PREDICTING MECHANICAL VENTILATION DURATION USING DEEP LEARNING MODELS: A COMPARATIVE ANALYSIS

by

HUANG ZHIWEN

(Department of Electrical & Computer Engineering)

A THESIS SUBMITTED

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF BACHELOR OF ENGINEERING

in

NATIONAL UNIVERSITY OF SINGAPORE

Supervisors:

Professor Prahlad Vadakkepat

Dr.Andi Sudjana Putra

Examiner:

Associate Professor MAMUN, Abdullah AL

Abstract

In this thesis, the goal is to predict mechanical ventilation (MV) duration for patients using deep learning models. A dataset consisting of 36 patients, some of which had multiple intubations, was carefully preprocessed to handle missing data and outliers. Only three features were selected for prediction due to the presence of many missing values in other features: Ventilator Reading End-Tidal Carbon Dioxide (ETCO2), Ventilator Reading Respiratory Rate (breaths/min), and Ventilator Reading Tidal Volume (VT) Exhaled. Group Shuffle Split was employed to split the dataset into training and validation sets, and a 5-fold cross-validation was used to evaluate model performance. Two baseline models were used: the mean value of the training set and a decision tree regressor. Four deep learning models were compared against the baselines: Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), Temporal Convolutional Networks (TCN), and Gated Recurrent Units (GRU). The CNN model demonstrated the best performance in predicting mechanical ventilation duration. The results suggest the potential of deep learning techniques, particularly CNN, for predicting mechanical ventilation duration in critical care settings. Limitations of the study include a small dataset with missing data and some intubations being re-intubations, which may affect the predictions.

Acknowledgments

I would like to express my heartfelt gratitude to Professor Prahald for his invaluable guidance and support throughout my final year project. His mentorship and expertise have been instrumental in shaping the direction of this work.

I would like to express my deepest gratitude to Dr. Andi for his exceptional guidance, support, and encouragement throughout the course of my research. His unwavering commitment and dedication to helping me achieve my goals have been invaluable. Dr. Andi played a pivotal role in securing the dataset from MOHH, and his expertise and insights were instrumental in guiding my FYP. I am truly grateful for his patience, understanding, and mentorship, which have been crucial in helping me navigate the complexities of this research.

Furthermore, I would like to thank Dr.Amanda from MOH Holdings for providing me with access to the ventilator dataset used in this study. Without her support and contribution, this research would not have been possible.

Finally, I would like to express my appreciation to all those who have supported me throughout this journey. Your encouragement and belief in me have been a constant source of motivation and inspiration.

Contents

List of Figures

List of Tables

List of Abbreviations

ARDS Acute Respiratory Distress Syndrome

CNN Convolutional Neural Network

COPD Chronic Obstructive Pulmonary Disease

GRU Gated Recurrent Unit

ICU Intensive Care Unit

LSTM Long Short-Term Memory

MAE Mean Absolute Error

MOHH Ministry of Health Holdings

MV Mechanical Ventilation

RNN Recurrent Neural Network

TCN Temporal Convolutional Network

Chapter 1

Introduction

1.1 Background and Problem Statement

In the first semester, the student focused on classifying three disease states: Acute Respiratory Distress Syndrome (ARDS), Chronic Obstructive Pulmonary Disease. and normal lung (COPD) and normal lung, using deep learning models. This work provided valuable experience in preprocessing and analyzing time-series medical data and building deep learning models. A simulated ventilator dataset was used to gain an understanding of how each feature of the ventilator affected or indicated a patient's condition, while a electrocardiogram signal dataset helped the student to build a data pipeline to preprocess time-series data and build different deep learning models to conduct experiments. However, the dataset provided by MOHH did not include disease labels, which required a shift in project goal. After discussion with doctors from Ministry of Health Holdings (MOHH), the project focus is shifted to mechanical ventilation (MV) duration prediction.

MV is a life-sustaining treatment that helps patients breathe when they are unable to do so on their own, as shown in Figure 1.1. However, prolonged MV

can cause a range of complications and is associated with increased mortality rates. Accurately predicting the duration of MV can help clinicians make more informed decisions about patient care and reduce the risk of complications associated with prolonged MV [1].

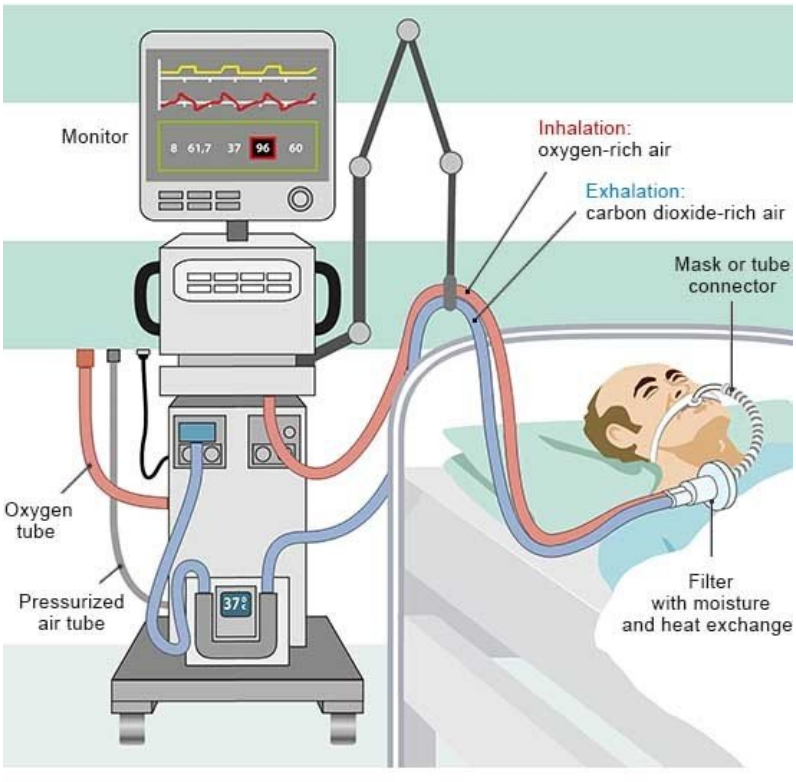


Figure 1.1: A mechanical ventilator provides vital respiratory support to critically ill patients. [2]

This problem is of great significance in the medical field, as respiratory failure is a leading cause of mortality and morbidity worldwide. A study has shown that in the United States, respiratory failure is the third most common cause of death, with a mortality rate of 29.3% for patients requiring MV [3]. Therefore, developing an accurate and reliable method to predict the duration of MV is essential to improve patient outcomes and reduce healthcare costs.

Several studies have explored the application of machine learning techniques to predict mechanical ventilation outcome or duration. However, it is still unclear if machine learning contributes to a higher extubation success rate [4]. Figueroa-Casas et al. developed predictive models to determine if a mechanical ventilation is prolonged or not, but they did not predict the exact duration [5] [6]. Mohammed Sayed utilized machine learning to predict the duration of mechanical ventilation for ARDS patients, but the results were not good enough for practical applications and deep learning models were not used [1]. Therefore, it is imperative to explore the potential of deep learning models in predicting the duration of mechanical ventilation.

1.2 Project Goal

In this project, several experiments were done to compare the performance of different deep learning models, including Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), Temporal Convolutional Networks (TCN), and Gated Recurrent Units (GRU), on a dataset of 26 patients requiring MV, with some having multiple intubations. The dataset originates from Ministry of Health Holdings (MOHH) and contains information on 39 patients, of which 26 are included in the analysis after filtering out patients with a lot of missing or incomplete data. A detailed description of the dataset, including the process of filtering out unsuitable cases and a table showing the basic information of each patient, will be provided later in this thesis. Due to the presence of many missing values in other features, only three features were selected for prediction: Ventilator Reading End-Tidal

Carbon Dioxide (ETCO2), Ventilator Reading Respiratory Rate (breaths/min), and Ventilator Reading Tidal Volume (VT) Exhaled.

The student utilized Group Shuffle to partition the dataset into separate training and validation sets, ensuring that data from the same patient was not present in both sets simultaneously. The performance of four deep learning models (LSTM, CNN, TCN, and GRU) was compared against two baseline models, the mean value of the training set and a decision tree regressor, using a 5-fold cross-validation approach.

1.3 Results

Results revealed that the CNN model exhibited the most superior performance in predicting the duration of mechanical ventilation. mong the deep learning models, CNN achieves the best overall performance, as demonstrated by its small mean MAE across different window sizes. Especially, when the window size is increased to 60, the mean Mean Absolute error (MAE) across five folds of CNN is 37.06 hours, which is 41% less than the mean MAE of the mean value baseline. This emphasizes the potential of deep learning techniques, particularly convolutional neural networks, for accurately predicting MV duration in critical care settings with limited feature sets. The study contributes to the ongoing exploration of deep learning applications in healthcare and provides insights that may be relevant for future research and clinical practice.

1.4 Thesis Organization

The rest of the report is organized as follows. Chapter 2 provides a comprehensive literature review of previous work on medical time series data processing, deep learning models in medical areas, predicting MV duration, and related topics. Chapter 3 provides the statistics of the dataset, including a table showing the demographics of 36 patients in the dataset, and details the data preprocessing steps, including data extraction, data transformation, and data cleaning used in this project. Chapter 4 presents the methodology used for train-validation set splitting, sliding window method for data augmentation, selection of two baseline models, model structure of four deep learning models, model training, and evaluation. Chapter 5 presents the results of the experiments and discusses their implications. Finally, Chapter 6 provides a conclusion, discusses some limitations, and suggests future work.

Chapter 2

Literature Review

2.1 Deep Learning Models in Medical Area

Recent advances in deep learning techniques have demonstrated remarkable success in various healthcare applications, including time series analysis and prediction tasks [7]. However, the literature specifically focusing on MV duration prediction using deep learning methods remains limited. In light of these developments, this study aims to investigate the potential of deep learning models in predicting MV duration based on a limited set of patient time series data.

2.1.1 Convolutional Neural Network

The architecture consists of several layers, including input, convolutional, activation, pooling, fully connected, and output layers.

Convolutional layers detect features within the input data using filters or kernels, which slide over the input data, generating feature maps. The filter size, stride, and padding parameters impact the dimensions of the output feature map.

Activation functions, such as Rectified Linear Unit (ReLU) and sigmoid, introduce

non-linearity into CNNs, allowing the model to learn complex patterns. Pooling layers reduce dimensionality and computational complexity by summarizing information within the feature maps.

Fully connected layers integrate learned features and connect them to the output layer, often employing dropout for regularization. The output layer provides final predictions based on the features extracted and processed throughout the network.

Loss functions and optimization techniques are used to train the network by minimizing the error between predictions and ground truth values. An example of CNN architecture is shown in Figure 2.1. Knowing the role of these parameters and layers is helpful for model tuning.

CNN has been widely used in image classification and recognition tasks, but their effectiveness in analyzing medical time series data has also been demonstrated in recent years. CNNs can extract features automatically from raw data, which is particularly useful in medical applications where feature engineering can be challenging due to the complexity of the data [8].

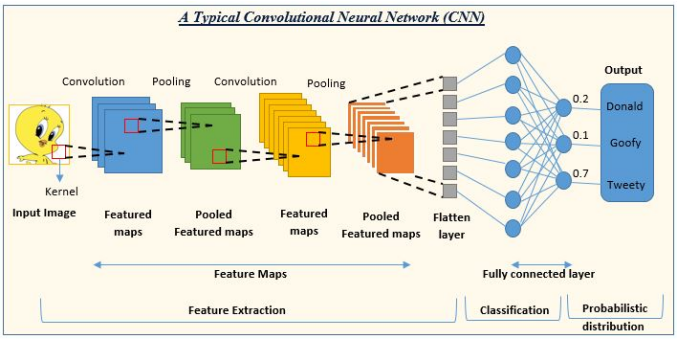


Figure 2.1: Convolutional Neural Network Architecture[9]

In the medical field, CNNs have been applied to various time series classification tasks such as electrocardiogram (ECG) classification [10]. The success of CNNs

in these applications is due to their ability to capture local patterns and temporal dependencies in the time series data [8]. In their review article, Anwar et al. discussed the application of convolutional neural networks (CNNs) in medical image analysis. They highlighted the effectiveness of CNNs in detecting and classifying various medical images, such as mammograms, X-rays, and MRI scans. The authors noted that CNNs have shown promising results in medical image segmentation, which is a crucial task in medical image analysis for identifying and locating abnormal regions in an image. Anwar et al. also discussed the challenges and limitations of using CNNs in medical image analysis, such as the need for large datasets and the interpretability of the model's decisions [11].

These studies demonstrate the effectiveness of CNN in analyzing medical time series data and their potential to improve clinical decision-making. The ability of CNN to learn and extract relevant features from raw data can reduce the need for manual feature engineering and enable more accurate predictions.

2.1.2 Long Short-Term Memory Neural Network

LSTM is a type of recurrent neural network (RNN) architecture that is designed to handle the issue of vanishing gradients that occurs with traditional RNNs.

LSTM architecture comprises memory cells, input gates, forget gates, and output gates. These components work together to enable the network to learn and retain crucial information from input sequences while discarding less important data.

Memory cells store and manage information over time, allowing the network to remember patterns from the input sequence.

Input gates control the flow of new input data into the memory cells, deciding which information to store based on its relevance.

Forget gates determine which information to retain or discard from the memory cells, ensuring that only relevant information is used for predictions. Output gates regulate the flow of information from the memory cells to the output layer, providing the final predictions based on the extracted features. An example of LSTM architecture is shown in Figure 2.2.

LSTMs is well-suited for modeling temporal dependencies and have been widely used in various applications, including natural language processing, speech recognition, and time series analysis [8].

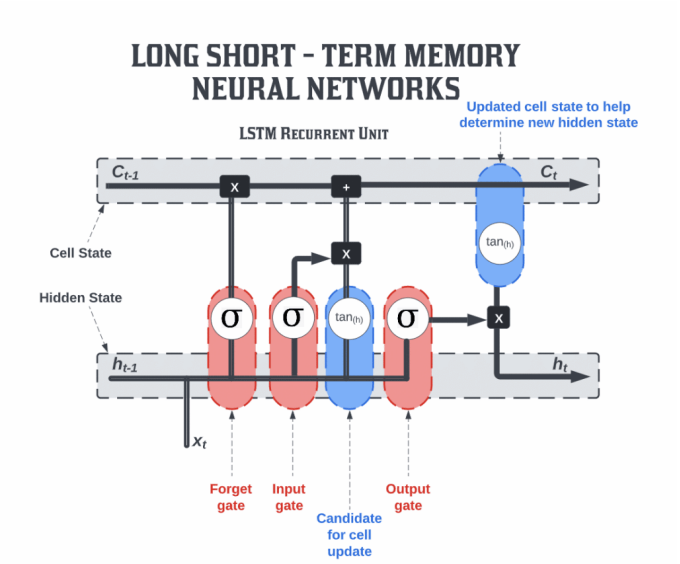


Figure 2.2: Long Short-Term Memory (LSTM) [12]

Lu used LSTM to predict hospital-acquired acute kidney injury onset in COVID-19 patients. The inclusion of demographics and comorbidities with longitudinal laboratory data improved prediction accuracy, yielding an AUC of 0.965 and accuracy of 89.57% for the best model on the Montefiore validation dataset. LSTM models

of longitudinal clinical data could help identify patients for early interventions to prevent long-term renal complications [13].

2.1.3 Temporal Convolutional Networks

TCN is a class of deep neural networks designed for processing time series data. TCN use 1D convolutional layers to process input sequences, employing causal convolutions, dilation, and residual connections for efficient learning.

Causal convolutions maintain the temporal structure of the data, processing the input sequence in a forward direction. Dilation enables the model to capture dependencies at different time scales, efficiently processing input sequences of varying lengths. Residual connections mitigate the vanishing gradient problem and allow deeper network stacking, enhancing the model's ability to capture complex patterns [14]. An example of TCN architecture is shown in Figure 2.3.

Similar to CNNs, TCNs are characterized by their ability to learn local patterns in the data, but in contrast to LSTMs, TCNs do not have a memory cell. Instead, they use dilated convolutions to capture long-term dependencies and make predictions based on the entire history of the input sequence [14]. Zhao proposes a hierarchical attention-based temporal convolutional network (HA-TCN) architecture for myotonic dystrophy diagnosis using handgrip force time series data. The study compares the HA-TCN model to benchmark TCN models, LSTM models, and SVM approaches, finding that deep learning models outperform SVM, and the HA-TCN model outperforms its TCN counterpart in terms of computational efficiency and performance. Additionally, the HA-TCN model can consistently identify

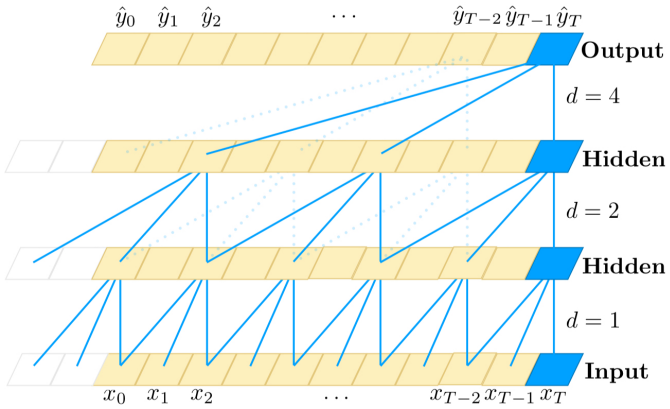


Figure 2.3: Temporal Convolutional Network (TCN) [15]

relevant time series segments and exhibits increased robustness to noise compared to attention-based LSTM models [16].

2.1.4 Gated Recurrent Unit

GRU is another variant of the RNN architecture that is a simpler version of LSTM that has fewer parameters, which makes it computationally more efficient [17].

GRU consists of three key components: reset gates, update gates, and hidden states. Reset gates control the flow of past information, update gates manage the balance between the previous hidden state and new input data, and hidden states maintain extracted features and information from the input sequence. Understanding these components is crucial for designing and optimizing GRU models for various applications. An example of GRU architecture is shown in Figure 2.4.

Li discusses the use of GRU with attention mechanism for biomedical event extraction, specifically the Bacteria Biotope event extraction (BB) task in the BioNLP Shared Task 2016. Biomedical event extraction is a crucial task in biomedical

text mining and requires fine-grained information extraction. The presented approach achieved an F-score of 57.42% in the test set, outperforming previous state-of-the-art official submissions to BioNLP-ST 2016 [18].

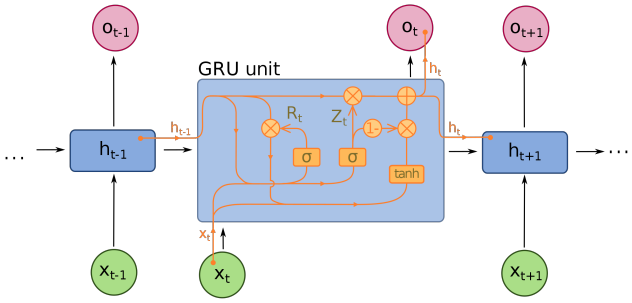


Figure 2.4: Gated Recurrent Unit (GRU) [19]

2.2 Medical Time Series Data

Due to its complexity and high dimensionality, medical time series data presents a significant challenge for feature extraction. As a result, existing approaches of mv duration prediction often rely on clinicians' expertise, which may be subjective and can lead to inconsistencies in patient management [20]. In medical time series data, feature extraction is particularly challenging because the data is often noisy, non-stationary, and may contain a lot of miss values.

To address these challenges, deep learning models have been developed to automatically extract features from medical time series data and demonstrated remarkable success in various healthcare applications, including time series analysis and prediction tasks. These models are trained to learn a set of hierarchical representations of the input data, with each layer of the model capturing more abstract and complex features [7].

Chapter 3

Dataset and Data Preprocessing

3.1 Datasource and Description

The dataset used in this study was collected by doctors from MOHH. It consists of data from 36 patients who underwent mechanical ventilation in intensive care units (ICUs). The dataset covers various time periods, with each patient having different durations in the ICU. Some patients have multiple intubations, while others do not have any intubation records. Table 3.1 gives an overview of patients' characteristics. A histogram in Figure 3.1 was used to show the distribution of MV duration.

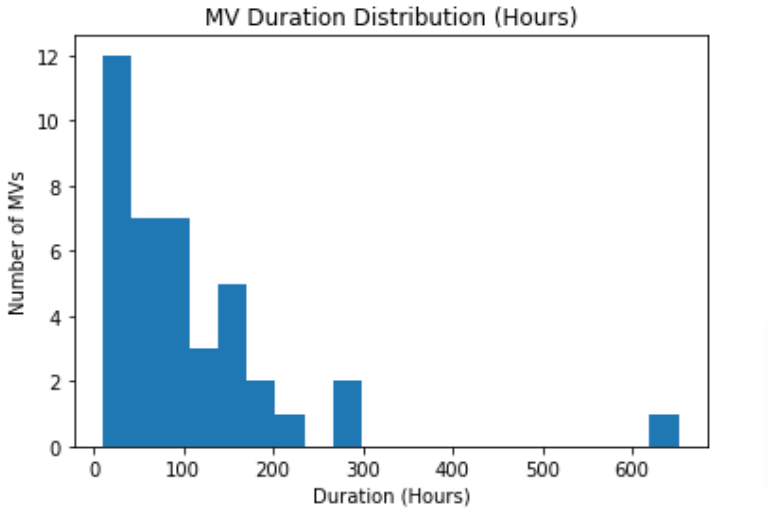


Figure 3.1: MV duration histogram.

3.2 Data Preprocessing

In the preprocessing stage, the dataset was cleaned and organized by the author. Firstly, a pandas dataframe was utilized to store the dataset. Duplicated rows were removed, as the dataset contained multiple duplicated data, especially for patients with multiple ICU admissions and surgeries. Rows with null values were also dropped. The data was sorted by 'Case Number', 'Created Datetime', and 'Item Description' columns. Each intubation was separated for model training purposes since some patients had multiple intubations. A new 'Hours' column was added to represent the time of sampling for each intubation, ensuring the time started from zero for each intubation. Nine cases (case 2, 3, 4, 12, 13, 18, 20, 33, 36) without ventilator data and case 01 with an outlier MV duration of over 600 hours were excluded from the dataset. Lastly, intubations with an MV duration of less than 10 hours were removed as they were deemed too short for training with the sliding window method.

3.3 Iutubation Seperation

This section introduces the procedure of intubation separation applied to the dataset. Since some patients have multiple intubations, each with different durations of mechanical ventilation, it is necessary to separate each intubation individually to be used in the model training process. Figure 3.2 and Figure 3.3 are included to illustrate the differences before and after intubation separation for case 025. It can be observed that two intubation with different MV duration was separated from

case 025.

Figure 3.2: Mechanical ventilation data of case 025 before intubation separation.

3.4 Data Pipeline

To ensure that the above data preprocessing techniques can be used for new data that will be added in the future, a data pipeline has been created. This pipeline includes all the necessary preprocessing steps, such as data cleaning, imputation of missing values, data visualization, feature selection, and intubation separation. This pipeline has been designed to be scalable and flexible, so that it can be applied to different datasets with varying characteristics. By using this data pipeline, new data can be easily processed and integrated into the existing dataset, and the resulting data can be used for further analysis and modeling. Overall, this pipeline ensures

the reproducibility and reliability of the data preprocessing process, and helps to facilitate the analysis and interpretation of the data.

3.5 Feature Selection

Although the ventilator machine gives many reading and setting data, only three features were selected for prediction. The reason for selecting these three features is that other ventilator reading and setting data have a lot of missing values, making them unsuitable for model training. The features selected as predictors are:

- **Ventilator Reading ETCO2**: End-tidal carbon dioxide (ETCO2) is the partial pressure or maximal concentration of carbon dioxide (CO2) at the end of an exhaled breath, which is considered to be a noninvasive estimate of the arterial carbon dioxide (PaCO2) level in the body [21].

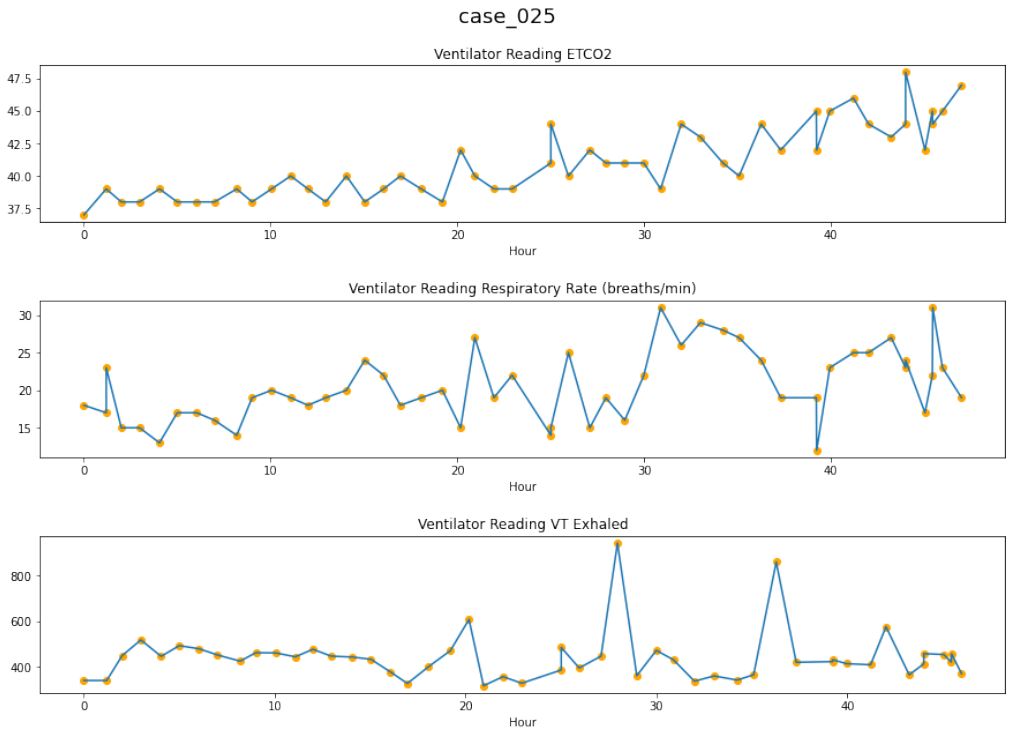
- **Ventilator Reading Respiratory Rate (breaths/min)**: Respiratory rate (RR) is the number of breaths that a person takes per minute. It is usually measured when a person is at rest and simply involves counting the number of breaths for one minute by placing a hand on the chest or abdomen and counting each time the chest rises.

- **Ventilator Reading VT Exhaled**: Exhaled tidal volume (VT) is the amount of air that is exhaled during each breath. It is usually measured using a spirometer, which measures the amount of air that a person breathes in and out of their lungs [22].

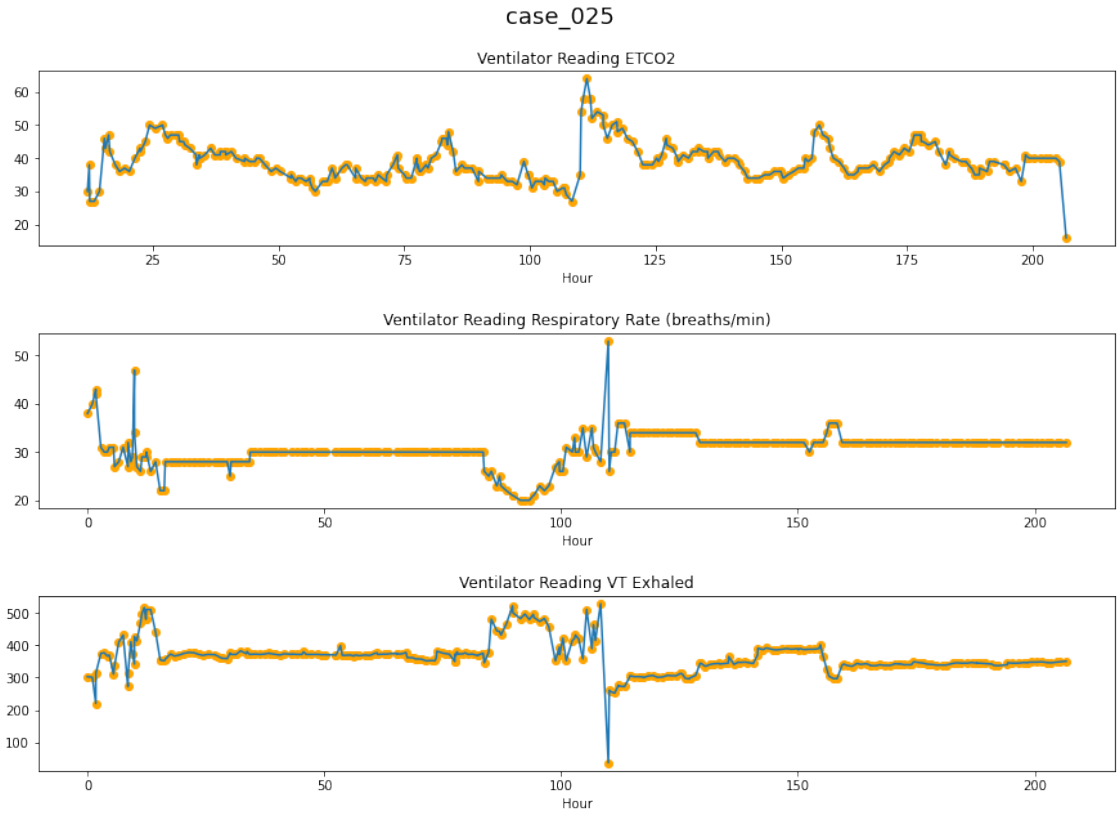
These three predictors were also confirmed by doctors that they may be helpful for MV duration prediction as they are relevant indicators of lung function and the effectiveness of MV support.

Table 3.1: Demographics of 36 patients

Case	Age	Gender	Race	Weight (kg)	Height (cm)	BMI (kg/m2)	Smoking History
1	69	Female	Chinese	48.5	160.0	18.9	Current smoker
2	91	Female	Indian	57.8	155.0	24.1	Non-smoker
3	42	Female	Indian	79.9	166.0	29.0	Non-smoker
4	97	Male	Chinese	51.7	156.0	21.2	Ex-smoker
5	45	Male	Malay	46.0	173.0	15.4	Non-smoker
6	67	Male	Chinese	44.8	168.0	15.9	Not asked
7	84	Male	Chinese	53.6	155.0	22.3	Non-smoker
8	71	Female	Chinese	64.0	153.0	27.3	Non-smoker
9	82	Female	Chinese	47.5	153.0	20.3	Not asked
10	66	Male	Chinese	53.7	171.0	18.4	Non-smoker
11	71	Male	Chinese	56.6	169.0	19.8	Non-smoker
12	75	Male	Chinese	46.6	163.0	17.5	Ex-smoker
13	75	Female	Chinese	34.6	146.0	16.4	Non-smoker
14	84	Male	Chinese	64.9	167.0	23.3	Not asked
15	69	Male	Chinese	46.6	164.0	17.3	Not asked
16	83	Male	Chinese	58.5	158.0	23.4	Non-smoker
17	63/64	Female	Malay	97.6	150.0	43.4	No data
18	76	Male	Chinese	76.0	174.0	25.1	Ex-smoker
19	67	Male	Chinese	43.9	164.0	16.3	Current smoker
20	77	Female	Chinese	94.4	161.0	36.4	Non-smoker
21	80	Male	Chinese	64.6	161.0	24.9	Non-smoker
22	82	Male	Chinese	NaN	NaN	NaN	Non-smoker
23	81	Female	Chinese	37.2	160.0	14.5	Non-smoker
24	90	Female	Chinese	53.7	145.0	25.5	Non-smoker
25	41	Female	Chinese	73.0	173.0	24.4	Non-smoker
26	89	Male	Chinese	57.4	152.0	24.8	Current smoker
27	72	Male	Chinese	80.1	165.0	29.4	Non-smoker
28	76	Male	Chinese	51.8	159.0	20.5	Ex-smoker
29	69	Female	Chinese	51.4	156.0	21.1	Non-smoker
30	69	Male	Chinese	44.0	162.0	16.8	Non-smoker
31	85	Male	Malay	60.9	157.0	24.7	Non-smoker
32	65	Male	Chinese	60.3	160.0	23.6	Current smoker
33	65	Male	Chinese	65.1	NaN	NaN	Non-smoker
34	67	Female	Chinese	75.4	162.0	28.7	Non-smoker
35	67	Female	Chinese	56.7	153.0	24.2	Not asked
36	69	Male	Chinese	61.6	163.0	23.5	Non-smoker



(a) First intubation extracted from case 025



(b) Second intubation extracted from case 025

Figure 3.3: Mechanical ventilation data of case 025 after intubation separation.

Chapter 4

Methodology

4.1 Train and Validation Data Split

In order to evaluate the performance of the deep learning models, 40 MVs extracted from the dataset were split into training and validation sets. In time series data analysis, random splitting of data for training and validation cannot be used, as it may lead to information leakage. Information leakage occurs when data points in the validation set are correlated with the data points in the training set. This can result in the model overfitting to the training set and performing poorly on the validation set [23]. Hence, to maintain the integrity of the dataset and ensure that MVs belonging to the same patient do not appear in both the training and validation sets simultaneously, a Group Shuffle Split approach was employed. This method groups the MVs by their associated patient and shuffles them, maintaining the patient grouping throughout the process.

Test size was set to be 0.3, which means 30% of the data is used for validation, while the remaining 70% is used for training the models. To further assess the effectiveness and robustness of the models, a 5-fold cross-validation technique was also used. This means that the dataset is divided into five equally-sized folds, and

the model is trained and evaluated five times, each time using a different fold as the validation set and the remaining four folds as the training set.

By using this approach, the risk of overfitting was minimized and can better generalize the model's performance across different patients. This helps the author to assess the effectiveness of the models in predicting mechanical ventilation duration for patients in a real-world setting.

4.2 Sliding Window Method

To capture the temporal dependencies and patterns within the mechanical ventilation data, a sliding window method was used to preprocess the data before feeding it into the deep learning models. The sliding window method enables me to analyze the variations in the data and uncover the inherent relationships between the features over time.

Given the sample rate of the data is one hour, the sliding window's slide was set to one hour as well. This means that the window moves forward one hour at a time, extracting a time series from the data at each step. The sliding window method is applied separately to the training and validation sets after the train-validation split has been performed. This ensures that no information leakage occurs between the two sets during the preprocessing step.

In order to determine the optimal window size for the models, the author experiment with different window sizes and analyze the influence of the window size on model performance. By selecting the appropriate window size, one can ensure that the models effectively capture the temporal dependencies in the data and

provide accurate predictions for mechanical ventilation duration. The experiments results visualized as plots and shown in the Results and Analysis part.

4.3 Baseline Model

In this study, two baseline models were employed to evaluate the performance of deep learning models in predicting the duration of mechanical ventilation. These baseline models provide a point of comparison to ensure that the deep learning models are effectively learning from the data and identifying relevant patterns.

4.3.1 Mean Value of the Training Set

The first baseline model is the mean value of the mechanical ventilation duration in the training set. This model represents a simple and intuitive guess of the MV duration. By comparing the performance of the deep learning models to this baseline, it can be determined whether the models are genuinely learning from the data and discovering patterns that contribute to better predictions.

4.3.2 Decision Tree Regressor

The second baseline model is the Decision Tree Regressor. A Decision Tree Regressor is a non-parametric supervised learning method that can be used for regression tasks. It works by recursively splitting the input space into non-overlapping regions and fitting a constant model (e.g., the mean) within each region. The tree structure provides a flexible and interpretable representation of the relationships between input features and the target variable [24].

The Decision Tree Regressor can be used as a baseline regression model for

time series data because it can capture complex, non-linear relationships between input features and the target variable without making any assumptions about the underlying data distribution. Although it may not be as powerful as more sophisticated deep learning models in capturing temporal dependencies, it provides a reasonable point of comparison to assess whether the deep learning models are indeed leveraging the time series nature of the data to make more accurate predictions.

4.4 Deep Learning Models

In this section, four deep learning models are introduced: LSTM, CNN, TCN, and GRU. For all models, the Adam optimizer is used for optimization and the MAE serves as the loss function. Early stopping and learning rate reduction are employed as callbacks during training to ensure optimal model performance. The early stopping callback restores the best model weights when the validation loss stops improving, while the learning rate reduction callback decreases the learning rate when the validation loss plateaus. The model structures presented here are the result of experimentation with different architectures, starting from a single layer and incrementally adding layers and units to fine-tune each model for the best performance. The model structures are presented in Figure 4.1, 4.2, 4.3, and 4.4 respectively.

4.4.1 LSTM Model

The Long Short-Term Memory (LSTM) model is a type of recurrent neural network (RNN) that is capable of capturing long-range dependencies in time series

data [8]. In this work, an LSTM model with 64 hidden units is used, followed by a dropout layer with a rate of 0.5 to prevent overfitting. The model then feeds into a dense layer with 32 units and a ReLU activation function, and finally, another dense layer with a single output unit and a linear activation function.

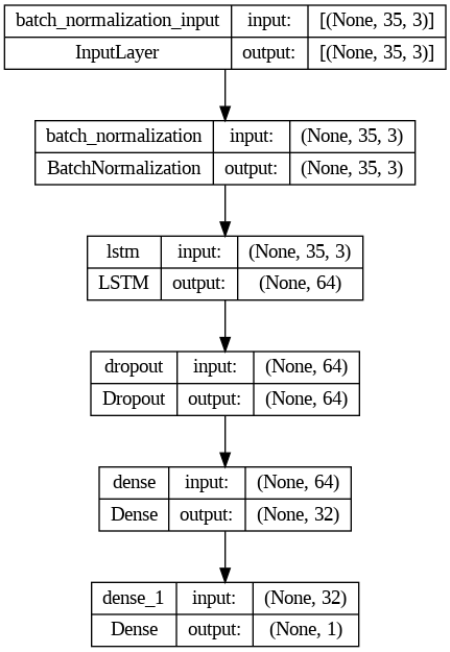


Figure 4.1: LSTM Model Structure (With a sliding window size equal to 35).

4.4.2 CNN Model

The Convolutional Neural Network (CNN) model is used to capture local patterns in time series data [8]. The CNN model consists of a 1D convolutional layer with 64 filters and a kernel size of 3, followed by a max-pooling layer with a pool size of 2, and a dropout layer with a rate of 0.3. The model then includes a flatten layer, a dense layer with 32 units and a ReLU activation function, another dropout layer with a rate of 0.3, and finally a dense layer with a single output unit and a linear activation function.

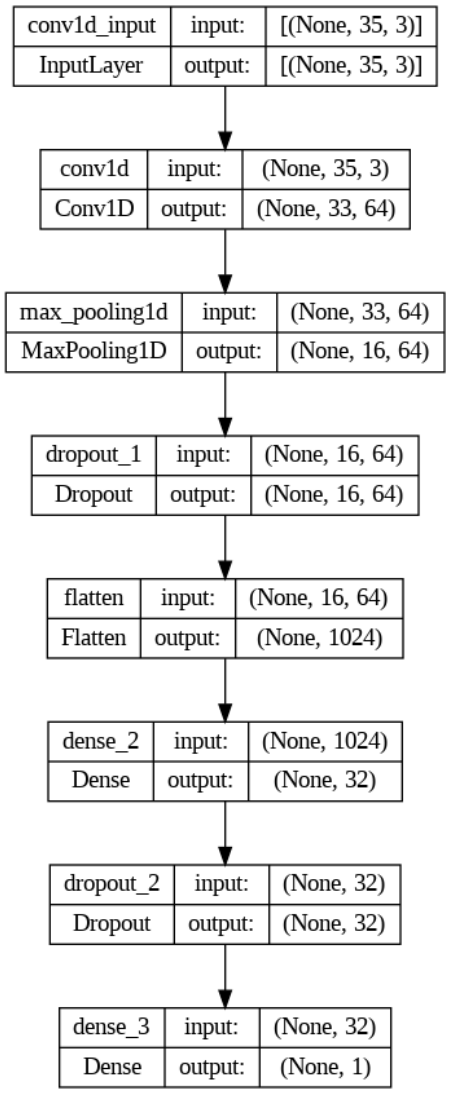


Figure 4.2: CNN Model Structure (With a sliding window size equal to 35).

4.4.3 TCN Model

The Temporal Convolutional Network (TCN) model is designed to capture both local and global patterns in time series data [14]. The TCN model uses dilated causal convolutions to handle long-range dependencies in the data. In this work, a TCN model with 32 filters, a kernel size of 3, and a dilation rate that increases exponentially from 1 to 256 is used. The model also includes skip connections to improve training efficiency. The model then feeds into a dense layer with a single

output unit and a linear activation function.

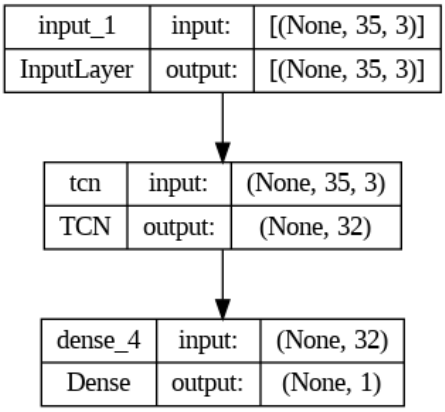


Figure 4.3: TCN Model Structure (With a sliding window size equal to 35).

4.4.4 GRU Model

The Gated Recurrent Unit (GRU) model is another type of RNN that is capable of capturing long-range dependencies in time series data [17]. The GRU model is used with 32 hidden units, followed by a dense layer with a single output unit and a linear activation function.

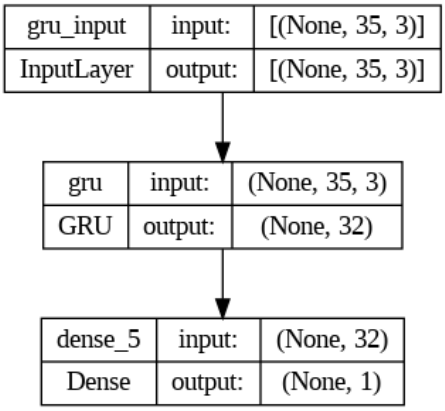


Figure 4.4: GRU Model Structure (With a sliding window size equal to 35).

Chapter 5

Results and Analysis

In this section, the performance comparison of the six models (two baselines and four deep learning models) for ten different window sizes: 15, 20, 25, 30, 35, 40, 45, 50, 55, and 60 is presented in Figure 5.2. For each window size, box plots were created to visualize the distribution of the MAE across the five cross-validation folds. These box plots allow for an assessment of the variability in performance and the stability of the models for each window size.

Additionally, the mean MAE of all models of different window sizes is shown in Table 5.1. The best result of each window size was highlighted in the table. To visualize these results, it was plotted in Figure 5.1. This plot demonstrates the impact of window size on the performance of the models and allows for a comparison of their performance across different window sizes. In the results, it can be observed that all deep learning models outperform the two baseline models. Among the deep learning models, CNN achieves the best overall performance, as demonstrated by its small mean MAE across different window sizes. Especially, when the window size is increased to 60, the mean MAE of CNN is 37.06 hours, which is 41% less than the MAE of the mean value baseline. TCN also exhibits strong performance, with a particularly noteworthy aspect being the focused MAE distribution across

Table 5.1: Mean MAE for each model and window size

Window Size	Mean Value	DTR	LSTM	GRU	CNN	TCN
15	64.62	77.17	**49.41**	54.57	50.75	54.57
20	64.46	77.06	48.84	53.70	**48.13**	52.39
25	64.43	73.95	53.71	53.87	**49.05**	51.15
30	64.26	70.55	51.17	53.23	**46.05**	48.27
35	63.88	71.01	**45.09**	53.60	46.87	48.80
40	63.49	74.37	**44.45**	54.38	45.40	51.20
45	63.08	72.09	48.53	54.88	**44.33**	51.55
50	62.85	66.14	**41.28**	56.13	44.80	47.33
55	62.81	66.83	43.99	61.77	**42.98**	48.86
60	62.59	67.81	45.42	62.90	**37.06**	49.03

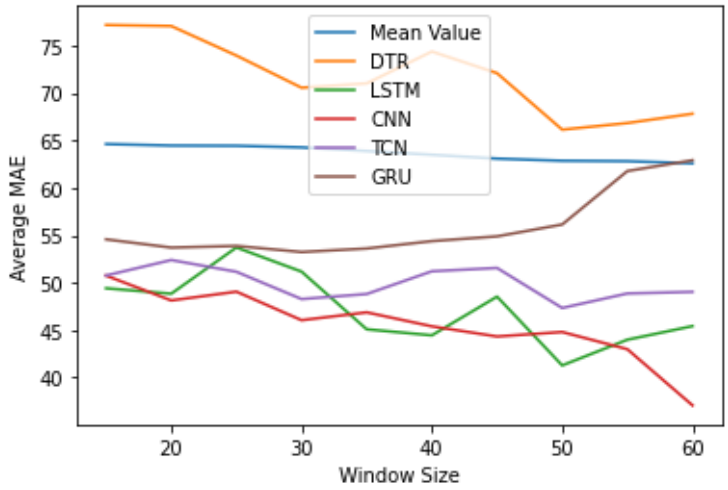


Figure 5.1: The impact of window size on the average mean absolute error (MAE) of each model. This line plot illustrates the relationship between the window size (15, 20, 25 ..., 55, 60) and the average MAE across five cross-validation folds for the six models (two baselines and four deep learning models).

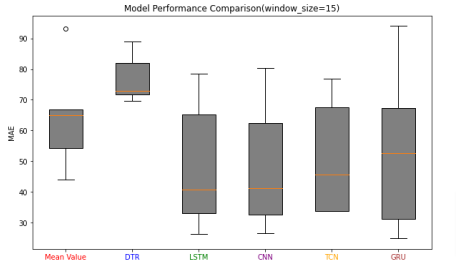
different folds. This implies that the standard deviation of TCN's performance is small, making it a more stable model.

The LSTM model performs between CNN and TCN, it performs worse than TCN when window size is small but outperforms TCN when window size is larger than 35. On the other hand, GRU model, although generally better than the baseline models, displays unstable performance across folds, resulting in a large standard deviation.

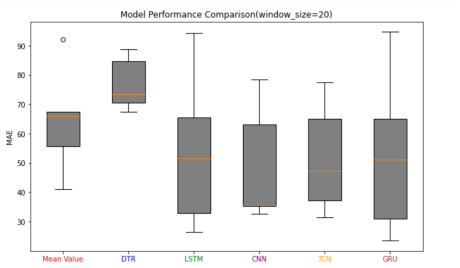
Interestingly, GRU's performance does not improve with increasing window size; in fact, it appears to worsen. This is in contrast to the other three deep learning models, which demonstrate better performance with larger window sizes.

This observation suggests that larger windows, which contain more information, enable the models to better identify patterns and dependencies within the data. However, this trend does not hold for the GRU model, indicating that there may be an underlying issue with its ability to utilize the additional information provided by larger window sizes effectively. Overall, the analysis highlights the strengths and weaknesses of each model and their performance in relation to window size, offering valuable insights for further model optimization and selection.

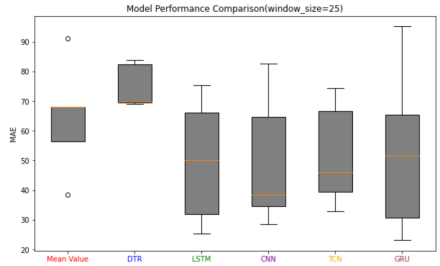
To visualize the predictied results of the CNN model, a figure was plotted when the window size was set to 60, as shown in Figure 5.3. The figure shows the actual MV duration and predicted MV duration with the sliding window as the input. It is worth noting that there is no patient overlap between the train and validation data, meaning the model has never seen the actual values before. From the figure, it can be observed that the CNN model has learned the internal pattern and has a very similar trend to the actual data, indicating the good performance of the model. Interestingly, upon closer examination, it was observed that the predicted values for each MV duration in the figure were decreasing. This may indicate that the model predicted the total MV duration calculated from the input window. This hypothesis could be verified by further investigation in future work.



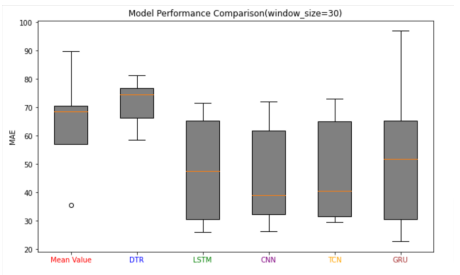
(a) window size = 15



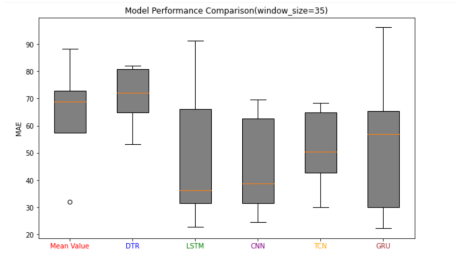
(b) window size = 20



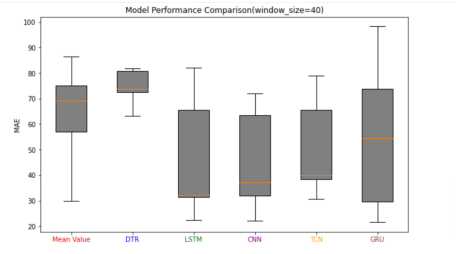
(c) window size = 25



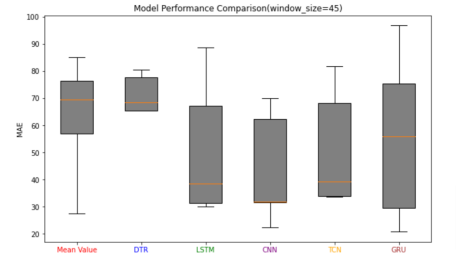
(d) window size = 30



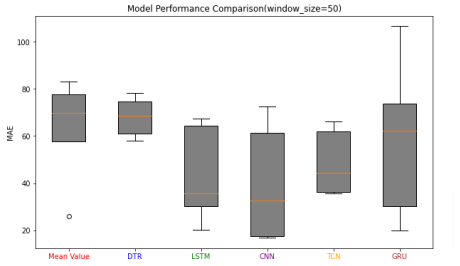
(e) window size = 35



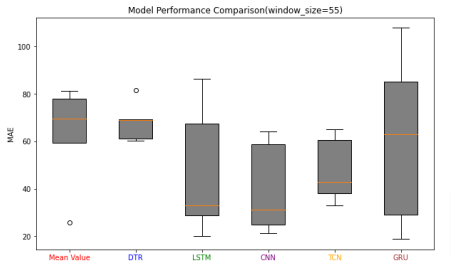
(f) window size = 40



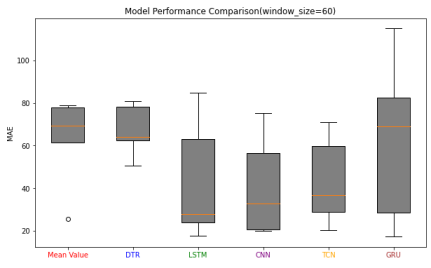
(g) window size = 45



(h) window size = 50



(i) window size = 55



(j) window size = 60

Figure 5.2: Boxplots of the average mean absolute error (MAE) of six models (two baselines and four deep learning models) across five cross-validation folds for ten window sizes ranging from 15 to 60. Each subfigure corresponds to a specific window size, illustrating the distribution and variability in performance for each model

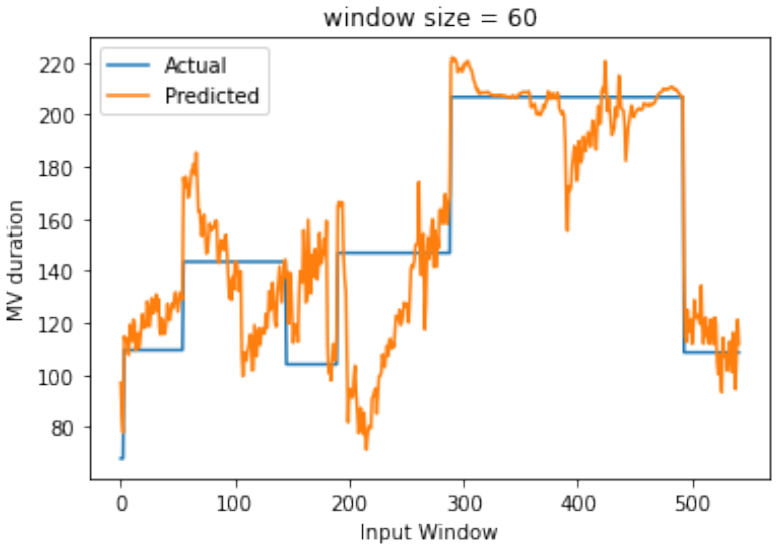


Figure 5.3: Comparison of Predicted and Actual Values for the CNN Model

Chapter 6

Conclusion and Future Work

In conclusion, this study investigated the use of various deep learning models, including CNN, TCN, LSTM, and GRU, to predict the duration of mechanical ventilation for patients. The results demonstrated that all deep learning models outperformed the two baseline models, with CNN generally showing the best performance. It was also observed that LSTM and TCN models performed well, potentially due to their ability to capture temporal dependencies and patterns within the time series data. These findings suggest that deep learning models can be effectively employed to predict mechanical ventilation duration, offering valuable insights for clinicians and healthcare providers.

However, several limitations should be considered when interpreting the results of the study. Firstly, the dataset is relatively small, with only 26 patients available for analysis. A limited sample size can reduce the generalizability of the findings and may lead to overfitting in the models, potentially affecting the performance of the models when applied to larger, more diverse patient populations.

Secondly, the study only utilizes three features from the ventilator machines, as a substantial amount of reading and setting data is either missing or not recorded. The limited availability of these features may hinder the models' ability to fully

capture the underlying patterns and dependencies within the data. The performance of the models could potentially be improved if additional relevant features were included in the analysis.

Lastly, the dataset consists of both successful and unsuccessful mechanical ventilation events, with some patients requiring re-intubation. This variability in the success of intubations may affect the models' ability to accurately predict the duration of mechanical ventilation. Future studies could potentially benefit from distinguishing between successful and unsuccessful intubations or focusing solely on successful cases to better understand the factors that contribute to the accurate prediction of mechanical ventilation duration.

Despite these limitations, the study provides a valuable starting point for future research aiming to improve the prediction of mechanical ventilation duration using deep learning techniques. Further studies could explore the incorporation of additional features, larger datasets, and more diverse patient populations to enhance the generalizability and performance of the models. Additionally, future research could investigate the impact of differentiating between successful and unsuccessful intubations on the accuracy of the predictions. Researchers could further explore the prediction of the remaining MV duration from the input time point rather than the total MV duration. This approach may offer a more accurate representation of the medical pattern within the ventilator data.

References

[1] M. Sayed, D. Riaño, and J. Villar, "Predicting duration of mechanical ventilation in acute respiratory distress syndrome using supervised machine learning", *Journal of Clinical Medicine*, vol. 10, no. 17, p. 3824, 2021.

[2] informedhealth.org, "How does mechanical ventilation work during an operation?" `https://www.informedhealth.org/how-does-mechanical-ventilation-work-during-an-operation.html`, n.d.

[3] I. L. Titlestad, A. T. Lassen, and J. Vestbo, "Long-term survival for copd patients receiving noninvasive ventilation for acute respiratory failure", *International Journal of Chronic Obstructive Pulmonary Disease*, vol. 8, p. 215, 2013.

[4] Y. Igarashi, K. Ogawa, K. Nishimura, S. Osawa, H. Ohwada, and S. Yokobori, "Machine learning for predicting successful extubation in patients receiving mechanical ventilation", *Frontiers in Medicine*, vol. 9, p. 252, 2022.

[5] J. B. Figueroa-Casas, S. M. Connery, R. Montoya, A. K. Dwivedi, and S. Lee, "Accuracy of early prediction of duration of mechanical ventilation by intensivists", *Annals of the American Thoracic Society*, vol. 11, no. 2, pp. 182–185, 2014.

[6] J. B. Figueroa-Casas, A. K. Dwivedi, S. M. Connery, R. Quansah, L. Ellerbrook, and J. Galvis, "Predictive models of prolonged mechanical ventilation yield moderate accuracy", *Journal of Critical Care*, vol. 30, no. 3, pp. 502–505, Jun. 2015.

[7] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, and J. Dean, "A guide to deep learning in healthcare", *Nature medicine*, vol. 25, no. 1, pp. 24–29, 2019.

[8] M. M. Taye, "Theoretical understanding of convolutional neural network: Concepts, architectures, applications, future directions", *Computation*, vol. 11, no. 3, p. 52, 2023.

[9] S. Shah, "Convolutional neural network: An overview", *Analytics Vidhya*, Jan. 2022. [Online]. Available: `https://www.analyticsvidhya.com/blog/2022/01/convolutional-neural-network-an-overview/`.

[10] M. Bakator and D. Radosav, "Deep learning and medical diagnosis: A review of literature", *Multimodal Technologies and Interaction*, vol. 2, no. 3, p. 47, 2018.

[11] S. M. Anwar, M. Majid, A. Qayyum, M. Awais, M. Alnowami, and M. K. Khan, "Medical image analysis using convolutional neural networks: A review", *Journal of Medical Systems*, vol. 42, no. 11, p. 226, 2018.

[12] A. Team, "Introduction to long short term memory (lstm)", `https://www.aiplus.info/blog/introduction-to-long-short-term-memory-lstm/`, Jun. 2022.

[13] J. Y. Lu, J. Zhu, J. Zhu, and T. Q. Duong, "Long-short-term memory machine learning of longitudinal clinical data accurately predicts acute kidney injury onset in covid-19: A two-center study", *International Journal of Infectious Diseases*, vol. 122, pp. 802–810, 2022.

[14] M. Nan, M. Trăscău, A.-M. Florea, and C. C. Iacob, "Comparison between recurrent networks and temporal convolutional networks approaches for skeleton-based action recognition", *Sensors*, vol. 21, no. 6, p. 2051, Mar. 2021.

[15] Naoki, "Temporal convolutional networks", *Medium*, Oct. 2022. [Online]. Available: `https://naokishibuya.medium.com/temporal-convolutional-networks-94293f1a83f8`.

[16] X. Zhao, W. Cheng, Y. Liu, W. Zhang, R. Huang, Y. Zheng, M. Li, and D. Shen, "Medical time series classification with hierarchical attention-based temporal convolutional networks: A case study of myotonic dystrophy diagnosis", *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 9, pp. 2662–2673, 2020.

[17] A. Gharehbaghi, R. Ghasemlounia, F. Ahmadi, and M. Albaji, "Groundwater level prediction with meteorologically sensitive gated recurrent unit (gru) neural networks", *Journal of Hydrology*, vol. 612, p. 128 262, 2022.

[18] L. Li, J. Wan, J. Zheng, and J. Wang, "Biomedical event extraction based on gru integrating attention mechanism", *BMC medical informatics and decision making*, vol. 20, no. 1, pp. 1–14, 2020.

[19] G. Loye, "Gated recurrent unit (gru) with pytorch", https://blog.floydhub.com/gru-with-pytorch/, Accessed: March 23, 2023, 2019.

[20] F. Taylor, "A comparative study examining the decision-making processes of medical and nursing staff in weaning patients from mechanical ventilation", *Intensive and Critical Care Nursing*, vol. 22, no. 5, pp. 253–263, Oct. 2006.

[21] S. Hunziker, M. Bivens, M. N. Cocchi, and A. C. Miller, "End-tidal and arterial carbon dioxide measurements correlate across all levels of physiologic dead space", *Respiratory care*, vol. 55, no. 3, pp. 288–293, 2010.

[22] S. Hallett, F. Toro, and J. V. Ashurst, "Physiology, tidal volume - statpearls - NCBI bookshelf", https://www.ncbi.nlm.nih.gov/books/NBK482502/, 2022.

[23] M. A. Lones, "How to avoid machine learning pitfalls: A guide for academic researchers", 2021. arXiv: 2108.02497v3 [cs.LG].

[24] W.-Y. Loh, "Fifty years of classification and regression trees", *International Statistical Review*, vol. 82, no. 3, pp. 329–348, 2014.