MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE RUSSIAN FEDERATION
Federal State Autonomous Educational Institution of Higher Education
“South Ural State University (National Research University)”

School of Electronic Engineering and Computer Science
Department of Computer Engineering

THESIS IS CHECKED
Reviewer,
____
______________ _. _. __________
“___”____________________ 2022 г.

ACCEPTED FOR THE DEFENSE
Head of the department,
Ph.D., Associate Professor
______________ D.V. Topolsky
“___”___________ 2022 г.

Digital model for prediction power consumption based on data mining technique

GRADUATE QUALIFICATION WORK
SUSU – 09.04.01.2022.308-643.GQW

Supervisor,
PhD, Associate Professor
_________________ D.V. Topolsky
“___”____________2022 г.

Author,
student of the group: CE-228
_________A. A. Al-Mahdawi
“___”____________2022 г.

Normative control
_________________ S. V. Siaskov
“___”_____________2022 г.

Chelyabinsk-2022

MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE RUSSIAN FEDERATION
Federal State Autonomous Educational Institution of Higher Education
“South Ural State University (National Research University)”

School of Electronic Engineering and Computer Science
Department of Computer Engineering

ACCEPTED FOR THE DEFENSE
Head of the department,
Ph.D., Associate Professor
______________ D.V. Topolsky
“___”___________ 2022 г.

TASK

of the master graduate qualification work

for the student of the group CE-228
A. A. Al-Mahdawi
in master direction 09.04.01
“Fundamental Informatics and Information Technologies”
(master program “Internet of Things”)

1. **The topic** (approved by the order of the rector from 24.5.2022): “Digital model for prediction power consumption based on data mining technique”.

2. **The deadline for the completion of the work:** 01.06.2022.
3. **The source data for the work:**

3.1. Windows 10.

3.2. The purpose of using the Windows 10 operating system is that it is the most common and easy to use and any user can install it and deal with it easily.

3.3. Power consumption in India (2019-2020) dataset. [Electronic Resource] URL: https :// www . kaggle .com / twinkle0705 / state-wise-power-consumption - in-india /activity.

3.4. LaTeX, MathCad® syntaxis, MathML, Wolfram Mathematica®.

3.5. We have used the Menedely as references management application.

4. **The list of the development issues:**

4.1. To deploy a library that allows for a high accuracy prediction of a real-world problem of power consumption of IoT data analysis power consumption in India.

4.2. To develop a library that operates on low-power and low-processing-capability devices.

4.3. To test the library and give an example of how to implement interfaces.

4.4. A real-world IoT power consumption dataset, named Power consumption in India (2019-2020), is adopted for the implementation of the proposed model and using data mining techniques to analyze data. A time series is a data type of the dataset that used in this work.

5.4. To employ the libraries that apply data mining methods in the field of IoT by making accurate decisions to identify the cities that use the most electric energy and help officials in developing future plans for optimal management of energy consumption in India.

6.4. Appling experiments on the proposed model using the most effective data mining techniques and methods to verify its effectiveness and compare the results of the proposed model with the previous studies.

5. **Issuance date of the task:** 25.12.2021.

Supervisor ________________________________/ *D.V. Topolsky* /

Student ________________________________/ *A. A. Al-Mahdawi* /

CALENDAR PLAN

Phase	Deadline	Supervisor's signature
Introduction and literature review	10.03.2022	
Development of the model, design of the system	21.03.2022	
Implementation of a system	04.04.2022	
Testing and debugging of the system, experiments	25.04.2022	
Full text, normative control	16.05.2022	
Proposal defense	24.05.2022	

Supervisor ___________________________________/ *D.V. Topolsky* /

Student ___________________________________/ *A. A. Al-Mahdawi* /

Annotation

A. A. Al-Mahdawi. Digital model for prediction power consumption based on data mining technique. Chelyabinsk: SUSU; 2022, 48 p., 17 pic., bibl. – 24.

This thesis consists of six main chapters: Introduction, definition of requirements, design and Implementation, deployment and testing, conclusion, and references.

In the first chapter, we will have the subject area analysis briefly then have an overview of analogues and the main technological solutions that I will use will be featured. All the different software platforms to be used will be adequately described.

In the second chapter, there is a description of the data set used in this work and the technique of statistical analysis. This chapter also includes an explanation of deep learning algorithms in detail and the metrics used to assess the performance of the predictive model.

In the third chapter will describe the design and implementation of the software and how the different components will interact with each other as well as the algorithms for tackling the problem and predicted power consumption.

In the fourth chapter, the implementation of the interface of important library used in the proposed model will be showed, the adding TanserFlow library and Scikit-learn library are adding into language program.

In the fifth chapter, the results of each step of the proposed system will be presented, testing the deep learning models and comparing the performance of these algorithms based on error measures.

Finally, in the sixth chapter we will have a conclusion for the thesis, with future improvements to the solution being discussed as well as opportunities.

CONTENTS

INTRODUCTION 8
1 . SUBJECT AREA ANALYSIS 11
1.1. OVERVIEW OF ANALOGUES 11
1.2 ANALYSIS OF THE MAIN TECHNOLOGICAL SOLUTIONS 17
1.3 CONCLUSION 20
2. DEFINITION OF REQUIREMENTS 21
2.1. FUNCTIONAL REQUIREMENTS 21
2.2 CORE REQUIREMENTS 22
2.3 POWER CONSUMPTION PREDICTION REQUIREMENTS 22
2.4 DOCUMENTATION REQUIREMENTS 23
2.5 CONCLUSION 23
3 DESIGN AND IMPLEMENTATION 24
3.1 ARCHITECTURE OF THE PROPOSED SOLUTION 24
3.2. DESCRIPTION OF IOT DATA 25
3.3 ALGORITHMS FOR SOLVING THE PROBLEM 27
3.4 CONCLUSION 32
4. IMPLEMENTATION 33
4.1. IMPLEMENTATION OF INTERFACES 33
4.2. CONCLUSION 36
5. TESTING 37
5.1. TESTING PREDICATION MODEL 37
5.2.CONCLUSION 42
6 CONCLUSION 43

Acknowledgements

I would like to acknowledge the role that God has played in my progress. For giving me this opportunity to further my studies. I would also like to thank my family for always being there for me this whole time.

My supervisor Dmitry Topolsky has been very instrumental and resourceful, my department has always been open and available to solving all my educational problems. My university has given me a conducive learning environment that I cannot take for granted.

Lastly, I would lastly like to thank the Russian Government for facilitating my stay in Russia.

INTRODUCTION

Electric power consumption is the extent of power used. IoT devices are being used to collect data in order to analyze the behavior and proper uses of power. Recently, it was observed that there has been a rapid rise in the consumption of power throughout the world []. Power predication is high importance to helps reduce power consumption and provides better energy, and cost savings [2].

Data mining is one of the effective methods to analyze the electricity consumption, discovering useful patterns from the database, and used to make good decisions. Therefore, power utilities and governments are always searching for intelligent models to improve the accuracy of prediction based on data mining methods [3].

There are several advantages to predication model for power consumption:

- in power planning, power consumption predictions is crucial, management, and conservation. Constantly improving the performance of prediction models is the key to ensuring the reliable functioning of power systems [4];
- it is necessary to build model based on data mining techniques that can high accurately predict energy consumption based on time series of energy consumption and give future indicators to determine the places that are more energy consuming and find appropriate planning to reduce this consumption [5];
- the economic and industrial development of countries depends on the production and management of power perfectly. This is done by relying on model that have the ability to predict quickly and with high accuracy. India is the third-largest producer in the world and the third-largest consumer of electricity, so it is necessary to use systems like this type of system [6].

Many studies and researches have offered several power consumption predication algorithms, datasets, and templets. These works have helped to proposed effective models to prediction power consumption using dataset that collected from different countries. Consider an apartment where every electrical powered device in the home is connected to a smart sensor that is able to send information within a certain time series to the Energy Management Center regarding the amount of electricity consumption. It would be simple to help professionals understand people's energy consumption behavior and result in good and efficient energy consumption management.

The purpose of the research:

To create a predication model to enable running of deep learning algorithms for power consumption predication on dataset with higher accuracy and faster predication. The tasks necessary to achieve the goal:

1. To deploy a library that allows for a high accuracy prediction of a real-world problem of power consumption of IoT data analysis power consumption in India.
2. To develop a library that operates on low-power and low-processing-capability devices.
3. To test the library and give an example of how to implement interfaces.
4. A real-world IoT power consumption dataset, named Power consumption in India (2019-2020), is adopted for the implementation of the proposed model and using data mining techniques to analyze data. A time series is a data type of the dataset that used in this work.
5. To employ the libraries that apply data mining methods in the field of IoT by making accurate decisions to identify the cities that use the most electric energy and help officials in developing future plans for optimal management of energy consumption in India.

6. Appling experiments on the proposed model using the most effective data mining techniques and methods to verify its effectiveness and compare the results of the proposed model with the previous studies.

Structure of the Thesis

This thesis consists of six main chapters: Introduction, definition of requirements, design and Implementation, deployment and testing, conclusion, and references.

In the first chapter, we will have the subject area analysis briefly then have an overview of analogues and the main technological solutions that I will use will be featured. All the different software platforms to be used will be adequately described.

In the second chapter, there is a description of the data set used in this work and the technique of statistical analysis. This chapter also includes an explanation of deep learning algorithms in detail and the metrics used to assess the performance of the predictive model.

In the third chapter will describe the design and implementation of the software and how the different components will interact with each other as well as the algorithms for tackling the problem and predicted power consumption.

In the fourth chapter, the implementation of the interface of important library used in the proposed model will be showed, the adding TanserFlow library and Scikit-learn library are adding into language program.

In the fifth chapter, the results of each step of the proposed system will be presented, testing the deep learning models and comparing the performance of these algorithms based on error measures.

Finally, in the sixth chapter we will have a conclusion for the thesis, with future improvements to the solution being discussed as well as opportunities.

1.1. OVERVIEW OF ANALOGUES

1.1.1. Short –Term Power Consumption Prediction

Short -term is type of predication methodology based on time series and its duration ranges (from one day to a month) and very short-term (within 24 hours). The Osman T. B. and Ahmet C. in [9] are focused on Turkey's Electrical Energy Consumption by using Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models, for short –term of time series data. One-hour and three-hour ahead forecasting are accomplished by using a historical dataset of electrical energy consumption in Turkey. Important features include:

- GRU and LSTM can reliably be employed to forecast electrical energy consumption to manage electrical energy strategies efficiently;
- dataset used in this works includes electrical energy consumption values in terms of MWh (megawatt hour) from December, 31- 2015 at 00:00 am to March, 2 - 2021;
- all the simulations are performed on a Colab (Colaboratory) Platform provided by Google. GPU, provided by Colab Platform, is also utilized in order to speed up the model training;
- Tensoflow v2.3.0 and Keras v2.4.3 libraries are used for model training;
- a four-layer network is used for LSTM and GRU models. All networks have three layers, each of which has 300 hidden units, and a dense layer [9].

1.1.2. Mid –Term Power Consumption Prediction

Mid -term is type of predication methodology based on time series and its duration ranges (from a month to a year). Heng S.et al.,[10] used a pooling-based deep recurrent neural network (PDRNN) to introduce deep learning for household load forecasting. Important features include:

- essentially the model could address the over-fitting issue by increasing data diversity and volume;
- this work reports the first attempts to develop a bespoke deep learning application for household load forecasting and achieved preliminary success;
- this model was evaluated on 920 smart metered consumers in Ireland using the Tensorflow deep learning platform;
- the dataset used from the Smart Metering Electricity Customer Behaviour Trials (CBTs) were initiated by Commission for Energy Regulation (CER) in Ireland. The trials took place between 1st July 2009 and 31st December 2010 with over 5000 Irish residential consumers and small and medium enterprises (SMEs) participating.

1.1.3. Long –Term Power Consumption Prediction

Long-term is type of prediction methodologies and it's in range (from year and more) . Guo, Z.et al.[11] used LSTM to exhibits better forecasting accuracy in terms of measuring electricity consumption .To verify the efficiency of the proposed methods, China Jiangsu Province Power based on daily electricity consumption data for 2014 are conducted. Important features include:

- LSTM has important to extract features of the power consumption and removing the noise;
- LSTM has exhibits a good forecasting accuracy in terms of measuring electricity consumption;
- the experiments of the LSTM model on daily electricity consumption data for China Jiangsu Province Power for 2014 are conducted;
- LSTM provides an approach to identifying the nature of the power consumption represented by the sequence of observations.

After comparing the different Short-term power consumption prediction, Mid-term power consumption prediction and Long-term power consumption prediction on Mean Absolute Error (MAE), Root mean squared error (RMSE) , execution time ,and power consumption predication methods to see how they compare to each other, I found out what are their weaknesses are and it helped me to build up on my proposed solution.

First, when on comparing short, mid, and long term power consumption prediction, all these methods are quite similar in their offering: developments of Deep Learning (DL) methods have resulted into powerful tools that can handle large data-sets. all power consumption prediction methods continually update their language models to decreasing error rate and execution time for power consumption prediction and make these modes suitable for IoT environments. This continuous development is a strong point for all, especially the introduction of decreasing error rate and execution time which can help with effectively managing energy consumption. all predication methods have need enhancement in term of error prediction rate and execution time: LSTM is effective method for short and mid -term prediction power consumption but has weakness performance with long -term power consumption prediction   , however [12].

On the other hand, Short-term memory was solved with the creation of LSTM and GRU. They have internal gates that allow them to regulate the flow of information. These gates can figure out which data in a series is important to keep or throw away. Long sequences are well processed by LSTM and GRU. LSTM and GRU do carry a lot of complexity, however, requiring a lot of time and effort to fully learn the quirks.

Using MSE error rate for power consumption prediction, which compares actual value and its corresponding prediction value, MSE is measured in units that are the square of the target variable. Short-term power consumption has MSE (83.10% (kWh)) [9], Mid-term power consumption has MSE (6.96% (kWh)) [10], and Long-term power consumption has MSE (594.84% (kWh)) [11].

I also compared them in terms of the RMSE error rate for power consumption prediction, which compares actual value and its corresponding prediction value, RMSE is measured in the same units as the target variable. Short-term power consumption has RMSE (85.09% (kWh))[9], Mid-term power consumption has RMSE (6.45% (kWh)) [10], and Long-term power consumption has RMSE (597.05% (kWh)) [11].

Table 1 – Comparison of analogues

Feature	**Short –term Power consumption prediction**	**Mid –term Power consumption prediction**	**Long–term Power consumption prediction**
Language support	Python with TensorFlow	Python with TensorFlow	Python with TensorFlow
Methods	LSTM +GRU	PDRNN	LSTM
Domain/Data	Turkey’s Electrical Energy Consumption	Smart Metering Electricity Customer Behavior Trial (CBT)	China Jiangsu Province Power
MSE, %(kWh)	83.10	6.96	594.80
RMSE, %(kWh)	85.09	6.45	597.05
Time-Scale	Short (30-min or 1-h)	Medium (1-month)	Long (1-year)

1. SUBJECT AREA ANALYSIS

The demand for power is based on various factors such as weather, occupancy, types of machines and appliances used. The dependency on high number of factors have made predicating techniques much complex. Accurate predictions of the electricity power consumption are important for efficient distribution. However, applying all the variables that effect electricity power consumption can create a complex predicating model which is unstable and unpredictable. Therefore, data-driven solutions to predict electricity power consumption focuses on time-series solutions [7].

To take advantage of time series attribute of the power consumption in India, it is imperative that we merge the IoT devices with deep learning since data is the most important parameter of any system. One of the most anticipated problems that this combination of disciplines can solve through predicting things based on past observations is reach a best accuracy of predict which cities in India consume the most electricity [8].

To do prediction power consumption using deep learning, a programmer should feed data into learning algorithm that discovers the relationship between power consumption and its time-series, which then builds a model based on the data provided through a process called training and finally data is then run through this model to make predictions, a process called inference [7].

The past few years there are various methodologies that are utilized for the predicating of power consumption based on IoT data and deep learning. All these methods use time series data of the prediction power consumption in their proposed models as will be discussed below. In most cases, power consumption prediction is done in the in several countries to cover the electricity consumption of homes, buildings, and cities for those countries, especially in India, where it ranks third in the world in the consumption of electrical energy, on powerful. The Deep learning DL models bring up accuracy issues, efficiency, and speed due to latency and high-power consumption sending a constant stream of data consumes a lot of energy [7].

This thesis will therefore focus on training a tiny model that used time series data for a power consumption   and testing on the publicly available dataset of the power consumption in India to achieve optimal accuracy and fast prediction time.

1.2. ANALYSIS OF THE MAIN TECHNOLOGICAL SOLUTIONS

1.2.1. Power Consumption dataset

The dataset that used in this work is Power consumption in India (2019-2020). The data is in the form of a time series for a period of 17 months beginning from 2nd Jan 2019 till 23rd May 2020. Rows are indexed with dates and columns represent states. Rows and columns put together, each datapoint reflects the power consumed in Mega Units (MU) by the given state (column) at the given date (row) [13].

The dataset is collect by Power System Operation Corporation Limited (POSOCO) is a wholly-owned Government of India enterprise under the Ministry of Power. It was earlier a wholly-owned subsidiary of Power Grid Corporation of India Limited. It was formed in March 2009 to handle the power management functions of PGCIL [13].

This data set include two sub-dataset : dataset-tk has 43 column or features represent India cities and row represents time series ,these features are :[tim, Punjab, Haryana, Rajasthan, Delhi, UP, Uttarakhand, HP, J&K, Chandigarh, Chhattisgarh, Gujarat, MP, Maharashtra, Goa, DNH, Andhra Pradesh, Telangana, Karnataka ,Kerala, Tamil Nadu, Pondy, Bihar, Jharkhand, Odisha, West Bengal, Sikkim ,Arunachal Pradesh, Assam, Manipur, Meghalaya, Mizoram, Nagaland, Tripura]. The second dataset called Long – data incudes 6 features are: [States, Regions, latitude, longitude, Dates, Usage] [13].

To make a strong case for the prototype, will be pre-processing power consumption dataset involves three sub-step (compute Mean Sequar Error (MSE) , Check Data Types where only numerical data (float64) will be used in predication modelling, and Check for Missing Data to ensure there are no missing values in the data) .

1.2.2. TensorFlow

TensorFlow is a free and open-source end-to-end platform for developing Machine Learning applications. It is a symbolic math library that employs dataflow and differentiable computation to handle a variety of tasks related to deep neural network training and inference. It helps developers to implement machine learning applications by utilizing a variety of tools, frameworks, and community resources [14].

TensorFlow is the finest library of them all since it is designed to be accessible for everyone. The TensorFlow library includes a variety of API for building large deep learning architectures such as CNN and RNN. TensorFlow is built on graph computation and allows developers to visualize the neural network's creation via the Tensor board. This application debugging tool is quite useful. Finally, TensorFlow is designed to be used in for many. It runs on both the CPU and the GPU. For instance, with artificial intelligence (AI), Google customers can get a faster and more precise search. When a user types a keyword into Google's search field, the search engine makes a suggestion for the next word [15].

Some supported TensorFlow algorithms include:

- Linear regression;
- Classification;
- Deep learning classification;

- Deep learning wipe and deep;
- Booster tree regression;
- Boosted tree classification [15].

1.2.3. Scikit-learn

Scikit-learn is an open source data analysis library, and the gold standard for Machine Learning (ML) in the Python ecosystem. Key concepts and features include [18]:

- algorithmic decision-making methods, including;
  - classification: identifying and categorizing data based on patterns;
  - regression: predicting or projecting data values based on the average mean of existing and planned data;
  - clustering: automatic grouping of similar data into datasets.
- algorithms that support predictive analysis ranging from simple linear regression to neural network pattern recognition;
- interoperability with NumPy, pandas, and matplotlib libraries.

When we want to get up and running fast, or are looking for the latest deep learning research tool, you will find that scikit-learn is both well-documented and easy to learn/use. As a high-level library, it lets you define a predictive data model in just a few lines of code, and then use that model to fit your data. It's versatile and integrates well with other Python libraries, such as matplotlib for plotting, numpy for array vectorization, and pandas for dataframes [18].

1.2.4. Programming Technologies

For this project we will use Python as our main programming languages for development. Development that tools are required to develop and test/debug the code include:

- compiler;
- debugger.

1.3. Conclusion

The following components are being used for this project. It is urgent and under development right now:

- power Consumption dataset in India;
- TensorFlow;
- Scikit-learn.

2. DEFINITION OF REQUIREMENTS

2.1. FUNCTIONAL REQUIREMENTS

2.1.1. Power Consumption Predication

The Power Consumption prediction model is a model that uses data mining techniques and methods to find the relationship between power consumption in Indian cities and time series in an IoT environment. In this case, the Central power stations for each city in India contain, it contains several sensors that receive millions of requests for power from different parties over a period of time (Long term. The Power consumption prediction model will be analyzing data collected from IoT devices (sensors) and then predication a city that has a peak in power consumption.

2.1.2. Collect IoT Data using Sensor

IoT data collection involves the use of sensors to track the performance of devices connected to the Internet of Things. The sensors track the status of the IoT network by collecting and transmitting real-time data that is stored and retrieved at any moment in time. These datasets are fundamentally different from other datasets which is containing power consumption smart sensors data from different cities of India over two years. This means that the proposed system was assessed from different viewpoints: analyzing all Indian cities and focusing on the minimum error prediction rate.

2.1.3. Internet independence

The combination between time series and machine learning model used by three effective data mining methods to predication Power consumption should run on the IoT devices offline.

2.1.4. Noise

Power consumption model should be able to work well in noisy environments by automatically identification and filtering out the noise.

2.1.5. Processor power

Power consumption predication model should run on an edge device with a 64-bit Intel(R) Core(TM) i7-3687U microprocessor running at 2.10 GHz of program memory and 6.00KB RAM.

2.2. Core Requirements

2.2.1. Power consumption predication model shall be able to classify attack in Real time.

2.2.2. Power consumption predication model shall be compatible with different operating system such as Windows 10 Pro.

2.2.3. Power consumption predication shall be easy to use by developers through bootstrapping of all the required code and shall have comments.

2.2.4. Power consumption predication prototype used for this thesis will be limited to Power consumption in India (2019-2020) Dataset to assess how effective the approach taken is, then be updated, by using a new dataset including short or middle time series.

2.3. Power Consumption Prediction Requirements

The following guidelines are specific to Power Consumption prediction:

2.3.1. Power Consumption prediction model SHALL use only The Power consumption in India (2019-2020) . The data is in the form of a time series for a period of 17 months beginning from 2nd Jan 2019 till 23rd May 2020.

2.3.2. Power Consumption prediction model SHALL provide output to indicate the minimum error rate of prediction of the power consumption.

2.3.3. Power consumption prediction model shall address the power consumption forecasting in the long short-term memory (LSTM) and Gated Recurrent Units (GRU) network due to its ability to deal with sequential data such as time-series data and prediction with higher accuracy the only for the scope of this thesis.

2.3.4. The dataset that is used in Power consumption prediction stored 43 column or features represent India cities and row represents time series. About 80% of these connections were used as training data, whereas the remaining 20% was used as testing data.

2.4. Documentation Requirements

Documentation of this proposed model will be delivered in various formats including pdf, ppt, html and docx and will be English language. This documentation will be the user manual for “power consumption prediction” outlining its various features and how to use the detection model for programming development.

2.5. Conclusion

In this chapter, we defined the various core and power consumption requirements for power consumption prediction. We also described how the various software components within it will interact with each other. Lastly, we gave the documentation requirements that described among other things the user manual and its contents.

3. DESIGN AND IMPLEMENTATION

3.1. ARCHITECTURE OF THE PROPOSED SOLUTION

The Power Consumption predication model is composed of various components, as shown in the figure below. First, the IoT senores receives power consumption data input then using three techniques to solved dataset problem are check missing data, series to supervised data, and normalization data. The interface is then run on the features outputting class probabilities as shown in the diagram below.

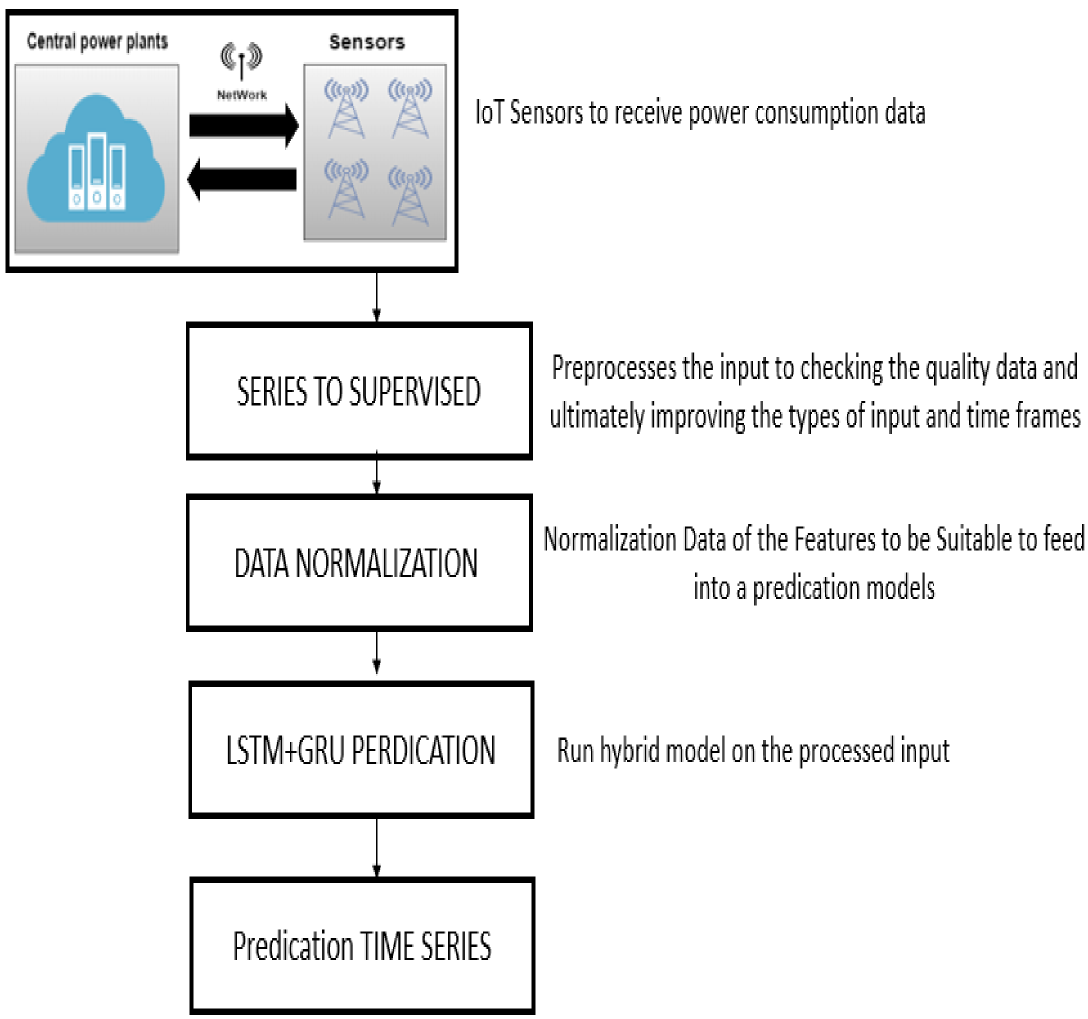

Figure 1 – Power Consumption Prediction Architecture

The model used in this thesis is trained to predication short-term memory for the power consumption. The model takes in data from IoT sensors devices and outputs the probability scores. This data is consumed by the model in terms of analysis and building of predication models.

3.2. DESCRIPTION OF IOT DATA

3.2.1. Data Collection

Time series is continuously sampled at real-world 987.42 kB using an IoT sensors device. The data is then temporarily stored in the dataset before being passed to the preprocess techniques which convert raw time series data 3-dimension into 2-dimension that will be run through the prediction model.

3.2.2. Data Usage

After the raw data is run through the prediction model, probabilities for the various classes of time series data and then calculate the error rate of the prediction class and displayed on the interface of predication model. At this point, a command predication could be programmed to undertake a certain command that the command responder will execute.

3.2.3. Data Storage

For the sake of device predications and service enhancements, the IoT sensors devices in central power plants generate and store various metadata. Because of the size of the devices that IoT sensors are targeting, time-series inputs, are the primary element of power consumption prediction model data. The neural network is used to analyze time series and then extract the user's power consumption. deep learning techniques are used by this system to better itself with each input.

3.2.4. Data Retention

Through IoT sensors in each central power plant can collect data for power consumption in this region and store it in the specific dataset, so the developer of the project decides to store data, and they are free to do so. As for the metadata, access is granted via specific, audited permissions, and access to customer data requires review and approval by the responsible managers.

Some model-level data is also stored in these data are stored in a CSV file, which has many attributes in addition to the date, entity, and code. The reports of people's power consumption in 17 months (from 2019 to 2020) for the specific country was delineated using a dataset of a multivariate time series. To use the data, some places with common characteristics are grouped into categories for power consumption control. For instance, buildings and companies are mixed, as both are also called important trips. This data collection aims to try to remedy the effect and pace of the power consumption in the places useful via controlling the power consumption.

We apply retention policies to data to minimize the (meta)data we retain. Data is retained when it serves a business purpose (including providing the service to customers and improving our systems) or as necessary to comply with law. We also offer debug interfaces like SWD and we also have disabled code readout on Arm platforms. Even though these measures are not perfect, they will raise the cost of an attack.

The power consumption of the multivariate time series in the power consumption prediction model is based on deep learning (DL) algorithms. Data sets from real use cases are fed into the various DL model to build new algorithms and improve existing algorithms.

3.3. ALGORITHMS FOR SOLVING THE PROBLEM

3.3.1. Power Consumption as Data

The first step to do power predication for our dataset will be to clear data. This is the identifying of irrelevant features and then dropping these features. The second step dropping features variables that have a high percentage of missing values. The third step is features rename that have space, '%' and unusual characters.

3.3.2. Handling Missing Data

After cleaning the data, the missing data have to be processed into the features, the missing rows are compensated by calculating the mean value or median, this is often a very preferred step since it has at least some support from the data. The null cell is checked and the mean value of the column is compensated and the remainder of the data was used to estimate the missing data. Null cell directly affects prediction results, their accuracy, and reliability (figure 2).

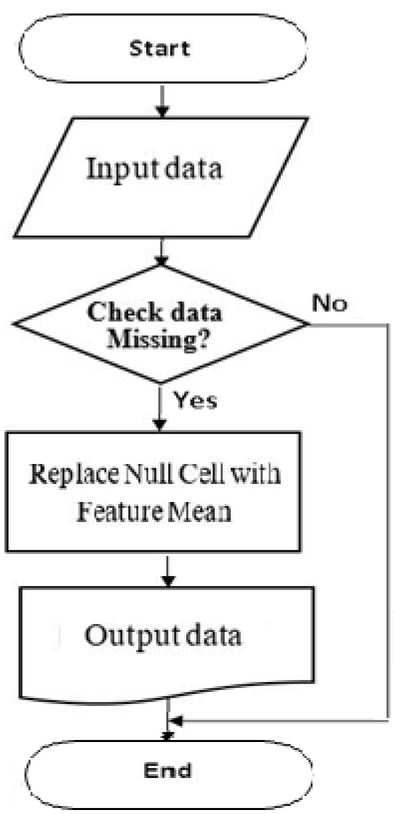

Figure 2 – The Flowchart for Check and Process the Missing Data

3.3.3. Series to Supervised

Series to supervised or Reshape Input Data is reshape the input data into 3 dimensions matrix [samples, time steps, features] to optimize preprocessing input data into prediction models. For the prediction of time-series data, the model has to estimate features from the previous values. So, if the desired output is (t), the input feature should be $(t - 1)$. Time series to supervised function is designed to take samples at time $\{t - n, t - n - 1, \ldots, t - 1\}$ as input for the machine learning and the output will be sampling at time $\{t, t + 1, \ldots, t + n\}$. For the current application, the power consumption for India cities data are sampled daily, and hence the input is the previous day and the output is the current day, so time steps are equal to one, and the Input matrix-shaped in the form [samples, 1, features]. Figure 3 illustrates time steps for the data.

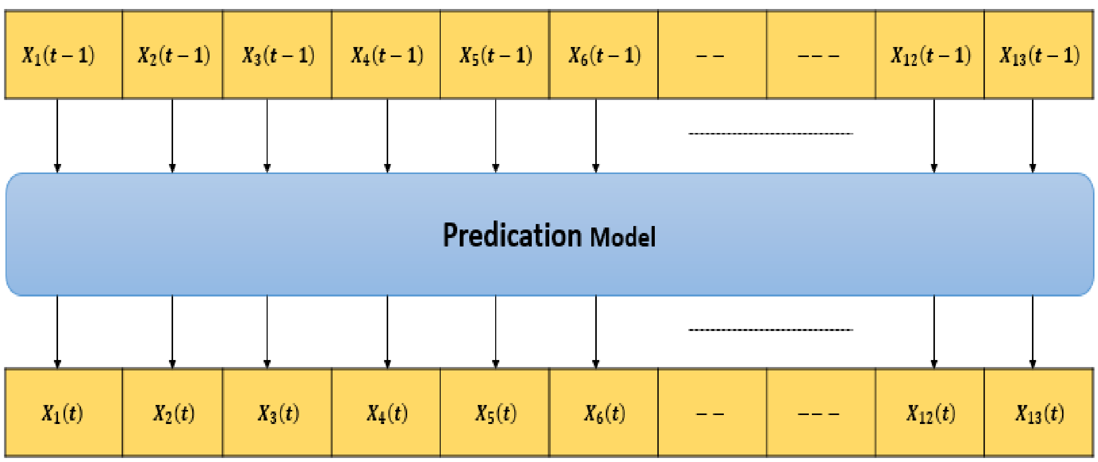

Figure 3 – Time steps for Data. input data at time (t-1) to the prediction model and the output data at time (t)

3.3.4. Data Normalization

Normalization of data is a kind of method in which data is so reorganized that consumers can use it correctly for further queries and analysis, which enhances the coherence of the types of input. Normalization helps to turn the value of the attribute into a limited set. It is the process of sending data to a particular range, such as between 0 and 1 or between -1 and 1 as in [16][17]. Normalization is needed where there are wide variations in the values of different feature ranges. Training time is hurried up by the data normalization, where the training time is started to access feature ranges of the same size.

3.3.5. Build Prediction Model

3.3.5.1. LSTM Model

Long Short-Term Memory Networks (LSTM) can hold information for long periods due to its chain-like structure, where it can solve the tasks that are difficult to implement using traditional RNN. LSTM neural networks are structured for sequential data processing. Network status at any time depends both on the present and preceding input of the network. The type of our model architecture is many-to-many [18]. LSTM consists of three main gates:

- forget gate: there is information that is no longer needed to complete the task, this gate removes it and this improves the performance of the network;
- input gate: through this portal, information is added to memory cells;
- output gate: this portal produces the necessary information in hidden layer output [19][20].

The LSTM cell is created from the input layer, the previously hidden cell $ht{-}1$ is entered and the new sequence x_t is entered, where the first step of this combined entry is that it is crushed through the tan_h layer where tan_h takes large or small variable numbers and converts them at a specific rate between (-1,1) to generate candidate memory cells $\tilde{C}$. As for the input gate, the gate is a layer of sigmoid activation nodes, whose output is multiplied by the output of the tan_h, the sigmoid of this input gate can stop any element of the input vector not required as this function outputs the values between 0 and 1.

In the forget gate, the question is whether the previous memory cell may be used to compute the current memory cell. As a result, the forget gate examines both the input and the previous hidden state. This addition process, instead of multiplication, helps reduce the risk of gradient vanishing. This gate allows the network to understand the state variables that must be remembered or forgotten. After completing the above parts, the cell state C_t of LSTM is updated. This equation connects the pre-state C_{t-1} and the present temporary-state $\tilde{C}$. Through the output gate, LSTM outputs the specified state, based on the cell status, where runs a sigmoid layer to determine the unit state section to be exported. And deals with current output ot and state C_t with a tan_h layer to write a new hidden layer state h.

In this model, the input is 33 features, two hidden layers each layer consisting of 1000 nodes of LSTM, and six outputs, connected in fully connected layers' form as shown in figure 4.

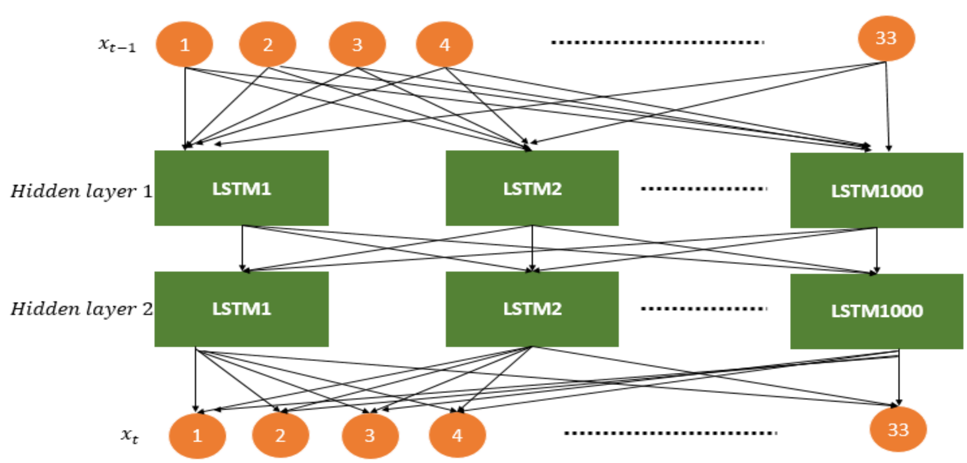

Figure 4 – The LSTM Structure

3.3.5.2. LSTM and GRU Model

In this model, the LSTM will take the output of GRU as inputs to improve the accuracy of prediction model. The gated recurrent units are a simplified version of LSTM and require less training time with improved network performance. There are two gates in the GRU unit: update and reset [21].

The reset gate is the first gate, and its primary role is to determine how much historical information can be stored. The update gate is the second of the gates. The main function of the update gate decides what past information can discard/keep being passed to the future. The added data is sent to the update gate and multiplied by the sigmoid function, the resulting value is between [0, 1] [22].

Generating new memory information continues to pass forward . When the reset gate is 0, the memory information is completely cleared. Conversely, when the reset gate is 1, it means that the memory information is all passed. Then calculates hidden state h_t at time t from the output of the update gate z_t, reset gate r_t, current input x_t, previously hidden state h_{t-1} [21].

In hybrid LSTM+GRU, firstly, the model adopted two hidden layers, the first layer includes 1000 GRU nodes and the second layer has 1000 LSTM nodes, the output of the first layer represents the input for the second layer was proposed.

In this model, the input is 33 features, two hidden layers, the first hidden layer consisting of 1000 nodes of GRU and the second hidden layer consisting of 1000 nodes of LSTM and 33outputs, connected in fully connected layers' form as indicated in the figure 5.

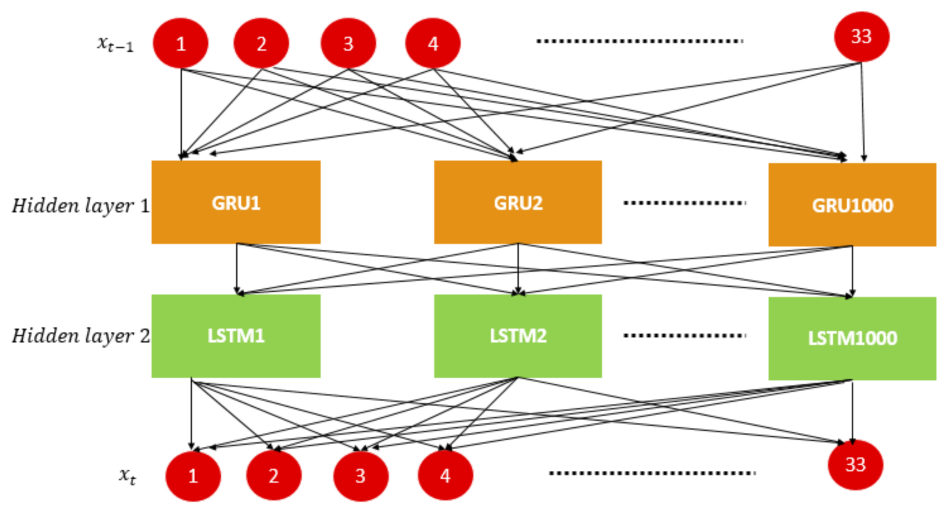

Figure 5 – The Hybrid Model Structure in the Proposed Power Consumption Prediction

3.3.5.3. The Predication

This phase flattens the sample into a 2-dimension feature vector then fed to a feed-forward neural network and back-propagation is applied to every iteration of training since the input feature has already been converted into a suitable form for our multivariate time series. The prediction model is able to find the relationship between the power consumption for cities and certain time series in instance and predication them using the hybrid LSTM and GRU.

3.3.5.4. Evaluation Model

To calculate the prediction efficiency, we used statistical measures to compare target values and predicted values, Mean absolute error (MAE) according to equation in [23] and root mean square error (RMSE) according to equation in [24] for performance evaluation of hybrid deep learning algorithms.

3.4. Conclusion

The model trained in this chapter will then be converted into a TensorFlow model which can now run on computer. This TensorFlow model will be the main inferencing component of the power consumption prediction interface, the focus in this thesis being enhancement accuracy of the prediction of the power consumption in India. The proposed model will deliver a analysis long-term data and efficient prediction application for the power consumption Dataset (2019-2020). The user will be able to use this application to perform deep learning on their India (2019-2020) for real-time power consumption prediction.

4. IMPLEMENTATION

4.1. IMPLEMENTATION OF INTERFACES

The platform follows the principles of Internet of Things (IoT) sensors through a link to Electric power station centers using IoT-related technologies, and all these devices transmit data to the database.

Hardware layer that rounds the sensors used to collect the requirement data. Power consumption prototype was developed to collect power consumption data via 17 months. The sensors that calculate the energy consumption were placed in electrical cabinets and took three-phase current readings, that is, each sensor measured three different alternate currents (AC) and calculated the electric power in kW (electric voltage).

Software layer that utilized by the power consumption prediction model for find the correlation between cities and power consumption. The predication model has been added to the TanserFlow Library to Pychram as shown in the list below. Pychram is a dedicated Python Integrated Development Environment (IDE) providing a wide range of essential tools for Python developers.

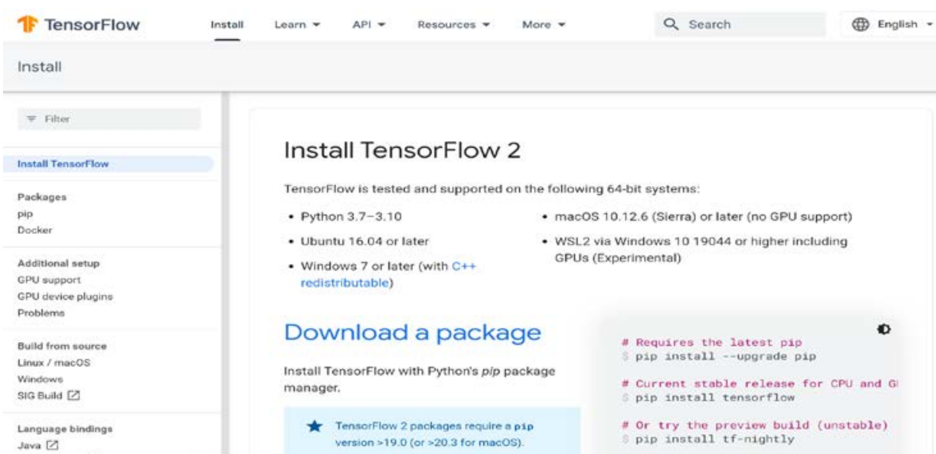

Figure 6- Interface of the Install TanserFlow

```
The following NEW packages will be INSTALLED:

    _tflow_1100_select: 0.0.1-gpu
    absl-py:            0.5.0-py36_0
    astor:              0.7.1-py36_0
    blas:               1.0-mkl
    certifi:            2018.8.24-py36_1
    cudatoolkit:        9.0-1
    cudnn:              7.1.4-cuda9.0_0
    gast:               0.2.0-py36_0
    grpcio:             1.12.1-py36h1a1b453_0
    icc_rt:             2017.0.4-h97af966_0
    intel-openmp:       2019.0-118
    libprotobuf:        3.6.0-h1a1b453_0
    markdown:           2.6.11-py36_0
    mkl:                2019.0-118
    mkl_fft:            1.0.6-py36hdbbee80_0
    mkl_random:         1.0.1-py36h77b88f5_1
    numpy:              1.15.2-py36ha559c80_0
    numpy-base:         1.15.2-py36h8128ebf_0
    pip:                10.0.1-py36_0
    protobuf:           3.6.0-py36he025d50_0
    python:             3.6.6-hea74fb7_0
    setuptools:         40.4.3-py36_0
    six:                1.11.0-py36_1
    tensorboard:        1.10.0-py36he025d50_0
    tensorflow:         1.10.0-gpu_py36h3514669_0
    tensorflow-base:    1.10.0-gpu_py36h6e53903_0
    tensorflow-gpu:     1.10.0-hf154084_0
    termcolor:          1.1.0-py36_1
    vc:                 14.1-h0510ff6_4
    vs2015_runtime:     14.15.26706-h3a45250_0
```

Figure 7 -Install TanserFlow entry in the Pychram Library

The power consumption prediction model library has been added to the scikit-learn Library to Python as shown in the list below.

Figure 8 – Interface of install scikit-learn Library

```
C:\Users\Geeks>python -m pip show scikit-learn
Name: scikit-learn
Version: 0.24.2
Summary: A set of python modules for machine learning and data mining
Home-page: http://scikit-learn.org
Author: None
Author-email: None
License: new BSD
Location: c:\users\geeks\anaconda3\lib\site-packages
Requires: joblib, threadpoolctl, numpy, scipy
Required-by:
```

Figure 9 – Install scikit-learn entry in the Python Library

In the implementation of the power consumption model, firstly running the training on the input power consumption dataset using training interface as illustrated in figure10.

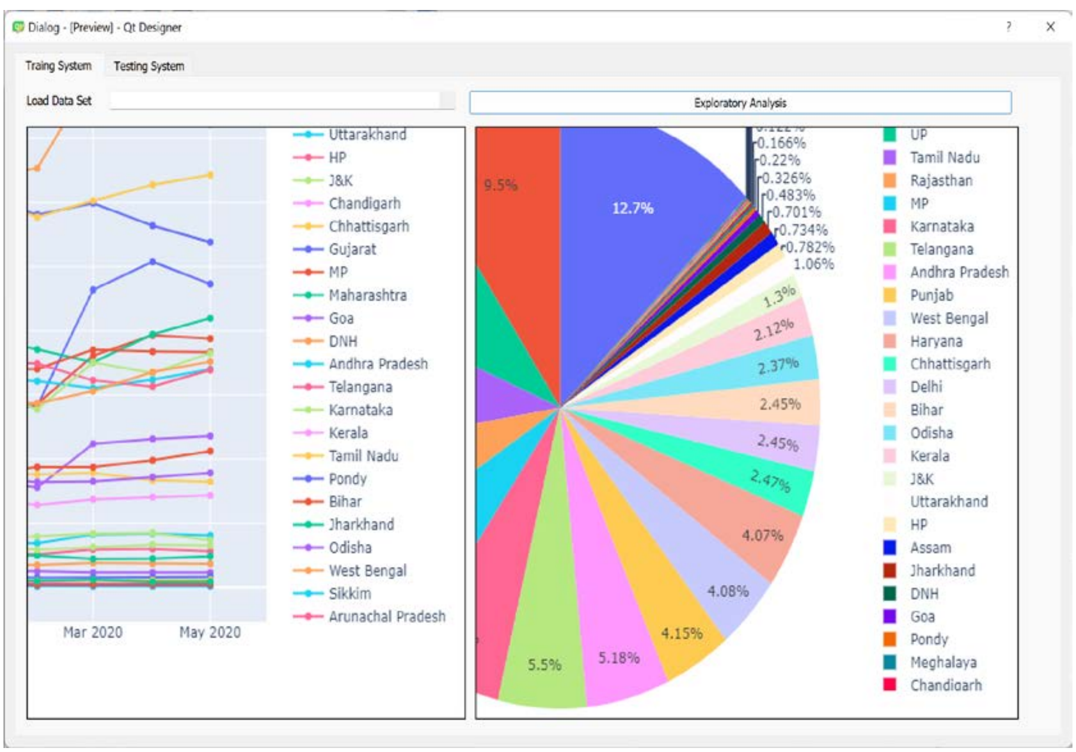

Figure 10 – Implementation of the Training interface

After training dataset then testing the power consumption prediction model by input instance from the testing dataset to prediction the input instance if has higher power consumption or not as shown in figure11.

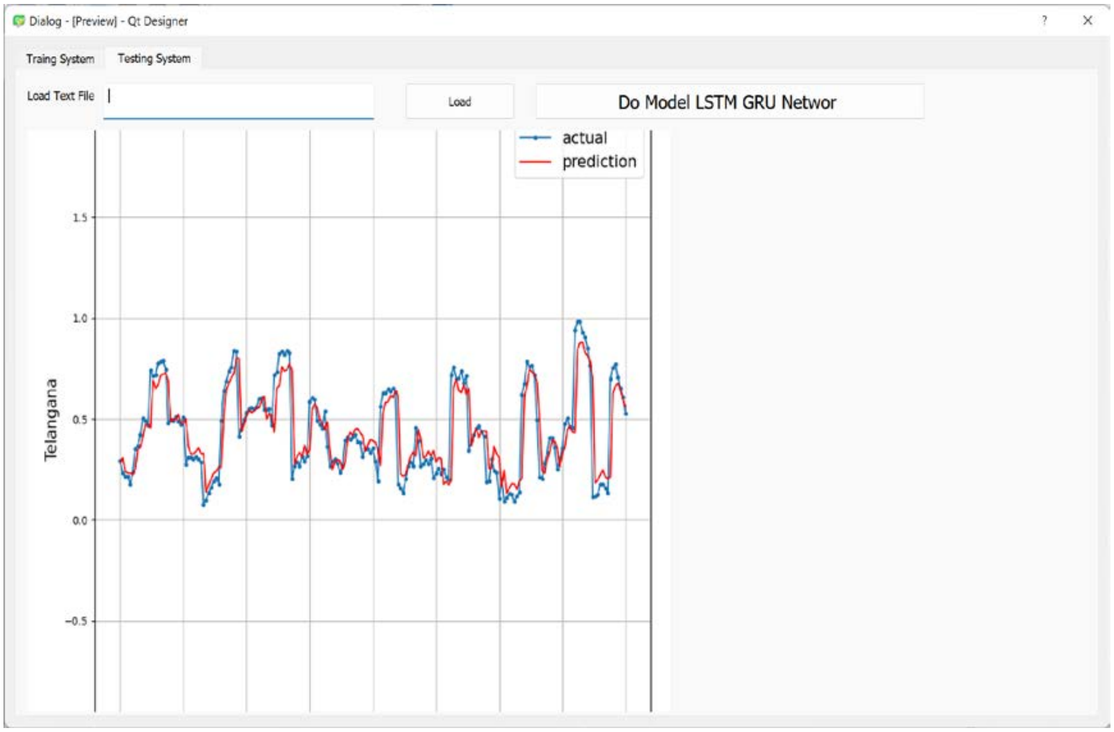

Figure 11 – Implementation of the Testing interface

4.2. Conclusion

In this chapter, will be present the hardware layer of the IoT sensors that used to collect IoT data of the power consumption in time series form. Furthermore, the level of software and libraries required by the suggested model in order to predict power consumption with high accuracy.

5. Testing

5.1. TESTING PREDICATION MODEL

For testing the hybrid Long Short Term Memory LSTM and Gated Recurrent Unit GRU, 80% of the data used for training and 20% of sample data that the LSTM+GRU used to evaluate how the model is performing. The metrics used to measure the accuracy of prediction are the mean absolute error MAE and root mean square error RMSE.

The hybrid model has two hidden layers; each layer contains 1000 nodes. To prevent overfitting, we used 100 epochs for training and batch size 64 for the training dataset. To obtain the most precise output prediction.

Figure 12 the Model summary (Hybrid) and the total number of parameters in the network and the number of parameters that have been trained. The hybrid Model, which has 4 layers that are fully connected. The first layer is the input layer which has 8 nodes. The second layer is hidden layer one which has 1000 nodes of GRU and the third layer is hidden layer two that has 1000 nodes of LSTM. In the first hidden layer, input and output data are in 3D format, while the second hidden layer, input data is in 3D format and output data is in 2D format, this is because the outputs from the first hidden layer are as inputs to the second hidden layer. The dense layer which is the last layer from which the output was obtained, has 8 nodes.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 1, 1000)           4036000

 gru (GRU)                   (None, 1000)              6006000

 dense (Dense)               (None, 8)                 8008

=================================================================
Total params: 10,050,008
Trainable params: 10,050,008
Non-trainable params: 0
```

Figure 12 – Model summary (Hybrid LSTM+GRU)

It is clear that the total parameters of the hybrid model are less than LSTM, but more than the total parameters of the GRU model. Errors for prediction in Hybrid model shows the MAE =0.075 and RMSE= 0.117. We notice that the sum of errors in the results of the hybrid model is small, as well as the model training time and the number of epochs are very small which refers to perfect predication .

Predictive time for each region (Feature) in dataset based on the hybrid predication model is shown in figure 13.

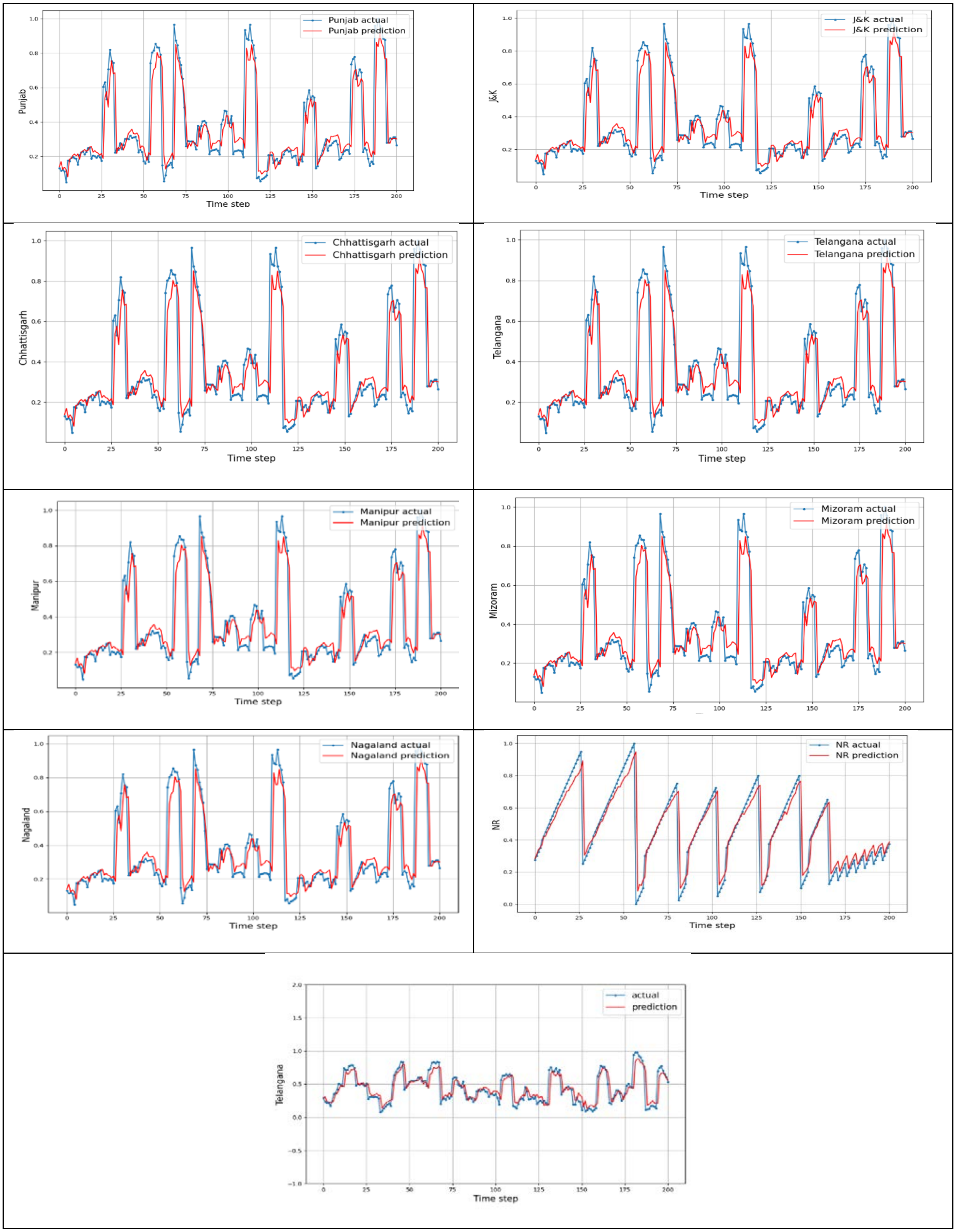

Figure 13 – Time Predicted Results in the predication model

By identifying the cities that have the highest rates of electrical energy consumption and the cities least consumption of electrical energy, we will have a clear vision for developing future plans to control energy consumption the most energy consuming cities.

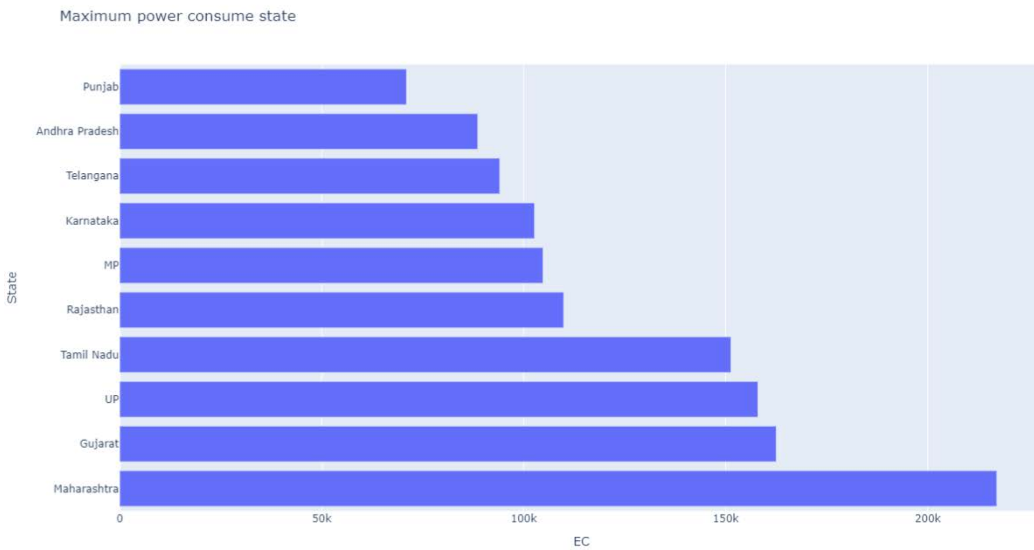

Figure 14 – Maximum of the Power Consumption

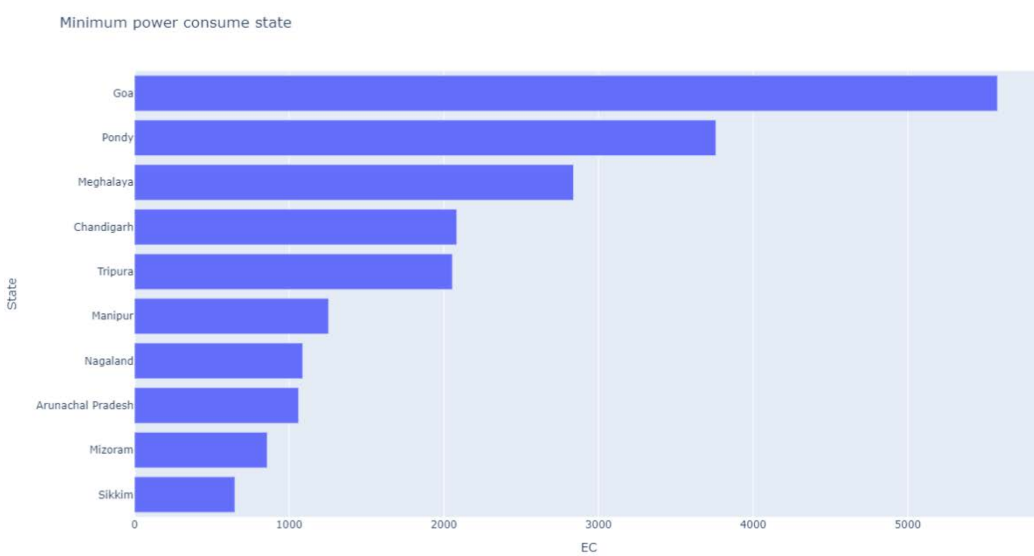

Figure 15 – Minimum of the Power Consumption

Figure bellow shows the data correlation between cities to understanding the relation between cities and power consumption.

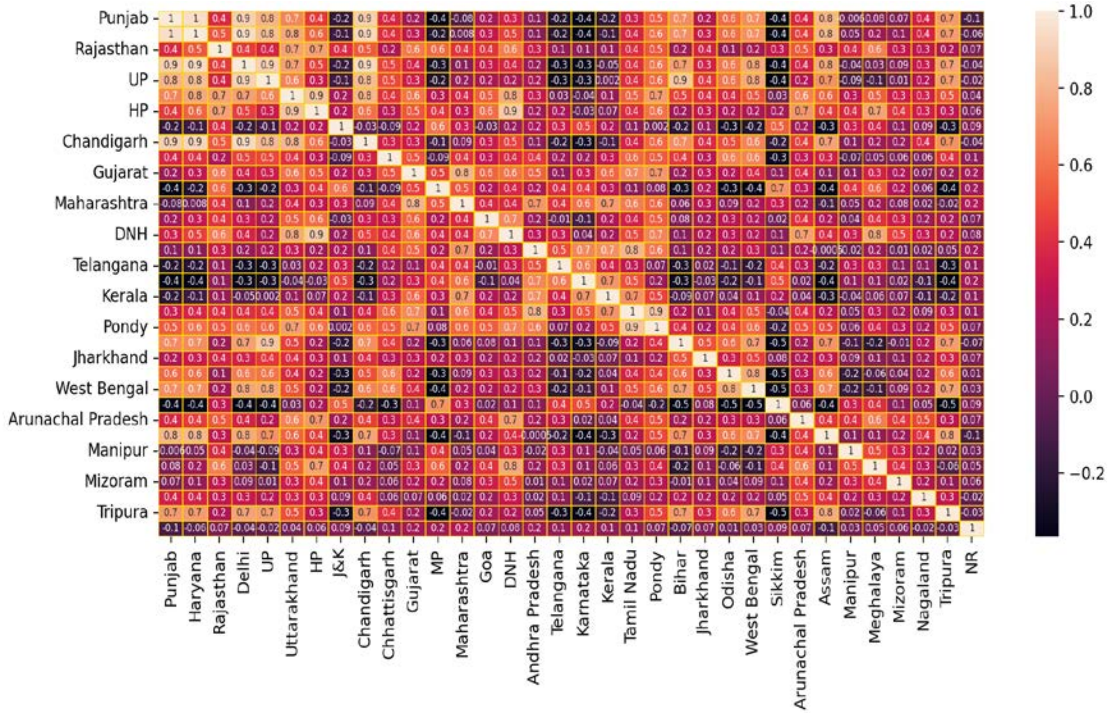

Figure 16 – The correlation between dataset

For the Hybrid model, as shown in figure 17, the loss is about 0.0057 for the training and 0.0121 for validation in 100 epoch by using MSE.

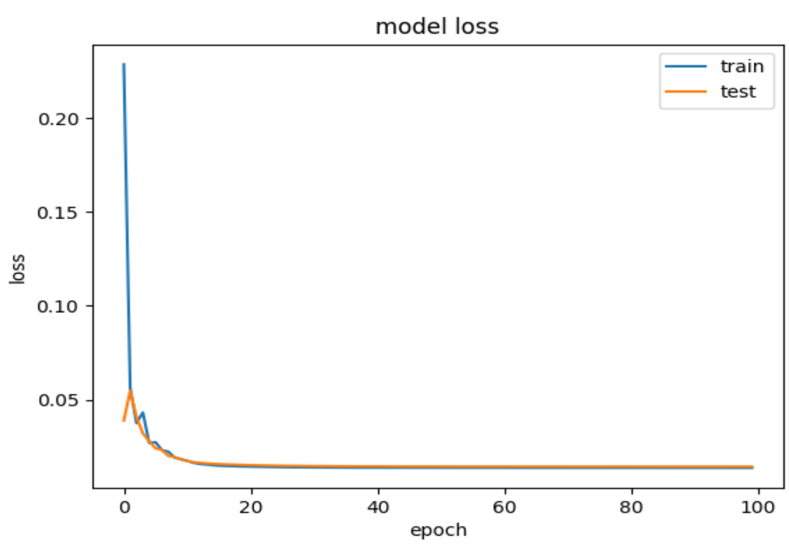

Figure 17 – The loss for the Hybrid model

5.2. Conclusion

In this chapter, we have looked at the analysis and understanding of IoT datasets using the python libraries and have also conducted testing for proposed hybrid prediction models using two deep learning models (LSTM and GRU) which returned the best accuracy value, where a value of MAE=0.75 and RMSE=0.117 which are optimal values compared with the original LSTM , the value of MAE of the LSTM = 1.98 and RMSE =1.023 . For that, we chose the hybrid model to power consumption prediction model. The prediction model is now ready for use by other developers.

6. Conclusion

This project undertakes a viable solution for the need of deep learning at the very basic level, that is, in low powered embedded devices. The project will enable us to help the institutions in India to give accurate indicators and predictions of the amount of electrical energy consumption for 13 cities in India and to know the most energy consuming city and thus help these institutions to develop future plans to control the consumption of electrical energy. It basically uses the power consumption dataset, which is carefully chosen and ideal for robotic applications.

The tasks solved in this thesis include

1. Development a library that allows for a high accuracy prediction of a real-world problem of power consumption of IoT data analysis power consumption in India. This is by developing a deep learning model that can predict different cities in the power consumption (2019-2020) dataset.
2. Development of a library that runs on devices that consume low power and have low processing capabilities by converting the deep learning model into a TensorFlow model that can run in very small devices.
3. Test the prediction model using error metrics and give an example of how to implement interfaces is added into the prediction model and can be accessed using the python language.
4. The prediction model using a real-world IoT power consumption dataset, named Power consumption in India (2019-2020) for analysis long-term time series data by using data mining techniques (handling missing data, series to supervised ,data normalization).
5. Development the libraries that apply data mining methods in the field of IoT by making accurate decisions to identify the cities that use the most electric energy

and help officials in developing future plans for optimal management of energy consumption in India.

6. Test two proposed prediction models LSTM and Hybrid (LSTM + GRU) based on the MAE and RMSE measures and compare the results of these models and choose the best model that got the least error in the prediction of power consumption.

Due to the successful running of the prediction model, the next steps will be to add the rest of the middle-term of time series in power consumption dataset as well will be done to tune the performance of the prediction model as shown below.

6.1. Optimizing Latency

Designing model architectures of   difficult and time-consuming, but there have recently been some advances in automating the process, such as Recurrent neural networks RNN models, using approaches like deep learning algorithms to improve network designs. These are still not at the point of entirely replacing humans.

I am therefore looking forward to using a ready deep learning method like LSTM and developing it by combining a LSTM with GRU that allows users to avoid many of the gritty details of training, be able to design the best possible model for your data and efficiency trade-offs solving latency issues.

6.2. Optimizing Power Usage

For this, I'll measure the latency for running one inference and multiply that by the system's average power usage for that period to get the energy usage. I can predict the time it will take to execute a model based on how many arithmetic operations it requires and how many operations per second a processor can perform.

6.3. Optimizing Model and Binary Size

Currently during training, weights are usually stored as floating-point values, taking up 4-bytes each in memory. Because space is such a constraint for embedded devices, I will use the compression utility in TensorFlow to I will used TanserFlow to supports deep learning. TanserFlow accepts data in the form of multi-dimensional arrays of higher dimensions called tensors. Multi-dimensional arrays are very handy in handling large amounts of data. TensorFlow works on the basis of data flow graphs that have nodes and edges. As the execution mechanism is in the form of graphs, it is much easier to execute TensorFlow code in a distributed manner across a cluster of computers while using GPUs. These codes are each stored in a byte, and arithmetic operations can be performed on them with a minimal loss of accuracy.

REFERENCES

1. Chammas, M. An efficient data model for energy prediction using wireless sensors / M. Chammas, A. Makhoul, J. Demerjian // Computers & electrical engineering. – 2019. – V.76. – P.249-57.
2. Runge, J. A Review of Deep Learning Techniques for Forecasting Energy Use in Buildings / J. Runge, R. Zmeureanu, // Energies. – 2021. – V.14. – P. 608.
3. Salam, A. Energy consumption prediction model with deep inception residual network inspiration and LSTM / A. Salam, A. El Hibaoui // Mathematics and Computers in Simulation. – 2021. – V.190. – P. 97-109.
4. Acakpovi, A. Time series prediction of electricity demand using Adaptive Neuro-Fuzzy Inference Systems / A. Acakpovi, A. T. Ternor, N. Y. Asabere, P., etc. // Mathematical Problems in Engineering. – 2020. – V.1. – P.14.
5. Tiewsoh, L. S. Electricity generation in India: Present state, future outlook and policy implications / L. S. Tiewsoh, J. Jirásek, M. Sivek // Energies. – 2019. - V.12. – P. 1361.
6. Ahmad, T. A critical review of comparative global historical energy consumption and future demand: The story told so far / T. Ahmad, D. Zhang // Energy Reports. – 2020. – V.6. – P.1973-1991.
7. Nugaliyadde, A. Predicting electricity consumption using deep recurrent neural networks / A. Nugaliyadde, U. Somaratne, K. W. Wong // arXiv preprint arXiv. – 2019. – V.1909. – P. 81-82.
8. Mohapatra, S. K. A Pragmatic Investigation of Energy Consumption and Utilization Models in the Urban Sector Using Predictive Intelligence Approaches / S. K. Mohapatra, S. Mishra, H. K. Tripathy, etc. // Energies. - 2021. – V.14. – P. 3900.

9. Biskin, O. T. Forecasting of Turkey's Electrical Energy Consumption using LSTM and GRU Networks / O. T. Biskin, A. Cifci // Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Dergisi. – 2021. – V. 8 – P. 656-667.
10. Shi, H. Deep learning for household load forecasting – A novel pooling deep RNN / H. Shi, M. Xu, R. Li // IEEE Transact. Smart Grid. – 2017. – V.9 – P. 5271–5280.
11. Estebsari, A. Single residential load forecasting using deep learning and image encoding techniques / A. Estebsari, R. Rajabi // Electronics 2020. – V.9 – P.68.
12. Bu, S. J. Time series forecasting with multi-headed attention-based deep learning for residential energy consumption / S. J. Bu, S. B. Cho // Energies. - 2020. – V. 13 – P. 4722.
13. Power consumption in India (2019-2020) [https://www.kaggle.com/twinkle0705/state-wise-power-consumption-in-india/activity] (accessed: 22.03.2022).
14. Abadi, M. Tensorflow: Large-scale machine learning on heterogeneous distributed systems / M. Abadi, A. Agarwal // Tensorflow arXiv preprint arXiv. – 2016. – V.1603. – P.4467.
15. Goldsborough, P. A tour of tensorflow / P. Goldsborough // arXiv preprint arXiv. – 2016. – V.1610. - P.1178.
16. Al Shalabi, L. Normalization as a preprocessing engine for data mining and the approach of preference matrix / L. Al Shalabi, Z. Shaaban // International conference on dependability of computer systems. – 2006. – P. 207-214
17. Han, J. Data mining: concepts and techniques / J. Han, J. Pei, M. Kamber // Elsevier. – 2011. – P.1.
18. Barzola-Monteses, J. Energy consumption of a building by using long short-term memory network: a forecasting study / J. Barzola-Monteses, M. Espinoza-

Andaluz, M. Mite-León // International Conference of the Chilean Computer Science Society. – 2020. – P. 1-6.
19. Somu, N. A hybrid model for building energy consumption forecasting using long short term memory networks / N. Somu, K. Ramamritham // Applied Energy. – 2020. – V. 261. – P.114131.
20. Luo, X. J. Forecasting building energy consumption: Adaptive long-short term memory neural networks driven by genetic algorithm / X. J. Luo, L. O. Oyedele // Advanced Engineering Informatics. – 2021. – V. 50. – P. 101357.
21. Wang, B. Air quality forecasting based on gated recurrent long short term memory model in Internet of Things / B. Wang, W. Kong // Access. – 2019. – V. 7. – P. 69524-69534.
22. Kala, K. U. Gated recurrent unit architecture for context-aware recommendations with improved similarity measures / K. U. Kala, M. Nandhini // Transactions on Internet and Information Systems. – 2020. – V. 14. – P. 538-561.
23. Kassa, Y. Short term wind power prediction using ANFIS / Y. Kassa, J. H. Zhang // International conference on power and renewable energy. – 2016. – P. 388-393.
24. Dhupia, B. Ensemble Machine Learning Modelling for Medium to Long Term Energy Consumption Forecasting / B. Dhupia // Turkish Journal of Computer and Mathematics Education. – 2021. – V. 12. – P. 459-463.