Using Deep Learning Model RNN to Detect Anomalies in Heartbeat data

Table of Contents

Abstract

The heart is a compound organ and around is several diverse and new kinds of Anomalies that can occur. An electrocardiogram is a simple test that can be used to check your heart's rhythm and electrical activity Electrocardiography (ECG) signals often analyzed manually by a medical professional to detect any arrhythmia that the patient may have suffered by different reasons. In the present age, the most common reason of death is heart disease. Existing systems have used ECG datasets and used CNN But in this study Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) auto encoder was used to detect anomaly in normal heartbeats. RNN model is a deep learning-based model and was used as a classifier to classify heart beat rate like normal, abnormal, and PVC heartbeats. In this work different ECG data sets was collected from different resources and preprocess them. Using this dataset, model was designed that can differentiate between normal and abnormal heartbeats. Also, no preceding information was required about abnormal signals, as required for the other networks as they were trained only on normal data. So, the proposed model was able to predict any heartbeat rates other than normal behavior. This study was help to detect the heart diseases at early stages.

Keyword: ECG, Heartbeat, LSTM, RNN

Chapter: One

Introduction

1.1 Background of study

According to the World Health Organization (WHO), almost 17 million people worldwide die each year from cardiovascular disease, which is linked to quicker or slower heart beats. According to a research released by the United Nations in 2018, the world's ageing population is forecast to grow each day, and by 2030, the number of old persons aged 60 and above would have increased by 56.00 percent, from 901 million to 1.4 billion. Furthermore, by 2050, the expanding number of people aged 60 and up is expected to reach 2.1 billion. The globe faces an economic and health-care dilemma as the world's population ages (NATIONS, 2018).

As a human's cardiovascular system becomes more susceptible to disease and weakens with age, the arteries and muscle wall of such left ventricle thicken and shrink, leading in a decrease in the compliance of blood vessels in the arteries (Rajendra, Lih, & Yuki, 2017). As a result, it's critical to be able to recognize and understand any cardiac disorders based on the shape of ECG signals.

The electrocardiogram (ECG) is a vital screening method in cardiology for diagnosing and analyzing a patient's cardiac state, particularly life-threatening heart illnesses and irregularities like arrhythmias (Ali I; Selen O, 2017). Arrhythmia is a rhythmic irregularity of the heartbeat. It occurs when the heart fails to efficiently pump blood throughout the body (Johns Hopkins Medicine, 2018).

Due to human nature and the muscular anatomy of the heart, cardiologists might occasionally fail to discern between normal and arrhythmic beats, even after years of training. As a result of research and development in this field of biotechnology, it has been discovered that there is a simpler technique to identify arrhythmias (Singha & al, 2018).

Many automated analyses and interpretations of arrhythmias in ECGs have been proposed in recent years, but deep learning, a machine learning subfield that learns using the Deep Neural

Network technique, is progressively being used in this area, and it has demonstrated to be an effective technique for the study of ECG signal data thus far.

For the ECG arrhythmia detection analysis, I was use a deep learning network, 1D - Convolutional neural network, using a dataset picked from Physionet's MIT-BIH arrhythmia database (Physionet, 2005).

Cardiac arrhythmia is a disorder in which the heart beats irregularly. According to the World Health Organization (WHO), around 17 million people die each year from cardiovascular illnesses throughout the world. This accounts for around 31% of all fatalities worldwide. According to the American Heart Association (AHA), cardiovascular illnesses cause one out of every three fatalities in the United States. Cardiovascular disease is responsible for more fatalities than all forms of cancer and chronic lower respiratory disorders combined. According to a 2014 study, atrial fibrillation affects around 2% to 3% of adults in North American and European countries (Gerc, 2020).

A fast heart rate (over 100 beats / minute in adults) is referred to as tachycardia, whereas a slow heart rate (below 60 beats per minute) is referred to as bradycardia. Premature contraction occurs when the heartbeat is too fast. Fibrillation or flutter is the medical term for an irregular heartbeat. Aside from heart rate criterion, there are a variety of other categories for cardiac arrhythmia based on other sorts of criteria. Another categorization is based on the location of the irregular heart rate's onset. The atrioventricular (AV) node is where atrial arrhythmias begin. The AV node is located between the atria and the ventricles (the atria are the two upper compartments of the heart from which blood is transported to the ventricles) (Swapna, 2018).

Atrial arrhythmias include atrial fibrillation (AF), atrial tachycardia, atrial flutter, premature atrial contractions, and sinus bradycardia. Arrhythmias such as atrial fibrillation and atrial flutter are instances of significant arrhythmias. The atrium contracts rapidly and irregularly in AF, with electrical impulses from the heart originating from a separate portion of the atria or neighbouring pulmonary veins rather than the sino-atrial (SA) node (Rajdev, 2012). The atria's walls fibrillate (quiver rapidly) instead of beating normally, preventing the atria from effectively pumping blood into the ventricles. Atrial fibrillation can cause stroke and heart failure, to name a few consequences. AF can be caused by a variety of factors, including excessive blood pressure, an enlarged thyroid gland, and coronary and rheumatic heart disease.

Atrial flutter has many of the same symptoms and problems as atrial fibrillation. In atrial flutter, however, electrical impulses from the heart travel around the atria in a rapid and regular pattern rather than the erratic pattern seen in AF. Ventricular arrhythmias are abnormally fast heartbeats caused by an ectopic ventricular focus. Ventricular arrhythmias include ventricular tachycardia, ventricular fibrillation, and premature ventricular contractions. Some arrhythmias are asymptomatic and have no life-threatening consequences. However, certain asymptomatic arrhythmias can cause major problems such as blood clots, strokes, heart failure, and sudden cardiac death (Calkins, 2007).

Arrhythmias arise when the electrical impulses sent to the heart to regulate heartbeats are disrupted. The examination of an electrocardiogram (ECG) and confirmation that the ECG is not suggestive of cardiac arrhythmia is the first step in diagnosing this issue. The electrocardiogram (ECG) is a bio signal that represents the action of an autonomous nervous system (ANS), which controls heart rhythm. ECG records the electrical activity of the heart in this way. It's a quick and painless way to examine heart rhythms and identify arrhythmias. The ECG signal is created as a result of the mechanisms listed below.

The SA node, located in the right atrium of the heart, generates the heartbeat as an electric pulse. This electric pulse contracts both atria and subsequently stimulates the atrioventricular (AV) node, which electrically links the atria and ventricles. Following that, both ventricles are activated. The ECG waveform represents the whole cardiac activity. Cardiac arrhythmias are indicated by abnormalities with in morphology of ECG waveforms. The ECG waveform is examined to determine the risk of any sort of arrhythmia.

Arrhythmia detection has been the subject of extensive investigation. The works listed below are for a dangerous kind of arrhythmia described as myocardial infarction (MI), sometimes recognized as a heart attack. The accuracy of MI detection was 94.74 percent when data from a single lead ECG was used (Naser Safdarian, 2014). The same goal was achieved with 96 percent accuracy using multiscale Eigen space analysis on 12 lead ECG data (Sharma, 2015). The discrete wavelet transform (DWT) was used to analyses 12 nonlinear characteristics taken from 12 lead ECG data to diagnose MI with an accuracy of 98.8%. (Rajendra Acharya et al., 2016).

Deep learning techniques are currently being used more often in this field. With an accuracy of 95.22 percent, CNN was used to detect normal and MI in an automatic manner (Rajendra Acharya et al., 2017). The identification of inferior MI in ECG using CNN has an accuracy of 84.54 percent (Tahsin Reasat, 2017). With the MIT BIH data set and another dataset as input, four forms of arrhythmia were diagnosed with 99.38 percent accuracy (Abhinav-Vishwa, 2011). The artificial neural network (ANN) was used to classify the MIT Arrhythmia database of ECG into normal and abnormal, with an accuracy of 96.77 percent (Babak Mohammadzadeh-Asl, 2006). Many studies have used ECG as a normal input data to classify particular forms of cardiac arrhythmia. These specific cardiac arrhythmia instances, which have been addressed in the majority of past research, are frequently dangerous arrhythmia kinds such as myocardial infarction. In summary, studies were carried out to categories normal ECG and various forms of arrhythmia-affected ECG.

Cardiac arrhythmia is caused by abnormalities in the heart and is diagnosed by an irregular cardiac rhythm. These abnormalities produce structural variances in the atria and ventricles, resulting in alterations in activation, depolarization, and repolarization. The ECG waveform deviates from its typical shape and size as a result of these modifications. Different forms of cardiac arrhythmia are induced by different reasons, resulting in different abnormalities in the ECG wave shape (Sujadevi, 2017; Rahul et al., 2017).

The goal of this research is analyze the deep learning model RNN to detect anomalies in heartbeat. We use a two-class classification system to determine whether or not a cardiac arrhythmia is present in the ECG signal. In Physionet, we use ECG records from the MIT-BIH arrhythmia database, which is open to the public (Goldberger, 2000). The MIT-BIH arrhythmia database was the first widely available dataset for evaluating the effectiveness of cardiac arrhythmia detection systems. We use CNN, CNN-RNN, CNN-LSTM, and CNN-GRU as deep learning-based analytical tools. Our study may be useful to cardiologists as an automated method for the first screening of persons with cardiac arrhythmia.

1.2 objective of study

- To investigate possible quality issues with real-world ECG data and how to enhance the quality of these signals such that follow-up anomaly identification activities are made easier?
- To analyze the deep learning model RNN to detect anomalies in heartbeat?

1.3 Research Questions

1. What are possible quality issues with real-world ECG data and how to enhance the quality of these signals such that follow-up anomaly identification activities are made easier?
2. How to using deep learning model RNN to detect anomalies in heartbeat?

Chapter: Two

Literature Review

2.1 ECG Monitoring and Its Signals

2.1.1 Standard 12-lead ECG

A conventional 12-lead ECG shows frontal and horizontal images of the heart, as well as perspectives of the left ventricle's surfaces from 12 distinct angles. Six limb leads (I, II, III, aVF, aVL, and aVR) and six chest leads (V1–V6) make up a 12-lead ECG. For chest pain or distress, electrolyte imbalances, electrical injuries, pharmaceutical overdoses, ventricular failure, syncope, stroke, and unstable patients, the conventional 12-lead ECG is employed as a standard clinical dysrhythmia analysis tool. It is commonly used to diagnose cardiac problems in clinics and hospitals (Hampton, 2013). A 12-lead ECG, on the other hand, is impracticable when the patient has to be monitored constantly since the patient must be linked to 10 electrodes.

2.1.2 Three-lead vs. 12-lead ECGs

Because the normal 12-lead ECG is inconvenient for continual ECG recording, three (3)-lead ECGs are commonly employed in portable ECG equipment for a 24-hour recording. Frank's lead system is a three-lead system that may be used in the clinic (Frank, 1956). Furthermore, much research has been conducted to demonstrate that a 3-lead ECG may be used to establish a reliable diagnosis. In an older population, Antonicelli (2012) was able to confirm the accuracy of a 3-lead telecardiology (tele)-ECG vs a 12-lead tele-ECG. Their research found that ECG diagnosis utilizing a basic home telecardiology equipment (3-lead tele-ECG) and more advanced instruments, such as the 12-lead tele-ECG, as well as the traditional 12-lead ECG, had a high degree of concordance. The study also showed that a basic 3-lead tele-ECG may be utilized to identify cardiac disturbances such as atrioventricular blocks, arrhythmias, atrioventricular blocks, and re-polarization abnormalities, with good agreement with the findings of a 12-lead tele-ECG and a regular 12-lead ECG.

Kristensen et al. also compared an affordable portable three-lead ECG monitor (PEM) to a typical 12-lead ECG in detecting individuals with atrial fibrillation (AF) (Kristensen, 2016). When contrasted to a 12-lead ECG, the analysis indicate that identifying AF using PEM recordings had a sensitivity of 86.7 percent and a specificity of 98.7 percent. The misclassification of three PEM recordings, according to cardiologists, was attributable to interpretation problems rather than the PEM recording itself. They stated in their study that PEM devices may be utilized to diagnose AF. Dehnavi et al. looked into 3-lead vectorcardiogram (VCG) data to see if they might identify cardiovascular disorders (Mehri-Dehnavi, 2013). The scientists tested diagnosing ischemia using a VCG algorithm with 3-lead and 12-lead ECGs and found that the results were identical in both scenarios.

Many scientists have also attempted to recreate a 12-lead ECG signal from 3-lead readings. The Dower & Levkov transformations were examined and contrasted between 3-lead VCG and 12-lead ECGs by Piotr Augustyniak (Augustyniak, 2001). The author then compared a real 12-lead ECG to a synthetic 12-lead ECG and a VCG. The synthetic 12-lead ECG was 10.08 percent distorted, and the synthesized vectocardiogram was 6.347 percent distorted, according to the findings. Atoui (2010) proposed a neural network-based methodology for converting a serial 3-lead ECG to a regular 12-lead ECG. As a consequence, when compared to the real 12-lead ECG, the ANN model's produced 12-lead ECG has an average correlation value of 0.93.

Iigueriedo et al. suggested synthesizing a 12-lead ECG with a 3-signal-lead sensor (Figueiredo, 2012). To generate the 12-lead ECG, the authors employed a linear equation to merge the gathered data from a 3-signal-lead sensor. H. Zhu et al. (Zhu, 2018) developed a new, lightweight synthetic approach for reconstructing the typical 12-lead ECG from three leads: I, II, and V2. The adaptive area segmentation-based piece wise linear (APSPL) approach is the name of the suggested method. Adaptive area segmentation, ECG sequence restoration, and linear regression are all part of it.

Furthermore, Nelwan et al. (2004) and Drew et al. (1997; 2002) have done multiple experiments proving that a reduced lead set ECG may be used to recreate a regular 12-lead ECG. I. Tomasic et al. investigated how a regression trees technique may be utilized to convert a 3-lead ECG into a synthetic 12-lead ECG (Tomasic, 2002). Their research showed that the regression trees technique can produce a precise 12-lead reconstruction and as such the reduced ECG lead set has adequate information to diagnose most cardiac abnormalities.

2.1.3 Normal ECG Signals

To spot abnormalities on ECG data, you must first understand how a typical heartbeat appears. A normal rhythm (see Figure 1) is described as an electrical impulse that originates in the sinoatrial (SA) node, travels through the heart muscles, and finally reaches the participant's chest. A normal rhythm is made up of the P wave created by atrial depolarization, the QRS complex produced by ventricular depolarization, and the T wave and U wave formed by ventricular re-polarization in that order. The P wave, and T wave, and QRS complex in a typical ECG signal should be comparable across time at a frequency of 60 to 100 beats per minute. The PR durations in a normal ECG signal should be 0.12–0.2 s, and the QT intervals should be less than half of the RR interval. In addition, the difference between the smallest PP interval/RR interval and the greatest PP interval/RR period in a normal ECG signal might be less than 0.04 s. (see Figure 2).

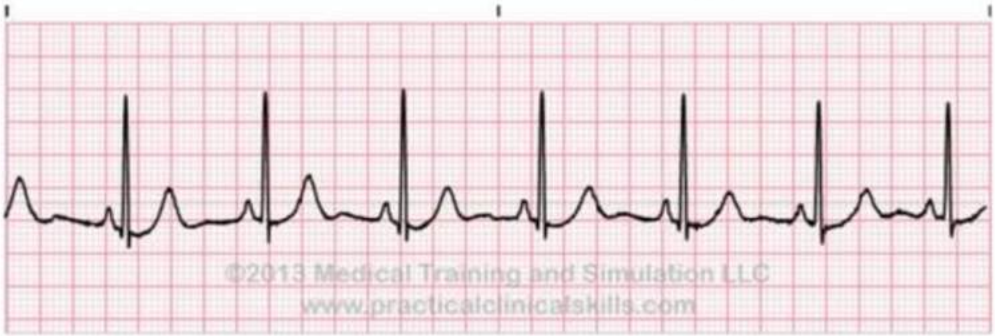

©2013 Medical Training and Simulation LLC
www.practicalclinicalskills.com

Figure 1. Normal sinus rhythm (NSR)

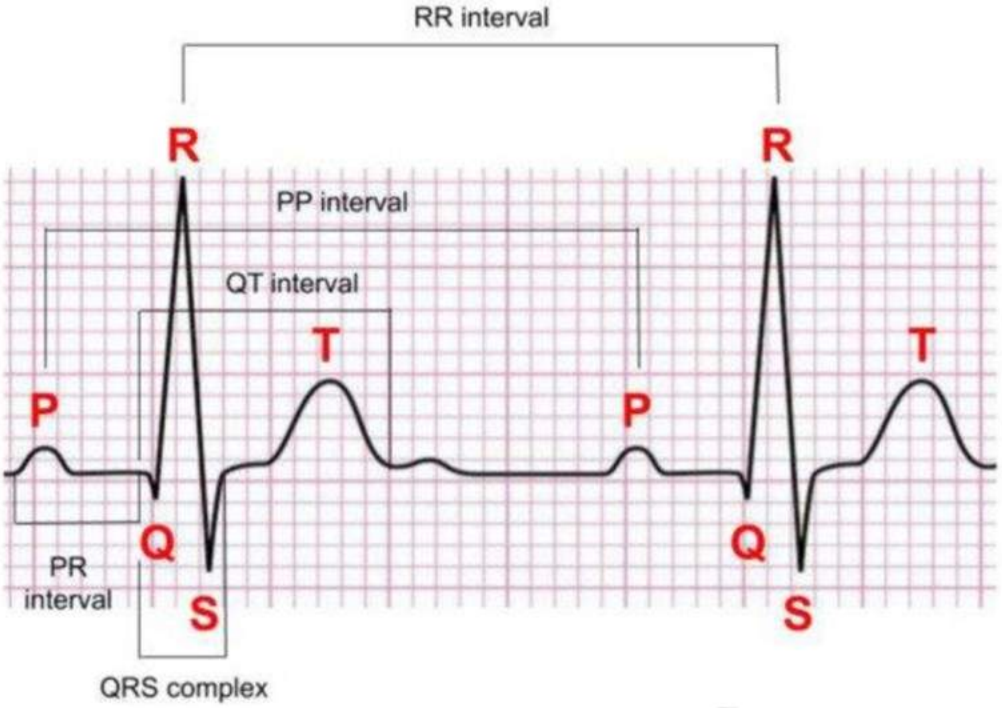
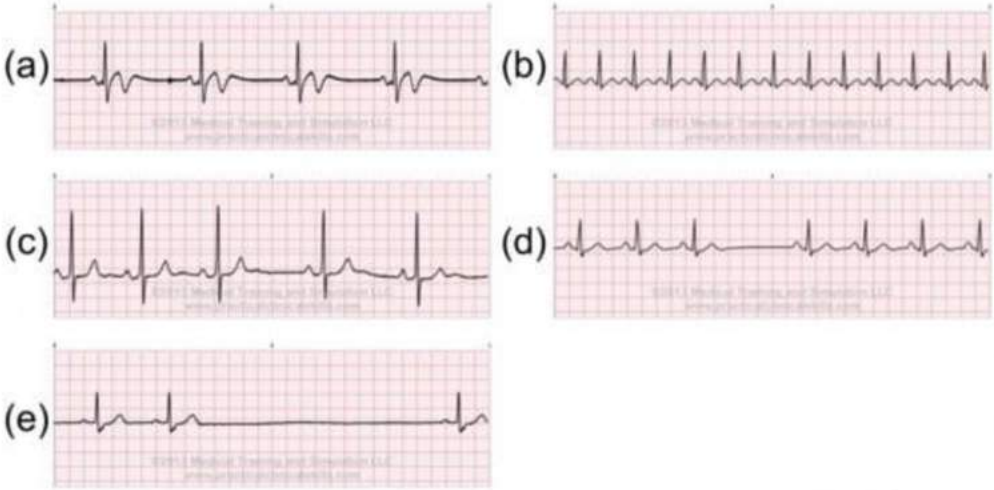


Figure 2. A normal electrocardiogram (ECG) signal and the corresponding notation (Training, 2019).

2.1.4 Abnormal ECG Signals

There are three types of abnormalities in ECG signals: irregular heart rate, irregular rhythm, and ectopic rhythm. The PP/RR intervals on the ECG might be used to calculate the heart rate. A lengthy PP/RR interval suggests a low heart rate, while a short period indicates a rapid heart rate. If the heartbeats begin at the SA node but the PP/RR intervals are greater than 1 second, this might be a sign of sinus bradycardia (Figure 3a), which means the heart is beating too slowly. Sinus tachycardia can be detected when the PP/RR durations are less than 0.6 seconds (Figure 3b). Furthermore, substantial differences between the PP/RR intervals may suggest Sinus Block, Sinus Arrhythmia, or Sinus Arrest (Figure 3c–e).



(i) Sinus bradycardia, (ii) Sinus tachycardia, (iii) Sinus arrhythmia, (iv) Sinus block, (e) Sinus arrest (Training, 2019).

These ECG abnormalities might be a sign of a patient's current health. Hypothyroidism, sick sinus syndrome, hyperkalemia, sleep apnea syndromes, carotid sinus hypersensitivity syndrome, and vasovagal responses are all possible causes of Sinus Bradycardia. Anxiety, excitement, pain, pharmacological responses, fever, pulmonary embolism, congestive heart failure, acute myocardial infarction, hyperthyroidism pheochromocytoma, intravascular volume loss, and alcohol intoxication or withdrawal are all typical causes of sinus tachycardia. Hypoxia, myocardial ischemia and infarction, digitalis poisoning, and a toxic reaction to medications can all produce sinus block and arrest (Goldberger, 2017).

2.1.5 Automatic Heart Anomaly Detection

Finding irregular heart rates, heartbeats, and rhythms is the goal of detecting abnormalities in ECG readings. To do this, an anomaly detection system should be able to identify them across all heartbeat sequences, allowing the needed metrics to be obtained. In addition, the system examines the full recording for any unusual rhythm segments, such as an uneven R-R interval or ectopic rhythms. Noise reduction, heartbeat detection, heartbeat segmentation, heartbeat classification, and rhythm classification are the five subsystems that make up an anomaly detection system.

Figure 9 depicts a typical heartbeat anomaly detection system. The noise reduction procedure aims to reduce the impact of the recording equipment or the patient's movement on signal interpretation. The goal of heartbeat detection is to locate the heartbeats in order to compute the heart rate. Based on a known heartbeat location, the heartbeat segmentation retrieves the whole heartbeat. The heartbeat categorization examines the ECG data for any aberrant pulse form. The irregular heart rhythm classification is related to the heartbeat classification, only it examines a period signal on the ECG record rather than just one heartbeat form. The following subsections introduce relevant research from the literature that relates to the five sub-systems.

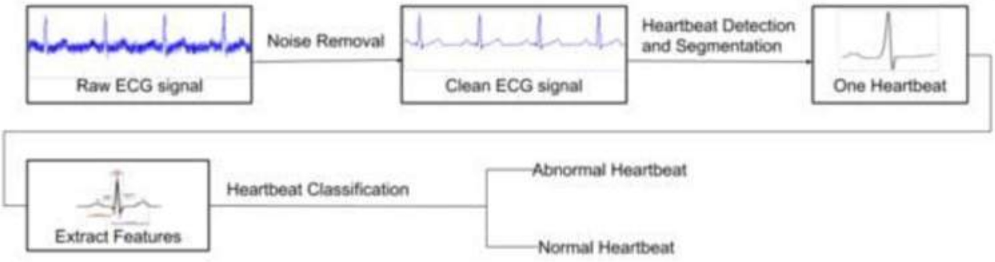


Figure 9. Typical Heartbeat Anomaly Detection.

2.2 Deep Learning Approaches

In a manner that ML-based techniques cannot, DL technology has a remarkable capacity to learn from movies, pictures, and unstructured data. DL-based techniques have a greater capacity to uncover valuable patterns and have a greater potential to generate accurate findings, which is important for applications in medicine and healthcare. Following that, we give a review of the literature on several effective DL frameworks and implementations of finding patterns in ECG modality for cardiovascular disease diagnosis.

Deep learning has achieved amazing results in the field of artificial intelligence during the past several years (AI). The use of deep learning techniques in the healthcare industry is one such element. By lowering costs and improving service efficiency, the introduction of automated health-maintenance technology into medical facilities has changed the industry. The field of cardiac healthcare has been the subject of a significant amount of study.

Eight heartbeats may be identified using the practical LSTM-based method suggested by Gao et al. (2019). For the precise identification of arrhythmias, many condensed LSTM layers were layered; however, the suggested method needed an enormous amount of labelled data. The model also took a while to train and yield notable results because of the heavily nested design. The time-series sequence of ECG data pertaining to normal and irregular heartbeats was categorised by Rana (2019). The focus of the design was on design simplicity while creating a unique LSTM layer-based classifier. The classifier correctly identified five arrhythmias with a 95% accuracy rate.

The classifier's total training time, which plays a crucial role in reducing the approach's overall cost, hasn't been acknowledged or taken into consideration in the research. Multiple deep LSTM models were thoroughly analysed by Hiriyannaiah et al. in order to extract temporal relationships from ECG data. Three stacked LSTM and one stacked BiLSTM model's performance were compared. According to benchmark statistics based on publicly accessible datasets, the bidirectional LSTM-based model outperformed all-LSTM stacked models with a 95% accuracy rate. However, the use of bidirectional LSTM significantly increased the training time per epoch, which raised the computational cost of the method.

The conventional deep-learning algorithms, including CNN, RNN, Auto-Encoders, and Deep Belief Networks, were reviewed by Parvaneh et al. in 2019. The advantages of deep neural networks over simple machine learning methods were stressed in the article. A statistical analysis of the most popular neural network for classifying heartbeat and arrhythmias was also completed. The results showed that most research projects to date have included convolutional neural networks (CNN). The report also emphasised CNN's strengths, which allowed it to be the most frequently employed in order to classify categorization issues.

Five distinct heartbeats were categorised by Acharya et al. (2017) using a densely layered convolutional neural network. In order to address the issue of the original data's class imbalance, synthetic data were produced based on a few resampling approaches. The experimental research showed that the model could reach a classification accuracy of 93.5% without any preprocessing and 94% with noise reduction.

Another method (2020) suggested a deep neural network (DNN)-based architecture to lessen the challenges associated with arrhythmia detection. The strategy combined a rigorous feature-extraction procedure with a learning step. A genetic algorithm was used to aggregate the combined subset of the best attributes. As an analytical module for the detection of abnormalities in certain medical diseases, the framework was created. The technique was successful in classifying five super-classes of arrhythmia with an accuracy of 94.00%.

Using a 12-lead ECG dataset, Chen et al. (2020) categorised nine micro-classes of arrhythmias. The strategy resulted in the formation of a 1D deep convolutional neural network, which also helped it win the China Physiological Signal Challenge (CPSC). The suggested model indicated an average recognition rate of 97% in relation to the clinical observations and the data supplied. Based on the F1 score, which was determined to be 0.84, the benchmarking process was carried out against the referred methodologies.

Wu et al. (2021) constructed a 12-layered deep 1D convolutional neural network for the accurate extraction of five arrhythmia micro-classes in more recent work. The model's classification accuracy was 97.40%, however because of the model's intricate architecture, training took an additional 120 minutes and took an additional 11 hours to complete the ten-fold cross-validation. A few performance criteria, such as accuracy, sensitivity, specificity, etc., were used as a benchmark with the other techniques. The data was improperly preprocessed in order to set up a controlled, supervised learning environment. The only adaptation-related negatives the method shows are longer computing times, which push up the method's overall cost.

2.2.1 CNN

Rahhal et al. (2016) suggested an AL (active learning) model that allows for the selection among the most convenient samples to improve the model. It reduces the training sample size and sends ambiguous data to an expert for categorization, and it uses DNN for ranking. They tested their model against data from three databases: MIT-BIH, INCART, and the SVDB arrhythmia database. They evaluated the arrhythmia groups N, S, V, F, and Q, however raising the number of hidden layers in their suggested model resulted in fewer accurate results, and the AL approach is not regarded fruitful for complex architectures with limited training data.

Serkan et al. (2016) suggested a unique technique based on 1-D CNNs that is an ECG heartbeat classifier for particular patients. Every patient's data was used to train the CNNs. However, the difficulty with this strategy is that the data on which CNNs were trained does not offer promising results for critical anomalies such as S beat, and if it is not included during training, the model would fail to categories them.

Ali Isina and Selen Ozdali (2017) suggested a productive deep learning-based ECG arrangement framework, which is recognized for performing programmed ECG arrhythmia diagnostics by categorizing patient ECGs into 3 separate cardiovascular disorders: normal (N), paced (P), or right bundle branch block (RBBB) (RBBB). After obtaining ECG records from the MIT-BIH arrhythmia information repository online, they are segregated from commotions and QRS waves are detected to isolate the R-T parts of the ECG. AlexNet, which has been pre-programmed, is transferred and used as an element extractor for the ECG grouping operation nearby. The extracted highlights are fed into a simple back spread neural organization, which groups the ECG R-T segments into one of three cardiac states (N, RBBB, and P). Though the suggested framework fulfilled its primary goal of implementing a simple and reliable deep learning approach, it only classifies three classes.

Weifang et al. (2019) introduced a CNN that automates diagnosis using GLCM (gray-level co-occurrence matrix). The amplitude of the ECG signal is mapped into a fixed grey level matrix, and the wave shape change and description are described using GLCM. The categorization obtained by 3D multi-scale GLCM is then automated using the CNN technique. The results demonstrate that this technique can more accurately distinguish discrete morphological arrhythmia groups. Noise from non-stationary objects can be suppressed using the proposed approach. Gray level co-occurrence matrix may readily extract the features vector from different groups of arrhythmias from lead ECGs, resulting in improved diagnostic performance. The issue with GLCM is that the computation method is quite complicated.

Two convolutional neural network models were proposed by Xin et al. (2019). FDRes Net is one among them (the fast down-sampling residual convolutional neural network). MSResNet is the other (a multi-scale decomposition enhanced fast down-sampling residual convolutional neural network). Different sizes of recovered datasets are used to train the rapid down-sampled residual convolutional neural networks. Transfer learning is used to train multi-scale residual convolutional neural networks. Three parallel FDResNets are connected by a tiny neural network to form the multi-scale residual neural network. The three FDResNets have the very same structure and yet are trained on various scales by rebuilt samples.

The dataset they utilized was from the PhysioNet Challenge 2017, and it targeted AF (atrial fibrillation), with FDResNet and Multi FDResNet reaching accuracy of 87.12 percent and 92.1 percent, respectively. The issue with FDResNet is that the loss value is higher in the fast down module, but the multi fast down module, while good, is not stable when FDResNet is increased. The findings get dispersed, making the model unstable and lowering the classification accuracy.

For the identification of fibrillation and flutters, Hamido Fujita and Dalibor Cimr (2019) suggested an 8-layer convolutional neural network technique (A fib, an fl, V fibs).

For model assessments, they used data from the MIT-BIH arrhythmia database. The disadvantage of the proposed method is that too many large batches might result in poor model performance. The normalized signals indicate voltage difference after 4ms during normalization, which might result in the loss of crucial information from the raw input; the suggested model does not have the capacity to recognize artefacts.

Acharya et al. (2017) suggested a model using an eleven-layer CNN and a four-neuron output layer. The normal (Nsr), Afib (Atrial Fibrillation), Afl (Atrial Flutter), and Vfib (Ventricular Fibrillation) ECG classes are represented by the four neurons. For training, a large amount of data (big data) is required, and data training takes a long time. Performance is sluggish as compared to other models.

As in the previous study, Acharya et al. (2018) employed a convolutional neural network to categorise ventricular arrhythmia into shockable and non-shockable categories. After that, Acharya et al. (2017) suggested a CNN model with nine layers to automatically detect five different kinds of heartbeats in ECG data (N, S, V, F, and Q). Three convolution layers, three max-pooling layers, and three fully-connected layers are among the models they've shown. However, the difficulty with their suggested model was that it required a greater number of training hours, specialized hardware is required to successfully train GPUs, and its training is theoretically expensive. More photos are needed during training so that it can successfully distinguish multiple patterns.

2.2.2 DBN

Mathews et al. (2018) presented for the categorization of ventricular and supraventricular heartbeats, and unique strategy based on deep learning methodology has been developed. The use of Restricted Boltzmann Machines as well as deep belief networks to automate the categorization of single-lead ECG readings is being studied. The RBM is a bidirectional connected network of stochastic handling units which learns significant highlights of a hidden probability distribution using instances from that set.

A bipartite chart with a visible layer and a concealed layer can be used to represent an RBM. DBNs are a type of multi-layer generative neural network that are known for its ability to show and envision substantial level learnt highlights. It is composed of layered, computed RBMs, with the lowest level RBM learning a shallow model of the data. Their research focused on five classes, however only the SVEB and VEB classes are classified higher.

2.2.3 LSTM

Yildirim (2018) in his study, he employed the structures DBLSTM-WS and DULSTM-WS to classify ECG signals. Two bi-directional LSTMs, one for forward propagation and the other for backward propagation, were employed in DBLSTM-WS, together with two dense (completely linked) layers. DULSTM-WS, on the other hand, uses a unidirectional LSTM layer using two sense(fully linked) layers to categories five multiple kinds of heartbeats derived from the MIT-BIH arrhythmia database. NSR (normal sinus rhythm), PB (paced beat), VPC (ventricular premature contraction), LBBB (left bundle branch block), and RBBB (right bundle branch block) are the five kinds (right bun dle branch block)

2.2.4 Autoencoders

Ozal et al. (2019) have given a CAE (AE structure with several hidden layers) compression technique instead of regular AES. In the encoder phase, signals are compressed and encoded, whilst in the decoder part, low-feature reconstruction is conducted. A total of 27 layers were employed in the encoding and decoding phases, with 14 layers in the encoding phase and 13 layers in the decoding phase. The signs are reduced to low-dimensional vectors in the encoder part of this model, and the signs are rebuilt in the decoder segment. In the buried levels of the model, the profound learning technique portrays the low and indisputable degrees of indicators.

The original signal is created with little loss, and unlike traditional straight change procedures, a profound pressure approach implies that it can automatically figure out how to use various ECG recordings. The model is evaluated using data from 4800 ECG sections taken from 48 exceptional clinical cases. The proposed model's pressure rate (CR) was 32.25, and the usual PRD esteem was 2.73 percent. Their major focus was on developing a method that could be used/embedded in wearables and other hardware devices, as well as a model that might allow for safe information transfer to far-flung clinics using a low-dimensional structure.

Previously employed methods necessitated the need of a professional to oversee long-term ECG beat recording. To address this issue, Ozal et al. (2019) proposed a unique technique (CAE-LSTM) that compresses signals before classifying them. The CAE (Convolutional Autoencoders) model is used to compress ECG signals with the least amount of data loss, and then the LSTM model is used to detect the compressed signals. The deep CAE model has 16 layers, but the LSTM has just 5. The difficulty with the suggested strategy is that it necessitates a complicated deep learning model for compression, which loses certain characteristics during compression and requires processing on raw data, which takes time.

2.2.5 ANN

Heart RateVariability (HRV) employing ANN was provided by Sarah et al. (2018) as a very encouraging evaluation instrument for wellbeing. HRV refers to the time difference between one heartbeat and the next. HRV estimate is straightforward and unobtrusive, and it is obtained by recording ECG on freely moving people. The main goal of this study is to investigate the aspects of the pulse's autonomic guidance by employing recurrence and fleeting evaluation to correlate the HRV with these physiological patterns.

The framework is divided into four stages: data collection, preprocessing, highlight extraction, and characterization. Preprocessing the HRV data was a necessary step in order to be able to eliminate certain characteristics. The RR span tachogram was used to calculate the time and recurrence region proportions of HRV. In this study, many neural network architectures were implemented and studied. Then, with an accuracy of 88.7%, the detection of rest/alert states is completed, followed by a multi-classification of various types of activities such as resting, strolling, practicing, and eating.

For ECG categorization of normal and pathological beats, Sannino and De Pietro (2018) suggested a deep learning technique. They used data from the MITBIH Arrhythmia Database for the experiment. After that, they used experiments to build DNN. They manually constructed deep neural networks in each experiment by adjusting factors such as the number of hidden layers and the number of neurons in each layer. For ECG categorization of normal and pathological beats, Sannino and De Pietro (2018) suggested a deep learning technique. They used data from the MITBIH Arrhythmia Database for the experiment. After that, they used experiments to build DNN. They individually constructed deep neural networks in each experiment by adjusting factors such as the number of hidden layers and the number of neurons in each layer.

2.2.6 RBFN

Deng et al. (2018) suggested a dynamical neural learning method for human identification and categorization of cardiovascular diseases. The suggested approach consists of two stages: the first is preparation, and the second is testing. Using spiral premise work (RBF) neural networks and deterministic learning, cardiac components within ECG data are separated (approximated) exactly in the first step. The pieces of the cardiac framework that have been obtained are addressed and stored in RBF networks that are stable. The excised cardiac components are subsequently marked with an ECG mark along the intermittent ECG state directions. To handle the established stride designs, a bank of assessors is created using the extricated heart pieces. In the test stage, acknowledgement errors are made and used as a resemblance measure by comparing the cardiovascular aspects of the generated ECG designs to the cardiovascular elements of the test ECG designs.

2.3 Machine Learning approaches

AI plays a significant role in clinical prediction and categorization in the medical field. It provides the doctors with significant help in dealing with a vast amount of collected clinical data.

These methods can help identify diseases more quickly, which can reduce the need for expensive clinical trials and medication (Canlas, 2009).

The implementation of many machine-learning algorithms, such as naive Bayes, random forests, SVM, etc., was demonstrated by Gupta et al. in 2022. It was also suggested to create the learner module using a linear SVM and Random Forest. The model has a classification accuracy of 77.4% when evaluated on the openly accessible MIT-BIH datasets. In addition to somewhat improved classification outcomes, the suggested model also required less training time.

Six feature-extraction algorithms were put into practise by Luz et al. (2013) for the comparative examination of various performance measures. The accuracy of the developed optimum path forest classifier (OPF) was comparable to that of a customised SVM, while the total model training time was significantly shorter. For the purpose of identifying and detecting patterns, Sarfraz et al. (2014) used the Independent Component Correlation (ICC) method with the general ECG functions. Training and testing tests were created using the characteristics that the ICCA retrieved. The following approach demonstrated excellent accuracy and precision results; but, because human feature recognition and extraction were required, it was expensive and challenging to apply in practise.

Gradient boosting and SVM were used by Batra and Jawa (1975) for the effective identification of arrhythmia from ECGs. The suggested method was compared to existing machine learning techniques including gradient boosting, decision trees, random forests, etc. The raw data has undergone substantial processing and feature selection procedures before to the final training of the models. An overall recognition rate of 84.82% was attained by the model.

A unique feature selection method based on the best-first selection principle was created by Namrata and Pradeep in 2019. The MIT-BIH dataset's subset of ideal features was removed using the three-filter feature selection (TFFS) method. Three classifiers—JRip, SVM, and random forest—were fed the final input. With a classification accuracy of 85.58%, the comparison findings showed that random forest performed better than other methods.

For the categorization of heartbeats, Miquel et al. (2019) presented an Echo State Network (ESN) classifier. The suggested methodology was able to deliver precise findings while requiring just one ECG lead. Additionally, a multi-ensemble combination that took use of parallelism for faster training was presented. On two publicly accessible datasets, the suggested method was evaluated, and lead II showed the greatest accuracy of 98.6%. The total methodology included a significant amount of preprocessing and data cleaning, which eventually raised the cost of the suggested strategy. As a result, this section comes to the conclusion that while machine learning has paved the way for significant advancement in the field of medicine and health automation systems, traditional machine-learning algorithms still struggle with issues like overfitting, the curse of dimensionality, and manual feature recognition. In order to do this, cutting-edge tendencies have switched in favour of automating feature recognition using deep neural networks.

2.4 Neural Networks

For many decades, humanity has been faced with difficult tasks in their daily lives, particularly when they are repetitive or require needed to transform within chaos, leading to the development of AI, a solution that relies on computers to perform and solve activities that are difficult for humans but relatively simple for computers. On the other hand, simulating problem solving in basic activities that appear natural and effortless, such as facial recognition or calligraphy using machine learning, is a more difficult undertaking (Goodfellow, 2016).

With the advancement of technology, we now have the ability to analyses patterns in our daily lives using algorithms that can replicate human capabilities, allowing us to accomplish greater outcomes in less time. The ability of the human brain to calculate data in a nonlinear, complex, and parallel manner through neurons that link and work together in order to traverse the world surrounding, practically automatically, via perception and pattern recognition, prompted the development of ANN (Haykin, 1999).

The capacity to accomplish these cognitive processes is based on brain plasticity, which permits neurons to change and evolve in response to a person's experiences, allowing them to respond quickly to new inputs. NNs, on the other hand, may be seen of as computers that, via learning and generalization, imitate how the brain executes a desired task. The learning process relates to the training phase of a neural network (NN) in which raw input data is used to alter the network's synaptic weights in order to achieve a certain objective. The model may then be used by a functional network to generate generalizations, which can be explained as legitimate outputs for additional inputs. Machine learning is now widely used for a range of bio signal processing and classification tasks, since it achieves excellent results in complicated applications, even though the signals contain a significant amount of noise (Enderle, 2000).

2.4.1 Structure

ANNs are based on a simplified form of the biological neural networks found in the brain, which is made up of a network of neurons. A processing unit that performs simple functions makes up an artificial neuron. It comprises links that are distinguished by their weight, which denotes the connection's strength. In order to conduct a specific action, a given connection j, with weight wkj, that is coupled to neuron k, has an accompanying input signal, xj, that is weighted and contributed to the other existing input signals. This produces an output signal yk that is transmitted to the next neuron and is limited in amplitude by the activation function. Figure 2.11 shows a single neuron diagram, which includes the bias bk, an external independent neuron that tries to control the net input of the previously indicated activation function via the polarity of its output unit signal (Eluyode, 2013).

The linked neurons are structured in layers, with input, output, and one or more hidden layers between them. DNNs are generated as there is more than one hidden layer, since a larger number of layers provide greater memory to the DNN, allowing it to deal with more complicated data. Figure 2.12 illustrates the structural differences between a basic neural network and a DNN (Savalia et al., 2018).

2.4.2 Recurrent Neural Networks

Recurrent Neural Networks (RNN) are a sort of neural network that may be characterized as a dynamic sequential data processing model in which the current internal state is influenced by the preceding one, allowing for a memory-based method to learning and adapting through data. This recursive technique may be represented by unfolding it, resulting in an extremely deep neural network (Figure 2.13) (Pascanu, 2013).

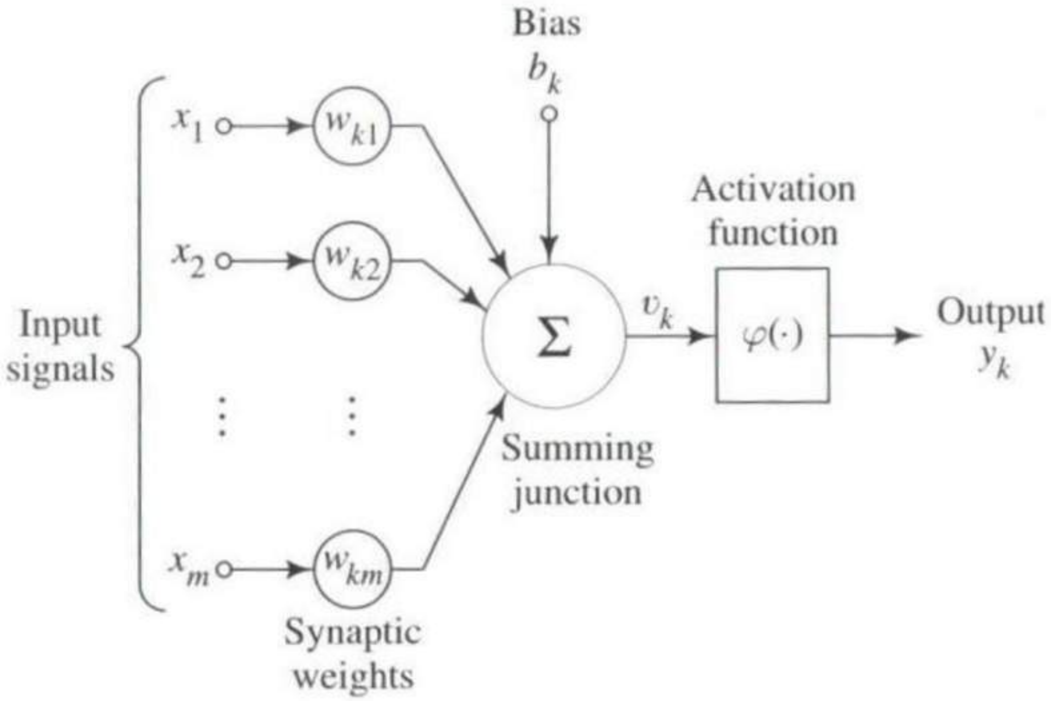


Figure 2.11: Schematic of a single neuron (Haykin, 1999).

Each iteration, the internal memory component contributes by changing the weights related with the connections, resulting in the desired output, which is subsequently fed into the network's input in the following phase. RNNs are commonly used in speech recognition, picture identification, machine translation, and spelling correction, feature definition, and text creation, and other pattern recognition-based applications today (Ghimes et al., 2018).

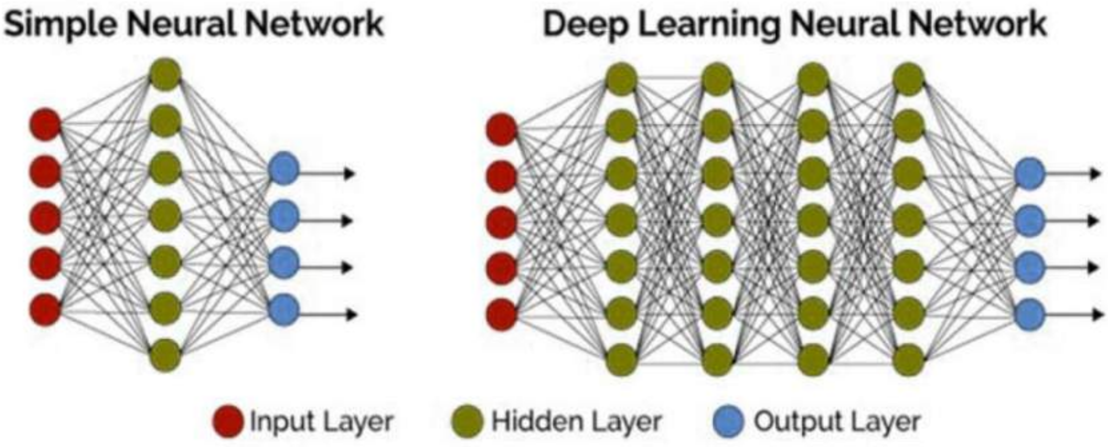


Figure 2.12: Comparison between a simple neural network and a DNN; both possess input and output layers, differing only in the number of hidden layers between them (Savalia, 2018).

2.4.3 Convolutional Neural Networks

The Convolution Neural Networks (CNN) model (Figure 2.14), developed by Fukushima (1980) and revised by LeCun et al (1998), is based on the brain's visual cortex neurobiology, and it connects convolutional layers to learn and organize itself by generating a mixture of feature maps which provide understanding to the pattern recognition at hand (Mahmud, 2018; Mao et al., 2017).

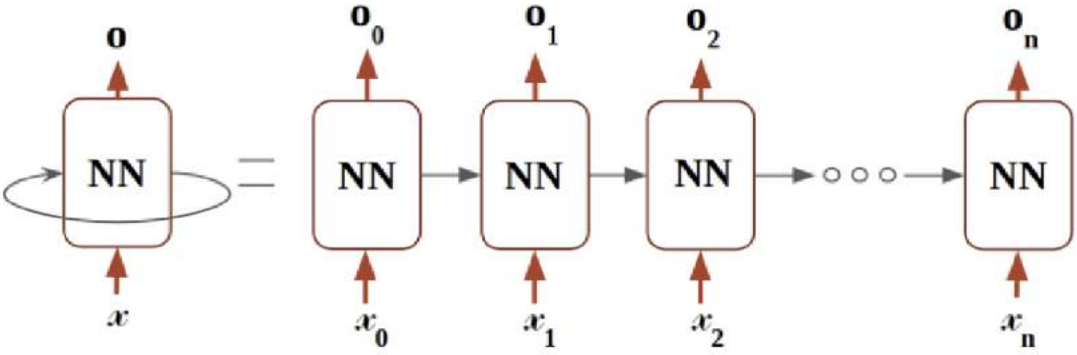


Figure 2.13: Unfolding of a recursive neural network

The convolutional layers use convolutional filters to build feature maps of the data, which are then aggregated and sent across the network to the pooling layers. The latter then applies statistical techniques on the feature maps in order to extract the most important characteristics (Cao et al., 2018). Finally, completely linked layers are those in which each neuron from the preceding layer is linked to the current layer, resulting in the number of classes being determined.

This sort of model has a wide range of applications in the real world, including object identification, picture classification, and handwriting classification, as well as diagnostic applications inside the medical industry (Acharya, 2017). Its primary benefit is that it demonstrates to be beneficial in data sets with a high number of nodes and parameters to be taught.

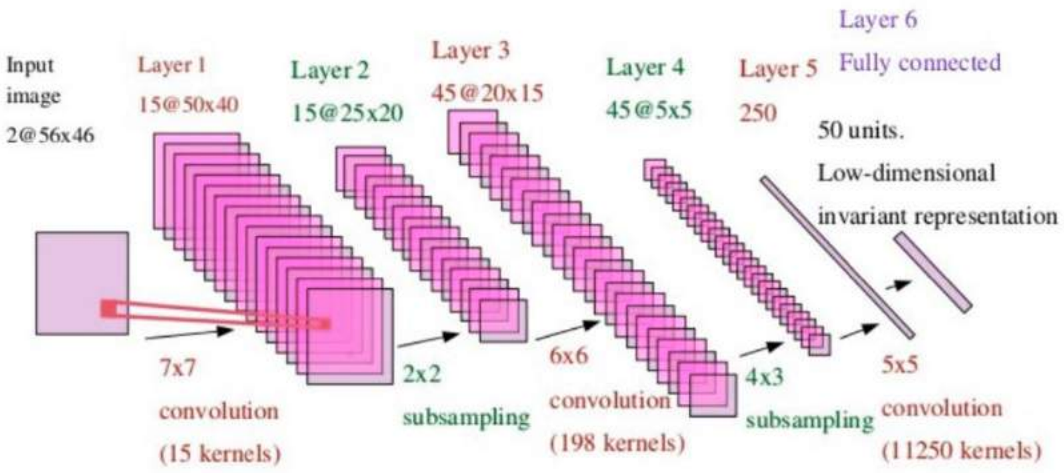


Figure 2.14: LeCun Net (LeCun, 1998).

Chapter: Three

Methods and Datasets

3.1 The Anomaly Detection Method

Two primary processes make up the project's anomaly detection method. In order to understand typical time series patterns and forecast future time series, a summary prediction model is first created. The next step in anomaly identification is to compute anomaly scores using the prediction errors.

3.1.1 Time Series Prediction Model

As a time series prediction model, we use LSTM RNN. The model generates q future values from the most recent p values as input. Parameters p and q are referred to as look back and look ahead, respectively. A hidden layer or layers, followed by an output layer, make up the network. Each dataset has a different number of hidden recurrent layers and units in each layer. The connections between two successive recurrent layers are complete. Dropout is utilized between two successive layers to prevent over fitting. A thick, fully linked NN layer serves as the output layer. One neuron is present in the output layer for each projected future value, which is equivalent to the look ahead value. We utilize linear activation in the output layer and MSE as the loss function since the model is used for regression. The prediction model is trained only on typical data devoid of any abnormalities so that it may understand how time series typically behave.

Trying to Anticipate Numerous Time Steps Ahead: We try to predict multiple time steps ahead (Malhotra, 2015). The model forecasts the following q values of the time series, i.e., t+1, t+2,..., t+q, with a look ahead of q at time t. There are two reasons why several time steps are predicted. As forecasting multiple future values is a more challenging task than one-step-ahead prediction, the first goal of this demonstration is to highlight LSTM's capacity as time series modellers. Second, forecasting several time steps offers a sneak preview of the behaviour to come. Even an early warning of an abnormality could be achievable. Think of a time series with a 5 minute scale. We can get a sense of how the time series washave over the following 30 minutes by projecting six time steps into the future. Early warnings may be provided if there is anything uncommon, such an extreme value. However, anomaly identification is only possible after the actual input value is made accessible. Prediction accuracy suffers when making numerous time step predictions. We only use a look ahead bigger than 1 in our experiments if the forecast accuracy is still respectable. The precise look ahead value utilized is selected at random.

3.1.2 Anomaly Detection

Prediction mistakes are used as anomaly indicators during anomaly detection. The difference between the input value received at time t and the prediction made at time t 1 is the prediction error. A Gaussian distribution is used to describe the prediction errors from training data. Using maximum likelihood estimation, the Gaussian's mean and variance are calculated (MLE). On fresh data, anomaly scores are derived from the log probability densities (PDs) of mistakes, with lower values suggesting a higher possibility that the observation is an anomaly. To define a threshold on log PD values that can distinguish anomalies from normal observations and produce the fewest false positives as feasible, a validation set made up of both normal data and anomalies is utilized. The model is assessed using a distinct test set.

3.1.3 Assumptions

The main assumption we make is that a prediction model trained on normal data should detect typical time series patterns. Areas with anomalies should have bigger prediction errors than regions without anomalies when the model is used to generate predictions on new data. We could then use the log PD values of errors as anomaly scores and set a threshold to differentiate abnormal data items from ordinary ones. Additionally, it is believed that the prediction errors follow a Gaussian distribution.

3.1.4 Algorithm Steps

Only normal data is used to train the LSTM RNN, which is then tuned for prediction accuracy. Each dataset is split into four subsets for this purpose: a training set (N), which contains only normal values; a validation set (VN), which contains only normal values; a second validation set (VA), which contains both normal values and anomalies; and a test set (T), which contains both normal values and anomalies. The algorithm works in the following way:

1. The prediction model is trained using Set N. To determine the ideal values for the following hyper-parameters: lookback, dropout, learning rate, and the network design, we used Bayesian

optimization (Snoek, 2012). (Number of hidden layers and units in each layer) Only in cases when the forecast accuracy is still acceptable do we employ a look ahead greater than 1. Lookahead may be set to 1 if just one time step has to be predicted and the greatest prediction accuracy is needed.

2. Early stopping using VN is used to stop the model from over fitting the training set of data.

3. A Gaussian distribution is used to describe prediction errors on N. MLE is used to estimate the distribution's mean and variance.

4. VA is subjected to the learned prediction model. The log PDs of the errors from VA are computed using the distribution parameters determined in the preceding phase. The log PD values are adjusted to a threshold such that the anomalies may be distinguished with the fewest number of false alarms as feasible.

5. The prediction errors from the test set T are used to assess the set threshold.

The experiments themselves needed a change. On set N, we train the model and improve prediction accuracy as per step 1 above. Assuming the model learns the typical time series behaviour, sets VA and T should have greater prediction errors, enabling us to perform anomaly detection. The optimal parameters for prediction, as mentioned in section 4.4.2, did not always provide accurate anomaly detection findings. In these situations, we started with the optimization phase's settings. Up until we were able to identify every anomaly in the set VA, the model was manually adjusted by repeating steps 1 through 4.

The GPyOpt1 package was used to do Bayesian optimization. For each hyper-parameter, we had to provide possible values as part of the optimization process. Learning rate, dropout, and other variables were given acceptable values in accordance with literature recommendations. We offered three arbitrary choices for the network design.

3.2 Keras

To develop and carry out the project's trials, we employed Keras and the Python programming language. Keras is an open source project that sits on top of existing deep learning libraries like

Tensor Flow and Theano2 and offers a high level API to create DNNs. We built on top of Tensor Flow using Keras. Since Keras has a straightforward API and is intended for quick development and experimentation, it was selected. By integrating various layers, activation functions, loss functions, optimizers, etc., one may modularly configure NNs. For the majority of the common deep learning building pieces, Keras offers ready-to-use solutions. However, Keras API could be somewhat constrained if someone wants to create a unique or customized implementation. Tensor Flow libraries might be a better option. As stated in, Keras includes an implementation of the LSTM with forget gates. To comprehend how LSTMs implemented in Keras work, it is essential to grasp two key points, which are described below.

3.2.1 BPTT Implementation

A modified version of BPTT is used in Keras. It is computationally inefficient to unroll RNN throughout the whole input sequence, which consists of hundreds or even thousands of time steps. As a result, up to a certain amount of time steps are unfurled in Keras RNN. The input method is used to supply this parameter. The input data is provided as a three-dimensional array with the following dimensions: batch size, lookback, and input dimension. The RNN is unfurled for the number of time steps that are specified by the second input, lookback. Sequences with an overlap and a time interval of one are created from the input data. Each sequence comprises one training sample for the RNN model and has a lookback number of consecutive time steps. Only individual samples are used for BPTT during training for lookback time steps. 3.2.2 State Preservation the LSTM state may be preserved in two distinct methods using Keras.

1. Default Mode: State is only kept throughout individual input sequences for lookback number of time steps and each sample in a batch is presumed to be independent.

2. Stateful Mode: In this setting, the multiple training batches all keep cell state. Ith sample of the current batch's end state serves as ith sample of the next batch's beginning state. Individual samples remain independent even within a batch. The training samples should never be shuffled while utilizing stateful mode. One should be cautious not to shuffle samples because a one-to-one mapping between samples of successive batches is anticipated to retain state across batches.

It may appear odd that different samples in a batch are independent of one another. The challenges of language modelling and voice recognition, which were major factors in the development and use of LSTM, served as the inspiration for this implementation. A brief lookback value equal to the maximum sentence length (in words) is sufficient to capture the essential sequential dependencies in many language modelling applications where the training data are individual phrases. As a result, various samples may be handled separately. However, this behaviour may be highly limiting for many domains and datasets.

3.3 Datasets

Lack of labelled benchmark datasets is a significant issue in anomaly detection research, as described in [2]. Many published studies either construct artificial datasets or exploit application-specific datasets [38]. Both strategies, however, have their drawbacks. It is challenging to determine how effectively an anomaly detection method was generalize to diverse datasets using an application-specific dataset. The performance of the method and the anomalies in synthetic datasets are not valid in the actual world. We selected real-world datasets from several areas to protect against these problems. These datasets have been used in earlier anomaly detection studies.

3.3.1 Numenta Machine Temperature Dataset

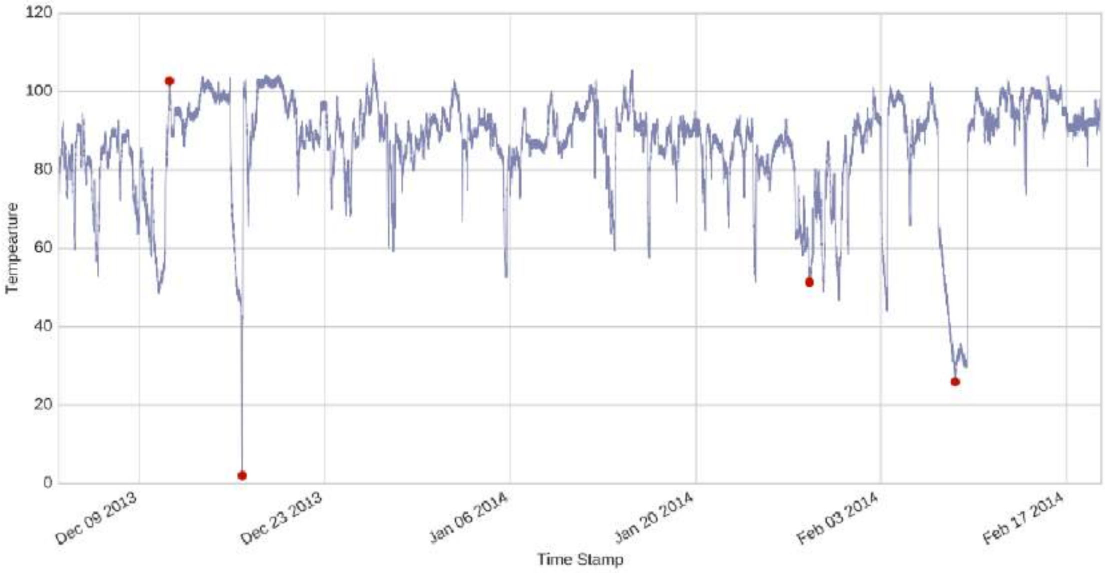
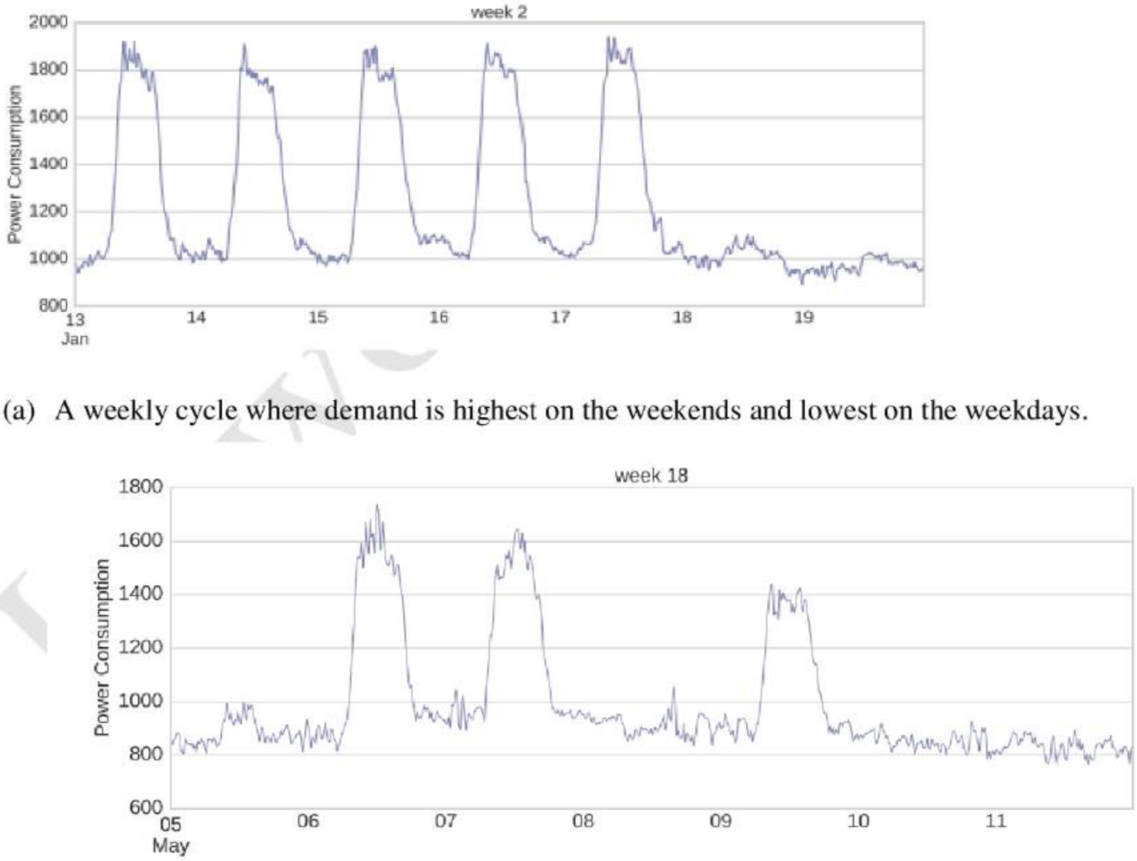


Figure 3.1: Machine temperature data from Numenta. Every five minutes, temperature values are included in this report. Red markers identify four recognised abnormalities. The Y-axis represents temperature, while the X-axis represents time increments.

This dataset, which can be found in Numenta's GitHub repository3, is extracted from (Lavin, 2015). The collection includes temperature sensor measurements from a major industrial machine's interior component. The measurements cover the time frame from December 2, 2013, to February 19, 2014. Every five minutes, 22695 readings are obtained in total. Four of the abnormalities have recognised root causes. Figure 3.1 displays the data, with abnormalities highlighted in red. A scheduled shutdown constitutes the first anomaly, while a catastrophic failure is the fourth. The other two abnormalities cannot be seen with the naked eye.

3.3.2 Power Demand Dataset



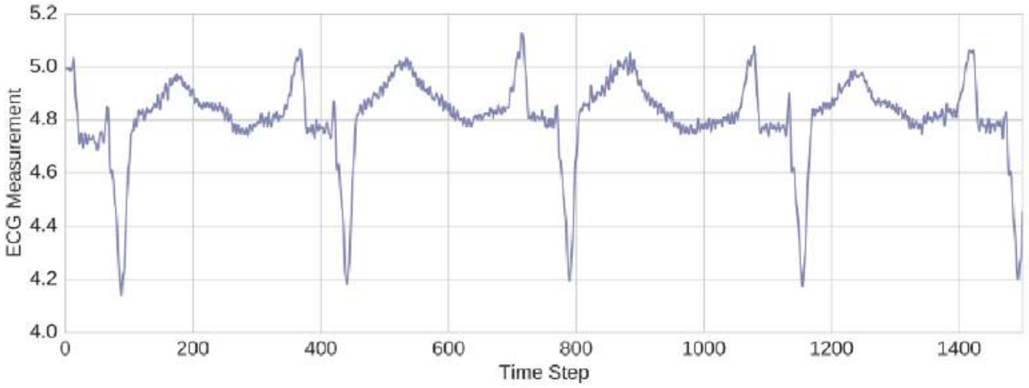
(a) A weekly cycle where demand is highest on the weekends and lowest on the weekdays.

(b) An unusual week with low demand between Mondays through Thursday.

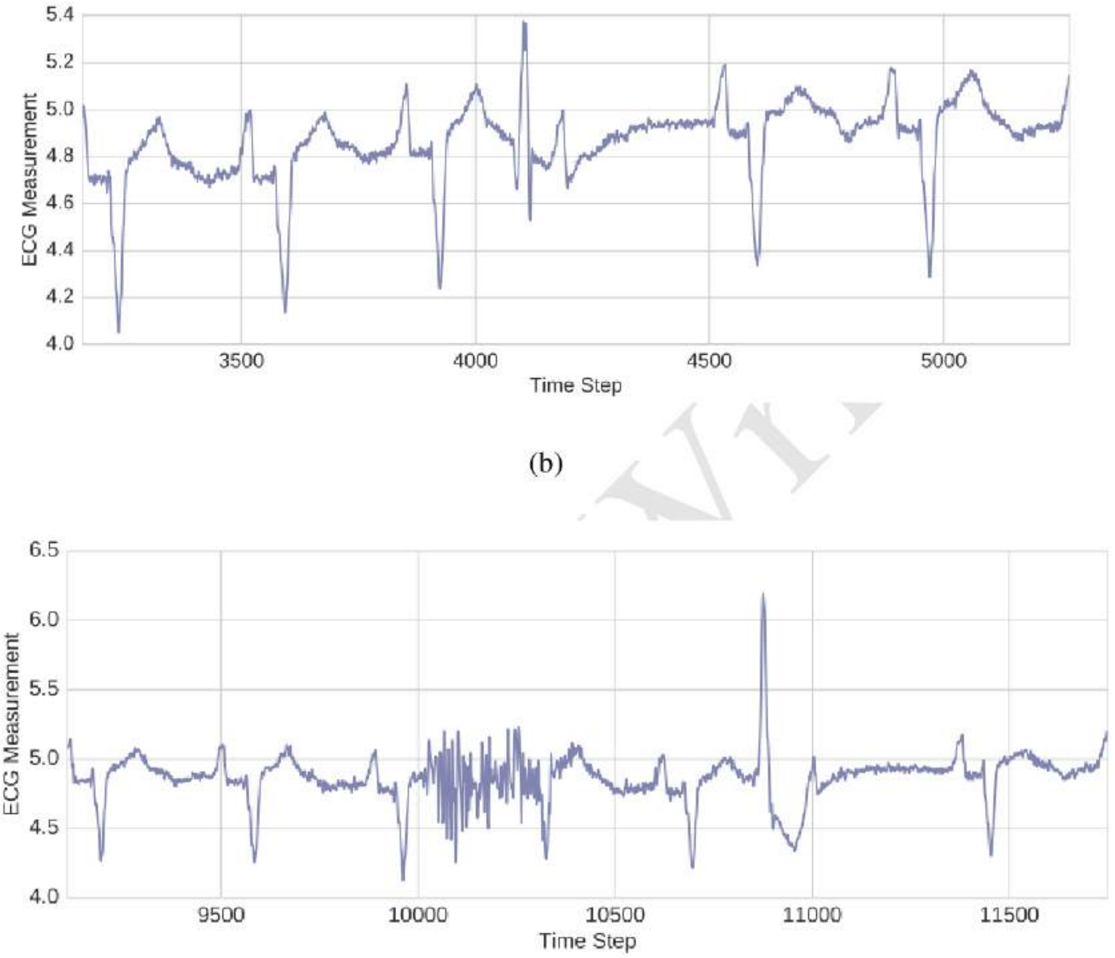
The second dataset shows the energy use of a Dutch research institution in 1997. Every 15 minutes, readings have been collected, totaling 35040 observations. The data comprises a lengthy weekly cycle of 672 time steps, five peaks, and two troughs, which correlate to high power usage during the weekdays and low power consumption over the weekend. The dataset was previously utilized in (Keogh, 2005), where low power demand weeks were regarded as anomalies. Holidays fell on these weekdays. The same may be said for weekends with a strong demand for electricity. We take the same method. In figure 3.2, examples of typical and unusual days are shown. The dataset is available for download at the companion webpage4.

3.3.3 ECG Dataset

Time series of the heart's electrical activity are captured by ECGs. ECG datasets from PhysioBank's archive 5 have been utilized for anomaly identification in a number of publications, including (Jones, 2014; Chuah, 2007). We selected the dataset from (Keogh, 2005) for this research since it offers tagged abnormalities with cardiologist-annotated annotations. In figure 3.3a, a sample of a dataset with typical patterns is shown. There are 18000 measurements in total with various oddities. Figures 3.3b and 3.3c display the three labelled anomalies that were determined to be the most uncommon sequences in (Keogh, 2005). Although there is a recurrent pattern in this dataset, the pattern's duration changes.



(a)



(b)

(c)

Figure 3.3: The ECG Dataset. A displays a typical heartbeat rhythm. (B) Displays the first abnormality. The other two anomalies are shown in (c). Time steps are shown on the X-axis, while ECG readings are displayed on the Y-axis. These graphics display portions of the information to make it simple to distinguish between regular and abnormal heartbeats.

Chapter Four

Experiments and Results

4.1 Main Results

The key findings of anomaly detection for each dataset are reported in this section. Table 4.1 provides an overview of the prediction models applied to various datasets.

4.1.1 Data Pre-processing

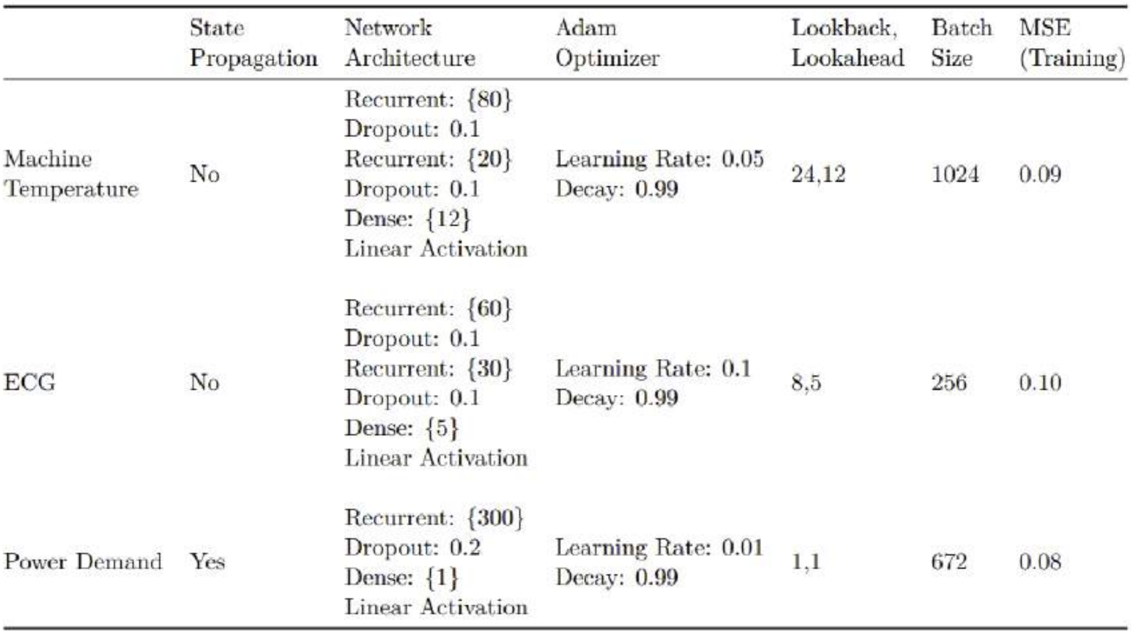
The anomalies were split into sets VA and T for each dataset. Then, regular data was added to these sets. Datasets with repeated cycles were partitioned in a manner that preserved the cycles. Sets N and VN were created from the remaining data. Additionally, we standardized the data to have a mean of 0 and a variance of 1. Other sets were normalized using the training set's mean and standard deviation. Each set was then modified to take on the format needed by the algorithm. Consequently, each input sample was made up of lookback time steps.

4.1.2 Numenta Machine Temperature Dataset

We evenly split the four anomalies into sets VA and T, each of which included around 20% of the data. The prediction model was trained using the remaining data.

Model Information The used LSTM RNN featured a dropout of 0.1, a lookback of 24, a look ahead of 12, two hidden recurrent layers with 80 and 20 LSTM units each, and 12 neurons in its dense output layer. With an Adam optimizer, we trained the prediction model with a learning rate of.05, decay of 0.99, and batch size of 1024. With early quitting, 200 epochs of training were completed. We did not keep the LSTM state across batches since the data does not include any recurrent patterns. On N, this model produced an MSE of 0.09. A threshold of 11 was imposed on the log PD values using the predetermined VA. Then, set T was used to assess the threshold.

Evaluation: In figures 4.1 and 4.2, respectively, the results of the anomaly detection on sets VA and T are shown. It took a threshold of 11 to find the Table 4.1: specifics of each dataset's prediction models. The kind of layers, the number of units in each layer, and the dropout values are all specified by network architecture. The output layer has been activated linearly. State propagation is the preservation of LSTM state between batches.



	State Propagation	Network Architecture	Adam Optimizer	Lookback, Lookahead	Batch Size	MSE (Training)

Machine Temperature	No	Recurrent: {80}
		Dropout: 0.1
		Recurrent: {20}
		Dropout: 0.1
		Dense: {12}
		Linear Activation
			Learning Rate: 0.05
			Decay: 0.99
				24,12	1024	0.09

ECG	No	Recurrent: {60}
		Dropout: 0.1
		Recurrent: {30}
		Dropout: 0.1
		Dense: {5}
		Linear Activation
			Learning Rate: 0.1
			Decay: 0.99
				8,5	256	0.10

Power Demand	Yes	Recurrent: {300}
		Dropout: 0.2
		Dense: {1}
		Linear Activation
			Learning Rate: 0.01
			Decay: 0.99
				1,1	672	0.08

We contrast the outcomes of the LSTM technique with those obtained by [39] using an anomaly detection method based on Hierarchical Temporal Memory (HTM) [43]. The central point of each fixed-size anomaly window described by the authors is an anomaly. A genuine positive is the first detection within a window, whereas a false positive is any detection outside the window. Early detection within a window is given a greater weight via the use of a weighted scoring procedure. Figure 4.3 displays the HTM algorithm's findings. HTM has five false positives and can identify three of every four abnormalities. Figure 4.3c's initial anomaly window shows how HTM makes the discovery far in advance of the actual anomaly. The false positive detections produced by the LSTM algorithm in figure 4.2 are covered by this anomaly window. In actuality, the majority of the LSTM algorithm's false positives correspond to the HTM anomaly windows. As a result, a window-based detection method may enhance the LSTM algorithm's performance and accurately identify all abnormalities. We didn't create such a detection mechanism, however, since the use case and goal should determine the specific detection method. We employed 70% of the data for training, but the HTM method has a clear benefit of employing a limited training window (purple shaded part in figure 4.3a).

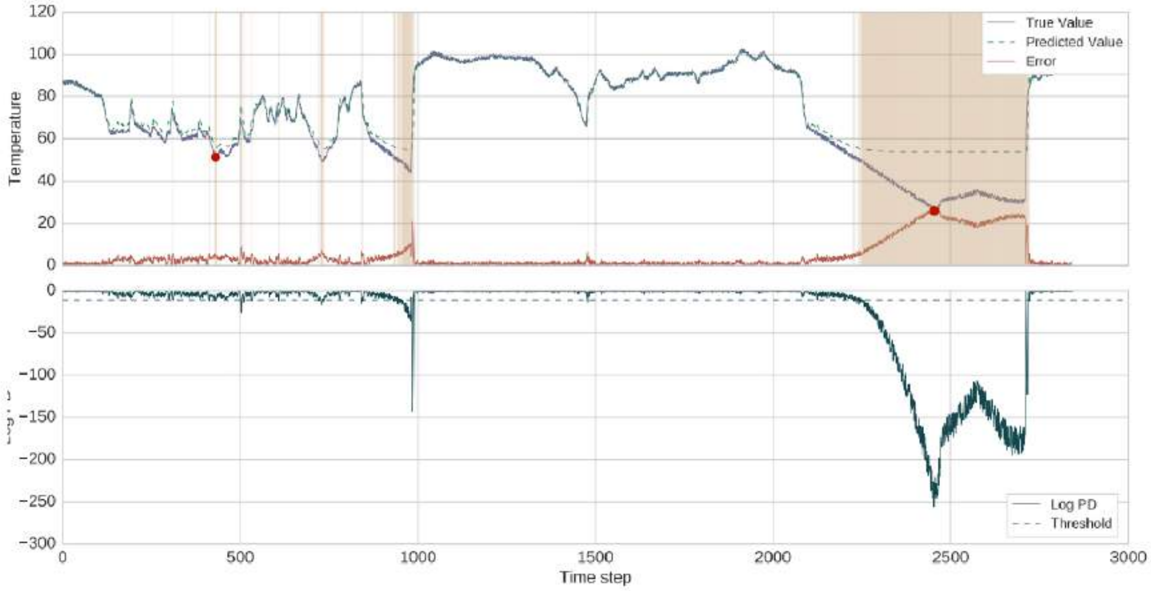


Figure 4.1: Results from the machine temperature dataset's validation set. The predictions made by VA are shown in the top plot along with the accompanying prediction errors. The threshold is set at 11, and the bottom plot displays the log PD values of the prediction errors. The Y-axis displays the matching metric value, while the X-axis displays time steps. Red markers indicate the locations of the two real anomalies. The top graph's shaded region represents detections performed by the LSTM algorithm.

4.1.3 Power Demand Dataset

The dataset includes readings for 52 consecutive weeks. We separated the data into sets N and VN, each of which had 32 weeks of data. Sets VA and T received equal distribution of the remaining weeks. There were a total of nine anomalies, eight of which corresponded to low-demand weekdays, and one related to a weekend with strong demand. Four of the nine anomalies were given to VA, while the other three went to T.

Model Information a single recurrent layer with 300 LSTM units, a dense output layer with 1 neuron, and a dropout of 0.2 made up the RNN. Lookahead plus Look aside equaled 1. We utilized Adam Optimizer with a learning rate of.01, decay of 0.99, and a batch size of 672 to train the RNN for 50 epochs with early halting. We kept the LSTM state constant across batches since the data had lengthy cycles. In figures 4.4 and 4.5, respectively, the outcomes for sets VA and T are shown. An MSE of.08 on N was provided by the prediction model. The prediction errors on VA were used to establish a threshold of 24.

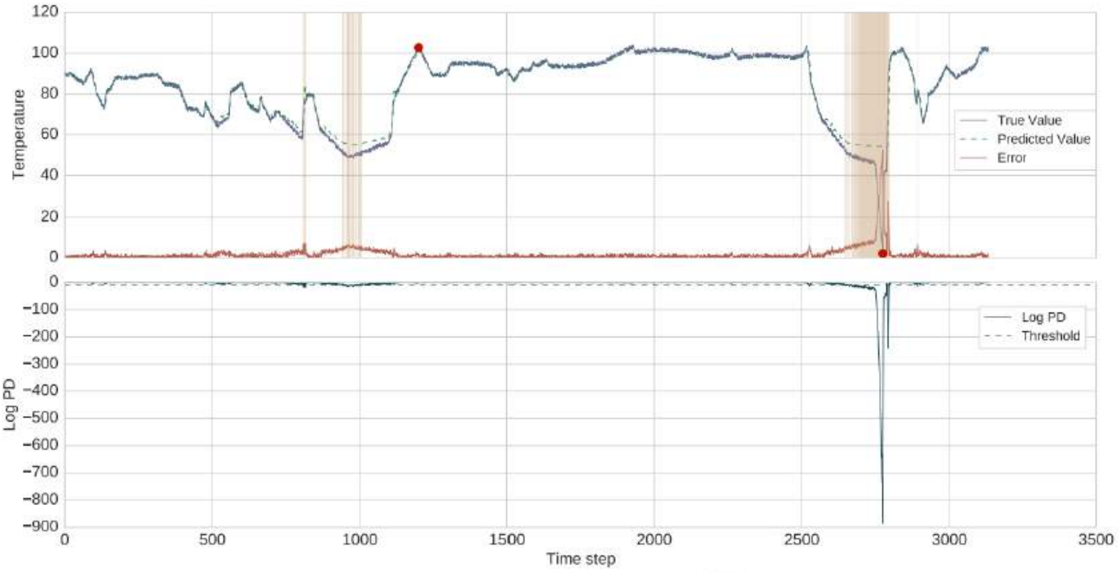
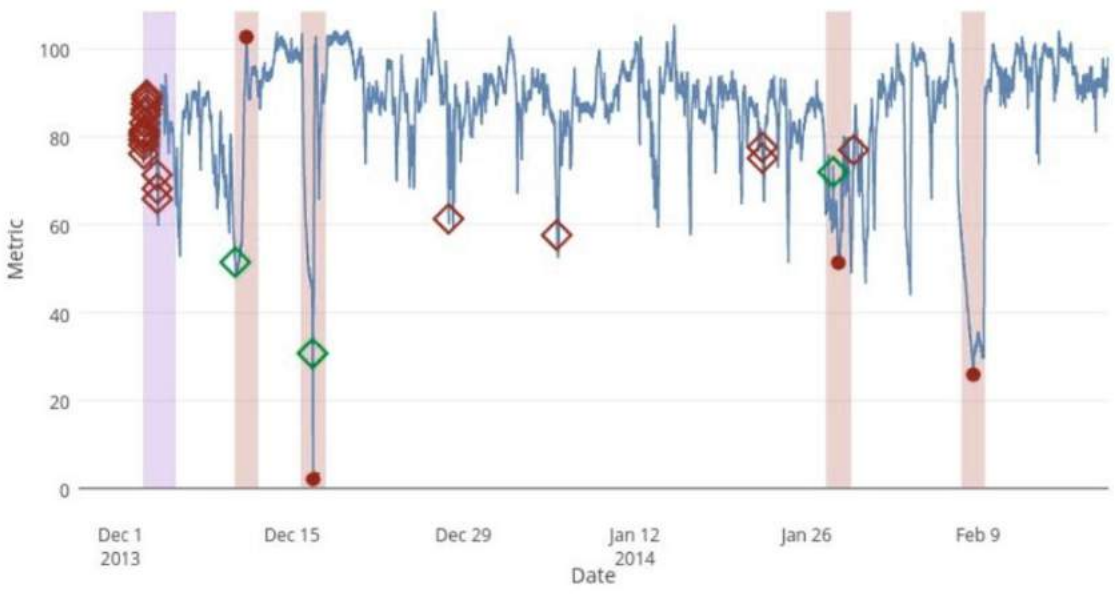


Figure 4.2: Results from a dataset measuring machine temperature. The predictions made on T and the accompanying prediction errors are shown in the top plot. The threshold is set at 11, and the bottom plot displays the log PD values of the prediction errors. The Y-axis displays the matching metric value, while the X-axis displays time steps. Two genuine abnormalities are shown by red markers. Regions with detections performed by the LSTM algorithm are shaded in the top graph.

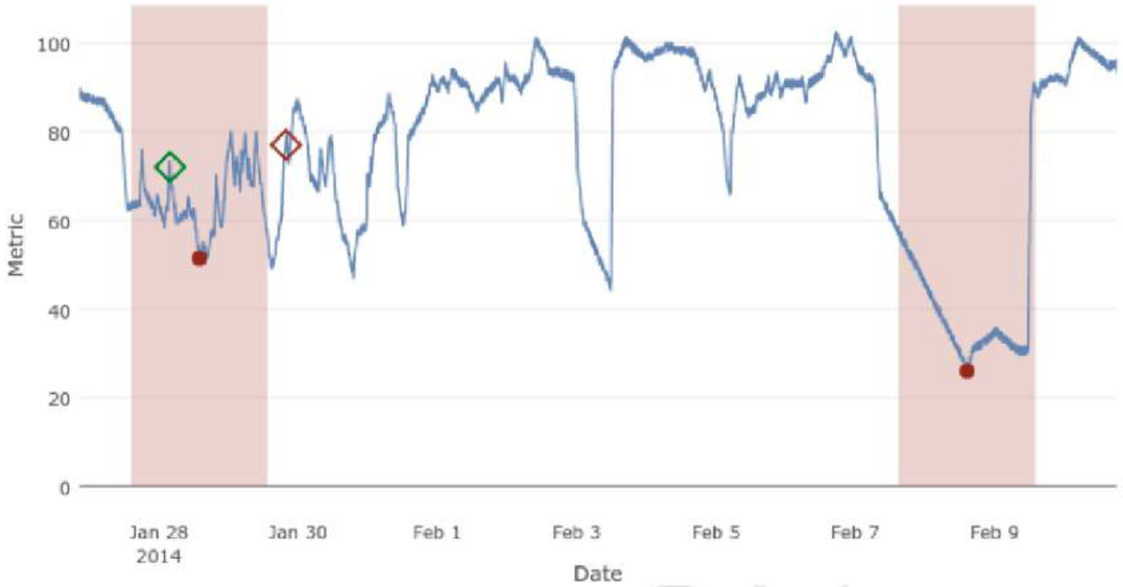
Evaluation: Figures 4.4 and 4.5 show the outcomes for the electricity demand dataset. All four anomalies are picked up by the threshold of 24 without generating any false positives. There is one false positive, but set T's five abnormalities are all found. This false positive was discovered to be a sudden peak upon closer examination. These studies: (Malhotra et al., 2015) all utilized this dataset. The three most uncommon weeks are all weeks with two holidays, according to (Keogh, 2005). Use LSTM networks and treat complete weeks with a holiday as an anomaly in (Singh, 2017). Both of them use F-score as a performance measurement tool. Due to the small number of anomalies and the fact that this should be determined by the use case, we did not utilize a particular measure for detection.

4.1.4 ECG Dataset

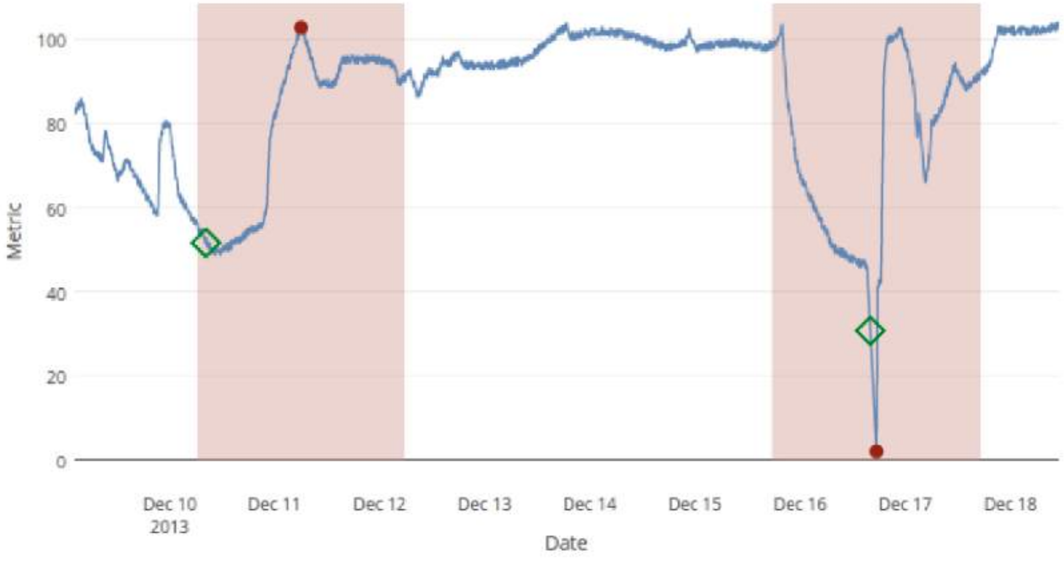
Out of 18000 samples in the ECG dataset, there are only three identified abnormalities. Therefore, we do not employ a predetermined VA to determine a threshold. Instead, the dataset was split into a test set (T) including all three anomalies, a normal training set (N), and a normal validation set (VN) for early stopping.



(a)



(b)



(c)

Figure 4.3: Data on machine temperature as a result of HTM. (A) Displays the findings for the complete dataset. The information matching to figure 4.1 is shown in (b). The findings matching to figure 4.2 are shown in (c). The Y-axis displays temperature, while the X-axis displays time.

Red markers indicate actual abnormalities. True/false positives are represented by green/red diamonds. Anomaly windows are those in the pink tinted areas. The testing window is shown in (a) by the purple-hued region. Plots produced using code found in the NAB GitHub source.

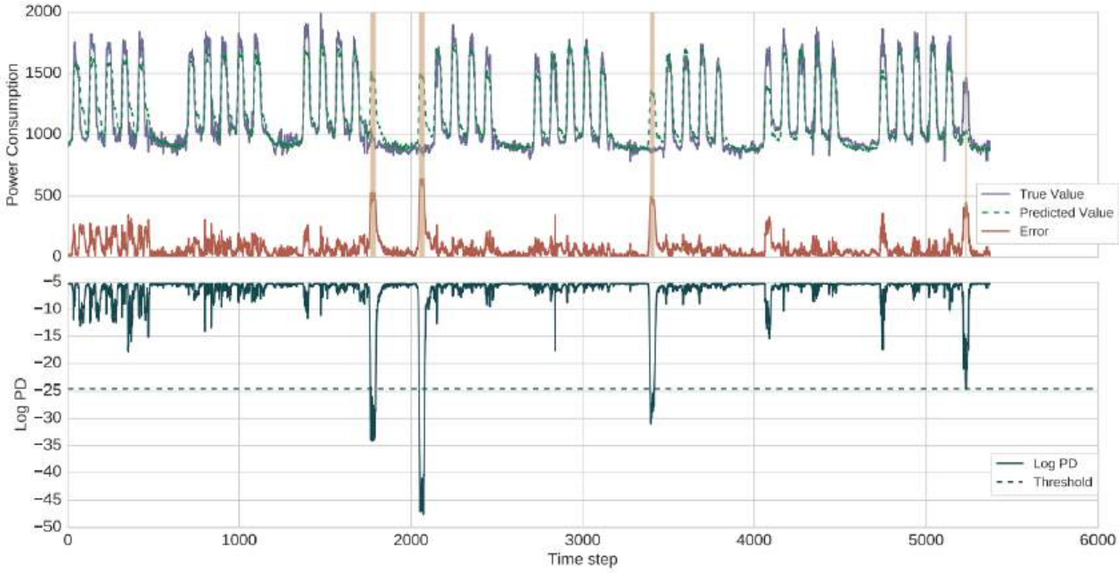


Figure 4.4 Results of the Power Demand Dataset Validation Set. The predictions made by VA are shown in the top plot along with the accompanying prediction errors. The threshold and log PD values for the prediction errors are shown in the bottom plot. The Y-axis displays the matching metric value, while the X-axis displays time steps. Eight weekly cycles make up the plots, and cycles 3, 4, 6, and 7 each include one abnormality. The final anomaly is a weekend (Saturday) with a high demand, whereas the previous three anomalies correspond to weekdays with low demand. To identify all abnormalities, the threshold is set at -24. The top graph's darkened area illustrates areas where anomalies were found by the algorithm.

Model Specifications: The prediction model employed was a stacked RNN with a lookback of 8, a look ahead of 5, and a dropout of 0.1, and two hidden recurrent layers with 60 and 30 LSTM units each. Between batches, LSTM state was not kept. Using the Adam optimizer, we trained the RNN with a learning rate of 0.1, decay of 0.99, and batch size of 256. With early halting, the model was trained for 50 epochs. A threshold of 23 was established on the log PD values of the set T prediction error values, and the MSE on set N was 0.10 overall.

Evaluation: Figure 4.6 displays the findings of the anomaly identification on set T. The model finds all three anomalies at a threshold of 23. These outcomes mirror those in (Qu, 2021), which identifies the three most peculiar sequences in the data. Additionally, the order of the log PD values is reflected in the ranks of the discords, with a lower value suggesting a less common sequence.

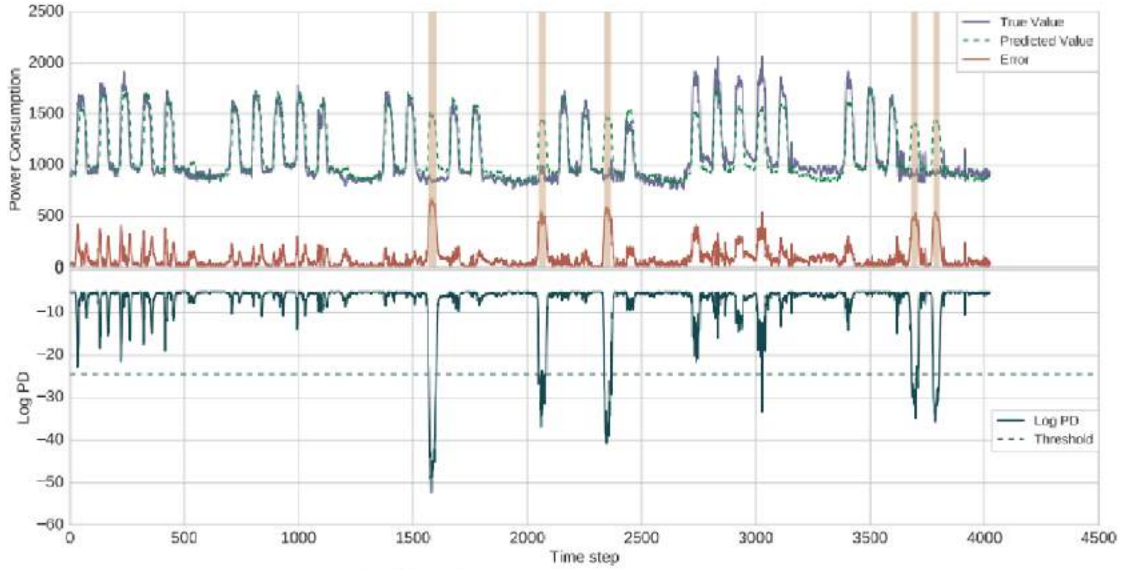


Figure 4.5: Results of a Test Set on a Dataset of Power Demand. The predictions made on T and the accompanying prediction errors are shown in the top plot. The log PD values of the prediction errors and the threshold set at 24 are shown in the bottom plot. The Y-axis displays the matching metric value, while the X-axis displays time steps. The graphs include six weekly cycles, with two anomalies in cycles 4 and 6 and one abnormality in cycle 3. The dark areas reflect the abnormalities that the algorithms have identified. The configured threshold detects all 5 abnormalities, although there is 1 false positive.

4.2 Maintaining LSTM State

The way LSTM state is handled plays a key role in helping LSTMs discover relevant patterns from input. We tested with several LSTM state construction techniques to see how they affected prediction and detection performance. In this part, we go through the findings. The machine temperature dataset did not seem to have any recurring patterns, and it was discovered that the model's performance was unaffected by the method used to preserve the state. Approximately 370 time steps are repeated throughout the ECG dataset, while the precise number of repetitions varies significantly. Although we anticipated that retaining state across batches would improve performance, both methods produced comparable outcomes. Even though the ECG dataset had a lengthy pattern, we think the relevant temporal linkages were only evident within a small number of recent time steps.

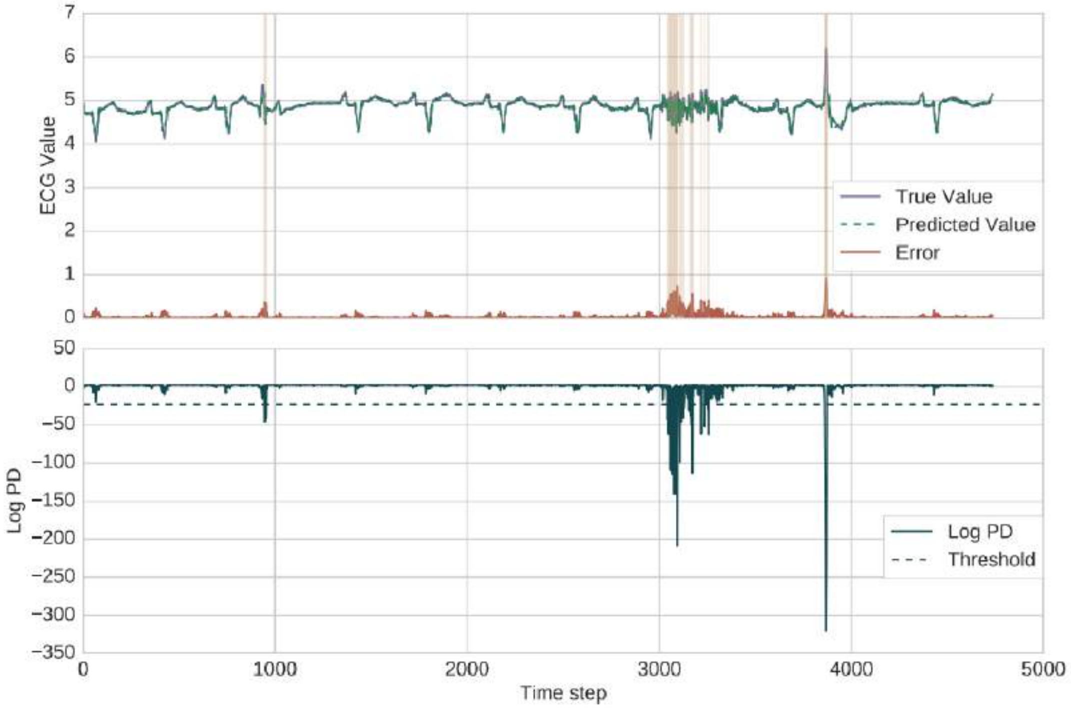
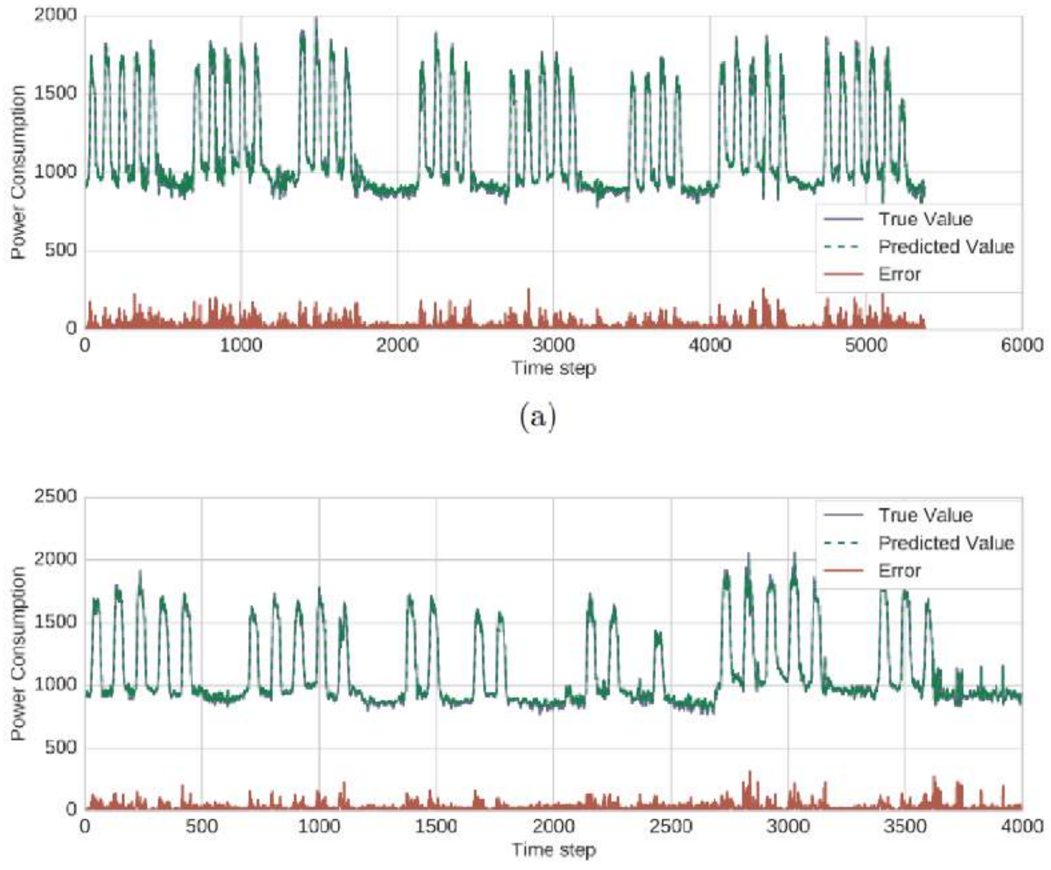


Figure 4.6: outcomes for the ECG dataset. The predictions made on T and the accompanying prediction errors are shown in the top plot. The log PD values of the prediction errors and the threshold of 23 are shown in the bottom plot. The Y-axis displays the matching metric value, while the X-axis displays time steps. Three abnormalities stand out. The dark areas reflect the abnormalities that the algorithms have identified.

To get the intended outcomes for the power demand dataset, state preservation was essential. First, we tested the default setting, which reset the status after each batch. The prediction model was identical to section 4.1.3 in every other way. On N, the model's MSE was 0.0283. The MSE for sets VA and T, which feature abnormalities, was, more critically, 0.017 and.019, respectively (contrast this with the results of section 4.1.3 where the MSE on N, VA, and T was 0.08, 0.15, and 0.26 respectively). Although these findings show a considerable increase in prediction, they cannot be used to discover anomalies. We need a low MSE on set N and a larger MSE on sets VA and T for anomaly detection to function. This is due to the fact that sets VA and T include abnormalities, and the model ought to perform worse when applied to data containing anomalies. Figure 4.7 displays the outcomes using the default option (no state maintenance between batches). It is obvious that the network is unable to learn the five peaks and two lows weekly recurring pattern. For instance, in figure 4.7a, the third weekly cycle only has four peaks and a low on Friday, but the model projections don't forecast a high power consumption for Friday; instead, they match the actual (anomalous) facts. As a consequence, there is no distinction between abnormal and typical data points in the resultant prediction errors.



(a)

(b)

Figure 4.7: prediction using a dataset of electricity demand without keeping state. Both (a) and (b) show the findings on VA and T, respectively. Due to the lack of updated state information, the model is unable to learn the weekly pattern. Time steps are shown on the X-axis, while power use is shown on the Y-axis. When we looked more closely, we discovered that there are many patterns in the data on power consumption at various time scales. There is a lengthy weekly pattern with 672 time steps, but there is also a daily pattern with 96 time steps.

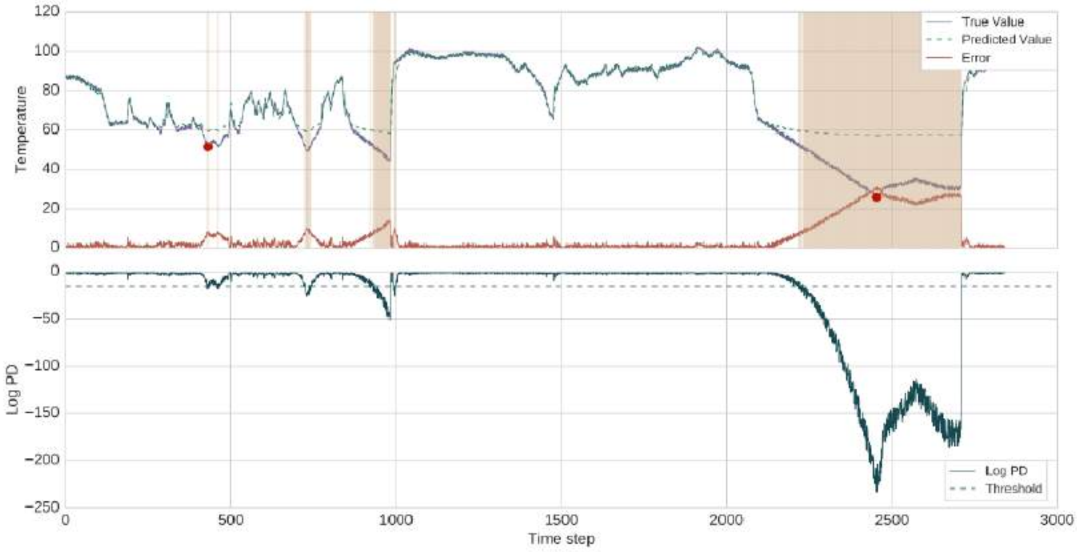
The LSTM state must create a memory of previous data in order to learn these various patterns and must offer context so that the model can correctly forecast the day of the week. The state is wiped clean after each batch in the default configuration. Model learning is almost totally impacted by recent observations in the absence of any valuable input from the state, and the weights are adjusted to reduce prediction errors. Due to its attempt to closely match current observations, this leads the model to fit even aberrant data. After that, we experimented with

Keras' stateful mode. The last model, which is shown in section 4.1.3, was created using this method.

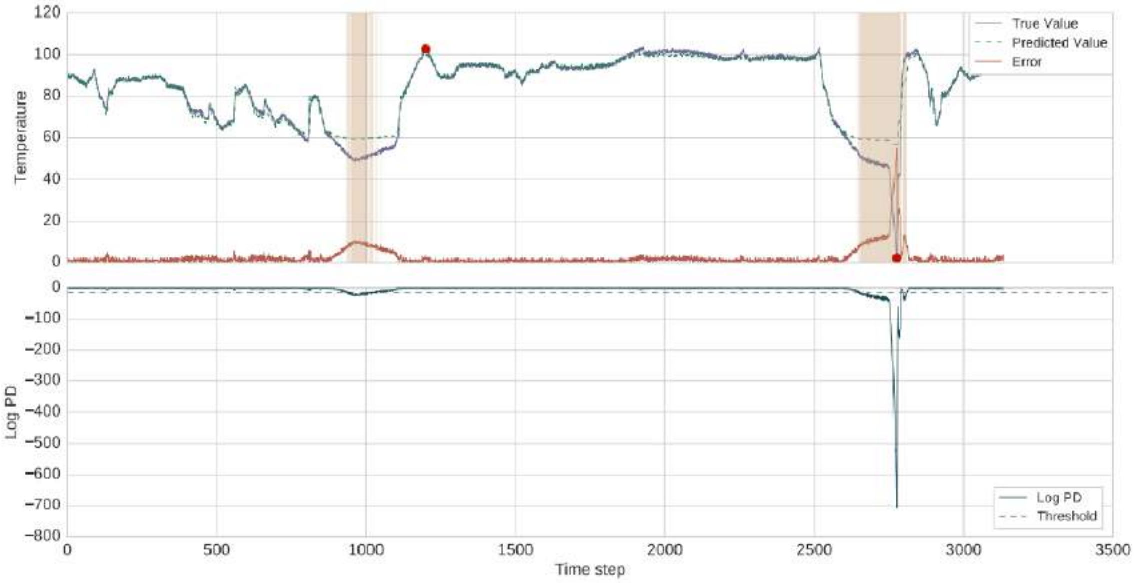
The state was kept constant throughout all training batches, and it was only reset at the beginning of a new epoch over training data. As a result, the model was able to construct the state for a full pass through the training data for all training batches. The state was reset before each epoch, but the model's weights have been fine-tuned over time, and the LSTM gates have learned to store pertinent activations in the cell state. Therefore, the state offers the required input for the model to pick up on various patterns and generate accurate predictions. Other intriguing findings in the stateful mode included a few. First, we discovered that the batch size had an impact on performance. The best results were obtained when the long-range pattern was learned in batches that were about the same size as the weekly cycle. No of the batch size, the LSTM model should be able to learn the patterns. Second, even if the state was not reset before each epoch, the network's performance remained consistent. The LSTM state should typically be reset after each epoch since it may continue to develop and become unstable. It is difficult to draw any firm conclusions, however, since the datasets we utilized were rather tiny (20000 to 40000 observations) and because of the specific Keras structures.

4.3 Feed-forward NNs with Fixed Time Windows

We discovered that the power demand dataset was the only one for which state preservation across batches was necessary. For the remaining two datasets, the state was only preserved for each training sample throughout the lookback number of time steps. Even yet, the results of our anomaly detection and prediction were respectable. We began to wonder whether there was any benefit to employing a model as complicated as the LSTM RNN and if feed forward NNs with fixed-size time windows could also capture the temporal relations in light of this. So, as a prediction model, we tried employing a feed-forward NN. Figures 4.8a and 4.8b for the machine temperature dataset illustrate the outcomes for sets VA and T, respectively. Two hidden layers with 32 neurons each, sigmoid activation, and no dropout make up the prediction model that was employed in this study. Using the Adam optimizer, the model was trained with a learning rate of.01, a batch size of 1024, and early stopping. The output layer, lookback, and look ahead, along with all other features, were identical to those in the LSTM model. The model had an MSE of.08 on set N after 100 training iterations. With VA, a threshold of 16 was chosen, and it produced results for anomaly detection that were comparable to those of the LSTM model. The neural network (NN) employed for the ECG dataset included two hidden layers with 64 and 32 neurons each, sigmoid activation, and no dropout. Using Adam optimizer, the model was trained with a learning rate of.001, a batch size of 1024, and early stopping. The other variables were all the same as before. The model produced an MSE of.07 on set N after 80 epochs of training. All anomalies were found using a threshold of 25 on the log PD values of the errors on set T. Figure 4.9 displays the ECG dataset findings, which are comparable to those for the LSTM model.



(a)



(b)

Figure 4.8: Data about machine temperature is the subject of feed-forward NN results. Results for set VA are shown in (a). The output for set T is shown in (b). The findings are comparable to figures 4.1 and 4.2 and were determined with a threshold of 16. The Y-axis displays the matching metric value while the X-axis displays the time step.

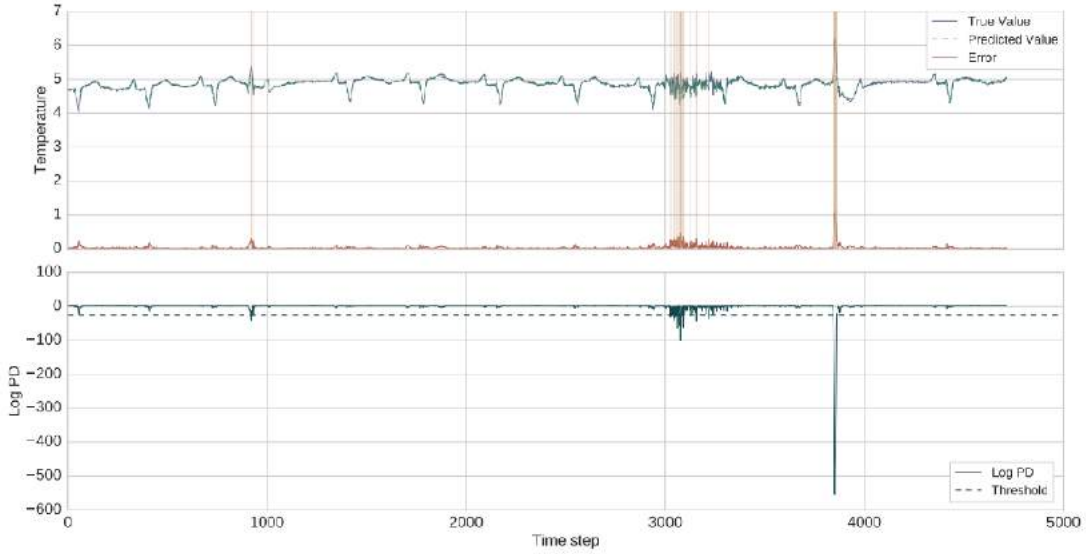


Figure 4.9: outcomes from using feed-forward NN on ECG data. A threshold of 25 finds all 3 abnormalities in the findings on set T. The Y-axis displays the matching metric value, while the X-axis displays the time step. The feed-forward NN provided decent prediction accuracy for the power demand dataset but failed to identify any abnormalities. We experimented with several architectures and 100 different window sizes. This was a similar issue to the LSTM algorithms when state was not maintained between batches. Long and complex patterns in the data were not learned by the NN.4.4 Other Experiments

4.4.1 Effect of Lookback

The amount of time steps for which the RNN is unfurled for back-propagation is the parameter lookback. The best value for unfolding the RNN relies on the kind of data and the temporal correlations that are present in it; there is no suggested value for this. For various jobs, lookback values between 8 and 200 have been employed effectively. LSTM RNNs can still learn patterns longer than lookback length despite lookback value being a significant learning-influencing parameter since the LSTM state may store data from any prior time step. RNNs trained using back-propagation over a single time step was shown in (Mikolov, 2012) to be capable of learning lengthier patterns for language modelling tasks. We discovered that raising the lookback values up to a certain point enhanced prediction performance, and further raising the lookback values had no positive effect. On a single time step prediction task, Figure 4.10 depicts the training MSE versus lookback for an ECG dataset (look ahead equal to 1). For a lookback value of 14, the MSE first decreases with a minimum of 0.07. Lookback values over 14 are not improved by increasing them. On further datasets, we discovered the same pattern.

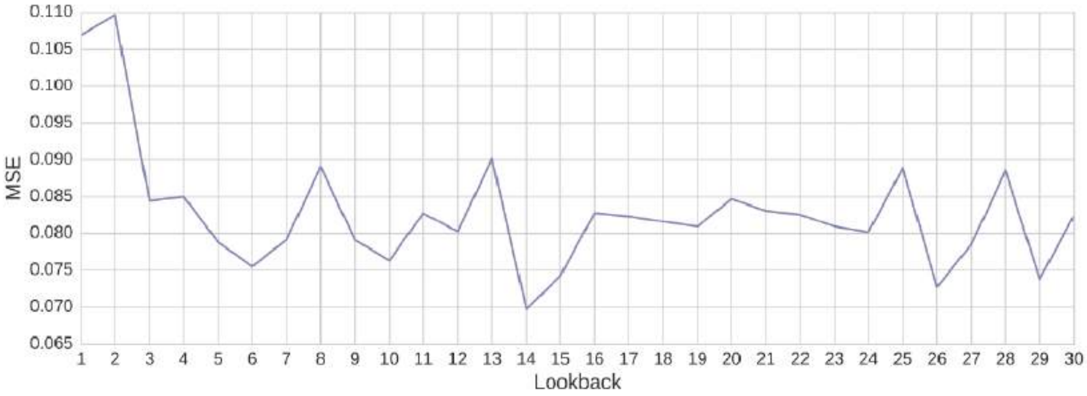


Figure 4.10: Backward and forward accuracy. The graphic displays the training MSE on the ECG dataset for various lookback levels. The MSE is not decreased by increasing lookback values beyond 14. The Y-axis corresponds to the respective MSE values, while the X-axis displays lookback values.4.4.2 Prediction Accuracy vs. Anomaly Detection

We saw a trade-off between forecast accuracy and anomaly detection outcomes in several of our trials. The best RNN for anomaly detection was not one that was designed for reducing the prediction MSE. Finding a threshold to distinguish anomalous from normal points without producing an excessive number of false positives was challenging since prediction errors were tiny over the whole set VA (and T). When we did not keep state between batches, this behaviour appeared in one case on the power demand dataset. Section 4.2 goes into further depth about this. The following provides an intuitive explanation for this phenomenon. When we are improving the model for prediction, a kind of over fitting takes place that cannot be stopped early since the MSE keeps dropping on both N and VN. The model's weights are adjusted such that only recent data have an impact on model predictions, and the model ignores any temporal linkages or information about its state. In order to achieve a low prediction error, the model attempts to fit even the anomalous data. Anomaly-related model training is unsupervised, and the model lacks an objective or feedback to prevent fitting anomalies. The only solution we discovered was to manually adjust the model to find abnormalities on a specific VA. It did, however, come at the expense of a rise in the MSE value. The majority of manual tuning involves lowering the value of one or more of the parameters lookback value, LSTM unit count per recurrent layer, or recurrent layer count.

Chapter Five

Discussion and Conclusion

5.1 Anomaly Detection Datasets, Metrics, and Evaluation

The selection of labelled datasets proved to be the most difficult part of the endeavour. Due to its applicability, anomaly detection is regarded as a significant subject and has received much research. It was thus unexpected that the lack of labelled and benchmark datasets continues to be a significant problem in anomaly detection research. Adding random noise to data points in a dataset that is known to include only normal observations is one approach to solving this problem. However, there are a number of issues with this strategy. First off, using manufactured anomalies makes it harder to gauge how well an algorithm performs on actual data. Second, supervised classification is preferable if there are enough labelled anomalies since it solves the issue considerably more easily than unsupervised anomaly detection. Additionally, anomalies are uncommon occurrences in the actual world as well, and it might be argued that the development process should reflect this reality.

We made the decision not to assess our approaches using metrics like F-score, accuracy, recall, etc. Given the rarity of anomalies, we think using a statistic to demonstrate strong performance would be deceptive. We decided to contrast our outcomes with those of earlier research since it is still important to establish the efficacy of our strategy. Our goal is to demonstrate the appropriateness of LSTM for temporal anomaly identification rather than to assert that it is a better method, even though we are aware that each dataset we choose has much too few abnormalities. A similar choice was to employ thresholding to separate anomalous from regular data rather than developing a specific anomaly detection technique. The intended use and area of anomaly detection should be taken into consideration while developing a particular anomaly detection technique. In the case of the power demand dataset, for instance, there are several methods to define an anomaly: one might consider a day, a week, or even a single observation to be an aberrant observation for real-time detection. Point anomalies and collective abnormalities are both present in the ECG dataset. The machine temperature dataset includes point anomalies that have been annotated. However, it is evident that the whole areas in and surrounding these sites are peculiar. For the purpose of detecting anomalies, the HTM algorithm employed windows at these sites. Without major modifications, the thresholding strategy we employed may be used to the identification of point anomalies. But for many applications, strange sequences come before failures.

A communal anomaly detection strategy might be more beneficial in these circumstances. Using a sliding window over prediction mistakes and computing anomaly scores based on the total of errors in the window is one potential strategy, similar to (Bontemps, 2016). The assessment of algorithms is one of the major problems in anomaly detection research that we have discussed throughout this study. The assessment issue is made more difficult by the fact that various algorithms have different trade-offs and identify abnormalities in different ways. This makes comparing various algorithms pointless. There have been several initiatives to solve these issues. (Emmott, 2013) provided a criteria and method for generating anomaly detection datasets from classification datasets. The authors of (Campos, 2016) examined several measures and datasets that have been used to the assessment of outliers. They stressed the need of the scientific community adopting consistent criteria.

According to the authors, each novel outlier detection technique should be validated by showing how well it performs over a range of datasets, metrics, and parameters. However, it is usual in practice to assert that a new method is better by testing it on extremely narrow datasets and using a limited number of metrics. The authors demonstrated the outcomes of 12 common algorithms on more than 20 datasets with various features and suggested using them to assess both new algorithms and datasets. They also discussed how previous studies were not reproducible. Although neither of these papers makes a claim to provide benchmark datasets for anomaly identification, we think they both constitute significant first efforts in creating the shared standards for anomaly detection assessment that are sorely lacking.

The normal distribution of residuals or prediction errors was one of the presumptions we made. We tested each dataset's residuals for normality using the Shapiro-Wilk method. We disproved the idea that the prediction errors were regularly distributed based on the results of the testing. The errors have broad tails and are somewhat skewed for all datasets. The procedure may still be employed, however, as we merely apply the normalcy assumption to generate a threshold to distinguish anomalies.

5.2 LSTMs vs Feed-forward NNs

There are a number of issues that worth consideration based on our examination of LSTMs and the tests we ran. Perhaps the most important one is how feed-forward NNs perform in comparison to LSTM RNNs. For time series prediction and anomaly detection on two of the three datasets, feedforward NNs with fixed size time windows performed similarly to LSTMs. The theory and mathematics behind the operation of LSTMs are more complicated and difficult to comprehend than those of feed-forward NNs. LSTM network training and tuning is also a slower and riskier procedure. This begs the issue of whether there is any benefit to employing LSTMs rather than feed-forward NNs. NNs that feed information forward lack explicit memories of the past. The breadth of the input time frame restricts their capacity to simulate temporal connections.

Input data is handled more like a multidimensional feature vector than a list of observations, even for a specific time frame. This approach would only work with short-term temporal relationships and simple time series when all the necessary data is available from a small number of recent inputs. With the advantage of hindsight, it can be deduced that datasets on machine temperature and ECG patterns are examples of straightforward time series. These two datasets vary visually. The machine temperature dataset does not seem to have any structure and seems to be very random, in contrast to the ECG dataset, which has a recurring pattern of 370 time steps. In this context, the findings from the power demand dataset become more important since they demonstrate that LSTMs are better than feed-forward NNs at modelling complicated time series. The power demand dataset demonstrates distinct patterns at various time scales, including a daily pattern and a weekly pattern, in addition to a lengthy repeating pattern (670 time steps). When utilizing feed-forward NNs, we were unable to get any decent anomaly detection results.

Learning pertinent patterns and spotting abnormalities required careful LSTM cell state maintenance. (Gers, 2001), where the authors used LSTMs for prediction on two benchmark time series datasets, provides some evidence in favour of our approach. In their trials, the feed-forward NNs were given windows of successive time steps, but the LSTM model only had access to the current time step. It was discovered that feed-forward NNs outperformed the LSTM model for the particular datasets. The authors came to the conclusion that LSTM's capacity for long-term memory does not provide any advantages since many time series jobs only call for a few recent time steps. They advised use LSTMs only when time window-based techniques failed to provide the intended outcomes. In order to further minimize the prediction errors, it was also suggested to combine the two methods by first training feed-forward NNs and then using LSTMs.

It may be challenging to predict in advance the kind of associations that exist in data or if the data calls for a model as intricate as LSTM. Testing simpler models first before going on to more complicated ones is often a good idea. Machine learning may benefit from Occam's razor in the same way that other disciplines do. One of the objectives of this endeavour was to research and examine LSTMs. Only after doing LSTM trials did we decide to utilize feed-forward NNs. However, employing LSTMs without first exploring easier methods seems to be a trap. LSTMs have gained recent appeal because to their effectiveness in solving a variety of sequential and temporal challenges. Use LSTMs for time series anomaly detection, for instance (Malhotra, 2016), but don't compare them to more straightforward methods.

5.3LSTMs for Temporal Anomaly Detection

Three distinct datasets were used in our research. These datasets vary not just in terms of the patterns they include, but also in terms of the types of anomalies they contain. Anomalies in the machine temperature dataset are indicated. It looks to be random and lacks any recurring patterns. Two anomalies with extraordinary values stand out. The other two seem to be very random. According to (Lavin, 2015), these later abnormalities are exceedingly subtle and are preceded by minute deviations from the norm that are invisible to the naked eye. The authors emphasize how useful anomaly windows are for spotting these abnormalities. The LSTM system detected these two anomalies in our studies as well; however it missed the real abnormalities. However, as mentioned, a window-based strategy would provide outcomes comparable to (Lavin, 2015).

The documented anomalies in the ECG dataset include both extreme values and odd sequences. It features a pattern that repeats, however the pattern's duration fluctuates. All three abnormalities were found using our approach. The power demand dataset includes noise, shows patterns at various time scales, and has the longest dependencies. Even if there are no labels, we can discern between typical and uncommon patterns by using the day's week as background. The model learnt the various patterns by carefully preserving the LSTM state, and all anomalies were found. Additionally, the anomaly ratings provided a decent indicator of the severity or likelihood of aberration. We think that LSTMs are good temporal anomaly detectors based on our studies. We were successful in fine-tuning LSTMs to recognize various kinds of abnormalities in datasets with distinct properties. Because LSTMs have memory, they can make predictions while taking into account context and input data. An LSTM network set for prediction accuracy, however, is not always an effective anomaly detector, as we have shown. Utilizing LSTM memory is crucial for ensuring that it retains pertinent historical data.

For multiple time step prediction, we employed LSTMs. One time step prediction is adequate for anomaly detection. The effectiveness of LSTMs as time series modellers was shown via the use of multiple time step prediction. As a crucial initial step in transitioning from anomaly detection to anomaly prediction and even predictive maintenance, predicting several time steps may provide an early warning of unexpected behaviour.

5.4 LSTMs: Evolution and Future

RNNs can now learn long-term dependencies thanks to LSTMs. In situations where the significant events (commonly referred to as teacher signals) were more than a few time steps ago, it was shown that vanilla RNNs perform poorly. The issue of exploding/vanishing gradients, which inhibits the model learning to converge to optimum parameters, affects plain vanilla RNNs. By including a memory cell that guarantees continual error flow and gating units to regulate information flow in and out of the cell, LSTMs solve this issue. When the input data's sequences had well defined bounds, the original LSTM design performed admirably; otherwise, it fell short.

The solution was forgetting gates, which periodically updated the LSTM memory. Since then, other modifications have been put forward. The most notable one allows LSTMs to learn accurate intervals and is peephole connections. All of this refinement, however, comes at the expense of increased complexity. It is well known that since so many distinct parameters need to be tweaked, LSTMs are challenging to train and optimize. In reality, one may get decent results by using trained models and known parameter values, but it takes a lot of work to comprehend how LSTMs function and what their constraints are. To comprehend the function of the various LSTM components and determine if the various LSTM versions genuinely provide any advantages, studies like (Jozefowicz, 2015) have been carried out. Gated recurrent unit (GRU), a novel, more straightforward design, was put out in (Cho et al., 2014) and has since gained a lot of traction. This is just another example of how the normal LSTM may be fairly complex, and it suggests that in the future, we was likely see more straightforward forms.

In this thesis, we used LSTM networks to create a temporal anomaly detection system. We began by looking into LSTMs to understand their intricate architecture, capacity to recognize distant relationships, and various methods of LSTM state maintenance. Then, using a summary prediction model as the foundation, we created an anomaly detection technique. We quickly discussed a number of problems that are common in anomaly detection research. We chose three real-world datasets that have been utilized in prior studies and exhibit various forms of anomalies to get around these issues. Last but not least, we ran tests on these datasets. We draw the conclusion that LSTMs are efficient time series modellers and anomaly detectors based on our findings. However, as feed-forward NNs are often adequate, we suggest attempting them first. Because of their increased complexity, LSTMs are only advantageous when the data contains several patterns and long-range relationships.

5.6 Future Work

A few of the difficulties in anomaly identification, such as dataset selection and algorithm assessment, have been discussed in (Petneházi, 2019). Even if none of the suggestions are for time series, we think it could be worthwhile to give them a go. Our approach for detecting anomalies directly detects abnormalities at points. A technique that makes use of the total prediction errors in a sliding window may be created to find collective anomalies. In order to further highlight the advantages of LSTMs, we would also want to deal with more complicated datasets that include long-range connections and patterns over different time scales. Finally, because GRUs streamlines the LSTM design, trying them out might be fascinating. GRUs does not keep an explicit memory cell in place and instead combine the input and forget gates into a single update gate. Due to their more straightforward construction, GRUs is more effective computationally than LSTMs. They have also been shown to generate outcomes comparable to LSTMs (Chung, 2014) and are rising in popularity as a substitute.

Reference

A. F. Emmott, S. Das, T. Dietterich, A. Fern, and W.-K. Wong, “Systematic Construction of Anomaly Detection Benchmarks from Real Data,” in Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description, ser. ODD ’13. New York, NY, USA: ACM, 2013. doi: 10.1145/2500853.2500858. ISBN 978-1-4503-2335-2 pp. 16–21. [Online]. Available: http://doi.acm.org/10.1145/2500853.2500858

A. Isina and S. Ozdalili, “Cardiac arrhythmia detection using deep learning,” Procedia Computer Science, vol. 120, pp. 268–275, 2017. [22] W. Sun, N. Zeng, and Y. He, “Morphological arrhythmia automated diagnosis method using gray-level co-occurrence matrix enhanced convolutional neural network,” IEEE Access, vol. 7, no. 67123, pp. 2169–3536, 2019.

A. Lavin and S. Ahmad, “Evaluating Real-Time Anomaly Detection Algorithms - The Numenta Anomaly Benchmark,” in 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), Dec 2015. doi: 10.1109/ICMLA.2015.141 pp. 38–44.

A. M. Ghimes, A. M. Avram, and A. V. Vladuta. “A Character Prediction Approach in a Security Context using a Recurrent Neural Network.” In: 2018 International Symposium on Electronics and Telecommunications (ISETC) (Nov. 2018). doi: 10. 1109/ISETC.2018.8584007.

Abhinav-Vishwa, Lal M.K., Dixit S, and Vardwaj P. (2011) “Clasification of arrhythmic ECG data using machine learning techniques.” International Journal of Interactive Multimedia and Artificial Intelligence1(4):67-70.

Acharya, U.R.; Oh, S.L.; Hagiwara, Y.; Tan, J.H.; Adam, M.; Gertych, A.; San Tan, R. A deep convolutional neural network model to classify heartbeats. Comput. Biol. Med. 2017, 89, 389–396.

Alfaras, M.; Soriano, M.C.; Ortín, S. A fast machine learning model for ECG-based heartbeat classification and arrhythmia detection. Front. Phys. 2019, 7, 103.

Antonicelli, R.; Ripa, C.; Abbatecola, A.M.; Capparuccia, C.A.; Ferrara, L.; Spazzafumo, L. Validation of the 3-lead tele-ECG versus the 12-lead tele-ECG and the conventional 12-lead ECG method in older people. J. Telemed. Telecare 2012, 18, 104–108.

Ary L. Goldberger, Luis A. N. Amaral, Leon Glass, Jeffrey M. Hausdorff, Plamen Ch. Ivanov, Roger G. Mark, Joseph E. Mietus, George B. Moody, Chung-Kang Peng, and H. Eugene Stanley. (2000) "PhysioBank, PhysioToolkit, and PhysioNet." Circulation101 (23):215-220.

Atoui, H.; Fayn, J.; Rubel, P. A novel neural-network model for deriving standard 12-lead ECGs from serial three-lead ECGs: Application to self-care. IEEE Trans. Inf. Technol. Biomed. 2010, 14, 883–890.

Augustyniak, P. On the Equivalence of the 12-Lead ECG and the VCG Representations of the Cardiac Electrical Activity. In Proceedings of the 10th International Conference on System-Modelling-Control, Zakopane, Poland, 21–25 May 2001; pp. 51–56.

Babak Mohammadzadeh-Asl and Seyed Kamaledin Setarehdan. (2006) "Neural network based arrhythmia classification using Heart Rate Variability signal." In 14th EuropeanSignal Processing Conferencepages 1-4.

Batra, A.; Jawa, V. Classification of arrhythmia using conjunction of machine learning algorithms and ECG diagnostic criteria. Train. J. 1975, 1, 1–7.

C. Cao, F. Liu, H. Tan, D. Song, W. Shu, W. Li, Y. Zhou, X. Bo, and Z. Xie. "Deep Learning and Its Applications in Biomedicine." In: Genomics, Proteomics Bioinformatics 16.1 (2018), pp. 17 –32. issn: 1672-0229. doi: https://doi.org/10.1016/ j.gpb.2017.07.003. url: http://www.sciencedirect.com/science/article/ pii/S1672022918300020.

Calkins, H., Brugada, J., Packer, D. L., Cappato, R., Chen, S. A., Crijns, H. J., ... & Shemin, R. J. (2007). HRS/EHRA/ECAS Expert Consensus Statement on Catheter and Surgical Ablation of Atrial Fibrillation: Recommendations for Personnel, Policy, Procedures and Follow-Up: A report of the Heart Rhythm Society (HRS) Task Force on Catheter and Surgical Ablation of Atrial Fibrillation Developed in partnership with the European Heart Rhythm Association (EHRA) and the European Cardiac Arrhythmia Society (ECAS); in collaboration with the American College of Cardiology (ACC), American Heart Association (AHA), and the .... Europace, 9(6), 335-379.

Canlas, R. Data Mining in Healthcare: Current Applications and Issues. Master's Thesis, School of Information Systems & Management, Carnegie Mellon University, Adelaide, Australia, 2009.

Chen, T.M.; Huang, C.H.; Shih, E.S.; Hu, Y.F.; Hwang, M.J. Detection and classification of cardiac arrhythmias by a challenge-best deep learning neural network model. Iscience 2020, 23, 100886.

Drew, B.J.; Adams, M.G.; Pelter, M.M.; Wung, S.F.; Caldwell, M.A. Comparison of standard and derived 12-lead electrocardiograms for diagnosis of coronary angioplasty-induced myocardial ischemia. Am. J. Cardiol. 1997, 79, 639–644.

Drew, B.J.; Pelter, M.M.; Brodnick, D.E.; Yadav, A.V. Comparison of a new reduced lead set ECG with the standard ECG for diagnosing cardiac arrhythmias and myocardial ischemia. J. Electrocardiol. 2002, 35, 13.

E. Keogh, J. Lin, and A. Fu, "HOT SAX: Efficiently Finding the Most Unusual Time Series Subsequence," in Fifth IEEE International Conference on Data Mining (ICDM'05), Nov 2005. doi: 10.1109/ICDM.2005.79. ISSN 1550-4786 pp. 8

Eluyode, O. S., & Akomolafe, D. T. (2013). Comparative study of biological and artificial neural networks. European Journal of Applied Engineering and Scientific Research, 2(1), 36-46.

F. A. Gers, D. Eck, and J. Schmidhuber, Applying LSTM to Time Series Predictable through Time-Window Approaches. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 669–676. ISBN 978-3-540-44668-2. [Online]. Available: http://dx.doi.org/10.1007/3-540-44668-0_93

Figueiredo, C.; Mendes, P. Towards Wearable And Continuous 12-Lead Electrocardiogram Monitoring-Synthesis of the 12-lead Electrocardiogram using 3 Wireless Single-lead Sensors. In International Conference on Biomedical Electronics and Devices; SCITEPRESS: Setúbal, Portugal, 2012; Volume 2.

Frank, E. An accurate, clinically practical system for spatial vectorcardiography. Circulation 1956, 13, 737–749.

G. O. Campos, A. Zimek, J. Sander, R. J. G. B. Campello, B. Micenková, E. Schubert, I. Assent, and M. E. Houle, "On the Evaluation of Unsupervised Outlier Detection: Measures, Datasets, and an Empirical Study," Data Mining and Knowledge Discovery, vol. 30, no. 4, pp. 891–927, Jul 2016. doi: 10.1007/s10618-015-0444-8. [Online]. Available: http://dx.doi.org/10.1007/s10618-015-0444-8

G.Sannino and G. Pietro, "A deep learning approach for ecg-based heartbeat classification for arrhythmia detection," Future Generation Computer Systems, vol. 86, no. 10, pp. 446–445, 2018.

Gao, J.; Zhang, H.; Lu, P.; Wang, Z. An effective LSTM recurrent network to detect arrhythmia on imbalanced ECG dataset. J. Healthc. Eng. 2019, 2019, 6320651.

Gerc, V., Masic, I., Salihefendic, N., & Zildzic, M. (2020). Cardiovascular diseases (CVDs) in COVID-19 pandemic era. Materia Socio-Medica, 32(2), 158.

Goldberger, A.L.; Goldberger, Z.D.; Shvilkin, A. Clinical Electrocardiography: A Simplified Approach E-Book: A Simplified Approach; Elsevier Health Sciences: Philadelphia, PA, USA, 2017.

Gupta, A.; Banerjee, A.; Babaria, D.; Lotlikar, K.; Raut, H. Prediction and classification of cardiac arrhythmia. In Sentimental Analysis and Deep Learning; Springer: Berlin, Germany, 2022; pp. 527–538.

H. Fujita and D. Cimr, "Computer aided detection for fibrillations and flutters using deep convolutional neural network," Information Sciences, vol. 486, pp. 1231–239, 2019. [25] U. R. Acharya, S. L. Oh, Y. Hagiwara, J. H. Tan, M. Adam, A. Gertych, and R. S. Tan, "A deep convolutional neural network model to classify heartbeats," Computers in Biology and Medicine, vol. 89, pp. 389–396, 2017.

Hammad, M.; Iliyasu, A.M.; Subasi, A.; Ho, E.S.; Abd El-Latif, A.A. A multitier deep learning model for arrhythmia detection. IEEE Trans. Instrum. Meas. 2020, 70, 1–9.

Hampton, J. The ECG Made Easy E-Book; Elsevier Health Sciences: St. Louis, MO, USA, 2013.

Hiriyannaiah, S.; GM, S.; MHM, K.; Srinivasa, K. A comparative study and analysis of LSTM deep neural networks for heartbeats classification. Health Technol. 2021, 11, 663–671.

I. Goodfellow, Y. Bengio, and A. Courville. Deep Learning. MIT Press, 2016. url: http://www.deeplearningbook.org.

J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," arXiv preprint arXiv:1412.3555, 2014. [Online]. Available: https://arxiv.org/abs/1412.3555

J. D. Enderle, S. M. Blanchard, and J. D. Bronzino. Introduction to Biomedical Engineering. San Diego, USA: Academic Press, 2000. isbn: 0-12-238660-4.

J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian Optimization of Machine Learning Algorithms," in Advances in Neural Information Processing Systems 25. Curran Associates, Inc., 2012, pp. 2951–2959. [Online]. Available: http://papers.nips.cc/paper/ 4522-practical-bayesian-optimization-of-machine-learning-algorithms.pdf

K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," arXiv preprint arXiv:1406.1078, 2014. [Online]. Available: https://arxiv.org/abs/1406.1078

K. Fukushima. "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position." In: Biological Cybernetics 36.4 (1980), pp. 193–202. issn: 1432-0770. doi: 10.1007/BF00344251. url: https://doi.org/10.1007/BF00344251.

Kristensen, A.N.; Jeyam, B.; Riahi, S.; Jensen, M.B. The use of a portable three-lead ECG monitor to detect atrial fibrillation in general practice. Scand. J. Primary Health Care 2016, 34, 304–308.

L. Bontemps, V. L. Cao, J. McDermott, and N.-A. Le-Khac, Collective Anomaly Detection Based on Long Short-Term Memory Recurrent Neural Networks. Springer International Publishing, 2016, pp. 141–152. ISBN 978-3-319- 48057-2. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-48057-2_9

Luz, E.J.d.S.; Nunes, T.M.; De Albuquerque, V.H.C.; Papa, J.P.; Menotti, D. ECG arrhythmia classification based on optimum-path forest. Expert Syst. Appl. 2013, 40, 3561–3573. [CrossRef]

M. A. Rahhal, Y. Bazi, H. AlHichri, N. Alajlan, F. Melgani, and R.R.Yagerc, "Deep learning approach for active classification of electrocardiogram signals," Information Sciences, vol. 345, pp. 340–354, 2016

M. C. Chuah and F. Fu, ECG Anomaly Detection via Time Series Analysis. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 123–135. ISBN 978-3-540-74767-3. [Online]. Available: http://dx.doi.org/10.1007/ 978-3-540-74767-3_14

M. Deng, C. Wang, M. Tang, and T. Zheng, "Extracting cardiac dynamics within ecg signal for human identification and cardiovascular diseases classification," Neural Networks, vol. 100, pp. 70–83, 2018.

M. Jones, D. Nikovski, M. Imamura, and T. Hirata, "Anomaly Detection in Real-Valued Multidimensional Time Series," in International Conference on Bigdata/Socialcom/Cybersecurity. Stanford University, ASE, 2014. [Online]. Available: http://www.merl.com/publications/docs/TR2014-042.pdf

M. Mahmud, M. S. Kaiser, A. Hussain, and S. Vassanelli. "Applications of Deep Learning and Reinforcement Learning to Biological Data." In: IEEE Transactions on Neural Networks and Learning Systems 29.6 (2018), pp. 2063–2079. issn: 2162-237X. doi: 10.1109/TNNLS.2018.2790388. [

M. Training and S. LLC. Ekg reference. 2019. Available online: https://www.practicalclinicalskills.com/ekg-reference (accessed on 4 March 2020).

Mehri-Dehnavi, A.; Salehpour, N.; Rabbani, H.; Farahabadi, A.; Farahabadi, E. Automatic Analysis of Vectorcardiogram Signal for Detection of Cardiovascular Diseases. Ph.D. Thesis, Isfahan University of Medical Sciences, Isfahan, Iran, 2013.

Naser Safdarian, Nader Jafarnia, and Gholamreza Attarodi. (2014) "A new pattern recognition method for detection and localization of myocardial infarction using T-wave integral and total integral as extracted features from one cycle of ECG signal." Journal of Biomedical Science and Engineering 7(10):818-824

Nelwan, S.P.; Kors, J.A.; Meij, S.H.; van Bemmel, J.H.; Simoons, M.L. Reconstruction of the 12-lead electrocardiogram from reduced lead sets. J. Electrocardiol. 2004, 37, 11–18.

O. Yildirim, "A novel wavelet sequence based on deep bidirectional lstm network model for ecg signal classification," Computers in Biology and Medicine, vol. 96, pp. 189–202, 2018. [30] O. Yildirim, R. S. Tan, and U. R. Acharya, "An efficient compression of ecg signals using deep convolutional autoencoders," Cognitive System Research, vol. 52, pp. 198–211, 2018.

O. Yildirim, U. B. Baloglu, E. J. Ru-San Tan, and U. R. Acharya, "A new approach for arrhythmia classification using deep coded features and lstm networks," Computer Methods and Programs in Biomedicine, vol. 176, pp. 121–133, 2019.

P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long Short Term Memory Networks for Anomaly Detection in Time Series," in Proceedings. Presses universitaires de Louvain, 2015, p. 89. [Online]. Available: https://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2015-56.pdf

P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long Short Term Memory Networks for Anomaly Detection in Time Series," in Proceedings. Presses universitaires de Louvain, 2015, p. 89. [Online].

Parvaneh, S.; Rubin, J.; Babaeizadeh, S.; Xu-Wilson, M. Cardiac arrhythmia detection using deep learning: A review. J. Electrocardiol. 2019, 57, S70–S74.

Petneházi, G. (2019). Recurrent neural networks for time series forecasting. arXiv preprint arXiv:1901.00069.

Qu, L., Lyu, J., Li, W., Ma, D., & Fan, H. (2021). Features injected recurrent neural networks for short-term traffic speed prediction. Neurocomputing, 451, 290-304.

R. Jozefowicz, W. Zaremba, and I. Sutskever, "An Empirical Exploration of Recurrent Network Architectures," in Proceedings of the 32nd International Conference on Machine Learning (ICML-15), 2015, pp. 2342–2350.

R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio. "How to Construct Deep Recurrent Neural Networks." In: CoRR (2013).

Rahul K. Pathinarupothi , Vinayakumar R., Ekanath Rangan, Gopalakrishnan E., and Soman. K.P.(2017) " Instantaneous heart rate as a robust feature for sleep apnea severity detection using deep learning." In IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)pages 293-296.

Rajdev, A., Garan, H., & Biviano, A. (2012). Arrhythmias in pulmonary arterial hypertension. Progress in cardiovascular diseases, 55(2), 180-186.

Rajendra Acharya U., Hamildo Fujita, Shu Lih Oh, Yuki Hagiwara, Jen Hong Tan, and Muhammad Adam. (2017) "Application of deep convolutional neural network for automated detection of myocardial infarction using ECG signals." Information Sciences415: 190-198.

Rajendra Acharya U., Hamildo Fujita, Vidya K Sudarshan, Shu Lih Oh, Muhammad Adam, et al. (2016) " Automated detection and localization of myocardial infarction using electrocardiogram: A comparative study of different leads." Knowledge-Based Systems99: 146-156.

Rana, A.; Kim, K.K. ECG heartbeat classification using a single layer lstm model. In Proceedings of the 2019 International SoC Design Conference (ISOCC), Jeju, Korea, 6–9 October 2019; pp. 267–268.

Rana, A.; Kim, K.K. ECG heartbeat classification using a single layer lstm model. In Proceedings of the 2019 International SoC Design Conference (ISOCC), Jeju, Korea, 6–9 October 2019; pp. 267–268.

S. Haykin. Neural networks. Upper Saddle River, US: Prentice Hall International, 1999.

S. Kiranyaz, T. Ince, and M. Gabbouj, "Real-time patient-specific ecg classification by 1-d convolutional neural networks," IEEE Transactions on Biomedical Engineering, vol. 63, pp. 664 – 675, 2016.

S. M.Mathews, C. Kambhamettu, and K. E.Barner, "A novel application of deep learning for single-lead ecg classification," Computers in Biology and Medicine, vol. 99, pp. 53–62, 2018.

S. Savalia and V. Emamian. "Cardiac Arrhythmia Classification by Multi-Layer Perceptron and Convolution Neural Networks." In: Bioengineering 5.2 (2018), p. 35. issn: 2306-5354. doi: 10 . 3390 / bioengineering5020035. url: http : / / www . ncbi.nlm.nih.gov/pubmed/29734666http://www.pubmedcentral.nih.gov/ articlerender.fcgi?artid=PMC6027502http://www.mdpi.com/2306-5354/5/ 2/35.

Sarfraz, M.; Khan, A.A.; Li, F.F. Using independent component analysis to obtain feature space for reliable ECG Arrhythmia classification. In Proceedings of the 2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Belfast, UK, 2–5 November 2014; pp. 62–67.

Sharma L.N., Tripathy R.K., Samarendra Dandapat. (2015) "Multiscale energy and eigenspace approach to detection and localization of myocardial infarction." IEEE Transactions on biomedical engineering62(7):1827-37.

Singh, A. (2017). Anomaly detection for temporal data using long short-term memory (lstm).

Singh, N.; Singh, P. Cardiac arrhythmia classification using machine learning techniques. In Engineering Vibration, Communication and Information Processing; Springer: Berlin, Germany, 2019; pp. 469–480.

Sujadevi V.G., Soman K.P., and Vinayakumar R. (2017) "Real-time detection of atrial fibrillation from short time single lead ECG traces using recurrent neural networks." In The International Symposium on Intelligent Systems Technologies and Applications pages 212-221 Springer.

Swapna, G., Soman, K. P., & Vinayakumar, R. (2018). Automated detection of cardiac arrhythmia using deep learning techniques. Procedia computer science, 132, 1192-1201.

T. Mikolov, "Statistical Language Models Based on Neural Networks," PhD thesis, Brno University of Technology., 2012. [Online]. Available: http://www.fit.vutbr.cz/~imikolov/rnnlm/thesis.pdf

Tahsin Reasat and Celia Shahnaz. (2017) "Detection of inferior myocardial infarction using shallow convolutional neural networks."arXiv preprint arXiv:1710.01115.

Tomasic, I.; Trobec, R.; Lindén, M. Can the regression trees be used to model relation between ECG leads? In International Internet of Things Summit; Springer: Rome, Italy, 2015; pp. 467–472.

U. R. Acharya, H. Fujita, O. S. Lih, Y. Hagiwara, J. H. Tan, and M. Adam. "Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network." In: Information Sciences 405 (2017), pp. 81–90. issn: 0020-0255. doi: 10.1016/J.INS.2017.04.012. url: https: //www.sciencedirect.com/science/article/pii/S0020025517306539.

U. R. Acharya, H. Fujita, S. L. Oh, U.Raghavendra, J. H. Tan, M. Adam, A. Gertych, and Y. Hagiwara, "Automated identification of shockable and non-shockable life-threatening ventricular arrhythmias using convolutional neural network," Future Generation Computer Systems, vol. 89, pp. 952–959, 2018.

U. R. Acharya, H. Fujita, S. L. Oh, Y. Hagiwara, J. H. Tan, and M. Adam, "Automated detection of arrhythmias using different intervals of tachycardia ecg segments with convolutional neural network," Informstion Science, vol. 405, pp. 81–90, 2017.

Wu, M.; Lu, Y.; Yang, W.; Wong, S.Y. A study on arrhythmia via ECG signal classification using the convolutional neural network. Front. Comput. Neurosci. 2021, 14, 564015.

X.-C. Cao, B. Yao, and B.-Q. Chen, "Atrial fibrillation detection using an improved multi-scale decomposition enhanced residual convolutional neural network," IEEE Access, vol. 7, pp. 89 152–89 161, 2019.

Y. LeCun and Y. Bengio. "The Handbook of Brain Theory and Neural Networks." In: ed. by M. A. Arbib. Cambridge, MA, USA: MIT Press, 1998. Chap. Convolutional Networks for Images, Speech, and Time Series, pp. 255–258. isbn: 0-262-51102-9. url: http://dl.acm.org/citation.cfm?id=303568.303704.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition." In: Proceedings of the IEEE 86.11 (1998), pp. 2278–2324. issn: 0018-9219. doi: 10.1109/5.726791.

Z. Mao, W. X. Yao, and Y. Huang. "EEG-based biometric identification with deep learning." In: 2017 8th International IEEE/EMBS Conference on Neural Engineering (NER). 2017, pp. 609–612. doi: 10.1109/NER.2017.8008425.

Zhu, H.; Pan, Y.; Cheng, K.T.; Huan, R. A lightweight piecewise linear synthesis method for standard 12-lead ECG signals based on adaptive region segmentation. PLoS ONE 2018, 13, e0206170.