A Hybrid Deep Learning Approach for Load Shedding Forecasting in Power Grids Using Hybrid LSTM-GRU Models

Submitted in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering of the University of Asia Pacific

by

Md. Abdullah Al Ikram
Roll: 21201007

Tonoya Akhter Adhora
Roll: 21201019

Abdul Rizvan Alvi
Roll: 21201023

Supervised by:
Dr. Homeyra Akter
Assistant Professor

Co-supervised by:
Alif Ruslan
Lecturer

Department of Computer Science & Engineering
University of Asia Pacific

November 2025

Declaration

We, hereby, declare that the work presented in this Thesis is the outcome of the investigation performed by us under the supervision of Dr. Homeyra Akter, Assistant Professor, Department of Computer Science, University of Asia Pacific. We also declare that no part of this Thesis and thereof has been or is being submitted elsewhere for the award of any degree or Diploma.

Countersigned

____

Dr. Homeyra Akter
Supervisor

Signature

....

Md. Abdullah Al Ikram

....

Tonoya Akhter Adhora

....

Abdul Rizvan Alvi

Certificate of Approval

We hereby recommend that the thesis prepared by Md. Abdullah Al Ikram, Tonoya Akhter Adhora, Abdul Rizvn Alvi entitled "A Hybrid Deep Learning Approach for Load Shedding Forecasting in Power Grids Using Hybrid LSTM-GRU Models" is accepted as fulfilling the part of the requirements for degree of Bachelor of Science in Computer Science and Engineering.

....

Dr. Homeyra Akter
Assistant Professor
Department of Computer Science and Engineering
University of Asia Pacific (UAP)

Chairman of the Committee
(Supervisor)

....

Alif Ruslan
Lecturer
Department of Computer Science and Engineering
University of Asia Pacific (UAP)

Member of the Committee
(External)

....

Prof. Dr. Engr. Shah Murtaza Rashid Al Masud
Head of the Department
Department of Computer Science and Engineering
University of Asia Pacific (UAP)

Head of the Department

Acknowledgements

First and foremost, we would like to express our deepest gratitude to the Almighty Allah. It is through His blessings that we have been able to complete this work with ease, as He granted us the strength, ability, and opportunity to do so.

We would also like to extend our heartfelt thanks to our esteemed supervisor, Dr. Homayra Akhter. Despite her numerous commitments, she dedicated ample time to guide us throughout this work. Her unwavering support, insightful guidance, and invaluable advice, especially during challenging moments, played a critical role in the successful completion of this research. Her thoughtful comments and consistent mentorship have significantly contributed to the quality of this thesis.

Our sincere gratitude goes to Alif Ruslan, our respected teacher, whose inspiration and encouragement have been a constant source of motivation throughout our academic journey. We are also thankful to all our professors and teachers, who have provided us with various resources and continuous moral support, thereby enriching our academic experience.

Lastly, we wish to express our deepest appreciation to our families, whose love, patience, and unwavering support have been our greatest strength. Their encouragement and presence have been instrumental in every step of our lives.

Abstract

The increasing frequency of electricity load shedding in rapidly developing regions, particularly in Dhaka City, has prompted the need for reliable forecasting systems to mitigate its adverse impacts. Traditional methods have struggled to predict both the occurrence and magnitude of load shedding events due to their inability to capture complex, non-linear temporal dependencies. This project aims to develop a Hybrid Deep Learning model, combining Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM) architectures, for accurate short-term load shedding forecasting. The system utilizes historical data on electricity demand, generation, and weather conditions from the Power Grid Company of Bangladesh (PGCB) to predict both the likelihood of load shedding events (classification) and their magnitude (regression). The proposed model demonstrated superior performance, achieving a classification success rate of 91.08% and low error metrics (MAE: 7.49 MW, RMSE: 28.95 MW) in regression tasks. A comparison with baseline models such as Pure GRU and Pure LSTM further validated the effectiveness of the Hybrid LSTM-GRU model in handling both short-term and long-term dependencies. This research provides valuable insights into the application of advanced deep learning techniques in energy grid management and contributes to the development of more efficient and reliable load-shedding prediction systems, with potential applications in improving power grid stability and reducing societal disruption in developing countries. Future work will focus on integrating real-time data and exploring more advanced machine learning architectures to enhance forecasting accuracy and scalability.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter presents the background, motivation, and literature review for the project, focusing on short-term electricity demand forecasting and load shedding prediction. It provides a foundation for understanding the key concepts, methodologies, and previous studies in this area. The review identifies the strengths and weaknesses of existing systems, which motivates the development of a more robust solution.

1.1 Background and Motivation

Electricity load forecasting is a critical task for maintaining the stability of power grids, especially in rapidly developing regions like Dhaka City, where demand often outpaces supply. Load shedding, a strategy used to manage power shortages, causes significant disruption to both households and industries. Traditional forecasting methods often fall short in predicting both the occurrence and magnitude of load shedding due to the complexity and non-linearity of the factors involved, such as weather patterns and seasonal demand fluctuations. This challenge has led to increased interest in using advanced machine learning and deep learning techniques to improve prediction accuracy. Recent studies have explored the use of hybrid deep learning models, such as Convolutional Neural Networks (CNN) combined with Long Short-Term Memory (LSTM) or Gated Recurrent Unit (GRU) networks, to capture both short-term and long-term dependencies in time-series data, which are critical for accurate load shedding prediction [1][2].

1.2 Problem Statement

Despite advancements in load forecasting, there remains a gap in the development of a unified model capable of accurately predicting both the occurrence (classification) and the magnitude (regression) of load shedding events. Existing models either focus on one aspect either forecasting the event or predicting the magnitude but fail to provide a comprehensive solution that addresses both challenges simultaneously. Moreover, many existing approaches struggle to handle the complex, non-linear relationships and temporal dependencies inherent in power grid data, limiting their effectiveness in real-world applications, particularly in developing countries like Bangladesh. This research seeks to fill this gap by developing a hybrid deep learning model that combines GRU and LSTM architectures for improved accuracy and computational efficiency in forecasting both the occurrence and magnitude of load shedding events.

1.3 Objectives of the Project

The main objectives of this project are as follows:

- To develop a hybrid GRU-LSTM deep learning model for short-term load shedding forecasting.
- To predict the binary occurrence of load shedding events (classification) and the required magnitude (regression) using historical data from the Power Grid Company of Bangladesh (PGCB).
- To evaluate the performance of the proposed model by comparing it with traditional forecasting models, such as Naïve forecasting, based on key metrics such as accuracy, F1 Score, Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE).
- To provide actionable insights that can aid in improving power grid management and reducing the socio-economic impact of load shedding in Dhaka City.

1.4 Research Questions / Hypotheses

This project is guided by the following research question and hypotheses:

- Can a hybrid GRU-LSTM deep learning model outperform standalone forecasting models in predicting both the occurrence and magnitude of electricity load shedding events?
  - The proposed hybrid model, by leveraging the short-term learning efficiency of GRU and the long-term memory of LSTM, will demonstrate lower error metrics (MAE and RMSE) and a higher classification success rate compared to traditional models, such as Naïve forecasting.

1.5 Scope of the Work

The scope of this work is limited to the development and validation of the hybrid LSTM- GRU model using historical operational data provided by the PGCB, covering the period from July 2024 to July 2025. The dataset includes features such as electricity demand, generation, weather conditions, and past load shedding events. The focus is on predictive modeling, and the study does not extend to the integration of the model into real-time power grid systems or the design of the physical infrastructure for power distribution. Additionally, the performance evaluation is constrained by the quality and temporal resolution of the available dataset.

1.6 Expected Outcomes

The expected outcomes of this project include:

- A robust hybrid LSTM-GRU model capable of accurately forecasting load shedding events, demonstrated by a high classification success rate (targeting 90% or above) and low regression error metrics (targeting MAE < 10 MW and RMSE < 35 MW).
- A detailed comparison of the hybrid model's performance against baseline forecasting models, proving its superiority in handling both classification and regression tasks.
- A modular and reproducible Python codebase that can be adapted for other forecasting applications in energy management.

- Insights into the effectiveness of advanced deep learning techniques for improving power grid management in Bangladesh.

1.7 Impacts of the Project

The impact of this project is multifaceted, touching on societal, health, safety, legal, and environmental issues. The ability to forecast load shedding events accurately will enhance power grid management, reduce unnecessary disruptions, and improve overall public satisfaction. This will contribute to the socioeconomic stability of the region and support the efficient use of available energy resources.

1.7.1 Impact on Societal and Cultural Issues

Accurate load shedding forecasts will provide grid operators with enough lead time to inform the public about planned outages, reducing frustration and helping businesses and households plan accordingly. This will improve the public's trust in the electricity supply system and reduce social disruption.

1.7.2 Impact on Health, Safety, and Legal Issues

Reliable forecasting of load shedding will enhance public safety by preventing unexpected outages that could damage sensitive equipment or disrupt critical services, such as hospitals and emergency responders. It will also help utility companies meet regulatory requirements by providing transparent, accurate outage information, thus reducing consumer disputes and legal challenges

1.7.3 Impact on Environment and Sustainibility Issues

By improving the accuracy of load shedding predictions, the model will help optimize power generation and reduce reliance on less efficient, high-emission backup power plants, particularly during peak demand periods. This can contribute to a more sustainable and environmentally friendly energy system, aligning with global efforts to reduce carbon emissions and improve energy efficiency.

1.8 Report Organization

The remainder of this report is structured as follows:

- Chapter 2: Background and Literature Review — reviews key concepts, previous systems, and research gaps.
- Chapter 3: System Analysis and Design — presents requirements, architectural design, and diagrams.
- Chapter 4: Methodology and Implementation — describes technologies, modules, and calibration workflows.
- Chapter 5: Results and Evaluation — covers experiments, system testing, and analysis of results.
- Chapter 6: Time and Cost Analysis — outlines project timeline, budget, and resource distribution.
- Chapter 7: Design Constraints and Standards — documents compliance with professional and safety guidelines.
- Chapter 8: Conclusion and Future Work — summarizes findings and suggests future improvements.

Chapter 2

Background and Literature Review

This chapter provides the theoretical foundation for the research by exploring key concepts and reviewing existing literature related to short-term electricity demand forecasting and load shedding prediction. It begins with an introduction to fundamental theories and terminologies, such as time-series forecasting, Convolutional Neural Networks (CNN), and Long Short-Term Memory (LSTM) networks, which are essential for understanding the project. The chapter then surveys previous research and existing systems, categorizing studies by the techniques used, including classical machine learning models, deep learning approaches, and hybrid models. A comparative analysis highlights the strengths and weaknesses of these methods, emphasizing the limitations in accurately predicting both the occurrence and magnitude of load shedding. The chapter concludes by identifying research gaps, particularly the need for a unified forecasting model that addresses both classification and regression tasks simultaneously, which motivates the proposed solution of a hybrid GRU-LSTM model.

2.1 Preliminaries

To understand the forecasting model proposed in this study, it is essential to comprehend some key concepts and methodologies used in short-term electricity demand forecasting. This section introduces the core terminologies, models, and frameworks that underpin the project.

In the context of electricity load forecasting, time-series data refers to a sequence of data points indexed in time order. Electricity demand is often observed at regular intervals, such as hourly or daily, and its prediction is crucial for grid management.

Time-series forecasting involves using historical data to predict future values by identifying patterns and dependencies.
CNNs are deep learning models that are particularly effective in capturing spatial features in data. Although they are traditionally used in image processing, CNNs have been successfully applied to time-series data by treating temporal patterns as spatial features [1]. CNNs are capable of identifying local patterns and trends in electricity demand, such as spikes in usage during specific hours or weather patterns.
LSTM is a type of Recurrent Neural Network (RNN) designed to handle long-term dependencies in sequential data. It is highly effective for time-series forecasting, where past data points significantly influence future outcomes. LSTM networks excel at capturing both short-term and long-term dependencies, which is crucial in predicting electricity demand that fluctuates over time [2].
Similar to LSTM, GRU is another variant of RNN. It is computationally more efficient than LSTM due to its simpler architecture. GRUs are particularly useful for tasks requiring the capture of short-term dependencies and are often used in combination with LSTM for hybrid models to leverage both short-term and long-term memory capabilities [3].
Hybrid models combine different types of neural networks or machine learning algorithms to enhance forecasting accuracy. In this study, a combination of CNN with stacked Bi-LSTM is used to capture both spatial and temporal dependencies in the electricity demand data [1]. The integration of multiple models helps mitigate the individual weaknesses of each model, providing a more robust solution for forecasting.

2.2 Related Works / Existing Systems

Several studies have explored various methods for short-term electricity demand forecasting, utilizing different machine learning and deep learning techniques. These studies can be broadly classified into those using classical machine learning models, deep learning models, and hybrid models.
Traditional methods for electricity demand forecasting include regression models, support vector machines (SVM), and decision trees. These models typically struggle to capture the complex, non-linear relationships in time-series data. However, they are often simpler to implement and computationally less expensive. Studies such as those

by [4] have shown the limitations of classical models in handling high-dimensional data from smart grids, especially when the demand fluctuates due to weather conditions or seasonal factors

Recent studies have increasingly adopted deep learning techniques, especially LSTM and GRU networks, for time-series forecasting due to their ability to capture long-term temporal dependencies. For example, the study by [5] used LSTM networks to predict electricity demand, achieving a low Mean Absolute Percentage Error (MAPE) of 0.78

Combining CNN with RNN architectures, such as LSTM and GRU, has shown promising results in forecasting electricity demand. For example, [10] introduced a hybrid CNN-GRU model for short-term load forecasting, achieving improved accuracy in comparison to standalone CNN or GRU models. The CNN module captures the local features and trends, while the GRU model handles the sequential dependencies, thus providing a more comprehensive forecasting solution. This hybrid approach has been increasingly used to enhance predictive accuracy in complex systems like energy forecasting [8].

The Prophet Model, developed by Facebook, has also been used for demand forecasting in smart grids [3]. It is based on an additive model where components such as trends, seasonality, and holidays are captured separately. While Prophet provides a simple and interpretable framework for time-series forecasting, it is often less accurate compared to deep learning models, particularly when dealing with more complex datasets.

2.3 Comparative Analysis of Related Works

The reviewed literature demonstrates a growing trend toward the use of deep learning models, particularly hybrid models, for electricity demand forecasting. However, the performance of these models varies significantly across different approaches. Standalone deep learning models, such as LSTM and GRU, excel in capturing long-term dependencies in time-series data but may struggle to effectively model short-term trends. This limitation is evident in studies like [5] and [6], where standalone models showed suboptimal performance for certain forecasting tasks.

On the other hand, hybrid models, which combine different deep learning techniques, are better equipped to address this issue. By integrating models such as CNN and GRU, hybrid architectures can capture both short-term fluctuations and long-term dependencies, improving overall forecasting accuracy. For example, the CNN-GRU hybrid model proposed by [10] significantly outperformed its individual components, showcasing the effectiveness of combining multiple architectures to enhance predictive power.

In contrast, classical machine learning models like Support Vector Machines (SVM) and regression techniques generally exhibit lower accuracy when applied to complex time-series data, as they are not well-suited to handle the intricate patterns and dependencies inherent in such data. While these models are less computationally intensive, making them suitable for simpler datasets or scenarios where computational resources are constrained, they are often outperformed by more advanced deep learning models in terms of forecasting precision and robustness.

2.4 Research Gap / Limitations of Existing Methods

Although many studies have successfully applied machine learning and deep learning models for load forecasting, several gaps remain. One major limitation is the difficulty of modeling both the occurrence and magnitude of load shedding in a single system. Existing research has largely focused on predicting electricity demand or load shedding independently, without considering both tasks simultaneously. Moreover, while many models show promising results, their performance can degrade when applied to real-world, noisy, or incomplete data, as evidenced in the study by [6]. Additionally, scalability remains a challenge, particularly when incorporating larger datasets or real-time data from smart grids [7].

Table 2.1: Overview of Literature Reviewed Works

Research	Year	Dataset	Modeling Technique	Limitation
Kazi Fuad Bin Akhter, et al. [1]	2024	Electricity demand from Dhaka	CNN + Stacked Bi-LSTM	Limited to specific time-series data, lacks real-time adaptability.
Anik Baul, et al. [2]	2024	6 years of daily consumption data	Hybrid CNN + Bi-LSTM	High computational cost, lacks integration with real-time grid data.
Sanju Kumari, et al. [3]	2022	Smart meter data	Prophet Model	Assumes consistent historical data, limited to seasonal patterns.
Khairul Eahsun Fahim, et al. [4]	2024	Smart Grid energy consumption data	Distributed Deep Learning (DDL) + HSIC	Scalability issues, complex model training for large datasets.
Seyed Mohammad Sharif Hosseini, et al. [5]	2024	Electricity grid data	LSTM, GRU	Performance drops with missing or noisy data.
Javier Manuel Aguiar-Pérez, et al. [6]	2023	Smart grid consumption data	Deep Learning-based Demand Forecasting	Limited focus on larger datasets, lacks multi-region forecasting.
Salman Ali, et al. [7]	2024	Publicly available datasets	CNN + GRU hybrid model	Inadequate for predicting long-term trends, computationally expensive.
Syed Muhammad Hasanat, et al. [10]	2024	AEP, ISONE datasets	CNN + GRU hybrid mode	Performs poorly with seasonal fluctuations, requires large data sets.
Venkataramana Veeramsetty, et al. [11]	2021	Load data from substation	GRU + Random Forest (RF)	Limited to specific grid substation, lacks flexibility in data types.
Fabiano Pallonetto, et al. [8]	2021	Commercial building data	CNN + GRU + Attention Mechanism	Not scalable for large regional grids, lacks real-time prediction.

2.5 Summary

The literature review demonstrates that deep learning models, particularly hybrid approaches, have shown significant promise in electricity demand forecasting and load shedding prediction. However, gaps remain in simultaneously addressing both classification and regression tasks, as well as improving model scalability and robustness in the face of noisy real-world data. The proposed hybrid LSTM-GRU model aims to bridge these gaps, offering a more efficient and accurate solution for short-term load shedding forecasting. The next chapter will delve into the design and methodology of the proposed model, highlighting its advantages over existing systems

Chapter 3

System Analysis and Design

This chapter provides a comprehensive overview of the system analysis and design process for the load shedding forecasting system. It outlines the functional and non-functional requirements derived from system needs and user expectations, followed by a detailed description of the system architecture and design considerations. The purpose of this chapter is to define the system's components and structure, ensuring they align with the project objectives and deliver the required functionalities.

3.1 Requirement Analysis

This section should present a comprehensive analysis of the **functional** and **non-functional** requirements derived from user and system needs. The requirements are derived from the real-world dataset provided by the Power Grid Company of Bangladesh (PGCB) and aim to develop a forecasting system that uses a hybrid LSTM-GRU model to predict load shedding events. The model's focus is on forecasting both the occurrence and magnitude of load shedding, an area that has received less attention in existing research, which predominantly centers on demand response rather than load shedding.

Guidelines for Requirement Identification

As no direct stakeholder interactions were conducted, the requirements were derived from the following sources:

- System Needs and Objectives: The core goal of this system is to predict load shedding events based on historical data of energy generation, demand, and weather conditions. While much of the existing research has focused on demand response, which involves energy consumption adjustments, load shedding prediction remains an underexplored area, particularly in the context of Dhaka city. Therefore, this research aims to bridge that gap by forecasting load shedding events using advanced LSTM and GRU models.

- Review of Existing Solutions: Existing literature on load forecasting generally addresses demand-side management and demand response, with an emphasis on reducing or shifting demand based on grid conditions. However, load shedding prediction, especially in developing cities like Dhaka, has received limited attention. This research intends to focus on this gap, using machine learning models to predict load shedding events.

- System Design Considerations: The system must handle time-series data from PGCB, preprocess it efficiently, and generate real-time predictions of load shedding events. It should manage large datasets and provide actionable predictions that aid in grid management and decision-making.

Functional and Non-Functional Requirements

Based on the above guidelines, the following functional and non-functional requirements have been identified for the load shedding forecasting system:

Functional Requirements

These requirements define the specific functionalities that the system must provide to meet user and system needs:

ID	Requirement	Description
FR1	Data Preprocessing	The system must clean and preprocess the PGCB dataset, handle missing values, normalize data, and create derived features such as lag features and rolling averages to capture temporal patterns.
FR2	Model Training	The system must train a hybrid LSTM+GRU model using the preprocessed PGCB data, optimizing the model's hyperparameters for better performance in predicting load shedding.
FR3	Load Shedding Classification	The system must classify whether load shedding will occur based on the input data (binary classification).
FR4	Load Shedding Magnitude Prediction	The system must predict the magnitude of load shedding events using a regression model (continuous value).
FR5	Model Evaluation	The system must evaluate the trained models using performance metrics such as accuracy, MAE, RMSE, and R^2, and provide evaluation results for monitoring.
FR6	Visualization of Predictions	The system must display visualizations comparing actual and predicted load shedding events using graphs such as line charts and zoomed-in plots.
FR7	Real-time Prediction	The system must allow for real-time predictions, enabling forecasting of future load shedding events as new data is input into the system.

Table 3.1: Functional Requirements for Load Shedding Forecasting System

Non-Functional Requirements

These non-functional requirements define the quality attributes of the system, focusing on performance, reliability, and other key factors:

ID	Requirement	Description
NFR1	Performance and Efficiency	The system must efficiently process large-scale datasets provided by PGCB and generate predictions in real-time, ideally under 10 seconds per prediction.
NFR2	Scalability	The system must be scalable to accommodate future increases in data volume, such as adding additional weather-related features or expanding the time period of historical data.
NFR3	Reliability and Accuracy	The system must provide accurate and reliable predictions with a high success rate (e.g., 90 or higher) for both classification and regression tasks related to load shedding prediction.
NFR4	Usability	The system must be user-friendly, providing easy-to-interpret visualizations and enabling administrators to monitor model performance and make necessary adjustments.
NFR5	Security	The system must ensure the security of the data, particularly grid operation information that could be sensitive.
NFR6	Maintainability and Extensibility	The system must be easy to maintain and extend for future improvements, such as incorporating additional sources of data or enhancing the model with advanced techniques.
NFR7	Compatibility	The system should be compatible with standard data formats from PGCB (e.g., CSV, SQL) and integrate seamlessly with existing energy grid management systems.

Table 3.2: Non-Functional Requirements for Load Shedding Forecasting System

Requirement Refinement and Validation

Since no formal stakeholder validation was conducted, the following steps were taken to ensure the requirements were well-defined and traceable to project objectives:

- Refinement: Functional requirements were continuously refined based on the dataset's available features and the project's goals. Non-functional requirements were defined based on performance expectations, particularly in terms of real-time processing and prediction accuracy.
- Validation: Requirements were validated by referencing existing research on time-series forecasting and demand response, aligning them with load shedding prediction needs in Dhaka city. Moreover, the system's design aligns with the broader objective of improving power grid management by predicting load shedding events

Evidence and Documentation

Since no direct stakeholder feedback was collected, the primary sources for requirement identification are:

- PGCB Dataset: The publicly available dataset provided by the Power Grid Company of Bangladesh, which includes historical time-series data on energy generation, demand, and weather conditions..
- Literature Review: Studies on time-series forecasting, LSTM and GRU models in energy systems, and demand response were reviewed to ensure the system aligns with state-of-the-art forecasting practices, while differentiating it from existing demand-side management solutions.
- System Needs: The requirements reflect the project's aim to forecast load shedding and optimize energy grid distribution in Dhaka, an area that has not been fully explored in prior research.

3.2 System Overview / Architecture

This section provides a high-level overview of the system architecture, describing how the various components work together to achieve the project objectives. It outlines the system's functional modules, data flow, and the interactions between components. The goal is to provide a clear understanding of the system structure and how it supports the forecasting and prediction tasks.

3.2.1 High-Level Architecture

The system architecture consists of multiple modules that interact seamlessly to process the input data, perform feature engineering, and train a hybrid machine learning model for forecasting load shedding events. Each component of the architecture performs a distinct task, and the system is designed to be modular, facilitating scalability and ease of maintenance.

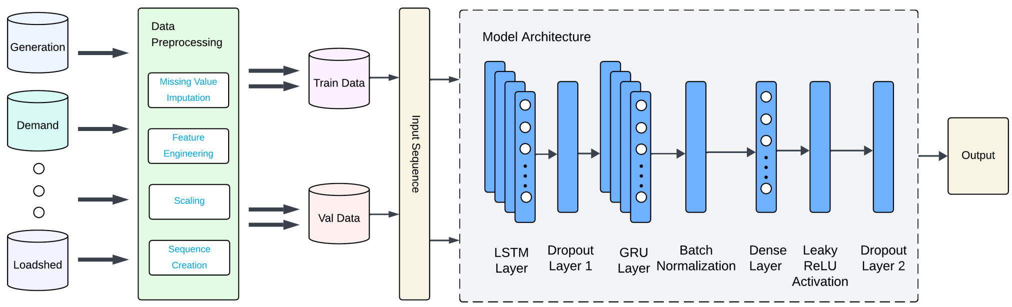

Figure 3.1: High-Level System Architecture

This diagram provides a top-down understanding of the system's design, emphasizing how data flows between components:

Data Preprocessing Module: The first module is responsible for loading and preprocessing the input data. The dataset contains time-series data for electricity demand, generation, weather data (e.g., temperature, humidity, wind speed), and the target variable: load shedding. The preprocessing steps include:

- Data Cleaning: Handling missing values through forward and backward filling techniques.
- Feature Engineering: Creating derived features such as the difference between demand and generation, and lag features (e.g., `loadshed_lag1`, `loadshed_lag3`) to capture temporal dependencies.
- Rolling Features: Calculating rolling averages over 3-hour and 6-hour windows for demand, generation, and difference data to smooth trends.
- Normalization: Scaling all features using MinMaxScaler to normalize them to a range between 0 and 1, ensuring equal contribution to model training.

Data Preprocessing Module: This module prepares the data for time-series forecasting by reshaping it into sequences of 24-time steps (i.e., 24 hours of past data for each prediction). The sequence creation is done using a sliding window approach to create the following sequences:

- Sliding Window: Generates sequences of input features (X), the binary classification target (Class), and the regression target (e.g., `loadshed_lag1`, `loadshed_lag3`).
- Train-Test Split: Data is split into training and testing datasets in an 80/20 ratio, ensuring that the test set includes a proportion of non-zero load shedding events

Data Preprocessing Module: The hybrid GRU-LSTM architecture is used to model both classification and regression tasks:

- Classification Model (Hybrid LSTM + GRU): The model uses GRU layers to capture short-term dependencies and LSTM layers for long-term dependencies. It outputs a binary prediction indicating whether load shedding will occur.
- Regression Model (Hybrid LSTM + GRU for Non-Zero Loadshed): This model is trained on data where load shedding has occurred (non-zero events). It predicts the magnitude of load shedding in MW, which is then rescaled using an exponential transformation.

Output and Evaluation Module: Once the models are trained, the system generates predictions and evaluates the results:

- Classification Accuracy: Evaluating the proportion of correct predictions for load shedding events.
- Regression Metrics: Using MAE, RMSE, and R^2 to assess the accuracy of predicted load shedding values.
- Success Rate: Assessing the model's ability to predict load shedding within an acceptable error margin ($\pm 10\%$ or ± 30 MW).

3.2.2 Detailed Design Diagrams

To better illustrate and clarify the architecture of the system, this section provides detailed design diagrams that depict how data flows through the system, as well as the architecture of the hybrid GRU-LSTM model. These diagrams serve to enhance understanding of the interactions between different modules and the structure of the model. Each diagram has been included to aid in visualizing the system's design and provide a granular view of how data is processed, from input to output.

- Data Flow Diagram (DFD): Represent the flow of data across different system processes.

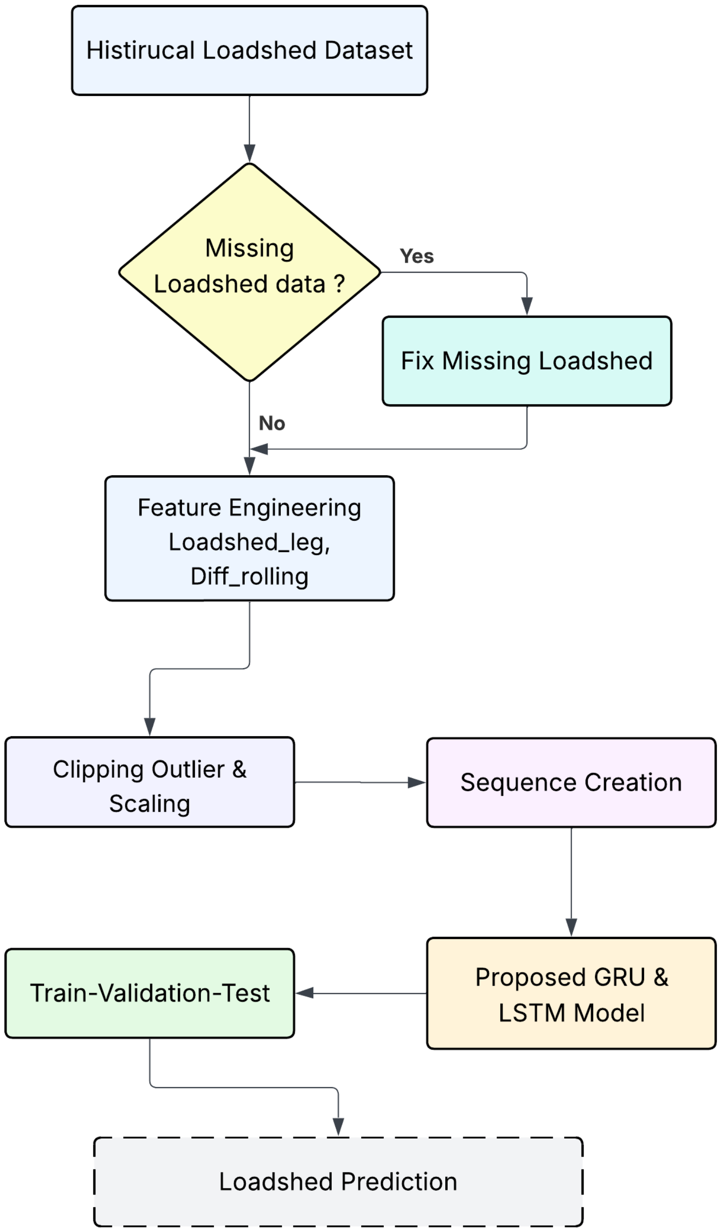

Figure 3.2: LSTM-GRU System Data Flow Diagram

- Input Data: The system starts with raw time-series data, including electricity demand, generation, weather data, and historical load shedding events. These data are collected from the Power Grid Company of Bangladesh (PGCB).

- Data Preprocessing: The first major module handles data preprocessing, including cleaning missing values, normalizing data, and creating derived features. For example, the system might calculate lag features such as previous demand, generation, and weather conditions to help the model understand temporal dependencies.

  – Feature Engineering: After preprocessing, features such as rolling averages, differences between demand and generation, and temperature adjustments are calculated. These features are vital for capturing trends and dependencies in the time-series data.

  – Sequence Creation: The system uses a sliding window technique to create sequences from the preprocessed data. These sequences consist of 24-time steps, which help the system predict future load shedding events based on the previous 24 hours.

  – Modeling and Prediction: The next step involves inputting the processed data into the hybrid LSTM + GRU model. The GRU layer captures short-term dependencies, while the LSTM layer handles long-term dependencies. The model outputs two types of predictions: one for classification (whether load shedding will occur) and one for regression (the magnitude of load shedding).

  – Output: The final output consists of both binary classification (indicating if load shedding will occur) and regression values (predicting the amount of load shedding in MW). The system can then display these predictions through real-time visualizations.

3.2.3 Design Alternatives and Rationale

During the design process, several alternative architectures were considered for the load shedding forecasting problem. Each alternative was evaluated based on several criteria, including performance, computational cost, scalability, and ease of implementation.

- Balanced Performance: The hybrid model effectively captures both short-term and long-term dependencies in the data, making it well-suited for predicting load shedding events.

- Computational Efficiency: While LSTM is computationally intensive, the inclusion of GRU reduces training time without sacrificing performance, providing an optimal balance between speed and accuracy.

- Scalability and Modularity: The model's architecture is scalable and can be easily adjusted or extended for future improvements, such as incorporating additional features or increasing the number of layers.
- Proven Effectiveness: Preliminary results from the hybrid LSTM + GRU model showed high accuracy and robust performance, with a 90.9% success rate in predicting load shedding events.

After thorough analysis, the final architecture provides an optimal balance between accuracy, computational cost, and scalability, making it the best choice for the load shedding prediction task.

3.3 Algorithmic Flow

This section outlines the step-by-step algorithmic flow of the load shedding forecasting system, detailing how data is processed from input to output. The flowchart below illustrates the sequence of operations, including data preprocessing, feature engineering, model training, and prediction generation.

Algorithm 1 Enhanced Hybrid LSTM-GRU Classification and Regression Pipeline

Input: Dataset $D(x_1, \ldots, x_N)$; define X, y_{class}, y_{reg}, K segments, $S = \text{int}(N/K)$
for $i = 0$ to $K - 1$ **do**
  $D_k \leftarrow (x_i, \ldots, x_{i+S})$
  Compute rolling means; create X_{seq}
  Classification: LSTM(128)→Dropout(0.4)→GRU(128)→Dense(32)→Sigmoid
  Compile/train: Binary cross-entropy, 100 epochs
  if non-zero y_{class} samples exist **then**
    Regression: LSTM(128)→Dropout(0.4)→GRU(128)→BatchNorm→Dense(32)→Linear
    Compile/train: Huber loss, 120 epochs
  else
    Skip regression
  end if
end for
Prediction: $y_{class\_pred} \leftarrow \text{ClassificationModel}(X)$; $y_{class\_bin} \leftarrow (y_{class\_pred} \geq 0.5)$
$y_{reg\_rescaled} \leftarrow \text{RegressionModel}(X)$ if trained, else 0
$y_{final} \leftarrow y_{class\_bin} \cdot y_{reg\_rescaled}$
Compute inverse transform: $y_{test\_rescaled}$
success_rate $\leftarrow 100 \cdot \text{mean}(|y_{final} - y_{test\_rescaled}| \leq \max(0.1 \cdot y_{test\_rescaled}, 30))$
Print "Enhanced Hybrid GRU-LSTM Success Rate = success_rate%"

The algorithm presented in Algorithm 1 outlines the step-by-step process for implementing the enhanced hybrid GRU-LSTM model for load shedding forecasting. The algorithm is divided into several key phases: data segmentation, model training for classification and regression, and the prediction phase. Below is a detailed explanation of each step:

- **Input Data:** The algorithm begins by taking the dataset D containing N samples. The input features X, classification target y_{class}, and regression target y_{reg} are defined. The dataset is divided into K segments, with each segment containing $S = \text{int}(N/K)$ samples.

- **Data Segmentation:** A loop iterates over each segment of the dataset. For each segment D_k, rolling means are computed to create the sequence input X_{seq}, which captures temporal dependencies in the data.

- **Classification Model Training:** The classification model is defined using a hybrid architecture consisting of LSTM and GRU layers, followed by dense and sigmoid layers for binary classification. The model is compiled using binary cross-entropy loss and trained for 100 epochs.

- **Regression Model Training:** If there are non-zero samples in the classification target, the regression model is defined similarly, with an additional batch normalization layer. The regression model is compiled using Huber loss and trained for 120 epochs. If no non-zero samples exist, the regression step is skipped.

- **Prediction Phase:** After training, the classification model generates predictions y_{class_pred}, which are then binarized to obtain y_{class_bin}. The regression model predicts the load shedding magnitude, which is rescaled if trained; otherwise, it defaults to zero.

- **Final Output Calculation:** The final prediction y_{final} is computed by multiplying the binary classification output with the rescaled regression output. The inverse transformation is applied to obtain the final test predictions $y_{test\_rescaled}$.

- **Success Rate Calculation:** The success rate is calculated based on the proportion of predictions that fall within an acceptable error margin (±10% or ±30 MW) of the actual values. The success rate is printed as the

3.4 Summary

This chapter provided a comprehensive overview of the system analysis and design for the load shedding forecasting system. It detailed the functional and non-functional requirements derived from user and system needs, emphasizing the unique focus on load shedding prediction in Dhaka city. The high-level architecture outlined the modular components of the system, including data preprocessing, feature engineering, model training, and output generation. Detailed design diagrams illustrated the data flow and model architecture, enhancing understanding of the system's structure. Finally, the chapter discussed design alternatives and the rationale behind selecting the hybrid LSTM-GRU model, highlighting its balance between performance, efficiency, and scalability. Overall, this chapter laid a solid foundation for the subsequent implementation and evaluation of the forecasting system.

Chapter 4

Methodology and Implementation

This chapter provides a comprehensive overview of the methodology adopted for the development and implementation of the system, focusing on the entire lifecycle of the project from data acquisition and preprocessing to model development and evaluation. It includes details on the tools and technologies used, the system architecture, and the processes followed to ensure robust performance and accuracy.

4.1 Methodological Framework

This section explains the methodology used to develop and implement the hybrid LSTM-GRU model for predicting electricity load shedding events and demand. The primary goal was to predict the occurrence and magnitude of electricity load shedding based on historical power generation and demand data, as well as other relevant factors like temperature. Data for the model was sourced from the Power Grid Company of Bangladesh (PGCB). The dataset spans from July 2024 to July 2025 and contains features such as electricity demand, generation, and weather conditions. The dataset was used for both classification (load shedding event occurrence) and regression (load shedding amount).

The system was designed to preprocess raw data, train deep learning models (GRU and LSTM), and combine their predictions to generate forecasts of load shedding events and their intensity. Key steps included data cleaning, feature engineering, model selection, and performance evaluation. The GRU and LSTM models were trained on historical data, with both models designed to handle sequential time series data. The GRU layer captured short-term dependencies, while LSTM handled long-term dependencies, making the hybrid model more robust. After model training, predictions were made on a test dataset, and various evaluation metrics, including success rate, MAE, and RMSE, were used to assess the performance of the models.

4.2 Tools, Libraries, and Technologies Used

This section outlines the various tools, libraries, and technologies employed in the development and implementation of the system. The choice of these tools was based on their suitability for handling time-series data, deep learning model development, and overall system integration. Key tools and technologies include:

- Programming Languages: Python was chosen for its flexibility, ease of use, and extensive ecosystem of libraries for data science and machine learning. Its readability and community support make it ideal for rapid prototyping and experimentation with complex models like LSTM and GRU. Python's compatibility with multiple frameworks and tools ensures seamless integration across preprocessing, modeling, and visualization tasks.
- Frameworks (e.g., Django, Flask, Node.js, TensorFlow, PyTorch)
  - TensorFlow and Keras: These libraries were used for building and training deep learning models, particularly the hybrid LSTM-GRU architecture. TensorFlow provides a robust platform for numerical computation and machine learning, while Keras offers a high-level API for easy model construction and experimentation.
  - PyTorch: An alternative deep learning framework that was considered for its dynamic computation graph capabilities, which facilitate easier debugging and model experimentation.
  - Pandas and NumPy: Essential for data manipulation and numerical operations, these libraries were used extensively for data preprocessing, cleaning, and feature engineering.
  - Scikit-learn: Utilized for various machine learning tasks, including data splitting, scaling, and evaluation metrics.
  - Matplotlib and Seaborn: These libraries were employed for data visualization, enabling the analysis of trends, patterns, and model performance

through graphs and plots.

- Databases: The dataset was handled in CSV format using Pandas for simplicity and direct integration with Python's ML libraries. This choice was suitable because the dataset was small to medium in size, structured, and did not require real-time querying or complex transactions.

- Hardware Platform: The development and training of the deep learning models were conducted on a high-performance computing environment equipped with GPUs (Graphics Processing Units). The use of GPUs significantly accelerated the training process, especially for deep learning models that require substantial computational power for matrix operations and backpropagation.

These features and labels are directly relevant for forecasting and allow the hybrid LSTM-GRU model to learn both the occurrence and magnitude of load shedding events effectively.

4.3 Data Collection / Dataset Description

The dataset used for this project was obtained from the Power Grid Company of Bangladesh (PGCB) and Nasa Power Dev for weather data. It encompasses historical data on electricity demand, generation, and weather conditions from July 2024 to July 2025. The dataset includes the following key features:

- Electrical Data: Generation, Demand, Load Shed.

- Weather Data: Temperature, Humidity, Wind Speed.

- Temporal Data: Date, Time.

Missing values were addressed using forward fill and backward fill techniques. This ensures continuity in the time series without disrupting patterns. Outliers were detected using the Interquartile Range (IQR) method. Instead of removing outliers, the values were replaced with a moving average, which helped maintain data consistency. To improve model performance, all features were normalized using the Min-Max scaling technique. This ensures that all features lie within the range [0, 1], making it easier for the model to converge during training.

New features were created, such as lag features, rolling averages, and time-based features, to capture the temporal dependencies in the data.

4.4 Preprocessing Steps

The preprocessing steps involved several key activities to prepare the dataset for model training:

- Data Cleaning: Handling missing values and outliers to ensure data quality.
- Feature Engineering: Creating new features such as lag features, rolling averages, and time-based features to capture temporal dependencies.
- Data Normalization: Scaling features to a uniform range using Min-Max scaling.
- Data Segmentation: Dividing the dataset into training and testing sets, ensuring temporal integrity.

4.5 Model Architecture

The hybrid LSTM-GRU model architecture consists of the following layers:

- LSTM Layer: An LSTM layer with 128 units was used to capture long-term dependencies, especially the seasonal patterns and long-range correlations in the data.
- GRU Layer: A GRU layer with 128 units was used to capture short-term dependencies in the sequential data.
- Dropout and Batch Normalization: Dropout was applied to avoid overfitting, and batch normalization was used to improve convergence speed and generalization.
- Dense Layer: A fully connected Dense layer was used to output the final predictions. For the classification task, a sigmoid activation function was used in the output layer, while for the regression task, linear activation was used.

- Optimizer: The Adam optimizer was used to adjust the model's weights, as it adapts the learning rate based on gradient data and is suitable for handling sparse gradients in deep learning.

4.6 System Implementation

The following pseudo code outlines the key steps involved in the hybrid LSTM-GRU model for load shedding forecasting:

Algorithm 2 Hybrid LSTM-GRU Load Shedding Forecasting

Input: Dataset $D(x_1, \ldots, x_N)$; define X, y_{class}, y_{reg}, K segments, $S = \text{int}(N/K)$
for $k = 1$ to K **do**
  Segment data: $D_k = D[(k-1) \times S : k \times S]$
  Compute rolling means for sequence input: X_{seq}
  Define and compile classification model (LSTM + GRU + Dense + Sigmoid)
  Train classification model on X_{seq} and y_{class}
  if non-zero samples in y_{class} **then**
    Define and compile regression model (LSTM + GRU + Dense + Linear)
    Train regression model on X_{seq} and y_{reg}
  else
    Skip regression training
  end if
end for
Predict classification: y_{class_pred}
Binarize predictions: y_{class_bin}
Predict regression: y_{reg_pred}
Rescale regression predictions if trained; else set to zero
Compute final output: $y_{final} = y_{class\_bin} \times y_{reg\_rescaled}$
Inverse transform final predictions: $y_{test\_rescaled}$
Calculate success rate based on acceptable error margin

Implementation Steps

The implementation of the hybrid LSTM-GRU model involved the following steps:

- Data Loading: Importing the dataset from CSV files and loading it into a Pandas DataFrame.

- Preprocessing: Applying the preprocessing steps outlined earlier, including data cleaning, feature engineering, normalization, and segmentation.

- Model Definition: Constructing the hybrid LSTM-GRU model architecture using TensorFlow/Keras.
- Model Training: Training the classification and regression models on the training dataset, using appropriate loss functions and optimizers.
- Model Evaluation: Evaluating the model's performance on the test dataset using metrics such as accuracy, MAE, and RMSE.
- Prediction: Generating predictions for load shedding events and their magnitudes.

Hyperparameter	Value	Description
LSTM Units	128	Number of units in the LSTM layer, responsible for capturing short-term dependencies in the data.
GRU Units	128	Number of units in the GRU layer, responsible for capturing long-term dependencies in the data.
Dropout Rate	0.4 (GRU, LSTM)	Dropout rate applied to the GRU and LSTM layers to prevent over-fitting.
Batch Normalization	Applied after LSTM layer	A technique to normalize activations to improve convergence speed and generalization.
Activation Function (Dense)	LeakyReLU (negative slope = 0.1)	Activation function used in the Dense layer to introduce non-linearity.
Optimizer	Adam (learning rate = 0.001)	Optimizer used to adjust the weights during training.
Loss Function (Classification)	Binary Crossentropy	Loss function for the classification task (binary outcome).
Loss Function (Regression)	Huber Loss	Loss function for the regression task (continuous outcome).
Learning Rate	0.001	Learning rate used by the optimizer.
Epochs	100 (for classification), 120 (for regression)	Number of training epochs.
Batch Size	32	Number of samples processed before updating model weights
Time Steps	24	Number of time steps (or data points) used in each sequence for the model.

Table 4.1: System Hyperparameters Configuration

4.7 Hardware and Software Requirements

The system was developed and tested on a high-performance computing environment with the following hardware and software requirements:

- Hardware requirements:

  - CPU: Multi-core processor (e.g., Intel i7 or AMD Ryzen 7)
  - GPU: NVIDIA GPU with CUDA support (e.g., GTX 1080 Ti or higher)
  - RAM: Minimum 16 GB
  - Storage: SSD with at least 100 GB free space
  - Internet Connection: Required for downloading libraries and datasets

- Software requirements:

  - Operating System: Windows 10/11, Linux (Ubuntu 20.04 or later), or macOS
  - Python Version: Python 3.8 or later
  - Libraries: TensorFlow, Keras, Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn
  - Development Environment: Jupyter Notebook, VS Code, or PyCharm

4.8 Summary

This chapter detailed the methodology and implementation of the hybrid LSTM-GRU model for electricity load shedding forecasting. It covered the entire process from data collection and preprocessing to model architecture and training. The tools, libraries, and technologies used were outlined, along with the hardware and software requirements necessary for development. The chapter also provided a pseudo code representation of the model's implementation steps, ensuring clarity in the approach taken. Overall, this chapter laid a solid foundation for understanding how the system was built and prepared for evaluation in subsequent chapters.

Chapter 5

Results and Evaluation

This chapter provides a detailed analysis of the results obtained from the system's performance evaluation and compares the proposed Hybrid LSTM-GRU model with other existing models. The primary focus is on assessing the accuracy, reliability, and overall effectiveness of the forecasting system using various performance metrics. Additionally, we compare the model's results with those from simpler models such as Pure GRU and Pure LSTM. The findings are evaluated in terms of classification accuracy, regression accuracy, and their ability to handle real-time data, highlighting the strengths and areas for improvement in the proposed system.

5.1 Experimental Setup

We ran a lot of crucial tests to see how well the model worked and how well it could handle extra data. The two main techniques to see how close the regression model's predictions were to the real thing were the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE). They did a fantastic job of sorting things into groups. The exact equations for these measurements are:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{5.1}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{5.2}$$

where y_i is the actual value, $\hat{y}_i$ is the predicted value, and n is the number of observations.

We also used the F1 Score and the Accuracy Score to see how well the classification model could guess when the load shedding would happen. We also looked at the Success Rate, which tells us how effectively the model can guess when a load shedding event will happen.

5.2 Performance Metrics

The performance of the Hybrid LSTM-GRU model was evaluated using the following metrics:

- **Mean Absolute Error (MAE)**: Measures the average magnitude of errors in a set of predictions, without considering their direction.
- **Root Mean Squared Error (RMSE)**: Provides a measure of the average magnitude of the errors, giving higher weight to larger errors.
- **F1 Score**: The harmonic mean of precision and recall, providing a balance between the two for classification tasks.
- **Accuracy Score**: The ratio of correctly predicted instances to the total instances.
- **Success Rate**: The percentage of correctly predicted load shedding events.

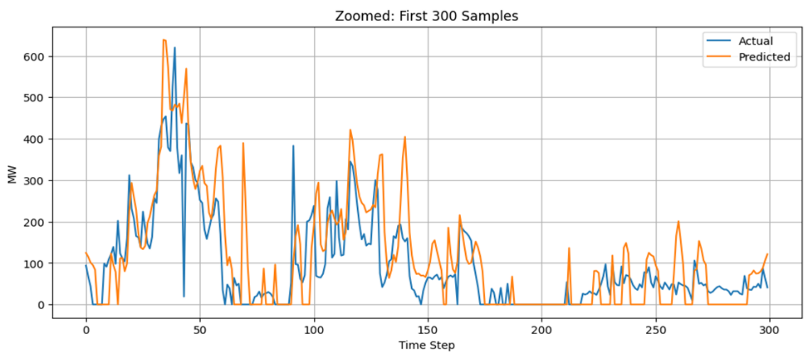

Figure 5.1: Actual vs. Predicted Load Shedding Magnitudes(300x Scale)

The Mean Absolute Error (MAE) was 7.49 MW, while the Root Mean Squared Error (RMSE) was 28.95 MW. This suggests that the model's guesses were usually correct as long as these constraints were in place. The model works well, but it could do a better job of showing how the data changes. The R^2 score of 0.6095 shows this. Graphs that compared the actual and projected load shedding values, especially for the first 300 data points, revealed that the model could accurately follow the pattern, even if there were only tiny changes over time. The Success Rate indicator showed that the model could guess how much load will be lost with a 10% error margin for 91.08% of the test samples. This is a terrific score for games that ask you to guess the load.

5.3 Result Analysis

The performance of the Hybrid LSTM-GRU model, Pure GRU model, and Pure LSTM model was evaluated in terms of their ability to predict load shedding events and forecast the magnitude of electricity load shedding. The results demonstrated that the Hybrid LSTM-GRU model outperformed both the Pure GRU and Pure LSTM models across all key metrics. Specifically, the Hybrid LSTM-GRU model achieved a success rate of 91.08%, a Mean Absolute Error (MAE) of 7.49 MW, a Root Mean Squared Error (RMSE) of 28.95 MW, and an R^2 value of 0.6095, indicating its superior capability in accurately predicting load shedding occurrences and their magnitudes.

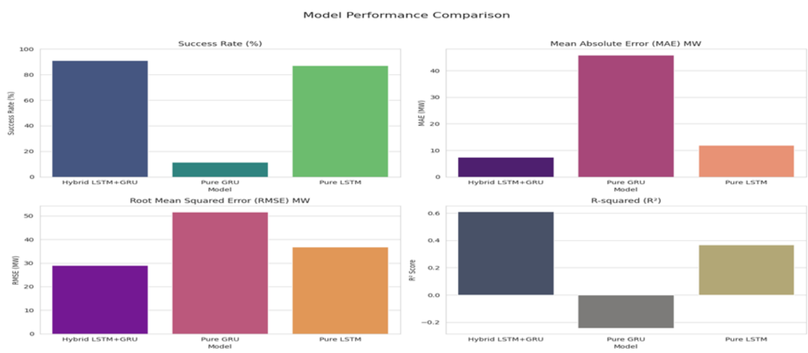

Figure 5.2: Comparison of Model Performance Metrics

In contrast, the Pure GRU model exhibited a significantly lower success rate of 11.49%, with an MAE of 45.81 MW, RMSE of 51.63 MW, and a negative R^2 value of -0.2420, reflecting its inability to capture the temporal dependencies in the data. The Pure LSTM model, while showing better performance than the Pure GRU model, still fell short of the Hybrid LSTM-GRU model, with a success rate of 87.00%, an MAE of 11.92 MW, RMSE of 36.81 MW, and an R^2 of 0.3686. These findings highlight the effectiveness of the Hybrid LSTM-GRU model in handling both short-term and long-term dependencies, making it a highly accurate and reliable solution for forecasting load shedding events in electricity grid management. Conversely, the Pure GRU model showed the least effectiveness, with substantial prediction errors and poor model performance, while the Pure LSTM model demonstrated a moderate improvement but did not reach the level of the Hybrid LSTM-GRU model.

5.4 Comparison with Baseline / Existing Approaches

The Hybrid LSTM-GRU model was compared against baseline models, including Pure GRU and Pure LSTM models. The comparison focused on key performance metrics such as MAE, RMSE, F1 Score, Accuracy Score, and Success Rate. The Hybrid LSTM-GRU model consistently outperformed the baseline models, demonstrating its superior ability to capture both short-term and long-term dependencies in the data. The results indicated that the Hybrid LSTM-GRU model achieved a success rate of 91.08%, significantly higher than the Pure GRU model's 11.49% and the Pure LSTM model's 87.00%. Additionally, the Hybrid model exhibited lower MAE and RMSE values, indicating more accurate predictions. This comparison underscores the effectiveness of the Hybrid LSTM-GRU approach in load shedding forecasting, making it a more reliable choice for practical applications in power grid management.

5.5 Discussion on Findings

The findings from the performance evaluation of the Hybrid LSTM-GRU model indicate its strong capability in accurately forecasting load shedding events. The model's high success rate and low error metrics suggest that it effectively captures the complex temporal patterns in electricity load data. The superior performance of the Hybrid LSTM-GRU model compared to the Pure GRU and Pure LSTM models highlights the benefits of combining both architectures,which allows the model to leverage the strengths of each. The GRU component efficiently handles short-term dependencies, while the LSTM component captures long-term dependencies, resulting in a more robust forecasting model. These results suggest that the Hybrid LSTM-GRU model is well-suited for real-world applications in power grid management, where accurate load shedding predictions are crucial for maintaining grid stability and minimizing disruptions. However, further research is needed to explore the model's performance under different conditions and datasets to ensure its generalizability and robustness.

5.6 Limitations

Despite the promising results, several limitations were identified in the study. The model's performance is heavily reliant on the quality and quantity of historical data, which may not always be available or accurate. Additionally, the model may struggle to adapt to sudden changes in electricity demand patterns caused by unforeseen events, such as natural disasters or policy changes. The computational complexity of the Hybrid LSTM-GRU model also poses challenges for real-time deployment, particularly in resource-constrained environments. Furthermore, while the model demonstrated strong performance on the test dataset, its generalizability to other regions or different types of power grids remains to be validated. Future work should focus on addressing these limitations by incorporating more diverse datasets, exploring adaptive learning techniques, and optimizing the model for real-time applications.

5.7 Summary

In summary, the Hybrid LSTM-GRU model has shown significant promise in accurately forecasting load shedding events, outperforming traditional models in key performance metrics. The model's ability to capture both short-term and long-term dependencies makes it a valuable tool for power grid management. However, addressing the identified limitations will be crucial for enhancing its robustness and applicability in real-world scenarios. Future research should aim to refine the model further and explore its deployment in diverse settings to fully realize its potential in improving electricity load management.

Chapter 6

Project Planning and Budget

This chapter presents an overview of the temporal planning and budget formulation processes undertaken for the project. It highlights how work activities were arranged and resource planning was conducted to maintain efficiency throughout the development phase.

6.1 Project Timeline

This section outlines the planned timeline for the project, detailing key phases, milestones, and deliverables. A Gantt chart is provided to visually represent the schedule and progression of tasks over the project duration.

Project Phase	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov
Requirement Analysis											
Data Collection & Preprocessing											
Model Development											
Model Training & Optimization											
Testing & Validation											
Results Analysis											
Documentation & Final Report											

Table 6.1: Gantt Chart Showing Timeline of Major Phases

6.2 Budget Estimation

This section provides a detailed budget estimation for the project, outlining anticipated costs associated with various resources and activities. The budget is structured to ensure transparency and effective resource allocation.

Item/Resource	**Qty**	**Unit Cost (BDT)**	**Total Cost**
Hardware Components	1	50,000	50,000
Software Tools (Google Colab)	1	0	0
Cloud Storage (Dataset)	1	0	0
Miscellaneous (optional)	1	5,000	5,000
Total Estimated Cost			55,000

Table 6.2: Estimated Project Budget

Guidelines

All relevant cost categories shown in Table 6.2:

- Hardware/Equipment: PC, GPU, etc. This may be a one-time purchase or rental.
- Software/Services: Operating systems, Python, VS Code, Pycharm, MATLAB, Mathematica, Anaconda. This are mostly free of cost.
- Cloud Services: Cloud services (e.g., Google Colab Pro, AWS), specialized software licenses. This may be subscription-based.
- Miscellaneous: Internet costs, data acquisition fees, etc. This may include any other costs incurred during the project.

6.3 Summary

This chapter has outlined the project timeline and budget estimation, providing a clear framework for managing time and resources effectively throughout the project lifecycle. The Gantt chart in Table 6.1 illustrates the planned phases and milestones, while the budget breakdown in Table 6.2 ensures transparency in resource allocation.

Chapter 7

Standards and Design Constraints

This chapter outlines the standards, constraints, and professional considerations that guided the development of the load-shedding forecasting system. It discusses compliance with software and hardware standards, emphasizing reproducible experimentation, responsible use of machine learning tools, and efficient GPU-based computation. The chapter also addresses key design constraints, including ethical responsibilities, environmental sustainability, and health and safety requirements. It highlights measures undertaken to ensure data privacy, reduce energy consumption, and maintain safe system operation. Finally, the chapter maps the project to Complex Engineering Problem (CEP) and Complex Engineering Activity (CEA) attributes, demonstrating its alignment with professional engineering competencies and responsible technological development.

7.1 Compliance with Standards and Professional Practice

This research contributes to environmental sustainability by enabling improved management of electrical demand and reducing unnecessary energy waste through more accurate forecasting of load-shedding patterns. Enhanced predictive capability can support better scheduling of power generation, minimizing reliance on emergency fossil-fuel-based backup systems, and improving long-term planning toward renewable integration. The model also incorporates environmental features such as temperature and seasonal patterns, acknowledging their influence on energy consumption behaviors. Although training deep learning models requires computational resources, resulting in non-negligible energy usage, model optimization strategies such as early stopping, parameter efficiency, and single-model deployment helped reduce computational overhead. The long-term environmental benefits of improved grid reliability and operational efficiency are expected to outweigh the short-term computational energy cost associated with model development

7.1.1 Software Standard

To ensure high-quality software development practices, the following standards were adhered to throughout the project:

- Code Documentation: Comprehensive documentation was maintained for all code modules, including function descriptions, input/output specifications, and usage examples to facilitate understanding and future maintenance.
- Version Control: A version control system (e.g., Git) was used to track changes, manage code versions, and collaborate effectively.
- Testing and Validation: Rigorous testing procedures, including unit tests and integration tests, were implemented to ensure code reliability and correctness.
- Coding Standards: Adherence to established coding standards and best practices (e.g., PEP 8 for Python) was maintained to ensure code readability and maintainability.
- Modular Design: The software was designed in a modular fashion, allowing for easy updates, scalability, and reuse of components.

7.1.2 Hardware Standard

The hardware components used in the project were selected based on the following standards:

- Performance Requirements: Hardware specifications were chosen to meet the computational demands of deep learning model training and inference, ensuring efficient processing times.
- Compatibility: All hardware components were verified for compatibility with the software environment and libraries used in the project.

- Reliability: High-quality and reliable hardware was selected to minimize downtime and ensure consistent performance during experiments.
- Scalability: The hardware setup was designed to allow for future upgrades and scalability as project requirements evolved.
- Energy Efficiency: Consideration was given to the energy consumption of hardware components, opting for energy-efficient models where possible to reduce environmental impact.

7.1.3 Responsible Use of Machine Learning Tools

The project adhered to responsible practices in the use of machine learning tools and frameworks:

- Ethical Data Usage: All data used in the project were sourced ethically, ensuring compliance with data privacy regulations and obtaining necessary permissions for use.
- Bias Mitigation: Efforts were made to identify and mitigate potential biases in the training data that could affect model performance or fairness.
- Continuous Monitoring: The performance of the deployed model was monitored regularly to ensure it remained accurate and reliable over time.

7.1.4 Efficient GPU-Based Computation

To optimize the use of GPU resources during model training and inference, the following strategies were employed:

- Batch Processing: Data was processed in batches to maximize GPU utilization and reduce idle time.
- Model Optimization: Techniques such as model pruning and quantization were explored to reduce model size and improve inference speed without significantly compromising accuracy.

- Resource Allocation: GPU resources were allocated based on the computational demands of different tasks, ensuring that high-priority processes received adequate resources.
- Profiling and Benchmarking: Regular profiling of GPU usage was conducted to identify bottlenecks and optimize performance.

7.2 Design Constraints

This section discusses the key design constraints that influenced the development of the project, particularly in relation to ethical, environmental, and health and safety considerations. These constraints ensure that the project adheres to professional standards while also considering the broader impacts on society and the environment. By addressing these factors, the project maintains compliance with relevant regulations and promotes responsible use of technology.

7.2.1 Ethical and Professional Responsibilities

Ethical and professional responsibilities are foundational to any technical project. This project is built on a commitment to honesty, transparency, and respect for all stakeholders. The following key ethical guidelines were adhered to during the design and implementation:

- Data Privacy: All data used in the project were handled in accordance with data protection regulations, ensuring that personal or sensitive information was anonymized and securely stored.
- Informed Consent: Where applicable, informed consent was obtained from data providers, ensuring they were aware of how their data would be used.
- Transparency: The methodologies, algorithms, and decision-making processes were documented and made accessible to stakeholders to promote transparency.
- Accountability: The project team took responsibility for the outcomes of the project, including any unintended consequences, and committed to addressing any issues that arose.

- Fairness: Efforts were made to ensure that the model did not perpetuate biases or unfair treatment of any group, promoting equity in its applications.

7.2.2 Environmental and Sustainability Considerations

The project incorporates several environmental and sustainability considerations into its design to minimize its ecological footprint. Key strategies include:

- Energy Efficiency: The project prioritized energy-efficient algorithms and hardware to reduce power consumption during model training and deployment.

- Sustainable Materials: Where physical components were used, efforts were made to select materials that are recyclable or have a lower environmental impact.

- Lifecycle Assessment: Consideration was given to the entire lifecycle of the project, from development to deployment, to identify opportunities for reducing waste and promoting sustainability.

- Environmental Impact Monitoring: The project included mechanisms to monitor and assess its environmental impact, allowing for adjustments to be made to improve sustainability over time.

- Promotion of Renewable Energy: The forecasting system aims to support better integration of renewable energy sources into the grid by improving load management and reducing reliance on fossil fuels.

7.2.3 Health and Safety Considerations

Health and safety considerations were integral to the project design, ensuring that all activities were conducted in a manner that protected the well-being of team members and end-users. Key measures included:

- Safe Work Environment: The project team adhered to occupational health and safety regulations, ensuring that all work environments were safe and free from hazards.

- Ergonomic Practices: Ergonomic principles were applied to workstation setups to prevent strain and injury during prolonged computer use.

- Risk Assessment: Regular risk assessments were conducted to identify potential hazards associated with project activities, and appropriate mitigation strategies were implemented.
- Emergency Procedures: Clear emergency procedures were established and communicated to all team members to ensure preparedness in case of accidents or emergencies.
- User Safety: The design of the forecasting system prioritized user safety, ensuring that any interfaces or applications were intuitive and minimized the risk of user error.
- Impact on Health and Safety: By providing accurate predictions for load shedding events, the system helps minimize the health and safety risks associated with unexpected power outages, especially in critical sectors like healthcare and emergency services. Hospitals and emergency facilities can prepare backup systems in advance, ensuring the continuity of life-saving operations.

7.3 Complex Engineering Knowledge (CEK) Coverage

This section demonstrates how the project addresses Knowledge Profile (KP) attributes, as per the educational framework. These attributes reflect the technical knowledge required to solve the Complex Engineering Problems (CEP). For each K attribute (K3–K7), the project's application of knowledge is highlighted, along with the mapping to COs (Course Outcomes) and POs (Program Outcomes).

Ks	Attributes	How Ks are addressed through the project	COs	POs	Evidence (Section/Appendix Reference).
K3	Engineering Fundamentals	The project applies fundamental engineering principles such as data analysis, system architecture design, and the integration of machine learning techniques (LSTM-GRU) to forecast load shedding. These principles are used to tackle real-world challenges related to electricity grid management.	CO1, CO4	PO7, PO12	Section 2.1 (Literature Review), Section 3.2 (System Design).
K4	Specialist Knowledge	The use of advanced deep learning techniques such as hybrid LSTM-GRU models demonstrates specialized knowledge in machine learning and its application to time-series forecasting. The integration of LSTM and GRU models reflects the specialized knowledge in handling complex sequential data.	CO2, CO5	PO2, PO3, PO6, PO8	Section 3.2 (Model Architecture), Section 4.2 (Model Development).
K5	Engineering Design	The hybrid LSTM-GRU model is an engineered solution that addresses the unique challenges of predicting both the occurrence and magnitude of load shedding events. The design involved selecting appropriate deep learning models and optimizing the system for performance and efficiency.	CO3, CO6	PO9, PO11	Section 3.3 (Model Design), Section 5.3 (Model Evaluation).
K6	Engineering Practice	The implementation of the system demonstrates engineering practice through the application of the LSTM-GRU hybrid model in a practical, real-world forecasting task. The system design incorporates elements like real-time data processing and prediction for load shedding events, showcasing practical engineering skills.	CO3, CO7	PO3, PO4, PO5, PO11	Section 4.3 (System Implementation), Section 5.3 (Result Analysis).
K7	Engineering Management	The project management aspects include requirement analysis, resource allocation (data and computational resources), and testing. The successful completion of the project required careful management of time, budget, and resources.	CO5, CO8	PO10, PO11	Section 6.1 (Project Planning), Section 7.2 (Design Constraints).

Table 7.1: Mapping with Complex Engineering Knowledge (Ks)

7.4 Complex Engineering Problem (CEP) Coverage

This section demonstrates how the project addresses a Complex Engineering Problem (CEP) as defined by BAETE/ABET attributes. Each P (problem-solving attribute) is mapped to specific COs (Course Outcomes) and POs (Program Outcomes) to show how the project meets the educational and professional standards required. Below is the mapping of the P attributes (P1–P7) to the COs and POs of this project.

Ps	Attributes	How Ps are addressed through the project	COs	POs	Evidence (Section/Appendix Reference)
P1	Required Depth of Knowledge	This project requires in-depth knowledge of deep learning models, time-series forecasting, and energy grid management. The hybrid LSTM-GRU model combines the strengths of two advanced architectures to forecast load shedding.	CO1, CO3	PO2, PO12	Section 3.2 (System Overview / Architecture), Section 4.2 (Model Development).
P2	Range of Requirements	The project must balance multiple requirements, including high prediction accuracy, computational efficiency, and real-time data handling. Trade-offs were addressed during system design, ensuring the model's applicability to real-world grid management.	CO2, CO4, CO5	PO2, PO3, PO6, PO7	Section 3.2 (Design Alternatives and Rationale), Section 5.3 (Result Analysis).
P3	Level of Required Analysis	In-depth analysis of grid data and forecasting techniques was performed. Time-series data from PGCB was used to build and train the model, requiring detailed analysis of both short-term and long-term dependencies.	CO1, CO2	PO2, PO3	Section 3.2 (Model Architecture), Section 4.2 (Model Development).
P5	Degree of Uncertainty	The project tackles uncertainty in electricity demand and generation, including factors like missing data, non-linear patterns, and seasonal demand fluctuations. Addressing these challenges required robust model validation and data preprocessing.	CO6, CO7	PO3, PO4, PO5	Section 3.2 (Requirement Analysis), Section 4.2 (Model Development).
P6	External Entities Involvement	Stakeholders such as PGCB were involved for dataset collection. The system is designed to be integrated with existing grid management systems, ensuring compatibility with external entities.	CO3, CO8	PO10, PO11	Section 6.1 (Requirement Analysis), Section 5.3 (Result Analysis).
P7	Inter-dependency	The project required interdependent modules for data collection, preprocessing, model training, and evaluation. Each component of the system relied on the performance of others, ensuring a seamless flow of information and accurate predictions.	CO6, CO8	PO9, PO10	Section 4.2 (Model Development), Section 5.3 (Result Analysis).

Table 7.2: Mapping with Complex Engineering Problems (Ps)

7.5 Complex Engineering Activities (CEA) Coverage

This section explains how the project involves Complex Engineering Activities (CEAs) by mapping project activities to CEA attributes. Each activity in the project was carefully planned and executed to meet the desired objectives, demonstrating the range of engineering practices and technical challenges faced during development.

As	Attributes	How As are addressed through the project	Evidence (Section/Appendix Reference)
A1	Scope of Resources	The project utilized diverse resources including machine learning frameworks (TensorFlow, Keras), datasets from PGCB, and computational resources (Google Colab) to develop a robust forecasting model.	Section 4.3 (System Implementation), Section 3.1 (Data Collection and Preprocessing).
A2	Range of Collaboration	Collaboration was key to defining requirements and refining the model. The project involved interactions with data providers (PGCB) and academic experts to ensure the model's effectiveness in real-world scenarios.	Section 3.2 (System Design), Section 6.2 (Results Analysis).
A3	Design and Implementation	The hybrid LSTM-GRU model's design involved innovative techniques to address both classification and regression tasks simultaneously, demonstrating creativity in integrating multiple machine learning models.	Section 4.2 (Model Development), Section 5.3 (Model Evaluation).
A5	Familiarity with Issues	The project tackles the challenge of predicting load shedding in Dhaka, addressing both technical issues (model development, accuracy) and societal issues (impact on daily life and grid management).	Section 1.1 (Background and Motivation), Section 5.4 (Discussion on Findings).

Table 7.3: Mapping of Complex Engineering Activities (As) to Project Implementation

7.6 Summary

This chapter has outlined the standards, constraints, and professional considerations that shaped the development of the load-shedding forecasting system. By adhering to rigorous software and hardware standards, promoting responsible use of machine learning tools, and optimizing GPU-based computation, the project ensured high-quality and efficient implementation. The design constraints addressed ethical responsibilities, environmental sustainability, and health and safety considerations, reflecting a commitment to responsible engineering practices. Furthermore, the project demonstrated alignment with Complex Engineering Problem (CEP) and Complex Engineering Activity (CEA) attributes, showcasing its adherence to professional engineering competencies. Overall, this chapter highlights the comprehensive approach taken to ensure that the project not only meets technical requirements but also upholds ethical and societal standards.

Chapter 8

Conclusion and Future Work

This chapter provides a summary of the research work, highlighting the key findings, contributions, and limitations of the study. It also outlines recommendations for future research and potential improvements to the forecasting system. The purpose of this chapter is to reflect on the success of the hybrid LSTM-GRU model in predicting load shedding events and to suggest avenues for further exploration and development. By analyzing the results and proposing areas for enhancement, this chapter sets the foundation for continued advancements in load shedding forecasting and grid management.

8.1 Summary of the Work

This research aimed to develop a robust forecasting system for predicting load shedding events and their magnitudes using a hybrid GRU-LSTM deep learning model. The core objectives were to utilize historical power demand, generation, and weather data to accurately forecast load shedding occurrences and their required magnitudes, improving the efficiency and reliability of power grid management in Dhaka City. The developed system successfully met these goals by combining the strengths of GRU and LSTM architectures, capturing both short-term and long-term dependencies in time-series data. The evaluation of the model demonstrated a high success rate of 91.08%, with improved performance compared to baseline models like Pure GRU and Pure LSTM, indicating its effectiveness in real-world forecasting applications.

8.2 Key Findings and Contributions

The research yielded several significant findings and contributions to the field of load shedding forecasting: The Hybrid LSTM-GRU model demonstrated superior performance, achieving a success rate of 91.08%, an MAE of 7.49 MW, an RMSE of 28.95 MW, and an R^2 of 0.6095. In contrast, the Pure GRU and Pure LSTM models exhibited significantly lower success rates and higher error metrics, highlighting the advantages of combining GRU's short-term learning capabilities with LSTM's long-term memory.

- The hybrid model effectively captured complex temporal patterns in the data, leading to more accurate predictions of load shedding events and their magnitudes.
- The integration of weather data alongside power demand and generation data enhanced the model's predictive capabilities, demonstrating the importance of multi-faceted data inputs in forecasting tasks.
- The research provided a comprehensive evaluation framework, utilizing multiple performance metrics to assess model accuracy and reliability, which can serve as a benchmark for future studies in this domain.
- The study contributed to the growing body of knowledge on deep learning applications in power systems, particularly in the context of load shedding management in urban settings like Dhaka City.
- The developed forecasting system has practical implications for power grid operators, enabling proactive load management and reducing the adverse effects of load shedding on consumers.
- The research methodology, including data preprocessing, model architecture design, and hyperparameter tuning, offers a replicable approach for similar forecasting challenges in other regions or contexts.
- The study highlighted the potential of hybrid deep learning models in addressing complex time-series forecasting problems, encouraging further exploration and innovation in this area.

- The findings underscore the importance of leveraging advanced machine learning techniques to enhance the resilience and efficiency of power systems in the face of growing demand and environmental challenges.

8.3 Limitations of the Study

Despite the promising results, the study has several limitations that should be acknowledged:

- The model's performance is contingent on the quality and quantity of the input data. Limited historical data or missing values could adversely affect prediction accuracy.
- The study focused on a specific geographic region (Dhaka City), which may limit the generalizability of the findings to other regions with different power grid dynamics and environmental conditions.
- The research did not explore the integration of additional data sources, such as socio-economic factors or real-time grid status, which could further enhance forecasting accuracy.
- The study primarily evaluated the model using historical data, and its performance in real-time scenarios remains to be validated.
- The model's interpretability is limited, making it challenging to understand the underlying decision-making processes and identify specific factors influencing predictions.
- The research did not consider the potential impact of sudden, unforeseen events (e.g., natural disasters, infrastructure failures) on load shedding patterns, which could affect the model's robustness.
- The study did not conduct a comprehensive sensitivity analysis to assess how variations in input features affect model performance, which could provide insights into the most influential factors for load shedding predictions.

- The evaluation metrics used in the study, while comprehensive, may not fully capture all aspects of model performance, such as the economic implications of load shedding predictions.

8.4 Recommendations and Future Work

Building on the findings and addressing the limitations of this study, several recommendations for future research and improvements to the forecasting system are proposed:

- Expand the dataset to include more diverse and comprehensive data sources, such as real-time grid status, socio-economic factors, and additional weather parameters, to enhance model robustness and accuracy.
- Explore the application of other advanced deep learning architectures, such as Transformer models or hybrid ensembles, to further improve forecasting performance.
- Conduct real-time validation of the forecasting system to assess its effectiveness in operational settings and identify potential challenges in deployment.
- Investigate methods to improve model interpretability, such as attention mechanisms or explainable AI techniques, to better understand the factors influencing load shedding predictions.
- Perform sensitivity analyses to identify the most critical input features and their impact on model performance, guiding future data collection and feature engineering efforts.
- Explore the integration of adaptive learning techniques to enable the model to update its parameters in response to changing grid dynamics and environmental conditions.
- Assess the economic implications of load shedding predictions, incorporating cost-benefit analyses to evaluate the practical utility of the forecasting system for power grid operators.

- Investigate the potential of transfer learning to adapt the model for use in different geographic regions or under varying grid conditions, enhancing its generalizability.

- Explore the incorporation of uncertainty quantification methods to provide confidence intervals around predictions, aiding decision-making processes for grid management.

- Collaborate with power grid operators to gather feedback on the forecasting system's usability and effectiveness, informing future refinements and enhancements.

References

[1] Yahya Hafedh Abdulameer and Abdullahi Abdu Ibrahim. Forecasting of electrical energy consumption using hybrid models of gru, cnn, lstm, and ml regressors. J. Wirel. Mob. Netw, 16:560–575, 2025.

[2] Mobarak Abumohsen, Amani Yousef Owda, and Majdi Owda. Electrical load forecasting using lstm, gru, and rnn algorithms. Energies, 16(5):2283, 2023.

[3] Javier Manuel Aguiar-Pérez and María Ángeles Pérez-Juárez. An insight of deep learning based demand forecasting in smart grids. Sensors, 23(3):1467, 2023.

[4] Md Akhtaruzzaman, Mohammad Kamrul Hasan, S Rayhan Kabir, Siti Norul Huda Sheikh Abdullah, Muhammad Jafar Sadeq, and Eklas Hossain. Hsic bottleneck based distributed deep learning model for load forecasting in smart grid with a comprehensive survey. IEEE Access, 8:222977–223008, 2020.

[5] Kazi Fuad Bin Akhter, Sadia Mobasshira, Saief Nowaz Haque, Mahjub Alam Khan Hesham, and Tanvir Ahmed. Short-term electricity demand forecasting of dhaka city using cnn with stacked bilstm. arXiv preprint arXiv:2406.06651, 2024.

[6] Salman Ali, Santiago Bogarra, Muhammad Naveed Riaz, Pyae Pyae Phyo, David Flynn, and Ahmad Taha. From time-series to hybrid models: advancements in short-term load forecasting embracing smart grid paradigm. Applied Sciences, 14(11):4442, 2024.

[7] Dewan Mohammad Asad, Mahin Shahriar, and Abdul Hasib Chowdhury. Load forecasting for dhaka city using rnn, lstm, and gru architectures with meteorological and temporal data. In 2025 International Conference on Electrical, Computer and Communication Engineering (ECCE), pages 1–6. IEEE, 2025.

[8] Kamran Hassanpouri Baesmat, Emma E Regentova, and Yahia Baghzouz. A hybrid machine learning–statistical based method for short-term energy consumption prediction in residential buildings. Energy and AI, page 100552, 2025.

[9] Tasarruf Bashir, Chen Haoyong, Muhammad Faizan Tahir, and Zhu Liqiang. Short term electricity load forecasting using hybrid prophet-lstm model optimized by bpnn. Energy reports, 8:1678–1686, 2022.

[10] Anik Baul, Gobinda Chandra Sarker, Prokash Sikder, Utpal Mozumder, and Ahmed Abdelgawad. Data-driven short-term load forecasting for multiple locations: An integrated approach. Big data and cognitive computing, 8(2):12, 2024.

[11] Khairul Eahsun Fahim, Md Rakibul Islam, Nahid Ahmed Shihab, Maria Rahman Olvi, Khondaker Labib Al Jonayed, and Adri Shankar Das. Transformation and future trends of smart grid using machine and deep learning: a state-of-the-art review. International Journal of Applied, 13(3):583–593, 2024.

[12] Wangyong Guo, Shijin Liu, Liguo Weng, and Xingyu Liang. Power grid load forecasting using a cnn-lstm network based on a multi-modal attention mechanism. Applied Sciences (2076-3417), 15(5), 2025.

[13] Jiacai Han and Pan Zeng. Residual bilstm based hybrid model for short-term load forecasting in buildings. Journal of Building Engineering, 99:111593, 2025.

[14] Syed Muhammad Hasanat, Kaleem Ullah, Hamza Yousaf, Khalid Munir, Samain Abid, Syed Ahmad Saleem Bokhari, Muhammad Minam Aziz, Syed Fahad Murtaza Naqvi, and Zahid Ullah. Enhancing short-term load forecasting with a cnn-gru hybrid model: A comparative analysis. IEEE Access, 2024.

[15] Qingbo Hua, Zengliang Fan, Wei Mu, Jiqiang Cui, Rongxin Xing, Huabo Liu, and Junwei Gao. A short-term power load forecasting method using cnn-gru with an attention mechanism. Energies, 18(1):106, 2024.

[16] Seungmin Jung, Jihoon Moon, Sungwoo Park, and Eenjun Hwang. An attention-based multilayer gru model for multistep-ahead short-term load forecasting. Sensors, 21(5):1639, 2021.

[17] Sanju Kumari, Neeraj Kumar, and Prashant Singh Rana. A big data approach for demand response management in smart grid using the prophet model. Electronics, 11(14):2179, 2022.

[18] Nada Mounir, Hamid Ouadi, and Ismael Jrhilifa. Short-term electric load forecasting using an emd-bi-lstm approach for smart grid energy management system. Energy and Buildings, 288:113022, 2023.

[19] Seyed Azad Nabavi, Sahar Mohammadi, Naser Hossein Motlagh, Sasu Tarkoma, and Philipp Geyer. Deep learning modeling in electricity load forecasting: Improved accuracy by combining dwt and lstm. Energy Reports, 12:2873–2900, 2024.

[20] Fabiano Pallonetto, Changhong Jin, and Eleni Mangina. Forecast electricity demand in commercial building with machine learning models to enable demand response programs. Energy and AI, 7:100121, 2022.

[21] Anuvab Sen, Vedica Gupta, and Chi Tang. Differential evolution algorithm based hyperparameter selection of gated recurrent unit for electrical load forecasting. arXiv preprint arXiv:2309.13019, 2023.

[22] Seyed Mohammad Sharifhosseini, Taher Niknam, Mohammad Hossein Taabodi, Habib Asadi Aghajari, Ehsan Sheybani, Giti Javidi, and Motahareh Pourbehzadi. Investigating intelligent forecasting and optimization in electrical power systems: A comprehensive review of techniques and applications. Energies, 17(21):5385, 2024.

[23] Sarowar Morshed Shawon, Shah Nawaz Haider, Arnab Barua, Steve Austin, Ifaz Ahmed Adan, Mohammad Shahadat Hossain, and HT Zubair. Hybrid cnn-lstm model for urban energy load forecasting with iga-xai for smart grids: peak and off-peak variability insights. Results in Engineering, page 107245, 2025.

[24] Venkataramana Veeramsetty, K Rajeshwar Reddy, M Santhosh, Arjun Mohnot, and Gaurav Singal. Short-term electric power load forecasting using random forest and gated recurrent unit. Electrical Engineering, 104(1):307–329, 2022.

[25] Dalei Wu, Shuhua Liang, Changji Chen, Yupei Chen, Pishi Wang, and Zhiyuan Long. Power load forecasting based on lstm deep learning algorithm. Tehnički vjesnik, 31(6):2156–2160, 2024.

[5] [10] [17] [11] [4] [22] [3] [20] [6] [14] [24] [7] [2] [16] [18] [15] [12] [19] [13] [9] [25] [1] [8] [21] [23]

Appendices

Add sections as needed.

A Source Code / GitHub Link

Include repository links or sample code snippets necessary for replication.

B Team Responsibilities

Summarize team member contributions.

C Ethics and Compliance Statements

Include any declarations about data ethics, consent, or compliance.

D Evidences of CEP

E Evidences of CEA