National College of Ireland

Electricity Consumption Anomaly Detection Model Using Deep Learning

MSc Research Project (MScDA_B)
MSc in Data Analytics

Chetan Banad Ramesh
Student ID: X17162939

School of Computing
National College of Ireland

Supervisor: Dr. Anu Sahni

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Chetan Banad Ramesh
Student ID:	X17162939
Programme:	MSc in Data Analytics **Year:** 2019
Module:	Research Project
Supervisor:	Dr. Anu Sahni
Submission Due Date:	12/08/2019
Project Title:	Electricity Consumption Anomaly Detection Model Using Deep Learning
Word Count:	7356 **Page Number:** 29

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Chetan Banad Ramesh

Date: 09/08/2019

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Contents

Electricity Consumption Anomaly Detection Model Using Deep Learning

Chetan Banad Ramesh
X17162939

Abstract

Electricity loss minimization is one of the major issues the service providers are facing, which needs to be addressed as soon as possible. Loss may be because of the technical or non-technical factor. The non-technical losses (NTLs) are the losses which are caused by human in form of illegal use of electricity, electricity theft and billing fraud etc. These losses should be minimized for the providers to be profitable. Smart meter plays a prominent role in monitoring energy theft and optimizing the usage of the electricity among the consumer. The smart meter records the consumption data of the consumer and when analysed give the usage pattern of the consumer which can be used to detect the anomaly in the usage pattern of the consumer. By detecting the anomaly, the provider can take necessary steps and minimize the loss. Many researchers have attempted to tackle this problem of detecting the anomaly in the consumption data using different time series model like ARIMA, ARMA etc to forecast the future consumption and thus detect anomaly. The researcher has also used deep learning models like LSTM, RNN etc. to predict the future consumption and detect anomaly. In this research we propose a novel computational effective anomaly detection model using the various version of GRU model.

Keywords: Non-technical losses (NTL), ARIMA (Auto-regression integrated moving average), ARMA (auto-regression and moving average), LSTM (long short-term memory), RNN (Recurrent neural networks).

1. Introduction

The rise in population and technological advancement has increased the necessity for the electricity. The upward trend for electricity is not going to diminish in the future but will surely rise with time. The electricity providers are putting utmost effort to fulfil these needs and to provide best service to the customers, but the service providers are facing losses in form of technical and non-technical forms. Technical losses are those caused by power outage, short circuit or by grid failure. Non-technical losses popularly known as the NTL's are those caused by humans in form of electricity theft, illegal usage of the electricity and avoiding the payment for the service etc. Recent studies in the field of electricity highlights that the electricity providers are losing lot of money due to these NTL's. The Northeast

Group, LLC conducted a research across 125 countries to estimate the NTL faced by different providers and it was estimated to be $96 billion dollars every year (Northeast-group.com, 2017).

Identifying the NTL's and mitigating the loss incurred by them is the major concern of many service providers. If the identification goes wrong the providers may have to incur more loss by losing a potential genuine customer. Differentiating a genuine customer versus fraud is a complex task. Many investigations have been carried out from the service providers to minimize the loss and to know the impact of the non-technical loss on the revenue. The investigation has been carried out in the areas of generation, transmission and distribution (Paul, 1987). The traditional approach to lower the NTL's which was followed earlier and it was as follows consumer with abnormal consumption behaviour was identified by monitored manually and then a physical inspection team is sent to the consumer house for further inspection. The consumption was considered fraudulent if the consumption changes drastically from the normal consumption in either way if it was increased or decreased. This approach required tedious man hour and money for detection of anomaly in the consumption and the approach had very low success rate of below 5% of physical inspection which added more burden to the providers revenue. The success rate was low because the traditional approach didn't record the consumption data or used it to its full potential (Guerrero et al., 2018).

With the advancement of time the traditional meters were replaced with the smart meter and advanced metering infrastructure and smart grid replaced the traditional grid. These smart grid and meter incorporated the ability to record and provide the real time consumption data of the consumers. By 2020, it is estimated that number of smart meters across the globe will exceed 800 million (Northeast-group.com, 2017). The data provided by the smart meter and smart grid should be analysed and utilised to know the consumption pattern of the consumer. By analysing the consumption pattern and the usage trend of the consumer, it will be possible to avoid energy wastage and anomaly detection. Solution from the emerging field of artificial intelligence is applied on the challenges faced by the providers which enables them with advanced monitoring capabilities. Anomaly detection is one of the major issues which impacts the company's revenue which can be solved by continuous monitoring of the consumption activities (Zanetti et al., 2019).

1.1. Research Question:

Can the anomalous consumption of electricity be identified using Gated Recurrent units (GRU) to lower the loss of the service providers?

To perform this, we propose implementing the Gated recurrent units. This model is used to identify the anomaly in the consumption pattern of the consumer, we use various type of GRU model to accurately identify the anomaly in the consumption data

The remaining section is divided into 5 parts: The first section gives an outline of the related work and the literature of the researcher who carried out the research in field of anomaly detection, theft-detection, Non-technical loss detection in the advanced meter infrastructure or smart grid. The next section is dedicated to the explanation of the methodology adopted to carry out this research, section 3 is dedicated to the evaluations and results, section 4 is dedicated to conclusion and the last section is dedicated for the future work.

2. Literature Review:

2.1. Forecasting Model:

In 2016, The authors (Yu, Mirowski and Ho, 2016) developed "A Sparse Coding Approach to Household Electricity Demand Forecasting in Smart Grids" the focus of the research was to forecast the household electricity load using sparse coding. The dataset used in this research has been considered from electric board of chattanooga from 2011 to 2013. The proposed model was able to achieve a MAPE of 20.64% which is much lower when compared to other statistical model like ARIMA, Holts-Winter for the same data.

2.2. Detection Model:

2.2.1. Statistical Model:

In 2014, The authors (Chou and Telaga, 2014) developed "Real-time detection of anomalous power consumption" the focus of the research was to identify the anomalous power consumption. The model had two phases: consumption prediction and anomaly detection. The researchers used ARIMA (Auto Regression Integrated Moving Average) hybrid with Neural nets to predict the consumption and used the two-sigma to predict the anomalies. The dataset used was gathered from 2012 to 2013 collected from the office of the researchers. The proposed hybrid model was able to achieve accuracy of 89.1% to 96.5% for 8-week window data and 86.8% to 94.72% for 4-week window data.

In 2016, The authors (Yijia and Hang, 2016) developed "Anomaly detection of power Consumption based on waveform feature recognition" the focus of the research was to identify the fraudulent power consumption. The authors proposed to use the feature extracted in waveform for anomaly detection. The author wanted to apply the vector space cosine similarity on the combination of power analysis and line loss data.

In 2016, The authors (Leong et al., 2016) developed "Detection of Anomalies in Activity Patterns of Lone Occupants from Electricity Usage Data" the focus of the

research was to detect anomaly in the lone user activity pattern. The data used for the research was obtained from the survey of three lone occupants for a month on hourly basis. The authors used fuzzy c-means clustering to cluster the scores which is obtained from the quantitative assessment of the consumption pattern of the occupant and the suspicion values of the occupant. The proposed model could identify the anomaly from the occupant electricity consumption pattern without need for external monitoring.
In 2017, The authors (Cui and Wang, 2017) introduced "Anomaly Detection and Visualization of School Electricity Consumption Data" the focus of the research was to detect anomaly in the building energy data. The researcher proposed a hybrid model of gaussian distribution and polynomial regression. The researcher used the school electricity consumption data for the model. In the research the variations throughout the year has been model from polynomial regression which is used to detect the anomaly in the weekend and anomaly in the weekday data has been detected from the gaussian distribution model. The hybrid model detects the anomaly in any day of the week.

In 2017, The authors (Fathnia, Fathnia and Javidi, 2017) proposed "Detection of Anomalies in Smart Meter Data: A Density-Based Approach" the focus of the research was to detect anomalies in the smart meter data. The researcher used the optics density-based approach to identify the abnormalities in the data. The researcher used the electricity consumption data provided by the London's metering data of January ,2013. The researcher used the Optics algorithms to detect the anomaly in the smart merter data and used the LOF index to enhance the performance of the model.

In 2017, The authors (Yip et al., 2017) proposed "Detection of energy theft and defective smart meters in smart grids using linear regression" the focus of the research was to study behaviour of consumer usage and evaluate the coefficient of anomaly to battle the energy theft by meter tampering and defective smart grids. The research used two linear regression-based algorithms. The researcher designed two algorithms LR-ETDM and CVLR-ETDM. The researcher found that the LR-ETDM was unable to identify the theft or faulty meter when there was a inconsistency in the energy leakage, hence they came up with a better model called CVLR-ETDM.

In 2017, The authors (Villar-Rodriguez et al., 2017) proposed "Detection of non-technical losses in smart meter data based on load curve profiling and time series analysis" the focus of the research was to detection of smart meter non-technical loss. The researcher used the data from the Spanish utility. The researcher used two different distance-based learning algorithms (LOF and LSA) which uses an inner classification model. The researcher used clustering to identify the non-technical losses.

In 2017, The authors (Singh, Bose and Joshi, 2017) developed "PCA based Electricity Theft Detection in Advanced Metering Infrastructure" the focus of the research was to detect electricity theft. The researcher used the principle component analysis (PCA) to convert the higher-dimensional data into lower-dimensional data. The researcher considered 5000 Irish home data from Sustainable Energy Authority of Ireland (SEAI) from 2009 to 2011. The proposed model can identify the normal and anomalous behaviour of the consumer.

In 2018, The authors (Qiu, Tu and Zhang, 2018) proposed "Anomaly Detection for Power Consumption Patterns in Electricity Early Warning System" the focus of the research was to detect anomaly in the consumption pattern of the user. The researcher proposed a novel approach called anomaly detection based on the log analysis (ADLA). Researcher first extract the different characteristics of the user consumption pattern and map that to two-dimensional plane using PCA. The model uses grid processing, principle component analysis and outlier calculation to detect the anomaly.

sIn 2018, The authors (Viegas, Esteves and Vieira, 2018) proposed "Clustering-based novelty detection for identification of non-technical losses" the focus of the research was to reduce the non-technical losses. The researchers used Gustafson-Kessel fuzzy clustering algorithm to reduce the non-technical losses. The proposed model works on smart meter consumption data which are in high resolution. The data is reduced into lower dimension and the consumption point is created using clustering algorithm on the data which has no NTL's. The proposed model outperformed SVM with a accuracy of 74.1% and the positive rate of 63.6 %.

In 2018, The authors (Yeckle and Tang, 2018) proposed "Detection of Electricity Theft in Customer Consumption using Outlier Detection Algorithms" the focus of the research was to detect theft using customer consumption data. The researcher used outlier detection algorithms to improve the security of the AMI. The proposed model uses 7 outlier detection algorithm and the k-means algorithm for the pre-processing. The researcher used electricity usages data of the Irish homes from 2009 to 2010. The researcher found that AUC performance increased when the number of metering reading reduced.

In 2019, The authors (Wang et al., 2019) proposed "Power Consumption Predicting and Anomaly Detection Based on Long Short-Term Memory Neural Network" the focus of the research was to predict the power consumption and detect anomaly based on the predicted values. The researchers used the long short-term memory accurately predict the consumption and anomaly detection. The proposed model shows a significant improvement in the accuracy than the ARIMA model with a 22% decrease in forecasting error.

2.2.2. Machine Learning Model:

In 2013, the authors (Jokar, Arianpoo and Leung, 2013) developed a "Intrusion Detection in Advanced Metering Infrastructure Based on Consumption Pattern" the focus of the research was to detect intrusion in the advanced metering systems (AMI). The researchers used Support vector machine algorithm on the dataset recorded by the residential energy consumption survey (RECS). The researchers were successful in detecting different types of activities which are malicious in AMI and the model was evaluated by using synthetic dataset.

In 2013, the authors (Fontugne et al., 2013) researched on "Mining Anomalous Electricity Consumption Using Ensemble Empirical Mode Decomposition" the focus of the research was to identify the abnormal device usage using ensemble empirical mode decomposition (E-EMD) which is used to estimate the correlation between the intrinsic inter-device. The data for the research has been collected from engineering building in university of Tokyo in 2011. The proposed model was good at analysing non-stationary data and it overcome the shortcomings of the classical empirical mode decomposition (EMD) models.

In 2015, The authors (Cody, Ford and Siraj, 2015) developed "Decision Tree Learning for Fraud Detection in Consumer Energy Consumption" the focus of the research was to detect anomalies in consumer usage pattern. The authors used M5P decision trees to profile energy consumption `of normal behaviour which are later used to detect fraudulent activity. The dataset used for the research are considered from 5000 residential and 650 business smart meter from 2009-2011. The proposed model was able to accurately predict the activity to be normal or fraudulent using historical data.

In 2016, The authors (Valdes, Macwan and Backes, 2016) developed "Anomaly Detection in Electrical Substation Circuits via Unsupervised Machine Learning" the focus of the research was to detect anomaly in the electricity circuits. The authors proposed a hybrid cyber-physical systems intrusion detection system (CPS-IDS). In this research the authors proposed the use of machine learning models to detect anomalous conditions.

In 2016, The authors (Jokar, Arianpoo and Leung, 2016) introduced "Electricity Theft Detection in AMI Using Customers' Consumption Patterns" the focus of the research was to detect electricity theft. The authors proposed an energy theft detector-based consumption pattern based (CPBETD) which can identify the consumer malicious and normal consumption pattern. The researcher used 5000 consumer data. The model used the support vector machine algorithms along with silhouette plots to identify the anomaly. The proposed model had the most less false positive rate of 11% when compared to other models.

In 2018, The authors (Aydin and Gungor, 2018) developed "A Novel Feature Design and Stacking Approach for Non-Technical Electricity Loss Detection" the focus of the research was to identify the non-technical loss in the electricity. The researcher used dataset provided by BEDAS in turkey which consists of 157 and 144 shopping malls data. The researcher used the stacking model of ensemble technique with the reduced set of parameters to boost the accuracy. The researcher used one-class SVM and the Clustering techniques.

In 2018, The authors (Zhang et al., 2018) proposed "Electricity Theft Detection Using Generative Models" the focus of the research was to detect electricity theft. The researcher proposed a semi-supervised generative and gaussian mixture model with the human input considered as indicators. The researcher used the data from commission for energy regulation (CER) project of smart metering from 2009 to 2010. The Proposed model S2G2M2 outperforms the one-class SVM and autoencoders in accurately detecting the electricity theft.

In 2018, The authors (Singh, Bose and Joshi, 2018) proposed "Entropy-based electricity theft detection in AMI network" the focus of the research was to detect electricity theft. The researcher proposed an entropy-based electricity theft detection model. The proposed model can track the consumption variation dynamically and the distance between consumption variation of probability distribution is computed. The proposed EBETD model outperforms the SVM Models.

In 2019, the authors (Buzau et al., 2019) proposed "Detection of Non-Technical Losses Using Smart Meter Data and Supervised Learning" the focus of the research was to detect non-technical loss in the smart grid. The researcher proposed a supervised model to identify the non-technical loss. The researcher used the XGBoost algorithm to identify the non-technical losses. The researcher used the data of the customer who were at least inspected once. The proposed model outperformed the other model with AUC score of 0.91 and precision approximately equal to 21%.

2.2.3. Deep Learning Model:

In 2014, The authors (Ford, Siraj and Eberle, 2014) developed "Smart grid energy fraud detection using artificial neural networks" the focus of the research was energy fraud detection. The author proposed a novel approach of using artificial neural network model on the smart meter data. The data used in the research were collected between 2009-2011 from 5000 residential and 600 business buildings. The proposed model outperformed the existing model at that time.
In 2015, The authors (Yuan and Jia, 2015) developed "A Distributed Anomaly Detection Method of Operation Energy Consumption Using Smart Meter Data" the focus of the research was operation anomaly detection. The authors used

stacked sparse autoencoders to get high level representation of the data and used softmax classifier to detect anomaly. The researcher gathered data from operation energy management systems (OEMS) from 2011 to 2013. The proposed model was able to achieve 95.96% accuracy.

In 2017, The authors (Bhattacharya and Sinha, 2017) developed "Intelligent Fault Analysis in Electrical Power Grids" the focus of the research was to detect the fault in the power grid. The research used PSS/E siemens software to create conditions like generator output fluctuations, faults and load fluctuation and the data is fed into the different classifiers like SVM and LSTM. The researcher using the proposed mode were able to forecast the maximum voltage using the historic fault data and find the location which is at fault.

In 2018, The authors (Zheng et al., 2018) proposed "Wide and Deep Convolutional Neural Networks for Electricity-Theft Detection to Secure Smart Grids" the focus of the research was to detect theft of electricity in smart grid. The researcher proposed a deep and wide convolution neural network model to address the shortcoming of the traditional model which used 1-dimensional data and failed to capture consumption periodicity. The model was able to capture the non-periodicity and periodicity of the 2-dimensional consumption data using deep neural network. The model was able to capture the 1-dimensional consumption feature using wide neural network.

In 2019, The authors (Fenza, Gallo and Loia, 2019) proposed "Drift-Aware Methodology for Anomaly Detection in Smart Grid" the focus of the research was to detect anomaly in the smart grid using the consumer behaviour change. The researchers proposed a model with k-means and LSTM model which can accurately predict the anomaly. The researchers used the data from UCI repository from 2011 to 2014. The model is trained using different profiles so that it can accurately predict the anomaly.

2.3. Other Models:

In 2017, The authors (Cardenas et al., 2012) developed "A Game Theory Model for Electricity Theft Detection and Privacy-Aware Control in AMI Systems" the focus of the research was to detect anomaly in electricity consumption and to forecast the load. The proposed model had two goals: first is to detect theft in electricity, this is designed as a game electricity theft and electric utility. Second, was to tackle the privacy-preserving demand response. A unified cost model was used to tackle both the problem.

In 2018, The authors (Yip et al., 2018) proposed "Detection of Energy Theft and Metering Defects in Advanced Metering Infrastructure Using Analytics" the focus of the research was to identify electricity theft. The researchers used linear based

regression rig to accurately identify the theft and faulty meter. The rig can identify the location of the theft and fault meter.

In 2018, The authors (Zhang and Wang, 2018) proposed “Multi-feature Fusion based Anomaly Electro-Data Detection in Smart Grid” the focus of the research was to detect anomaly in the smart grid. The researcher proposed a multi-feature fusion detection algorithm. The researcher got the distribution of multi-feature data by pre-processed the electricity data. Then the data is fed into D-S evidence theory based multi-feature fusion model to accurately detect the anomaly.

3. Research Methodology

This section is dedicated and emphasized about the methods, models and assumption considered in the detection to be successful. It is also dedicated for describing the framework which has been considered for this research. This research follows KDD (Knowledge Discovery Database) approach which has 5 phases. The framework helps us to systematically approach the problem and it serves us as a guideline for the steps to be followed in order to successfully solve the problem. This framework inclines towards the knowledge discovery in databases and it gives a general idea about knowledge extraction from the data and teaches about the machine learning application. The framework gives a general overview of the steps to be followed while successfully implementing a solution from data collection to the model building and evaluating it. The below figure represents the anomaly detection in the electricity consumption which is embedded with the KDD outline.

As we can see in the figure below that the data contains various information about the household consumption of different household recorded in kilowatts, Time of consumption, CACI Acorn Group and the house number. We are trying to detect anomaly by considering the data of one household at a time, hence we considered the data separately and integrate the other feature of date into the data. In the third stage we pre-process the data to be of desired shape and we split the data into training, testing and validation dataset. Then we apply suitable model on the dataset to get the desired results and achieving the desired trend, patterns with which we can understand the solution better. In the final stage we have the solution with which the service providers can detect anomaly beforehand and take necessary steps to avoid it.

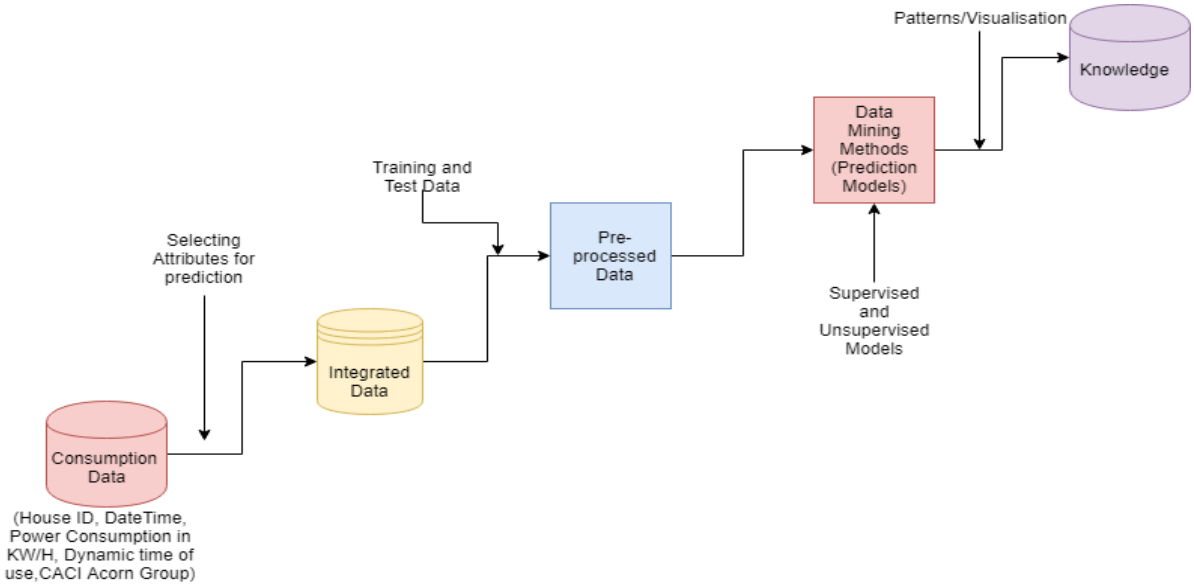

Figure 1: KDD Outline of the anomaly detection process.

This research also involves making choices in selecting the best set of models and techniques to be used on the data to get the best results. The data was considered from the London datastore which has the information about household electricity consumption data recorded between 2011 to 2014 by UK Power Networks. Data has the information about the consumption of electricity by the household in Kilowatts which is recorded every half hour interval by the smart meter of the consumer who participated in the Low Carbon London Project. The data has unique identifier for every household, Date and time of the moment when the consumption was recorded, CACI Acorn group the consumer belong and the standard energy price metric the customer belong. We have considered household consumption of one household for this study. The Figure represent the process flow of this research.

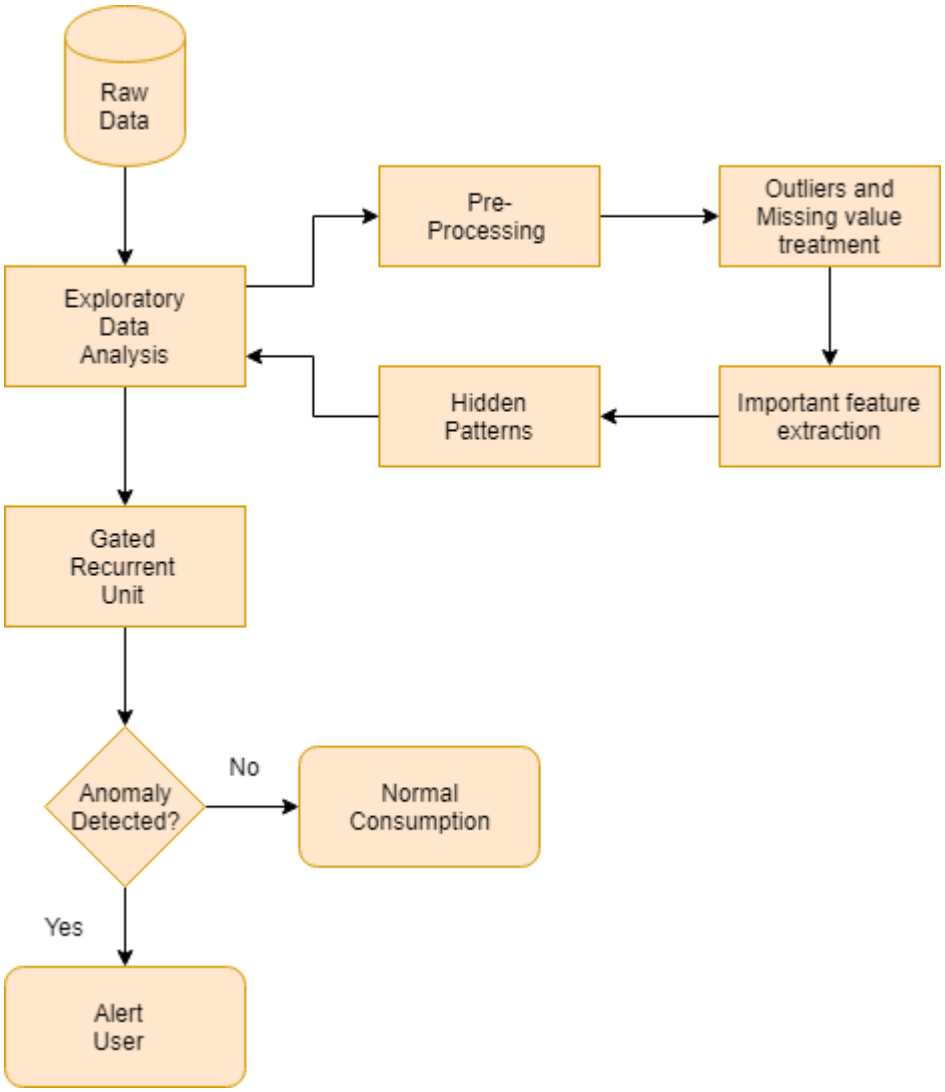

Figure 2: Flow of this research

In general, the aim of the research is to detect the anomaly in the consumption data using the deep learning models. In order to achieve this, we used different types of deep learning model of the given dataset. First, a simple model was executed on the dataset to set the benchmark for the deep learning model. This model considered the assumption that the data is a continuous time series model, hence the electricity consumption of tomorrow will be close to today's consumption. Thus, it is considered that the consumption in the next 24 hours will be equal to consumption right now. Once the benchmark is set then different deep learning model is used on the dataset and compared with this benchmark for the performance of the of those models.

3.1. Exploratory Data Analysis:

This stage of the research is about the collection of data from the data storage of the London datastore hoisted in the website: www.data.london.gov.uk. The dataset set consists of the consumption of electricity data, unique household number, date and time of the consumption recording and the CACI Acorn group the household belong

to. For this research we have considered only the date and time of consumption and the Power consumption which is recorded in the Kilowatt. The research started with choosing the best algorithm whether to choose the traditional time series model like ARIMA, SARIMA, Holts-Winter etc or to choose deep learning model like GRU, LSTM etc because the dataset being a continuous time series data. Seeing the complexity of the data deep learning model were chosen for the further research. Once the model was finalized then the data was processed with many pre-processing steps and feature extraction which has been briefed below:

Step 1: Collection of data from the UK government repository in CSV format.

Step 2: In this research only one household data was considered for further analysis. The unwanted variables were discarded from the final data.

Step 3: Pre-Processing:
lubridate () function was used to bring the date and time variable to the desired date format.
The NULL was were replaced with the average value of the time series, because the time series needs to be continuous with no NULL values or missing timestamp in the series.

Step 4: Checking the need for the vectorization (vectorization was not done because all data values are numerical). Checking for the normalization as the data varies little the normalisation was not necessary.

Step 5: Feature Extraction/ Engineering:
Many feature were extracted from the date and time column from the original dataset like date, month, year, day of the week, day of the year, hour, minutes, time of the day(morning, afternoon, evening, night, midnight), weekday or weekend, holiday or not.

Step 6: We try to visualise the data in hand in order to get a better understand of the data. We try to plot the data using ggplot2 library and ggplot function. The figure 3 represents the power consumption of a household during entire study. The figure 4 represents the power consumption of a household during first 20 days of the study. The power consumption is recorded every half hour interval.

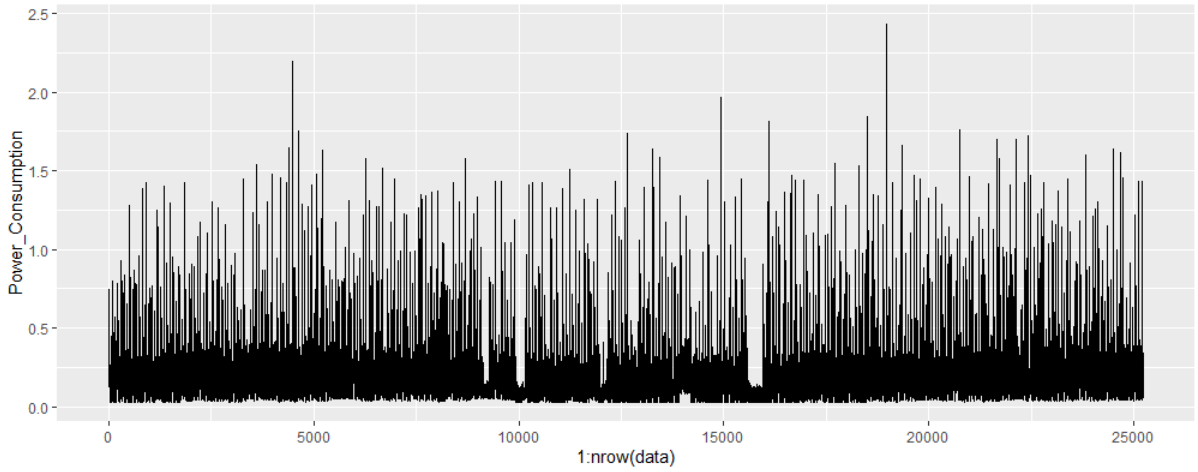

Figure 3:Power Consumption of household from 2012 to 2014 which is recorded at every half hour interval

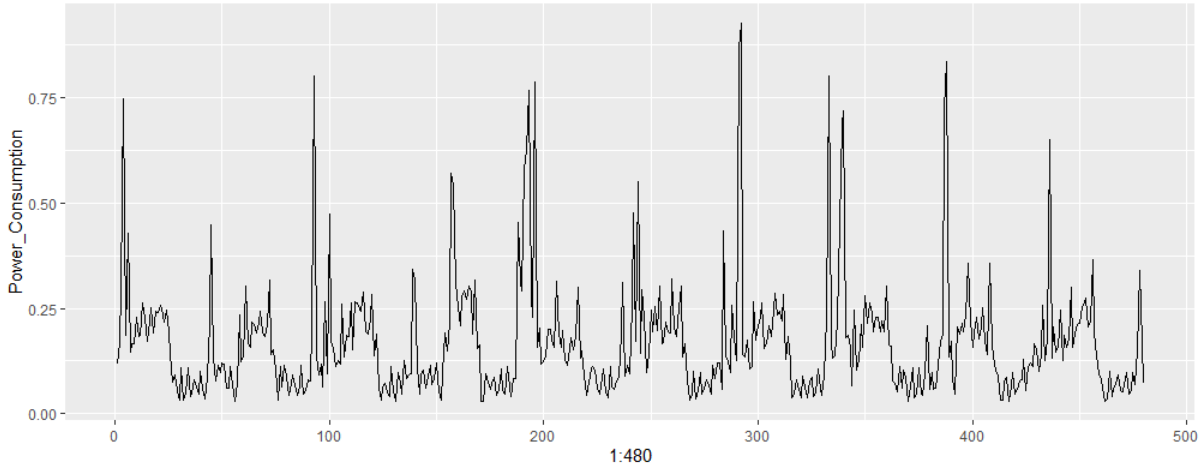

Figure 4:Power Consumption of a household in the first 20 days which is recorded at half hour interval

Step 7: The data is converted into matrix format because the model takes input in the matrix format only using matrix () function.
Step 8: After the pre-processing and feature extraction the data is split into three parts: Training dataset (~48%), Testing dataset (~28%) and Validation dataset (~24%).

In this research the important aspect is to understand what is meant by anomaly. Anomaly is also be referred as an outlier. If the behaviour of a person or a thing deviates from its normal behaviour, then it is called as an anomaly (Abraham and Chuang, 1989). Anomalies can exist in different data in different forms, these anomalies are identified using different information in various research and domains E.g.: If a computer network is getting a huge load which is not observed normally then it means that somebody has hacked the network and it is considered to be an anomaly in the network domain (Kumar, 2005).

3.2. Deep learning Model:

3.2.1. Gated Recurrent Unit:

Gated Recurrent Unit which was proposed by the (Cho et al., 2014) makes the recurrent unit capture the dependencies adaptively which are in different scale of time. Gated recurrent unit and the long short-term memory both works on the same principles, but the gated recurrent unit is more streamlined than the long short-term memory model. The gated recurrent unit require cheap computational power when compared to long short-term memory. The gated recurrent unit doesn't have representational power as much as the long short-term memory.

3.2.2. Dropout in Gated Recurrent unit:

The author (Gal et al., 2016) found out the correct way to dropout use in the recurrent network. The pattern of dropout should be same for all the timesteps (same dropout mask) instead of different pattern of dropout at different timestep. If we want to regularize the gated recurrent unit and long short-term memory model, then a dropout should be added to the inner most layer of recurrent network. If we use the same dropout at all the timestep then the ability of the network to propagate properly across the time will be boosted. A random dropout at a different timestep will harm the learning and the propagation process.

4. Design Specification

Based on the research of many researchers that has been reviewed, it looks like very few researchers have tried to tackle the anomaly detection in electricity consumption model using a computational effective gated recurrent unit model. The complexity and uncertainties in the electricity consumption pattern makes it difficult for the hypothesis formulation. Going through all the research and considered them, we have come up with a hypothesis which is presented below:

Step 1:
The Electricity consumption trend of the household is observed in terms of whether it is increasing or decreasing and then a sample of 20 days data (480 data points) are formed.

Step 2:
These 20 days data are used to forecast the electricity consumption of the consumer in the coming 24 hours iteratively until be reach the end of the data set.

Step3:

The forecast consumption data are compared with the actual consumption data (predicted-actual). if the error is less then it means that there is no anomaly in the data. If the error is more, we check if the forecasted data is within the 2 std dev of the past 20 days data distribution. If the forecasted point doesn't fall within this range, then it is considered as an anomaly.

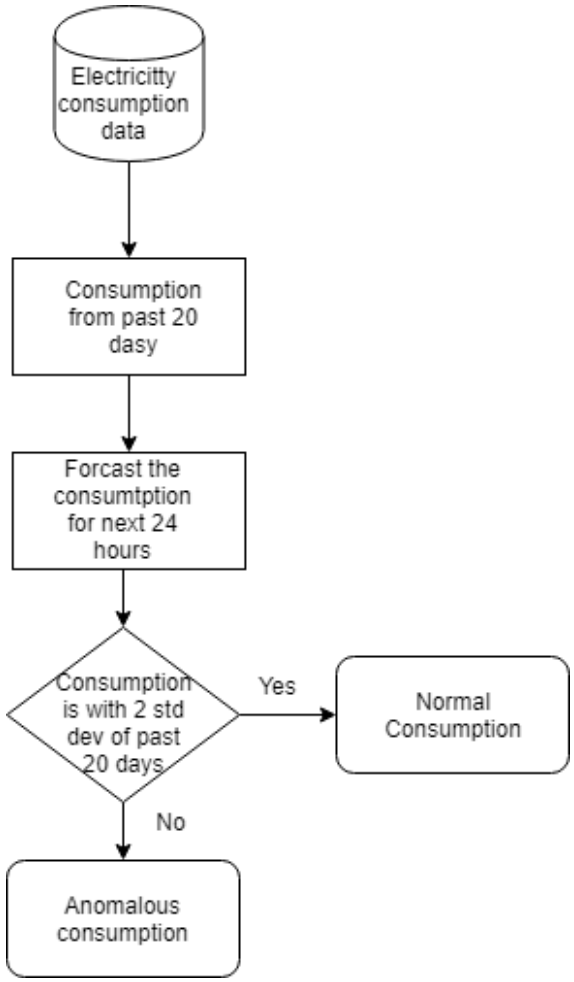

Figure 5: Hypothesis for the Anomaly detection

5. Implementation

Figure 6 represents the fully working model that will be implemented in this research. The data is collected from the datastore of the UK government. The datastore contains the information of the several households from the 2008 to 2014. The dataset contains the information of the electricity consumption, which was downloaded from the website in the form of zip file. The file was unzipped which resulted in numerous csv files containing information of several household electricity consumption information. The data was processed using the Microsoft excel and the information of one household was taken into consideration for this research.

The household information which was selected for the research has the electricity consumption information of that house from 2012 to 2014 which was recorded every half an hour interval. The data had many information regarding the house unique ID, CACI acorn information, date and time of the consumption recording, rates at which the customer were charged for the usage of the electricity. The information of only consumption and the date and time of the consumption were considered for this research. Many other features were extracted from the date and time variable like the day of the week, day of the year, day of the month, month, year, hour, minute, which part of the day the consumption happened, whether the day was weekday or weekend, weather it was a holiday on that day.

Once the feature engineering was done the data was checked for the missing values. If any missing values were found, they were replaced with the average consumption of the dataset. The dataset was then loaded into the RStudio for further analysis. The data was visualized to see the pattern. The entire consumption was plotted to the see the trend in the data, because we had lot of data points, we couldn't clearly see the trend, so the first 20 days consumption was plotted to see the data more clearly. Once the data was confirmed to have an anomalous consumption in it. The model was chosen to find the anomaly in the data and to predict the anomaly in the coming future. We had two option one to go with traditional model like ARIMA, ARMA, SARIMA, Holts-Winter etc. to forecast the future consumption and thus predict the anomaly or to go for more complex deep learning approach like RNN, LSTM and GRU etc. to predict the future consumption and to find the anomaly. The data in hand had many data points and it was recorded at a granular level, so we decided we go with deep learning model. The input of the deep learning model must be in a specific shape and type in order to it to work. So, the data was converted into matrix format and not normalization and vectorization has been carried out on the dataset because the data variables were numeric and the variation in the values between the variable were little.

The Processed data was split into 3 datasets: Training dataset, Testing dataset and validation dataset. Before using the deep learning model, a simple model was run to set the benchmark so that the model could compare in terms of MAE values. The model was incorporated with the assumption that the consumption of today will be same as the consumption which was 24 hours before. It also assumed that the consumption in the next 24 hours will be same as the present consumption. The base model was run, the benchmark was set. Then before going to the complex model a basic model with flatten input data was ran to check the MAE values. Then a recurrent baseline model was processed, the MAE values was compared with benchmark and the base model. Then the dropout was added to the model to check whether the MAE values of the model minimizes or not. The MAE values were again compared with the other model. Then the we used a stacked complex GRU model to check if the MAE values was increased and it was compared with the other models. The above models were travelling in one direction from start to end of the dataset by taking input of 20 days information and trying to predict the anomaly in the future 24 hours. The next model was a bidirectional GRU which takes the

input shuffle the sequence or reverse it to complete change the representation of the input fed into the model and thus minimizing the MAE values.

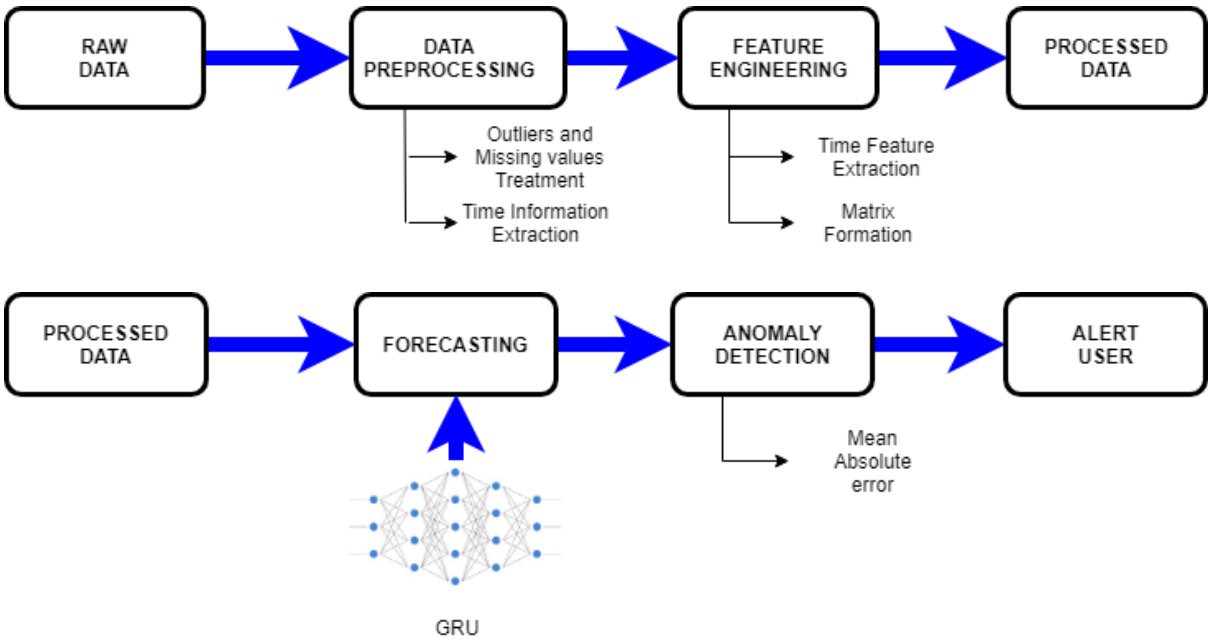

Figure 6: Anomaly Detection Architecture

6. Evaluation

We have chosen mean absolute error as error metric to measure the model performance in our research. Mean absolute error without considering the direction of the data measure the average set of predictions error magnitude.

$$\mathrm{MAE} = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

Figure 7:Mean Absolute Error

6.1.Benchmark model:

In order to set the benchmark for the error of the model to compare we executed a benchmark model which take in the small sample of input from the validation dataset got from the validation dataset generator function. This model has the assumption that the consumption of present is same as the consumption of 24 hours before and the consumption of next 24 hours will be same as the consumption of present. When we

ran the benchmark model and we tried to calculate the mean absolute error manually by subtracting the predicting with the actual then average it out to give the MAE value of 0.092 which will be the benchmark for the next models.

```
> library(keras)
> val_steps <- (18000 - 12001 - lookback) / batch_size
> test_steps <- (nrow(data) - 18001 - lookback) / batch_size
> library(keras)
> evaluate_first_method <- function() {
+   batch_maes <- c()
+   for (step in 1:val_steps) {
+     c(samples, targets) %<-% val_gen()
+     preds <- samples[,dim(samples)[[2]],2]
+     mae <- mean(abs(preds - targets))
+     batch_maes <- c(batch_maes, mae)
+   }
+   print(mean(batch_maes))
+ }
> evaluate_first_method()
[1] 0.09188633
```

6.2. Machine learning model:

Before going to the complex models, we tried simple basic model which takes the input in the flatten manner as in 2-dimensional dataset and tries to predict the future consumption while detecting the anomaly in the sequence. This model has 3 layers: first layer which accepts the 3-dimensional data and converts it into 2-dimensional data and feds that data into the second layer which has 32 neural network units to detect the anomaly and it pass the information to the third output layer which produces the output. This model had MAE value of 0.0986 for the training data and 18.337 for the validation data in the last run with is not performing good compared to the benchmark model. The model got MAE value of 36.1075 on the testing data.

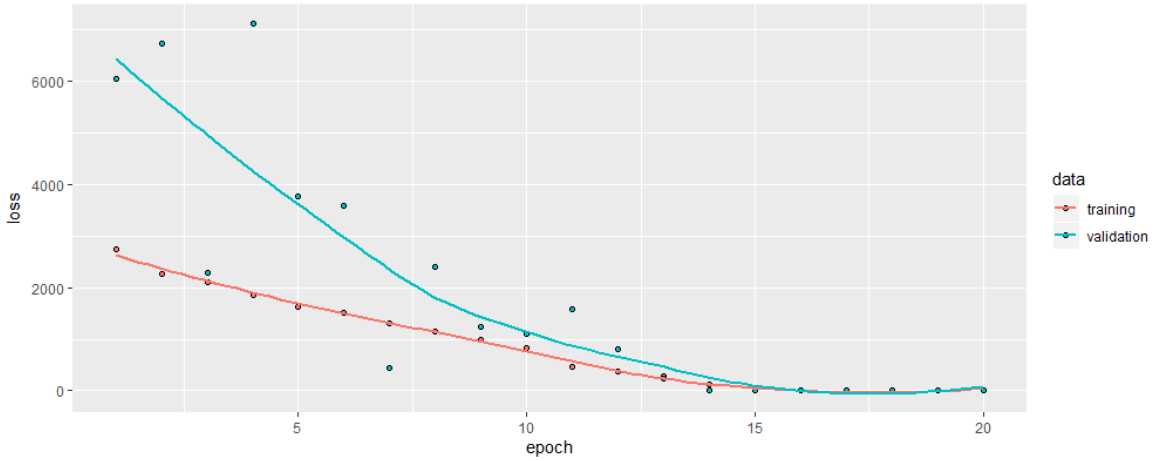

Figure 8: MAE Values for different epoch on training and testing data

```
> results1<-model %>% evaluate(Test[[1]],Test[[2]])
48/48 [==============================] - 0s 83us/sample - loss: 36.1075
```

Figure 9:MAE Value of the model on testing data

6.3. GRU Model with dropout:

This is the first deep learning model which is used on the dataset. This model has 2 layers in it the first layer consists of 32 neural networks which takes in the input in 3-dimensional and detects the anomaly and passes it to the next layer which has one neural net which output the results. This model has a dropout rate of 0.4 and recurrent dropout of 0.4 added to the above model which yields in the MAE value to be 0.1007 for the training dataset and 0.0844 for the validation dataset. The model got a MAE value of 0.1001 for the testing dataset.

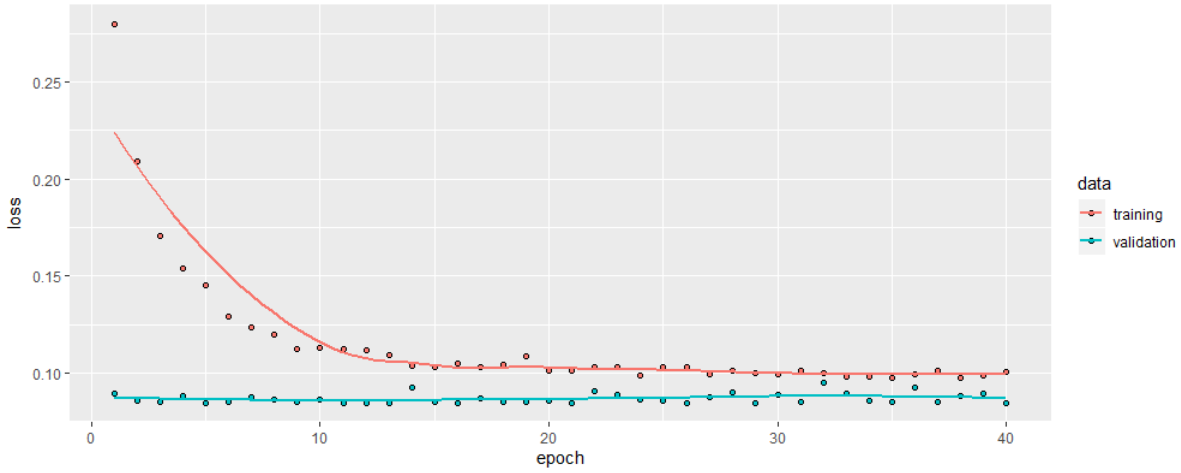

Figure 10:MAE Values for different epoch on training and testing data

```
> results3<-model %>% evaluate(Test[[1]],Test[[2]])
48/48 [==============================] - 0s 773us/sample - loss: 0.1001
```

Figure 11:MAE Value of the model on testing data

6.4. Stacked GRU Model:

This Model has two GRU layer stacked connected to one another to minimise the MAPE value.

The first layer has 64 neural networks with a recurrent dropout of 0.5 and dropout rate of 0.2 connected to the second layer which has the recurrent dropout of 0.5 with 64 neural network units. This model has MAE value of 0.1007 for the training data and 0.0844 for the training data for the validation data in the last epoch. The model got MAE value of 0.0983 on the testing data.

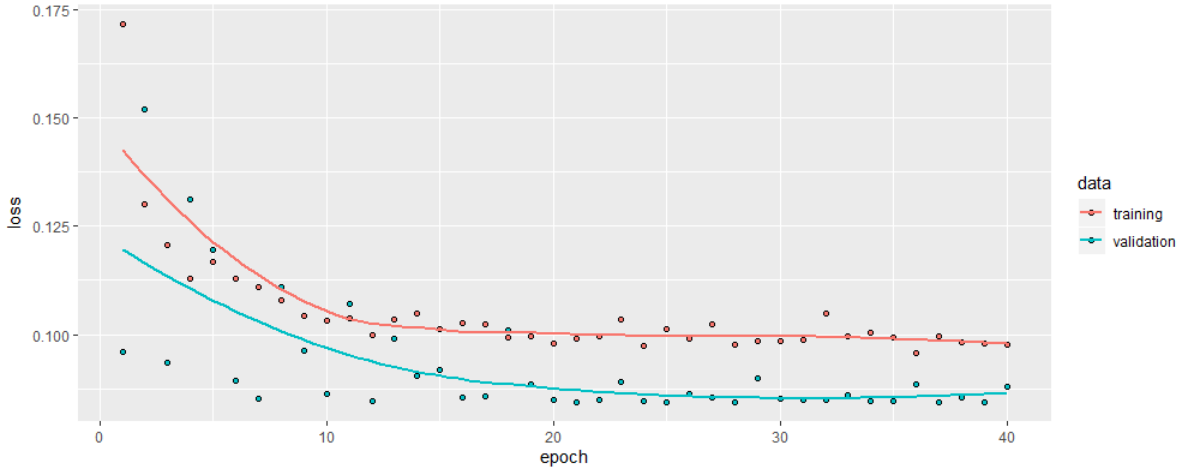

Figure 12:MAE Values for different epoch on training and testing data

```
> results4<-model %>% evaluate(Test[[1]],Test[[2]])
48/48 [==============================] - 0s 2ms/sample - loss: 0.0983
```

Figure 13:MAE Value of the model on testing data

6.5. GRU Model with Reverse data:

This model has layer of 32 units of neural network which takes in the input which is fed with the data which has been reversed. This layer is connected to the output layer which output the outputs. This model has MAE value of 0.0982 for the training data and 0.0953 for the training data for the validation data in the last epoch. The model achieved MAE value of 0.4039 on the testing data.

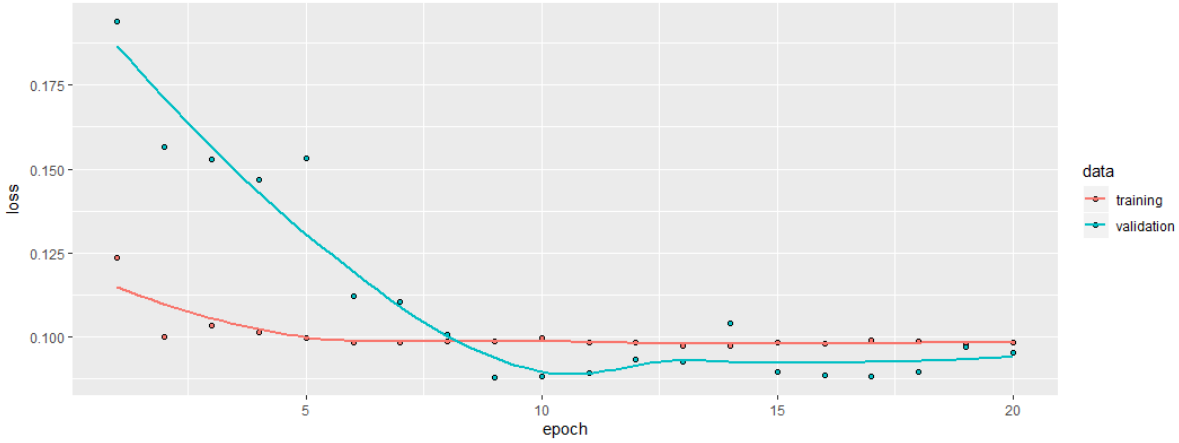

Figure 14:MAE Values for different epoch on training and testing data

```
> results6<-model %>% evaluate(Test[[1]],Test[[2]])
48/48 [==============================] - 0s 2ms/sample - loss: 0.4039
```

Figure 15:MAE Value of the model on testing data

6.6. Bidirectional GRU Model:

This model has layer of 32 units of neural network which takes in the input and shuffles the sequence or orders it backward to minimise the MAE value. This layer is connected to the output layer which is used to obtain the output. This model has MAE value of 0.0976 for the training data and 0.0976 for the training data for the validation data in the last epoch. The model's MAE value is 0.1003 on the testing data.

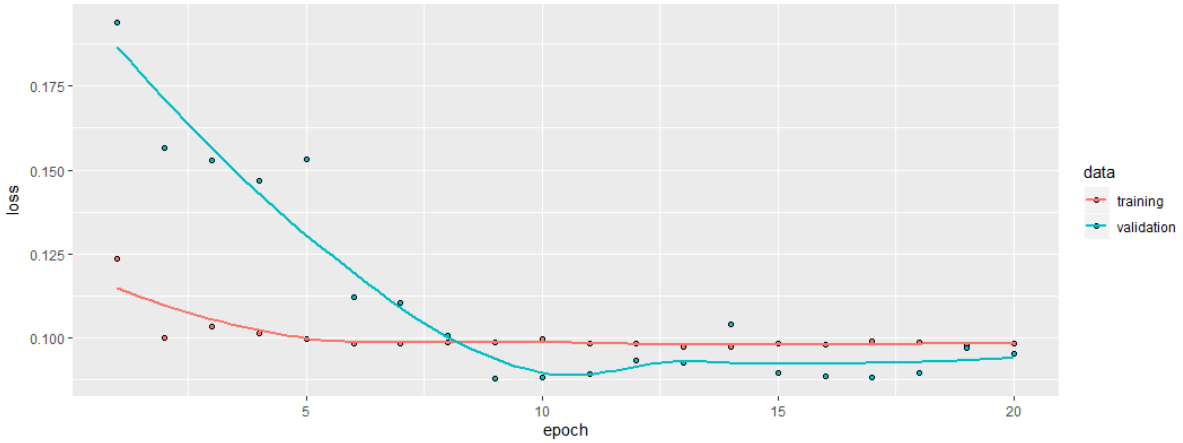

Figure 16:MAE Values for different epoch on training and testing data

```
> results7<-model %>% evaluate(Test[[1]],Test[[2]])
48/48 [==============================] - 0s 678us/sample - loss: 0.1003 - mean_squared_error: 0
.0313 - mean_absolute_error: 0.1003
```

Figure 17:MAE Value of the model on testing data

Below table represents the summary of all the models MAE values on training, testing and validation dataset:

Models	Training MAE	Validation MAE	Testing MAE
Base model	0.0986	18.337	36.1075
GRU Model	0.995	0.0911	0.112
GRU Model with dropout	0.1007	0.0844	0.1001
Stacked GRU Model	0.1007	0.0844	0.983
GRU Model With reversed data	0.0982	0.953	0.4039
Bidirectional GRU Model	0.0976	0.0976	0.1003

7. Discussion

Predicting the electricity consumption of the consumer on any given day is a very though process for the electricity providers as the consumption depends on lot of factors such as weather, time of the year, time of the day etc. Considering from the following study and the assumptions taken in the study while analysing the data seems to be easy task but trying to implement this methodology in the real time electricity data is bit challenging and might not work all the time. So, if the service provider wants to use this method to

detect the anomalies in the real time then the process must be automated and more sophisticated and computationally intensive system should be used. As this system needs to process a lot of information coming from different household at a time and detect the anomaly. In addition, in order to consider a household to be having anomalous consumption pattern more information about the house is necessary. The consumption may rise or fall depending on the number of persons in the house, if the person is using the property regularly or not etc.

As a part of implementation, we tried to extract as many features as possible to incorporate into the model to get better results, but because of limited resource and the short time duration for this resources we couldn't incorporate more feature about of the user behaviour into the model such as the number of consumer in the household, number of children in the household, age of the consumer, any pets in the household, how many times the household has guest in a year etc. This information will help to forecast the consumption better and thus help us to accurately predict the anomaly in the consumption. Due to lack of computationally intensive machine we restricted our resource to predicting anomaly in a single household at a time. The data used in the research is not a real-time data but was a recorded data.

8. Future Work

After having a deep insight on the research and related work in the field of the electricity consumption, the anomaly detection in the electricity consumption is in the niche phase. It can be improved with the upcoming technologies and methods to accurately identify the anomaly beforehand and take measure to stop it. In addition to various method and technologies to be implemented on the electricity data to accurately predict anomaly, various feature of the consumer like gender, age , number of person living in the house, it is a permanent house or a guest house, how they have children in the home etc. can be used to enhance the detection ability of model. The future researchers can use more sophisticated systems to detect anomaly in the real-time data and can detect anomaly across different household at a time.

9. Conclusion

From the research we can conclude that electricity consumption prediction and further drilling down to segregate the anomalous consumption pattern from normal consumption pattern is a model which is recommended for the service providers to correctly identify the anomaly from the normal consumption. Thus, from the above model we can say that the bidirectional GRU model perform well in detecting the anomaly with a MAE value of 0.0976 on training data, 0.0976 on validation data and 0.1003 on the testing data. Hence, we could say that the service providers can use the bidirectional GRU model to detect the anomalous consumption and stop it.

Acknowledgment:

I would like to devote my profound and sincere thanks to my mentor Dr. Anu Sahni who has tremendous knowledge and experience in teaching for her constant guidance and support which helped me for completing my research. She has always given her honest opinion, suggestion to me throughout my research and guided me towards the right approach whenever required.

I would also like to thank the National College of Ireland and be grateful for providing me with the necessary resources, faculty and knowledge. Finally, I would like to thank my family, wife and friends who constantly supported me and motivated me throughout my research duration. Without their motivation and support this accomplishment would have not been possible.

References

Abraham, B. and Chuang, A. (1989). Outlier Detection and Time Series Modeling. Technometrics, 31(2), pp.241-248.

Aydin, Z. and Gungor, V. (2018). A Novel Feature Design and Stacking Approach for Non-Technical Electricity Loss Detection. 2018 IEEE Innovative Smart Grid Technologies - Asia (ISGT Asia).

Bhattacharya, B. and Sinha, A. (2017). Intelligent Fault Analysis in Electrical Power Grids. 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI).
Buzau, M., Tejedor-Aguilera, J., Cruz-Romero, P. and Gomez-Exposito, A. (2019). Detection of Non-Technical Losses Using Smart Meter Data and Supervised Learning. IEEE Transactions on Smart Grid, 10(3), pp.2661-2670.

Cardenas, A., Amin, S., Schwartz, G., Dong, R. and Sastry, S. (2012). A game theory model for electricity theft detection and privacy-aware control in AMI systems. 2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton).
Cho, K., van Merrienboer, B., Bahdanau, D. and Bengio, Y. (2014). On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation.

Chou, J. and Telaga, A. (2014). Real-time detection of anomalous power consumption. Renewable and Sustainable Energy Reviews, 33, pp.400-411.

Cody, C., Ford, V. and Siraj, A. (2015). Decision Tree Learning for Fraud Detection in Consumer Energy Consumption. 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA).

Cui, W. and Wang, H. (2017). Anomaly detection and visualization of school electricity consumption data. 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA).

Fathnia, F., Fathnia, F. and Javidi, D. (2017). Detection of anomalies in smart meter data: A density-based approach. 2017 Smart Grid Conference (SGC).

Fenza, G., Gallo, M. and Loia, V. (2019). Drift-Aware Methodology for Anomaly Detection in Smart Grid. IEEE Access, 7, pp.9645-9657.

Fontugne, R., Tremblay, N., Borgnat, P., Flandrin, P. and Esaki, H. (2013). Mining anomalous electricity consumption using Ensemble Empirical Mode Decomposition. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing.

Ford, V., Siraj, A. and Eberle, W. (2014). Smart grid energy fraud detection using artificial neural networks. 2014 IEEE Symposium on Computational Intelligence Applications in Smart Grid (CIASG).

Gal, Y. and Ghahramani, Z., 2016, June. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In international conference on machine learning (pp. 1050-1059).

Guerrero, J., Monedero, I., Biscarri, F., Biscarri, J., Millan, R. and Leon, C. (2018). NonTechnical Losses Reduction by Improving the Inspections MAE values in a Power Utility. IEEE Transactions on Power Systems, 33(2), pp.1209-1218

Jokar, P., Arianpoo, N. and Leung, V. (2013). Intrusion detection in advanced metering infrastructure based on consumption pattern. 2013 IEEE International Conference on Communications (ICC).

Jokar, P., Arianpoo, N. and Leung, V. (2016). Electricity Theft Detection in AMI Using Customers' Consumption Patterns. IEEE Transactions on Smart Grid, 7(1), pp.216-226.

Kumar, V. (2005). Parallel and Distributed Computing for Cybersecurity. IEEE Distributed Systems Online, 6(10), pp.1-1.

Leong, K., Leung, C., Miao, C. and Chen, Y. (2016). Detection of anomalies in activity patterns of lone occupants from electricity usage data. 2016 IEEE Congress on Evolutionary Computation (CEC).

Northeast-group.com. (2017). Electricity Theft and Non-Technical Losses: Global Markets, Solutions, and Vendors. Brochure, 2017. [online] Available at: http://www.northeastgroup.com/reports/Brochure-Electricity%20Theft%20&%20Non-Technical%20Losses%20%20Northeast%20Group.pdf [Accessed 6 Aug . 2019].

Paul, C. (1987). System loss in a metropolitan utility network. Power Engineering Journal, 1(5), p.305.

Qiu, H., Tu, Y. and Zhang, Y. (2018). Anomaly detection for power consumption patterns in electricity early warning system. 2018 Tenth International Conference on Advanced Computational Intelligence (ICACI).

Singh, S., Bose, R. and Joshi, A. (2017). PCA based electricity theft detection in advanced metering infrastructure. 2017 7th International Conference on Power Systems (ICPS).

Singh, S., Bose, R. and Joshi, A. (2018). Entropy-based electricity theft detection in AMI network. IET Cyber-Physical Systems: Theory & Applications, 3(2), pp.99-105.

Valdes, A., Macwan, R. and Backes, M. (2016). Anomaly Detection in Electrical Substation Circuits via Unsupervised Machine Learning. 2016 IEEE 17th International Conference on Information Reuse and Integration (IRI).

Viegas, J., Esteves, P. and Vieira, S. (2018). Clustering-based novelty detection for identification of non-technical losses. International Journal of Electrical Power & Energy Systems, 101, pp.301-310.

Villar-Rodriguez, E., Del Ser, J., Oregi, I., Bilbao, M. and Gil-Lopez, S. (2017). Detection of non-technical losses in smart meter data based on load curve profiling and time series analysis. Energy, 137, pp.118-128.

Wang, X., Zhao, T., Liu, H. and He, R. (2019). Power Consumption Predicting and Anomaly Detection Based on Long Short-Term Memory Neural Network. 2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA).

Yeckle, J. and Tang, B. (2018). Detection of Electricity Theft in Customer Consumption Using Outlier Detection Algorithms. 2018 1st International Conference on Data Intelligence and Security (ICDIS).

Yijia, T. and Hang, G. (2016). Anomaly detection of power Consumption based on waveform feature recognition. 2016 11th International Conference on Computer Science & Education (ICCSE).

Yip, S., Wong, K., Hew, W., Gan, M., Phan, R. and Tan, S. (2017). Detection of energy theft and defective smart meters in smart grids using linear regression. International Journal of Electrical Power & Energy Systems, 91, pp.230-240.

Yip, S., Tan, C., Tan, W., Gan, M., Wong, K. and Phan, R. (2018). Detection of Energy Theft and Metering Defects in Advanced Metering Infrastructure Using Analytics. 2018 International Conference on Smart Grid and Clean Energy Technologies (ICSGCE).

Yu, C., Mirowski, P. and Ho, T. (2016). A Sparse Coding Approach to Household Electricity Demand Forecasting in Smart Grids. IEEE Transactions on Smart Grid, pp.1-11.

Yuan, Y. and Jia, K. (2015). A Distributed Anomaly Detection Method of Operation Energy Consumption Using Smart Meter Data. 2015 International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP).

Zanetti, M., Jamhour, E., Pellenz, M., Penna, M., Zambenedetti, V. and Chueiri, I. (2019). A Tunable Fraud Detection System for Advanced Metering Infrastructure Using Short-Lived Patterns. IEEE Transactions on Smart Grid, 10(1), pp.830-840.

Zhang, C. and Wang, F. (2018). Multi-feature Fusion Based Anomaly Electro-Data Detection in Smart Grid. 2018 15th International Symposium on Pervasive Systems, Algorithms and Networks (I-SPAN).

Zhang, Q., Zhang, M., Chen, T., Fan, J., Yang, Z. and Li, G. (2018). Electricity Theft Detection Using Generative Models. 2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI).

Zheng, Z., Yang, Y., Niu, X., Dai, H. and Zhou, Y. (2018). Wide and Deep Convolutional Neural Networks for Electricity-Theft Detection to Secure Smart Grids. IEEE Transactions on Industrial Informa tics, 14(4), pp.1606-1615.

Appendix:

```
> summary(model)
Model: "sequential_25"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
flatten_4 (Flatten)          (None, 4560)              0
_________________________________________________________________
dense_63 (Dense)             (None, 32)                145952
_________________________________________________________________
dense_64 (Dense)             (None, 1)                 33
=================================================================
Total params: 145,985
Trainable params: 145,985
Non-trainable params: 0
_________________________________________________________________
> # Get model configuration
> get_config(model)
{'name': 'sequential_25', 'layers': [{'class_name': 'Flatten', 'config': {'name': 'flatten_4', 'trainable':
 True, 'batch_input_shape': (None, 240, 19), 'dtype': 'float32', 'data_format': 'channels_last'}}, {'class_
name': 'Dense', 'config': {'name': 'dense_63', 'trainable': True, 'dtype': 'float32', 'units': 32, 'activat
ion': 'relu', 'use_bias': True, 'kernel_initializer': {'class_name': 'GlorotUniform', 'config': {'seed': No
ne, 'dtype': 'float32'}}, 'bias_initializer': {'class_name': 'Zeros', 'config': {'dtype': 'float32'}}, 'ker
nel_regularizer': None, 'bias_regularizer': None, 'activity_regularizer': None, 'kernel_constraint': None,
'bias_constraint': None}}, {'class_name': 'Dense', 'config': {'name': 'dense_64', 'trainable': True, 'dtype
': 'float32', 'units': 1, 'activation': 'linear', 'use_bias': True, 'kernel_initializer': {'class_name': 'G
lorotUniform', 'config': {'seed': None, 'dtype': 'float32'}}, 'bias_initializer': {'class_name': 'Zeros', '
config': {'dtype': 'float32'}}, 'kernel_regularizer': None, 'bias_regularizer': None, 'activity_regularizer
': None, 'kernel_constraint': None, 'bias_constraint': None}}]}
> # Get layer configuration
> get_layer(model, index = 1)
<tensorflow.python.keras.layers.core.Flatten>
> # List the model's layers
> model$layers
[[1]]
<tensorflow.python.keras.layers.core.Flatten>

[[2]]
<tensorflow.python.keras.layers.core.Dense>

[[3]]
<tensorflow.python.keras.layers.core.Dense>

> # List the input tensors
> model$inputs
[[1]]
Tensor("flatten_4_input:0", shape=(?, 240, 19), dtype=float32)

> # List the output tensors
> model$outputs
[[1]]
Tensor("dense_64/BiasAdd:0", shape=(?, 1), dtype=float32)
```

```
> summary(model)
Model: "sequential_26"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
gru_2 (GRU)                  (None, 32)                4992
_________________________________________________________________
dense_65 (Dense)             (None, 1)                 33
=================================================================
Total params: 5,025
Trainable params: 5,025
Non-trainable params: 0
_________________________________________________________________
>
> # Get model configuration
> get_config(model)
{'name': 'sequential_26', 'layers': [{'class_name': 'GRU', 'config': {'name': 'gru_2', 'trainable': True, '
batch_input_shape': (None, None, 19), 'dtype': 'float32', 'return_sequences': False, 'return_state': False,
  'go_backwards': False, 'stateful': False, 'unroll': False, 'time_major': False, 'units': 32, 'activation':
  'tanh', 'recurrent_activation': 'hard_sigmoid', 'use_bias': True, 'kernel_initializer': {'class_name': 'Gl
orotUniform', 'config': {'seed': None, 'dtype': 'float32'}}, 'recurrent_initializer': {'class_name': 'Ortho
gonal', 'config': {'gain': 1.0, 'seed': None, 'dtype': 'float32'}}, 'bias_initializer': {'class_name': 'Zer
os', 'config': {'dtype': 'float32'}}, 'kernel_regularizer': None, 'recurrent_regularizer': None, 'bias_regu
larizer': None, 'activity_regularizer': None, 'kernel_constraint': None, 'recurrent_constraint': None, 'bia
s_constraint': None, 'dropout': 0.0, 'recurrent_dropout': 0.0, 'implementation': 1, 'reset_after': False}},
 {'class_name': 'Dense', 'config': {'name': 'dense_65', 'trainable': True, 'dtype': 'float32', 'units': 1,
'activation': 'linear', 'use_bias': True, 'kernel_initializer': {'class_name': 'GlorotUniform', 'config': {
'seed': None, 'dtype': 'float32'}}, 'bias_initializer': {'class_name': 'Zeros', 'config': {'dtype': 'float3
2'}}, 'kernel_regularizer': None, 'bias_regularizer': None, 'activity_regularizer': None, 'kernel_constrain
t': None, 'bias_constraint': None}}]}
>
> # Get layer configuration
> get_layer(model, index = 1)
<tensorflow.python.keras.layers.recurrent.GRU>
>
> # List the model's layers
> model$layers
[[1]]
<tensorflow.python.keras.layers.recurrent.GRU>

[[2]]
<tensorflow.python.keras.layers.core.Dense>

>
> # List the input tensors
> model$inputs
[[1]]
Tensor("gru_2_input:0", shape=(?, ?, 19), dtype=float32)

>
> # List the output tensors
> model$outputs
[[1]]
Tensor("dense_65/BiasAdd:0", shape=(?, 1), dtype=float32)
```

```
> summary(model)
Model: "sequential_27"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
gru_3 (GRU)                  (None, 32)                4992
_________________________________________________________________
dense_66 (Dense)             (None, 1)                 33
=================================================================
Total params: 5,025
Trainable params: 5,025
Non-trainable params: 0
_________________________________________________________________
>
> # Get model configuration
> get_config(model)
{'name': 'sequential_27', 'layers': [{'class_name': 'GRU', 'config': {'name': 'gru_3', 'trainable': True, '
batch_input_shape': (None, None, 19), 'dtype': 'float32', 'return_sequences': False, 'return_state': False,
  'go_backwards': False, 'stateful': False, 'unroll': False, 'time_major': False, 'units': 32, 'activation':
  'tanh', 'recurrent_activation': 'hard_sigmoid', 'use_bias': True, 'kernel_initializer': {'class_name': 'Gl
orotUniform', 'config': {'seed': None, 'dtype': 'float32'}}, 'recurrent_initializer': {'class_name': 'Ortho
gonal', 'config': {'gain': 1.0, 'seed': None, 'dtype': 'float32'}}, 'bias_initializer': {'class_name': 'Zer
os', 'config': {'dtype': 'float32'}}, 'kernel_regularizer': None, 'recurrent_regularizer': None, 'bias_regu
larizer': None, 'activity_regularizer': None, 'kernel_constraint': None, 'recurrent_constraint': None, 'bia
s_constraint': None, 'dropout': 0.4, 'recurrent_dropout': 0.4, 'implementation': 1, 'reset_after': False}},
 {'class_name': 'Dense', 'config': {'name': 'dense_66', 'trainable': True, 'dtype': 'float32', 'units': 1,
'activation': 'linear', 'use_bias': True, 'kernel_initializer': {'class_name': 'GlorotUniform', 'config': {
'seed': None, 'dtype': 'float32'}}, 'bias_initializer': {'class_name': 'Zeros', 'config': {'dtype': 'float3
2'}}, 'kernel_regularizer': None, 'bias_regularizer': None, 'activity_regularizer': None, 'kernel_constrain
t': None, 'bias_constraint': None}}]}
>
> # Get layer configuration
> get_layer(model, index = 1)
<tensorflow.python.keras.layers.recurrent.GRU>
>
> # List the model's layers
> model$layers
[[1]]
<tensorflow.python.keras.layers.recurrent.GRU>

[[2]]
<tensorflow.python.keras.layers.core.Dense>

>
> # List the input tensors
> model$inputs
[[1]]
Tensor("gru_3_input:0", shape=(?, ?, 19), dtype=float32)

>
> # List the output tensors
> model$outputs
[[1]]
Tensor("dense_66/BiasAdd:0", shape=(?, 1), dtype=float32)
```

```
> summary(model)
Model: "sequential_31"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
gru_9 (GRU)                  (None, 32)                4992
_________________________________________________________________
dense_70 (Dense)             (None, 1)                 33
=================================================================
Total params: 5,025
Trainable params: 5,025
Non-trainable params: 0
_________________________________________________________________
>
> # Get model configuration
> get_config(model)
{'name': 'sequential_31', 'layers': [{'class_name': 'GRU', 'config': {'name': 'gru_9', 'trainable': True, 'batch_input_shape': (None, None, 19), 'dty
pe': 'float32', 'return_sequences': False, 'return_state': False, 'go_backwards': False, 'stateful': False, 'unroll': False, 'time_major': False, 'un
its': 32, 'activation': 'tanh', 'recurrent_activation': 'hard_sigmoid', 'use_bias': True, 'kernel_initializer': {'class_name': 'GlorotUniform', 'conf
ig': {'seed': None, 'dtype': 'float32'}}, 'recurrent_initializer': {'class_name': 'Orthogonal', 'config': {'gain': 1.0, 'seed': None, 'dtype': 'float
32'}}, 'bias_initializer': {'class_name': 'Zeros', 'config': {'dtype': 'float32'}}, 'kernel_regularizer': None, 'recurrent_regularizer': None, 'bias_
regularizer': None, 'activity_regularizer': None, 'kernel_constraint': None, 'recurrent_constraint': None, 'bias_constraint': None, 'dropout': 0.0, '
recurrent_dropout': 0.0, 'implementation': 1, 'reset_after': False}}, {'class_name': 'Dense', 'config': {'name': 'dense_70', 'trainable': True, 'dtyp
e': 'float32', 'units': 1, 'activation': 'linear', 'use_bias': True, 'kernel_initializer': {'class_name': 'GlorotUniform', 'config': {'seed': None, '
dtype': 'float32'}}, 'bias_initializer': {'class_name': 'Zeros', 'config': {'dtype': 'float32'}}, 'kernel_regularizer': None, 'bias_regularizer': Non
e, 'activity_regularizer': None, 'kernel_constraint': None, 'bias_constraint': None}}]}
>
> # Get layer configuration
> get_layer(model, index = 1)
<tensorflow.python.keras.layers.recurrent.GRU>
>
> # List the model's layers
> model$layers
[[1]]
<tensorflow.python.keras.layers.recurrent.GRU>

[[2]]
<tensorflow.python.keras.layers.core.Dense>

>
> # List the input tensors
> model$inputs
[[1]]
Tensor("gru_9_input:0", shape=(?, ?, 19), dtype=float32)

>
> # List the output tensors
> model$outputs
[[1]]
Tensor("dense_70/BiasAdd:0", shape=(?, 1), dtype=float32)
```