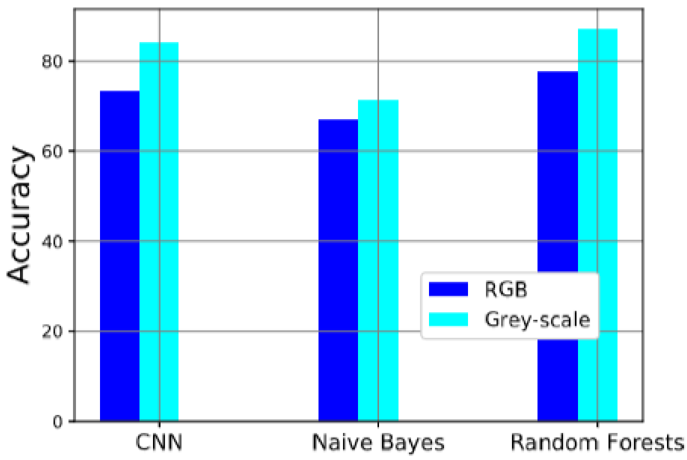


Fig. 5.1: Graph of Accuracy for Grey-scale and RGB images

The kernel size for Grey-scale image is k x k x 1 whereas it is k x k x 3 for an RGB image. Depending on the number of kernels, the number of parameters increases proportionally. Furthermore, large dimensional vectors are almost equally spaced and hence are difficult to separate by a classifier. This can be accredited to the curse of dimensionality phenomena. Although no substantial information is added in an RGB image, it in turn adds to more compute and memory intensive training time due to large number of input parameters. Hence, feeding these models with a Grey-scale image rather than an RGB image results in a better classification accuracy. The Random Forests model has the highest classification accuracy followed by CNN and Naïve Bayes. Some classifier combination techniques like ensembling, bagging and boosting may improve Naïve Bayes classification accuracy, but these methods would not help since their purpose is to reduce variance. Naïve Bayes has no variance to minimize. Accuracy of CNN saturates after a particular number of epochs as the model has limited number of data samples to train from. Neural Networks show best results when the number of data points or images are very large in number. Loss values of the CNN model on the test dataset are represented in fig 5.2.

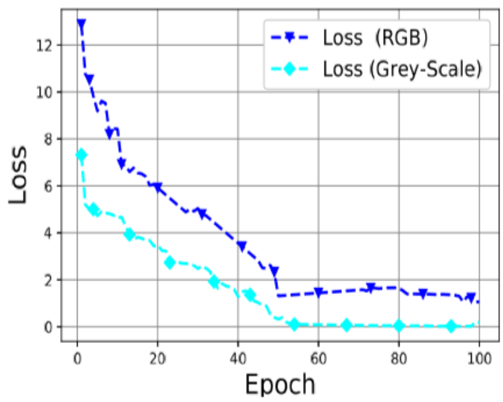


Fig. 5.2: Graph of loss values for Grey-scale and RGB images for CNN model

Classification accuracy of individual fault parameters obtained from Random Forests classifier are represented in fig 5.3. Classification accuracy for S.O. is the highest for RGB images while C.O. are better classified in Grey-scale images. Also, a very small degree of antenna tilt results in a noticeable change in the RSRP map images therefore enabling Random Forests to better classify them as compared to TxP faults.

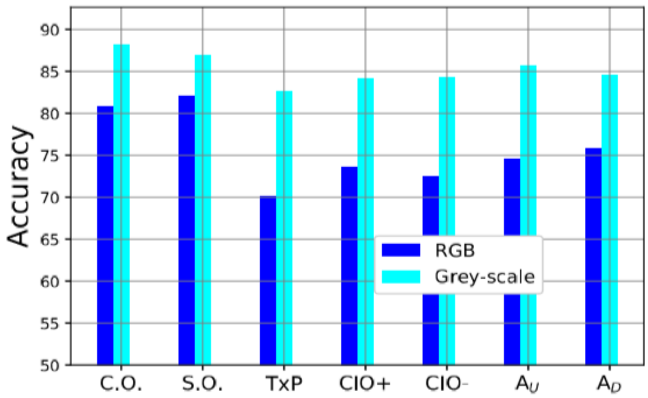


Fig. 5.3: Random Forests prediction accuracy for individual faults

CHAPTER 6

Conclusion

Fault diagnosis is a the highly desirable feature in emerging complex and dense cellular networks to not only reduce operational costs but also to improve quality of experience. In this thesis, a first of its kind CNN and Random Forests based fault diagnosis solutions that outperform the prevailing, Naïve Bayes fault diagnosis schemes in existing literature is presented. Another key advantage of proposed solution is that unlike existing solutions in literature, it does not require human domain knowledge for manual feature extraction. This advantage is achieved by feeding RSRP data in form of images to train the models instead of conventional approach of using raw fault data to train the models. In real networks, these images can be created using MDT reports or RSRP measurements gathered from other sources. The results indicate that the proposed model can flag all seven faults in the fed images. This framework can be used as a part of the self-healing module in emerging networks.

CHAPTER 7

Future Work

For future works, the performance of proposed solution in presence of multiple faults concurrently would be investigated. Furthermore, a neuromorphic computing model would be implemented to compare its performance to the previously researched models.

7.1 Neuromorphic AI empowered root cause analysis of faults in emerging networks

The goal of this approach is to assess how similar the decision mechanisms of convolutional networks are to the corresponding mechanisms in the human cortex. This offers a tool for neuroscience to understand the dynamic processes of learning and development in the brain and applying brain inspiration to generic cognitive computing.

This concept was developed by Carver Mead [19], in the late 1980s, describing the use of very-large-scale integration (VLSI) systems containing electronic analog circuits to mimic neuro-biological architectures present in the nervous system. In the medium term we may expect neuromorphic technologies to deliver a range of applications more efficiently than conventional computers, for example to deliver speech and image recognition capabilities in smart phones. (Currently such capabilities are available only using powerful cloud resources to implement the recognition algorithms). These will require small-scale neuromorphic accelerators integrated with the application processor, using a fraction of the resources of a single chip. Large-scale systems may be used to find causal relations in complex data from science, finance, business and government. Based on the causal relations detected such neuromorphic systems may be able to make temporal predictions on different time-scales.

7.2 Natural Intelligence vs Artificial Intelligence

Key advantages of neuromorphic computing compared to traditional approaches are energy efficiency, execution speed, adaptability robustness against local failures, the ability to learn and diverse cell types at individual nodes. Talking in terms of image recognition in the human brain, there are two complementary paths of scene perception in humans. First, an object-centered approach, in which components of a scene are segmented and serve as scene descriptors. And second, space-centered approach, in which spatial layout and global properties of the whole image or place act as the scene descriptors. In the proposed solution, the notion of which method of image recognition is utilized by the neuromorphic model would be discussed.

7.3 Nengo

Nengo [20] is a neural simulator based on a framework called the Neural Engineering Framework (NEF). The Neural Engineering Framework (NEF) [21] is one set of theoretical methods that are used in Nengo for constructing neural models. The NEF is based on Eliasmith & Anderson's (2003) book from MIT Press.

Nengo is highly extensible and flexible. You can define your own neuron types and learning rules, get input directly from hardware, build and run deep neural networks. It is a large-scale modeling approach that can leverage single neuron models to build neural networks with demonstrable cognitive abilities.

7.4 Nengo Neural Network [22]

Neurons communicate through unidirectional connections called synapses. When a neuron spikes, it releases neurotransmitter across the synapse, it causes some amount of current to be imparted in the postsynaptic (downstream) neuron. Many factors affect the amplitude of the imparted current; which can be summarized in a scalar connection weight representing the strength of the connection between two neurons. In order to compute any function, the connection weights are set between two populations to be the product of the decoding weights for that function in the first population, the encoding weights for the downstream population, and any linear transform.

The Node represents non-neural information, such as sensory inputs and motor outputs. It can be used to model a complex experimental environment that both provides input to the neural model and responds to the neural model's output.The Connection describes how nodes and ensembles are connected. The Probe gathers data during a simulation for later analysis. The Network encapsulates a functionally related group of interconnected nodes and ensembles.

The Model encapsulates a Nengo model. Several signals are created for each high-level Nengo object; for example, for each ensemble, the simulator creates signals that represent the high-level input signal that will be encoded to input currents, and the encoding weights. The ensemble also contains a neural population, for which the simulator creates signals that represent input currents, bias currents, membrane voltages, and refractory times for each cell.

7.5 Initial Results:

Image Type	Correct predictions	Accuracy
Grey-scale	1694	86.38
RGB	1535	78.27

Table 2: Image Classification Accuracy of a Neuromorphic Model

The classification accuracy on the test data set is represented in Table 2. As discussed in chapter 5, the model tends to classify Grey-scale images better than RGB images.

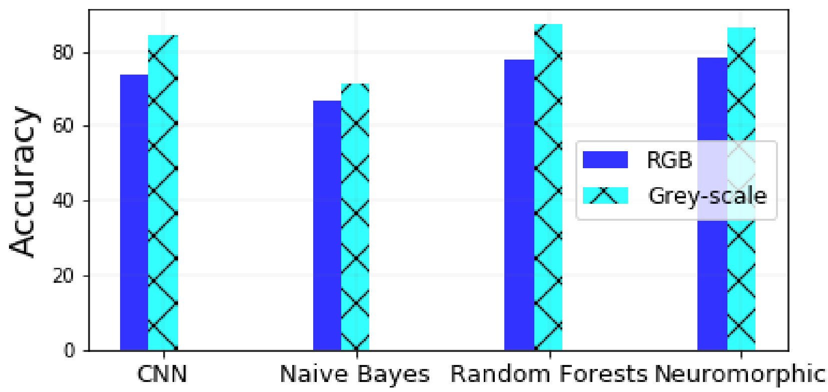


Fig 7.5: Classification Accuracy of Neuromorphic model as compared to previously researched models

Also, the neuromorphic model performs better than CNN, Naïve Bayes and Random Forests models. A representation of the accuracy comparison is shown in fig 7.5.

Bibliography

[1] A. Asghar, H. Farooq, and A. Imran, “Self-Healing in emerging cellular networks: Review, Challenges, and Research directions,” IEEE Communications Surveys Tutorials, vol. 20, no. 3, pp. 1682–1709, third quarter 2018.

[2] A. Imran, A. Zoha, and A. Abu-Dayya, “Challenges in 5G: how to empower SON with big data for enabling 5G,” IEEE Network, vol. 28, no. 6, pp. 27–33, Nov 2014.

[3] R. Barco, L. Nielsen, R. Guerrero, G. Hylander, and S. Patel, “Automated troubleshooting of a mobile communication network using Bayesian networks,” in 4th International Workshop on Mobile and Wireless Communications Network, Sep. 2002, pp. 606–610.

[4] E. J. Khatib, R. Barco, A. G´omez-Andrades, P. Mu˜noz, and I. Serrano, “Data mining for fuzzy diagnosis systems in LTE networks,” Expert Syst. Appl., vol. 42, no. 21, pp. 7549–7559, Nov. 2015. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2015.05.031

[5] R. M. Khanafer, B. Solana, J. Triola, R. Barco, L. Moltsen, Z. Altman, and P. Lazaro, “Automated diagnosis for UMTS networks using Bayesian network approach,” IEEE Transactions on Vehicular Technology, vol. 57, no. 4, pp. 2451–2461, July 2008.

[6] R. Barco, V. Wille, and L. Dez, “System for automated diagnosis in cellular networks based on performance indicators,” European Transactions on Telecommunications, vol. 16, no. 5, pp. 399–409. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/ett.1060

[7] M. Z. Asghar, S. Hmlinen, and T. Ristaniemi, “Self-healing framework for LTE networks,” in 2012 IEEE 17th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Sep. 2012, pp. 159–161.

[8] R. Barco, V. Wille, L. D´ıez, and M. Toril, “Learning of model parameters for fault diagnosis in wireless networks,” Wirel. Netw., vol. 16, no. 1, pp. 255–271,

Jan. 2010. [Online]. Available: http://dx.doi.org/10.1007/s11276-008-0128-z

[9] S. Rezaei, H. Radmanesh, P. Alavizadeh, H. Nikoofar, and F. Lahouti, "Automatic fault detection and diagnosis in cellular networks using operations support systems data," NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium, pp. 468–473, 2016.

[10] P. Szilagyi and S. Novaczki, "An automatic detection and diagnosis framework for mobile communication systems," IEEE Transactions on Network and Service Management, vol. 9, no. 2, pp. 184–197, June 2012.

[11] P. Szilagyi and S. Novaczki, "An automatic detection and diagnosis framework for mobile communication systems," IEEE Transactions on Network and Service Management, vol. 9, no. 2, pp. 184–197, June 2012.

[12] S. Novczki, "An improved anomaly detection and diagnosis framework for mobile network operators," in 2013 9th International Conference on the Design of Reliable Communication Networks (DRCN), March 2013, pp. 234–241.

[13] "Atoll RF design tool. Available at: https://www.forsk.com." [14] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, Nov 1998.

[15] A. Hassan and A. Mahmood, "Convolutional Recurrent Deep Learning model for sentence classification," IEEE Access, vol. 6, pp. 13949– 13957, 2018.

[16] I. Masi, Y. Wu, T. Hassner, and P. Natarajan, "Deep face recognition: A survey," in 2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Oct 2018, pp. 471–478.

[17] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The German traffic sign recognition benchmark: A multi-class classification competition," in The 2011 International Joint Conference on Neural Networks, July 2011, pp. 1453–1460.

[18] B. Abraham, A. Mandya, R. Bapat, F. Alali, D. E. Brown, and M. Veeraraghavan, "A comparison of machine learning approaches to detect Botnet traffic," in 2018 International Joint Conference on Neural Networks (IJCNN), July 2018, pp. 1–8

[19] http://www.carvermead.caltech.edu/index.html

[20] https://www.nengo.ai/

[21] https://www.nengo.ai/nengo/examples/advanced/nef_summary.html

[22] https://www.nengo.ai/nengo/