Studies On Multi - Modal Fake News Detection

Supriya Saha

Department of Computer Science and Engineering
National Institute of Technology Rourkela

Studies On Multi - Modal Fake News Detection

Progress Report - March 2025

submitted in partial fulfillment

of the requirements for the degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Supriya Saha

(Roll Number: 122CS0101)

based on research carried out

under the supervision of

Prof. Shyamapada Mukherjee

October, 2025

Department of Computer Science and Engineering

National Institute of Technology Rourkela

Department of Computer Science and Engineering
National Institute of Technology Rourkela

October , 2025

Certificate of Examination

Roll Number: *122CS0101*
Name: *Supriya Saha*
Title of Dissertation: *Studies On Multi - Modal Fake News Detection*

We the below signed, after checking the project report mentioned above and the official record book (s) of the student, hereby state our approval of the project report submitted in partial fulfillment of the requirements of the degree of *Bachelor of Technology* in *Computer Science and Engineering* at *National Institute of Technology Rourkela*. We are satisfied with the volume, quality, correctness, and originality of the work.

Shyamapada Mukherjee
Principal Supervisor

Dr. Bibhudatta Sahoo
Head of Department

External Examiner

Department of Computer Science and Engineering
National Institute of Technology Rourkela

Prof. Shyamapada Mukherjee
Professor

October , 2025

Supervisors' Certificate

This is to certify that the work presented in the progress report entitled *Studies On Multi - Modal Fake News Detection* submitted by *Supriya Saha*, Roll Number 122CS0101, is a record of original research carried out by her under our supervision and guidance in partial fulfillment of the requirements of the degree of *Bachelor of Technology* in *Computer Science and Engineering* . Neither this project report nor any part of it has been submitted earlier for any degree or diploma to any institute or university in India or abroad.

Shyamapada Mukherjee
Professor

Dedication

I dedicate this project to my cherished family and friends, whose love and support have been my guiding force throughout this B.Tech journey. Your encouragement, understanding and patience have fueled my determination, and your belief in me has been my greatest motivation.

To my family, for your endless sacrifices and encouragement and thank you for being my constant pillars of support.

To my friends, for the laughter, late-night talks, and constant motivation. Your presence made this journey truly special.

This work is a reflection of your belief in me, and I am deeply grateful to have you all by my side.

With heartfelt gratitude,
Supriya Saha

Declaration of Originality

I, *Supriya Saha*, Roll Number *122CS0101* hereby declare that this project report entitled *Studies On Multi - Modal Fake News Detection* presents my original work carried out as a student of NIT Rourkela and, to the best of my knowledge, contains no material previously published or written by another person, nor any material presented by me for the award of any degree or diploma of NIT Rourkela or any other institution. Any contribution made to this research by others, with whom I have worked at NIT Rourkela or elsewhere, is explicitly acknowledged in the thesis. Works of other authors cited in this thesis have been duly acknowledged under the sections ''Reference'' or ''Bibliography''. I have also submitted my original research records to the scrutiny committee for evaluation of my thesis.

I am fully aware that in case of any non-compliance detected in future, the Senate of NIT Rourkela may withdraw the degree awarded to me on the basis of the present thesis.

October 21, 2025
NIT Rourkela

Supriya Saha

Acknowledgement

I would like to express my sincere gratitude to everyone who has supported and encouraged me throughout the course of this project.

I am deeply grateful to my project supervisor, ***Professor Shyamapada Mukherjee***, for his invaluable guidance, suggestions, and constant support. His encouragement and mentorship have been a great help to me in navigating challenges and staying motivated to give my best.

I would also like to extend my sincere thanks to the ***Department of Computer Science and Engineering*** for providing me with the necessary resources to carry out this work.

Finally, I am very grateful to my parents and friends for their constant motivation and belief in me. I am also thankful to the **National Institute of Technology, Rourkela** for offering me the platform and facilities to pursue this project successfully.

October 21, 2025 *Supriya Saha*
NIT Rourkela Roll Number: 122CS0101

Abstract

The detection of misinformation on social media platforms often handles text and images separately or simply combines them by simple concatenation, yielding weak cross-modal alignment and limited understanding. This weak integration leads to poor alignment between the two modalities which results in less accurate predictions and limited understanding of how those predictions are made. When the text and image in a post contradict one another then systems tend to struggle and they provide little to no explanation for their decisions.

Recent advances in multimodal learning address these challenges by improving how text and images are aligned and fused. Contrastive learning helps bring related text–image pairs closer together in the model's understanding while pushing unrelated ones apart. Cross-modal attention then allows the system to focus on whichever modality is more important for each individual post.

In order to make the process transparent, the system uses explainability techniques such as Grad-CAM, which highlights the most influential regions of an image, and token-level SHAP, which shows which words in the text contributed most to the decision.

This project proposes building an end-to-end fake news detection system for Twitter posts that combines BERT for text understanding and ResNet50 for image analysis. Both components will be pre-trained with contrastive loss to align their representations, and their outputs will be fused using cross-modal attention before classification. This approach is expected to deliver more accurate predictions and offer clear visual explanations of how each decision was made.

Keywords: *Misinformation Detection*; *Multimodal Learning*; *Contrastive Learning*; *Cross-Modal Attention*; *BERT*; *ResNet50*; *Fake News Detection*; *Explainable AI (XAI)*; *Grad-CAM*; *SHAP*; *Text–Image Fusion*.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Misinformation spreads quickly on social media which often uses both text and images, that makes detection a multimodal challenge. Traditional approaches usually handle text and images separately or combine them in a basic, straightforward manner. This weak integration results in poor semantic alignment between modalities that struggles in cases where text and image contradict each other, and offers little transparency into why a prediction was made. Models such as SpotFake use BERT for text and VGG19 for images with equal-weight feature fusion, but this approach offers limited interaction between the modalities and often faces trouble to generalize to complex, real-world misinformation.

Recent advances in multimodal learning provide more effective solutions. First, **contrastive learning** (in a CLIP-style setup) aligns text and image embeddings by bringing related pairs closer and pushing unrelated pairs apart thus improving cross-modal understanding before supervised training. Second, **cross-modal attention** allows the model to selectively focus on the most informative modality or token for each post. Finally, **explainability techniques** such as Grad-CAM for images and token-level SHAP for text helps in finding the reasoning behind predictions, enhancing transparency and supporting more thorough error analysis. Additionally, replacing VGG19 with ResNet50 strengthens visual encoding by offering better representations and inductive biases.

This project integrates these advancements into a single, end-to-end pipeline for fake news detection on Twitter posts. The system employs BERT for textual encoding, ResNet50 for image encoding (with comparative analysis against VGG19), contrastive pre-training for cross-modal alignment, cross-modal attention for intelligent fusion, and explainability techniques to make the decision-making process interpretable.

1.2 Objectives

The primary objectives of this project are:

- To design a multimodal architecture combining BERT and ResNet50 (and compare against VGG19) for robust text–image encoding.
- To pre-train the text and image encoders using CLIP-style contrastive learning thus improving the alignment between modalities and enhancing overall performance on downstream tasks.
- To introduce cross-modal attention for observation-wise and selective fusion instead of simple concatenation.
- To integrate explainability methods such as Grad-CAM (for image regions), SHAP (for token-level importance), and attention heatmaps (for text–image focus) to make predictions transparent.

1.3 Organization of Project

This project is structured into six chapters and each chapter is built based on the previous to present a complete picture of the research process:

- **Chapter 1** introduces the problem of multimodal fake news detection, presents the motivation for this research, and defines the project objectives.
- **Chapter 2** reviews the related work on multimodal misinformation detection, contrastive pre-training, attention-based fusion, and explainability in vision–language models.
- **Chapter 3** details the proposed methodology that includes encoder selection (VGG19 vs ResNet50), CLIP-style contrastive pre-training, and cross-modal attention-based fusion.

- **Chapter 4** presents the experimental results, cross-modal alignment analyses, and explainability visualizations (Grad-CAM, SHAP, and attention maps).
- **Chapter 5** discusses the limitations of the current approach, ethical considerations, and potential future work, such as incorporating temporal signals, enabling few-shot learning, and enhancing robustness to distribution shifts.
- **Chapter 6** concludes the project with key findings and their practical implications for improving fake news detection systems.

Chapter 2

Literature Review

The spread of false information on social media has changed from simple text-based messages to more complex posts that include text, claims, and images. In the late, methods for detecting abbreviation mainly treated it as a text-focused problem, using word choices, writing styles, and how information spreads to classify false content. These techniques, which included simple word lists with basic models, recurrent and convolutional neural networks, and eventually transformer models, worked well with text-heavy datasets but struggled when images supported or added to the false claim. On the image side, tools that only used images, like those based on hand-coded features or CNN models like VGG and ResNet, could spot obvious changes or simple tricks in images. But they had a hard time matching the meaning of the image with the text that accompanied it.

As datasets that include both text and images started to appear (such as Twitter and Weibo rumors, FakeNewsNet, Fakeddit, and MM-COVID), study moved toward models that could feel at both text and images. The original type of these models used late fusion, where each part (text and images) was processed separately and then joined together before making a decision. This approach performed better than previous methods and was easy to use and fast. However, late fusion treated each part equally and couldn't handle situations where the two parts conflicted or differ. It also tended to favor whichever type of data was more useful in the training data.

To fix these issues, attention-based methods and cross-modal reasoning started to be used more. Techniques like co-attention, bilinear pooling, and transformer-style cross-attention let models weigh diverse parts of the data selectively, emphasize the most important text and image elements. Vision-language transformers like VisualBERT, ViLBERT, and LXMERT showed that having a shared understanding of both text and image data outperforms just joining them directly, especially when the linkage between text and picture is important for judging truthfulness. These models similarly made it easier to question how they make decisions, leading to further explainable approach for checking misinformation.

At the similar moment, contrastive pre-training greatly enhance how well text and images match. Training methods like CLIP use a loss function called InfoNCE to bring similar text and image pairs closer together and push different ones apart in a shared space. This creates strong encoders that work well for both text and image tasks, making it easier to apply them to new problems without much training. When used for detecting misinformation, this method helps reduce reliance on misleading patterns that only work in one type of data and makes the system more robust when dealing with new or different types of information. Even small amounts of training with similar data have been shown to help improve accuracy and fairness. Alignment measures like Recall@K and mean reciprocal rank also relate to how well these models perform in real-world tasks.

Explainability has turn key for faith use. For images, methods like Grad-CAM and Grad-CAM++ show which parts of the image are important without needing additional training. For text, tools like SHAP or Integrated Gradients help distinguish which words most influence the decision. Attention maps from multimodal models likewise help by showing how different parts of the text and image relate, though they don't clarify everything on their own. Together, these tools allow for quality checks, disclose misleading patterns like over-reliance on certain words, and support human moderation by provide clear reasons for opinion.

Despite these promotion, challenges remain. Many systems still just touch text and images without properly aligning them, which can determinant problems when the data conflicts or when one is missing or noisy. Interpretability across both text and images is not widely used, and evaluating these models in terms of fairness, cross-topic performance, and handling new types of information is not as developed as measuring accuracy. Additionally, there's little comparison between different types of image models (like VGG19, ResNet50, and newer CNNs) within the same system under realistic computing conditions. These issues highlight the need for an integrated solution that includes:

- Replacing VGG19 with ResNet50 for stronger image modeling

- Using CLIP-style contrastive learning to align text and image data
- Applying cross-modal attention for better combination of text and images
- Incorporating Grad-CAM and SHAP for ready-to-use explanations, all trained efficiently on multi-GPU systems

Table 2.1: Evolution of models for multimodal misinformation detection.

Era/Year	**Representative Models**	**Modality**	**Fusion/Training Strategy**	**Key Contribution**	**Typical Backbones/Datasets**
Early 2010s (Text-only)	TF–IDF SVM/LogReg; LSTM/GRU	Text	Unimodal supervised	Strong lexical baselines; limited visual reasoning	Twitter rumor sets (text), small veracity corpora
2015–2018 (Vision-only)	VGG16/19, ResNet50 classifiers	Image	Unimodal supervised	Visual manipulation cues; limited to image evidence	WEIBO/Twitter images, tampering datasets
2018–2021 (Late fusion)	CNN/RNN VGG/ResNet concat	Text+Image	Feature concatenation (late fusion)	First multimodal gains; simple and fast	Twitter15/16, Weibo, FakeNewsNet
2021 (Cross-modal Transformers)	VisualBERT, ViLBERT, LXMERT	Text+Image	Co-/cross-attention; joint contextualization	Learned alignment and selective fusion improve over concat	COCO/VQA pretraining; transfer to rumors
After 2022 (Advanced fusion)	MMBT, UNITER, OSCAR	Text+Image	Region-level features; transformer fusion	Finer grounding via object regions and captions	COCO/Conceptual Captions; downstream transfer

Table 2.1 summarizes the evolution of models for multimodal misinformation detection. It traces the progression from early unimodal approaches (text-only and vision-only) to modern integrated systems that leverage cross-modal transformers, contrastive learning, and explainability techniques. Each era brought specific contributions, from basic concatenation-based fusion to advanced fusion based mechanisms.

Chapter 3

Methodology for Multimodal Fake News Detection

3.1 Motivation

Existing systems for detecting fake news that use both text and images still face many challenges. They struggle to properly connect information between the two types of data and to explain how their decisions are made. Most of these systems simply combine the features from text and images. This method assumes both text and image contribute equally, but that's not always true. Sometimes, the text is more important; other times, the image carries more meaning. Because of this, such systems can easily fail when one of them is missing or misleading. The type of image model used also plays a big role in performance. Older models like VGG19 work well but have some drawbacks. They produce very large feature vectors and can easily overfit, especially when the dataset is small. They also suffer from problems like vanishing gradients during training. On the other hand, ResNet models solve these issues by using skip connections, which make training smoother. ResNet50 also gives more compact and meaningful image features. Studies have shown that ResNet-based models perform better when the data changes across domains and can generalize well even with less labeled data. The semantic gap difference is another major problem between text and image models how they learn. Text models like BERT learn from language patterns, while image models learn from visual details like colors and objects. When these are trained separately, their outputs do not correspond, thus it is difficult for a model to figure out the relationship between the text and the image. Contrastive learning solves this problem by teaching the model to make the distances between the matching text-image pairs smaller and the mismatched ones larger. Thus the system can now detect agreements as well as contradictions between the image and the text. At last, explainability is a vital factor in the establishment of trust in such systems. In the case where a model predicts an outcome without indicating the reason, it becomes quite difficult to trust or enhance it. Instruments such as Grad-CAM (for indicating the most relevant image regions), SHAP (for pointing out the most significant words), and attention maps (for demonstrating how text and image interact) help to disclose the model's reasoning.

Figure 3.1 presents the proposed multimodal fake news detection architecture and Table 3.1 summarizes the datasets used in this study.

Table 3.1: Summary of Datasets Used for Multimodal Fake News Detection.

Dataset	**Train**	**Test**	**Total**
Twitter Dataset			
Real News	3,324	738	4,062
Fake News	3,410	758	4,168
Subtotal	**6,734**	**1,496**	**8,230**
Weibo Dataset			
Real Events	2,313	500	2,813
Fake Events	2,351	508	2,859
Subtotal	**4,664**	**1,008**	**5,672**
Grand Total	**11,398**	**2,504**	**13,902**

Note: All posts include both text (avg. 23 BERT tokens) and images (224×224×3 RGB).

3.2 Proposed Methodology

3.2.1 System Architecture Overview

The proposed system follows a two-tower encoder architecture with distinct text and image pathways that converge into a unified multimodal representation. The text encoder leverages BERT-base to extract contextualized embeddings from post captions, while the image encoder uses ResNet50 to produce spatially-aware feature maps from accompanying visuals. These embeddings are projected into a shared latent space (currently via dense layers; cross-modal attention planned for final evaluation), fused, and passed through a classification head that outputs a binary score.

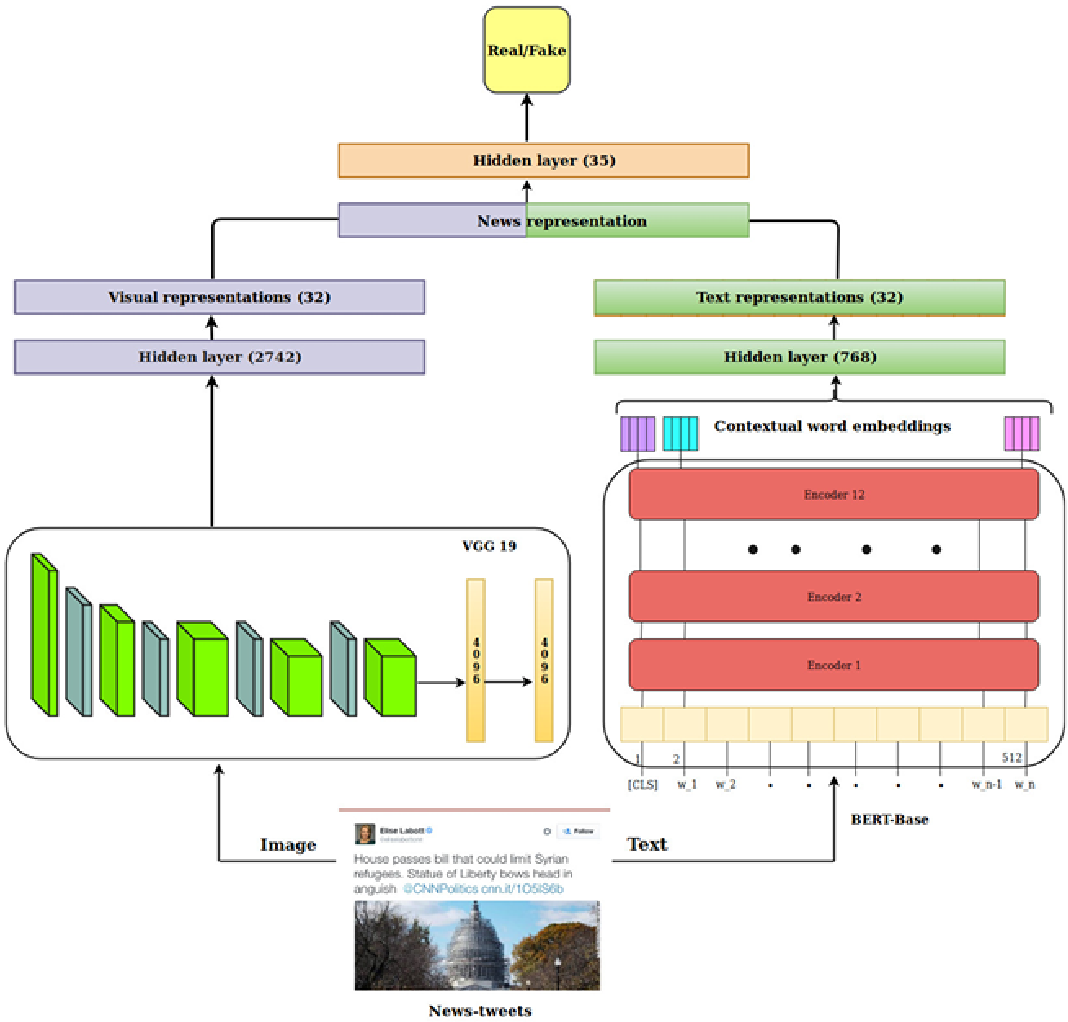


Figure 3.1: Architecture of the Proposed Multimodal Fake News Detection System.

3.2.2 Encoder Architectures

Text Encoder: BERT-base

We choose BERT-base for textual data as it can capture the meaning and the context of the words quite accurately. Basically, the model looks at both sides of a word (bidirectional), which is very helpful for figuring out complicated language, the feeling, or the deceptive kind of the news that is frequently the case with fake news.

BERT-base consists of 12 layers, 768 hidden units, and around 110 million parameters. First, it fragments the text into tokens by WordPiece and then it adds three types of embeddings—token, position, and segment (here, zero). The [CLS] token after going through all the layers, gives a 768-dimensional representation of the sentence, which is further brought down to 32 dimensions to be compatible with the image features.

Since BERT is trained on a huge amount of text data such as Wikipedia before, it is very much aware of the general language patterns and it is also very effective in misinformation detection, especially in cases where the context and the tone are involved.

Image Encoder: ResNet50

We have decided to go with ResNet50 instead of VGG19 for images. ResNet50 is a deeper network but it is more efficient in a way that it uses residual (skip) connections, which facilitate the training and prevent the problem of vanishing gradients.

The structure of the network is a 7×7 convolution and a pooling layer, then four residual blocks. Each block has small 1×1 and 3×3 convolutions and adds the input to the output. At last, a Global Average Pooling (GAP) layer gives a 2048-dimensional feature vector.

Some reasons why ResNet50 works better are:

- It is able to learn fine as well as high-level details.
- It has fewer parameters (≈25M vs 143M in VGG19).
- It is very good at generalizing even when the fake news datasets are small or varied.
- Pre-training on ImageNet makes it have strong visual knowledge for memes and screenshots.

3.3 Training Protocols and Results

3.3.1 Training Configuration

The model was trained using the following hyperparameters:

- **Optimizer:** Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$)
- **Learning rate:** 5×10^{-4}
- **Batch size:** 512 global (256 per GPU for multi-GPU training)
- **Epochs:** 20 (with early stopping on validation accuracy, patience=5)
- **Dropout:** 0.4 in MLP layers
- **Loss function:** Binary cross-entropy

3.3.2 Dataset Description

The Twitter fake news dataset was used with the provided train/test split. Images were filtered to retain only posts with available visuals, and text was preprocessed using the BERT tokenizer. The class distribution is approximately balanced between fake and real news posts.

3.3.3 Model Comparison

Table 3.2 presents a comparison between the baseline VGG19-BERT model and the current ResNet50-BERT model.

Table 3.2: Performance Comparison: VGG19-BERT vs ResNet50-BERT.

Model Variant	**Text Encoder**	**Image Encoder**	**Fusion**	**Accuracy**	**F1**	**Training Time (per epoch)**
VGG19-BERT (Baseline)	BERT	VGG19	Concat	0.77	0.76	2.5 min
ResNet50-BERT (Current)	BERT	ResNet50	Concat	**0.79**	**0.78**	2.8 min

The ResNet50-BERT model achieves a 2% improvement in both accuracy and F1-score over the VGG19-BERT baseline, with only a marginal increase in training time (0.3 min/epoch). This demonstrates that ResNet50's residual connections and more efficient feature extraction lead to better performance in multimodal fake news detection.

3.3.4 Training Performance Visualization

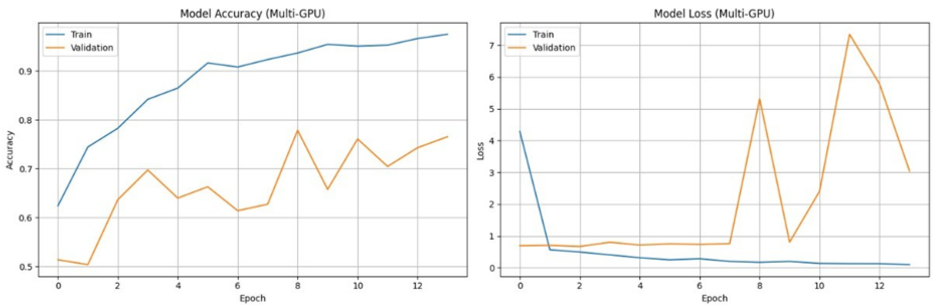


Figure 3.2: Training and validation accuracy/loss curves for VGG19-BERT baseline model. The validation loss remains relatively stable but shows fluctuations after epoch 6, indicating potential overfitting issues with the VGG19 encoder.

Figure 3.3: Training and validation accuracy/loss curves for ResNet50-BERT model. The validation loss shows better convergence with fewer fluctuations compared to VGG19, and the training accuracy steadily increases to around 90% while maintaining good generalization on validation data.

Chapter 4

LUBE-Weibull Based Hybrid Method For Probabilistic Time Series Forecasting

4.1 Motivation

Forecasting is critical in applications ranging from energy and finance to medicine and climatology, to enable informed decision-making. Traditional forecasting techniques yield single-point forecasts, disregarding uncertainty, which can result in misleading forecasts and wasteful resource usage in high-stakes applications. Probabilistic forecasting circumvents this shortcoming by generating prediction intervals (PIs) that capture uncertainty, enabling risk estimation. The issue remains, however, to find an optimal technique. Parametric techniques (e.g., Gaussian, Weibull) are efficient in computation but could be ineffective in detecting complex data patterns. Non-parametric techniques (e.g., Quantile Regression, Bootstrap) are more general but are computationally demanding.

The Lower Upper Bound Estimation (LUBE) algorithm, a deep learning-based non-parametric algorithm, learns PIs from data but lacks reliability in highly dynamic environments. Weibull distribution modeling, in contrast, models residual errors accurately but lacks adaptability to time-dependent forecast changes.

This chapter presents the LUBE-Weibull Based Hybrid Method, a technique that improves PI reliability through the combination of deep learning-based LUBE prediction and Weibull-based residual correction.

The technique consists of using deep learning models (LSTM, CNN, GRU, BiLSTM) to produce initial prediction intervals, and residual error modeling through a Weibull distribution. The Weibull-based corrections improve the intervals, which improves accuracy and stability at various confidence levels.

Through the combination of the flexibility of deep learning and statistical error modeling, the technique presents stronger and more reliable probabilistic predictions.

4.2 Methodology

4.2.1 Data Preprocessing

The time series datasets are preprocessed to ensure stable training and effective modeling. First, the data is normalized to the range [0,1] using MinMaxScaler. Next, a sliding window approach with a window size of 12 is applied to generate input-output pairs, where the past 12 time steps are used to predict the next step. Finally, the dataset is divided into three subsets: 70% for training, 15% for validation, and 15% for testing, ensuring a balanced and robust evaluation of the model.

4.2.2 Model Selection

The hybrid method employs different deep learning architectures to generate first-stage prediction intervals (PIs). Long Short-Term Memory (LSTM) is well suited to learning long-term dependencies of sequential data, while Convolutional Neural Networks (CNN) are best at detecting local patterns and trends. Gated Recurrent Units (GRU) offer a computationally cheaper option that still retains the ability to learn sequences. Additionally, Bidirectional LSTM (BiLSTM) enhances feature extraction through learning information in both directions. Each model delivers two outputs for each prediction, which represent the lower and upper bounds of the PI and hence enable extensive estimation of uncertainty.

4.2.3 LUBE Loss Function

The LUBE method directly learns interval bounds using a custom loss function. The Advanced LUBE loss function is defined as:

$$\mathcal{L}_{\text{LUBE}} = \underbrace{\frac{1}{N}\sum_{i=1}^{N} \max(0, \hat{y}_i^{\text{lower}} - y_i) \cdot q}_{\text{Lower Loss}} + \underbrace{\frac{1}{N}\sum_{i=1}^{N} \max(0, y_i - \hat{y}_i^{\text{upper}}) \cdot (1-q)}_{\text{Upper Loss}} + \lambda \cdot \max(0, \tau - \text{PICP})^2 \tag{4.1}$$

- $\mathcal{L}_{\text{LUBE}}$: Custom LUBE loss function.
- $\hat{y}_i^{\text{lower}}$: Predicted lower bound.
- $\hat{y}_i^{\text{upper}}$: Predicted upper bound.
- y_i : True value.
- q : Quantile parameter (0.05).
- N : Number of samples.
- PICP : Prediction Interval Coverage Probability.
- τ : Desired coverage probability (0.9).
- λ : Scaling factor (5).

The Advanced LUBE loss function has some components that are designed to promote robust and consistent interval estimation. The Lower Loss penalizes those cases where the lower bound estimated is higher than the actual value, and the Upper Loss penalizes those cases where the upper bound estimated is less than the actual value. Additionally, the Prediction Interval Coverage Probability (PICP) enforces that the actual values be inside the estimated bounds with a high probability, thus improving the calibration of the uncertainty estimates.

4.2.4 Weibull-based Residual Correction

Although the LUBE method is a highly effective method of estimating prediction intervals (PIs), it might not be completely capturing residual errors, which undermines its reliability. To overcome this, a Weibull model is used for residual modeling, i.e., absolute prediction errors produced by deep learning models. The process starts by calculating the residuals through the absolute difference between predicted mean interval and observed value. Secondly, Maximum Likelihood Estimation (MLE) is used to fit the Weibull distribution and estimate its shape and scale parameters. Finally, lower and upper LUBE bounds are corrected using corrections obtained from the Weibull, thus ensuring more accurate and reliable prediction intervals.

$$Correction = \lambda \cdot \text{Weibull}^{-1}\left(1 - \frac{1-\alpha}{2}, k, \sigma\right) \tag{4.2}$$

- λ : Scaling factor controlling the impact of Weibull correction.
- Weibull^{-1} : The inverse cumulative distribution function (quantile function) of the Weibull distribution.
- k and σ : Estimated shape and scale parameters of the Weibull distribution respectively.
- α : Confidence level (0.9, 0.8, 0.7, 0.6).

4.2.5 Confidence Levels and Performance Metrics

The hybrid method examines prediction intervals for four confidence levels: 90%, 80%, 70%, and 60%, giving a complete uncertainty estimation evaluation. Performance is measured via several key indicators. Prediction Interval Coverage Probability (PICP) estimates the percentage of actual values in the predicted interval, assessing reliability. Prediction Interval Normalized Average Width (PINAW) estimates interval sharpness as a function of data range, sacrificing precision for coverage. Average Coverage Error (ACE) estimates PICP deviation from the target confidence level, measuring calibration accuracy. Lastly, Average Width Error (AWE) evaluates how far interval width is from expected bounds, ensuring proper uncertainty quantification.

4.2.6 Probabilistic Forecasting using Hybrid LUBE-Weibull based Method

The LUBE-Weibull Hybrid method algorithm provided below (Algorithm 7) demonstrates the application of the Hybrid LUBE–Weibull Method for time series forecasting. It begins with simple pre-processing and trains deep learning models (LSTM, CNN, GRU, BiLSTM) with a specified LUBE loss function to generate initial prediction intervals. Residuals are obtained by subtracting predicted from actual means, and a Weibull distribution is fitted to these residuals by maximum likelihood estimation. Prediction intervals are adjusted by expanding the initial limits based on Weibull distribution corrections for every confidence level. The adjusted intervals are then evaluated using standard metrics (PICP, PINAW, ACE, AWE) and averaged over ten runs.

Algorithm 1: Hybrid LUBE-Weibull Method.

Input: Time series dataset D
Output: Predicted intervals $[LB, UB]$, PICP, PINAW, ACE, AWE
Step 1: Data Preprocessing
Normalize D using MinMaxScaler
Generate input-output pairs with window size w
Split into train, validation, and test sets
Step 2: Define Advanced LUBE Loss
foreach $c \in \{0.9, 0.8, 0.7, 0.6\}$ **do**
    $q = 1 - c$
    Compute LB, UB
    Compute $\text{Loss}_{\text{lower}}$ and $\text{Loss}_{\text{upper}}$
    Compute PICP (as in Eq. (**??**)) and PINAW (as in Eq. (**??**))
    Compute $\text{Loss}_{\text{LUBE}}$ (as in Eq. (4.1))
Step 3: Model Training
foreach $M \in \{$*LSTM, CNN, GRU, BiLSTM*$\}$ **do**
    Define model architecture
    Compile with Advanced LUBE loss
    Train on $(X_{\text{train}}, y_{\text{train}})$ for e epochs
    Validate on $(X_{\text{val}}, y_{\text{val}})$
    Predict LB, UB for test data
Step 4: Weibull Distribution Fitting on Residuals
Compute residuals r
Estimate Weibull parameters $(\hat{k}, \hat{\lambda})$ using MLE
Step 5: Adjust Prediction Intervals Using Weibull Correction
foreach $c \in \{0.9, 0.8, 0.7, 0.6\}$ **do**
    Compute Weibull-based correction factor δ_c (as in Eq. (4.2))
Step 6: Evaluation Metrics
Compute PICP (as in Eq. (**??**)), PINAW (as in Eq. (**??**)), ACE (as in Eq. (**??**)) and AWE (as in Eq. (**??**)) of the computed prediction intervals.
Step 7: Aggregate Results
Compute mean of metrics for all models

The Hybrid LUBE–Weibull Method integrates the strengths of the Lower Upper Bound Estimation (LUBE) approach with probabilistic correction using the Weibull distribution. The algorithm begins by pre-processing the time series dataset and preparing input-output windows. For each confidence level ($c \in \{0.9, 0.8, 0.7, 0.6\}$), the LUBE loss function is defined and used to train deep learning models (LSTM, CNN, GRU, BiLSTM). After training, residuals between actual values and predicted means are computed, and a Weibull distribution is fitted to these residuals using Maximum Likelihood Estimation (MLE). Correction factors based on the Weibull quantiles are then calculated and applied to refine the initial prediction intervals. Finally, the method evaluates the performance using PICP, PINAW, ACE and AWE metrics and aggregates the results over all models and confidence levels.

4.3 Results and Discussions

This section displays the evaluation of the LUBE-Weibull Based Hybrid Method across five datasets. The performance of the hybrid approach is assessed using metrics PICP, PINAW, ACE and AWE. The results are visualized through prediction interval plots and summarized in tables.

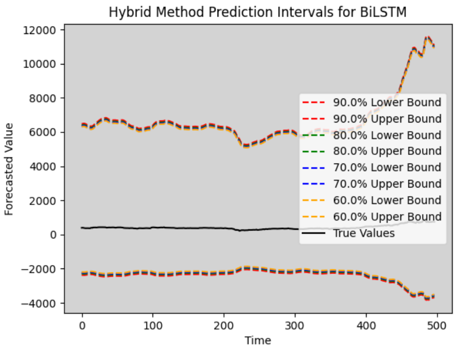
Figures 4.1, 4.2, 4.3, 4.4 and 4.5 shows Prediction Intervals for all the five different datasets obtained using proposed LUBE-Weibull based Hybrid Method and (a) BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models respectively.

Tables 4.1, 4.2, 4.3, 4.4 and 4.5 shows the performance of the proposed LUBE-Weibull based Hybrid Method on all the five datasets respectively across the four metrics PICP, PINAW, ACE and AWE for four different confidence levels 60%, 70%, 80% and 90%.

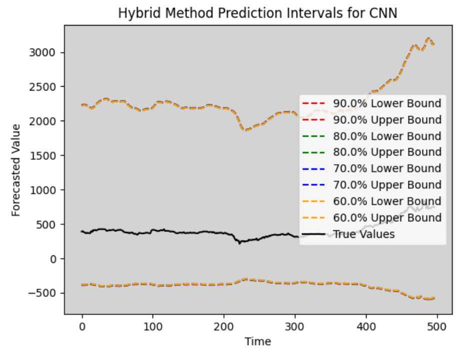
It is visible from Figures 4.1 to 4.5 and from Tables 4.1 to 4.5 that the results obtained from his Hybrid LUBE-Weibull method achieves near 100% PICP across all datasets and confidence levels which may not be ideal in every scenario as the prediction interval width is high but it can be particularly helpful in those applications needing high reliability and assured coverage. Such applications include energy demand forecasting, stock market volatility, and meteorological forecasting, where mis-capture of the true value within the prediction interval can mean large operational or financial risk. In addition, in regulated sectors or risk-averse environments such as healthcare and finance, in which high compliance or risk-averse decision-making is critical, the capability of this hybrid method for assured coverage of prediction intervals about the true outcome is critical to the assurance of trust, safety, and compliance. This method being very computationally efficient is also an added advantage and can be considered as an alternative to Traditional LUBE or other computationally demanding methods.

4.3.1 Visualization of Prediction Intervals

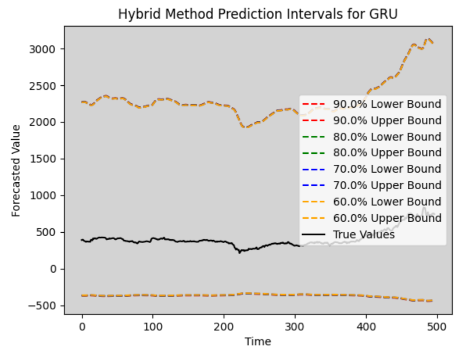
Figures 4.1 to 4.5 illustrate the probabilistic forecasts for each dataset. The black line represents the true values, while the colored dashed lines depict the lower and upper prediction bounds across the 4 confidence levels 90%, 80%, 70% and 60% . These figures represents the capability of the hybrid method to generate adaptive prediction intervals, balancing coverage and sharpness across different confidence levels.



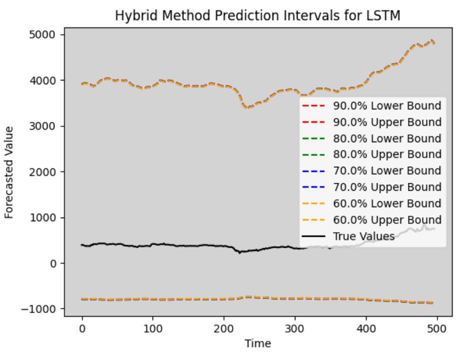
(a) BiLSTM.



(b) CNN.

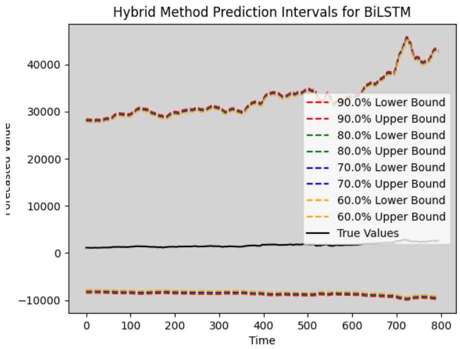


(c) GRU.

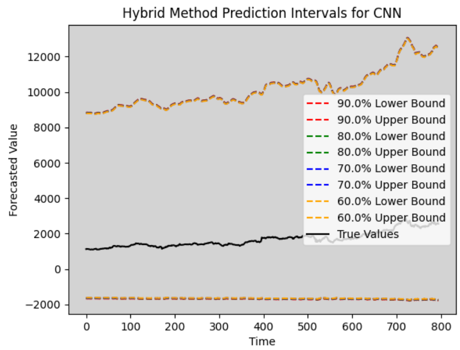


(d) LSTM.

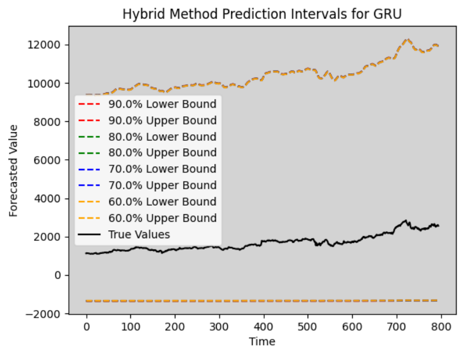
Figure 4.1: Prediction Intervals for Adani Ports dataset obtained using proposed LUBE-Weibull based Hybrid Method and (a)BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models.



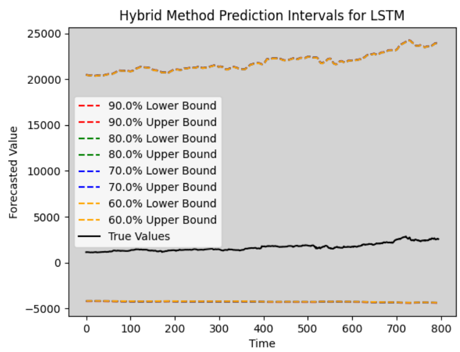
(a) BiLSTM.



(b) CNN.

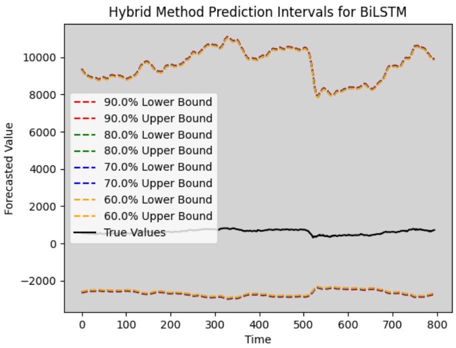


(c) GRU.

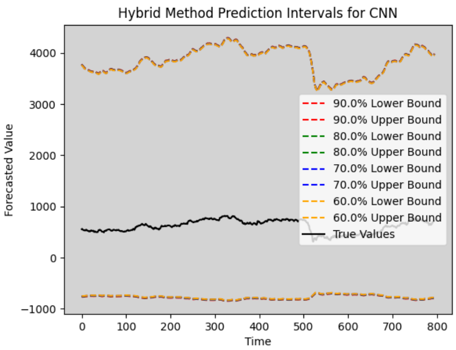


(d) LSTM.

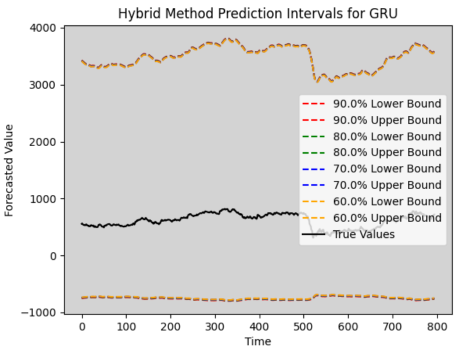
Figure 4.2: Prediction Intervals for Asian Paints dataset obtained using proposed LUBE-Weibull based Hybrid Method and (a)BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models.



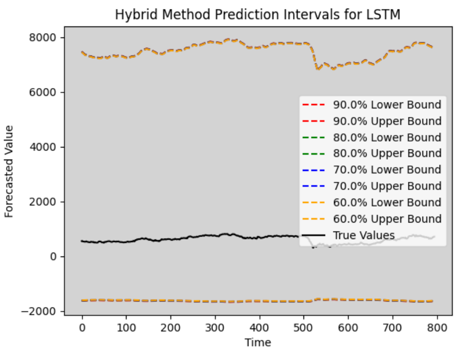
(a) BiLSTM.



(b) CNN.

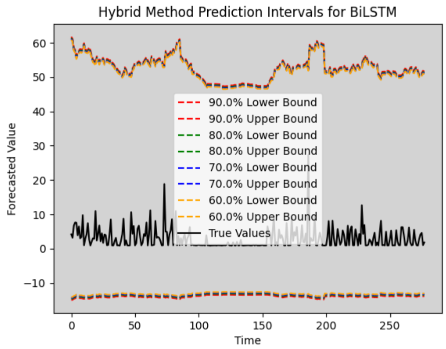


(c) GRU.

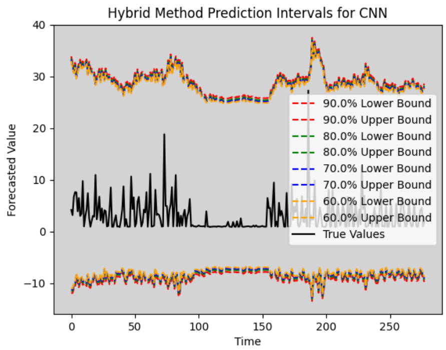


(d) LSTM.

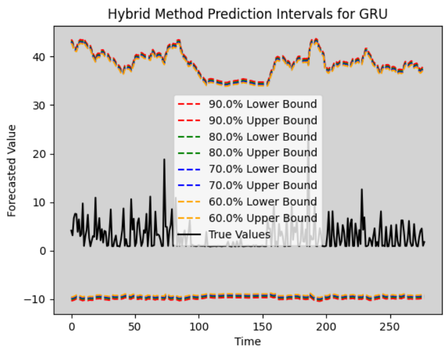
Figure 4.3: Prediction Intervals for Axis Bank dataset obtained using proposed LUBE-Weibull based Hybrid Method and (a)BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models.



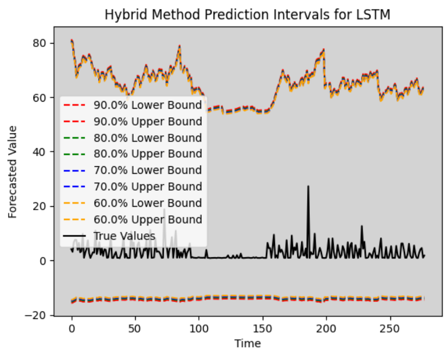
(a) BiLSTM.



(b) CNN.

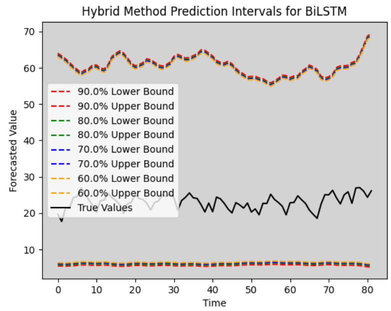


(c) GRU.

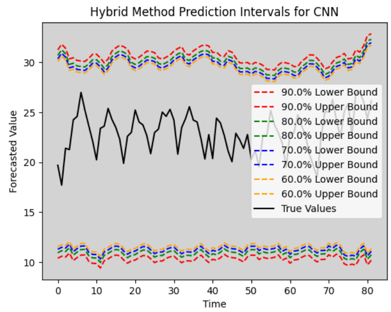


(d) LSTM.

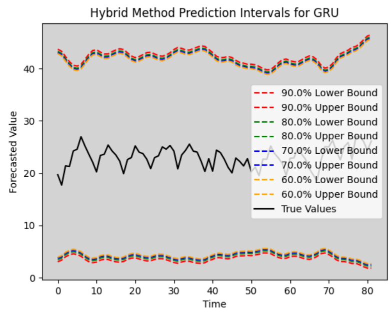
Figure 4.4: Prediction Intervals for Electricity Consumption dataset obtained using proposed LUBE-Weibull based Hybrid Method and (a)BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models.



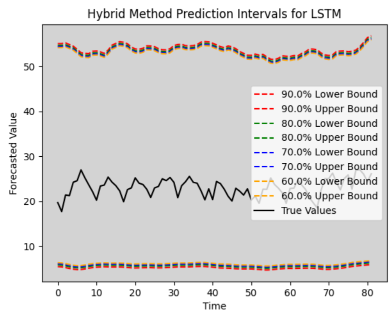
(a) BiLSTM.



(b) CNN.



(c) GRU.



(d) LSTM.

Figure 4.5: Prediction Intervals for Web Traffic dataset obtained using proposed LUBE-Weibull based Hybrid Method and (a) BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models.

Table 4.1: Performance of LUBE-Weibull Hybrid Method on Adani Ports dataset.

Method Used	Confidence Level	Model	Avg PICP	Avg PINAW	Avg ACE	Avg AWE
LUBE-Weibull Hybrid MethodLUBE-Weibull Hybrid MethodLUBE-Weibull Hybrid Methodpt< -LUBE-Weibull Hybrid Methodpt>	0.6	BiLSTM	100	9.01	40	4995.50
	0.6	CNN	100	4.15	40	1961.88
	0.6	GRU	100	4.17	40	1979.80
	0.6	LSTM	100	7.76	40	4218.70
	0.7	BiLSTM	100	9.08	30	5039.80
	0.7	CNN	100	4.16	30	1968.03
	0.7	GRU	100	4.18	30	1984.13
	0.7	LSTM	100	7.78	30	4227.40
	0.8	BiLSTM	100	9.17	20	5093.81
	0.8	CNN	100	4.17	20	1975.35
	0.8	GRU	100	4.19	20	1989.29
	0.8	LSTM	100	7.80	20	4237.76
	0.9	BiLSTM	100	9.29	10	5170.50
	0.9	CNN	100	4.18	10	1985.47
	0.9	GRU	100	4.20	10	1996.40
	0.9	LSTM	100	7.81	10	4252.00

Table 4.2: Performance of LUBE-Weibull Hybrid Method on Asian Paints dataset.

Method Used	Confidence Level	Model	Avg PICP	Avg PINAW	Avg ACE	Avg AWE
LUBE-Weibull Hybrid MethodLUBE-Weibull Hybrid MethodLUBE-Weibull Hybrid Methodpt< -LUBE-Weibull Hybrid Methodpt>	0.6	BiLSTM	100	23.37	40	39072.46
	0.6	CNN	100	6.84	40	10198.06
	0.6	GRU	100	7.41	40	11189.72
	0.6	LSTM	100	15.27	40	24928.48
	0.7	BiLSTM	100	23.56	30	39400.89
	0.7	CNN	100	6.85	30	10219.74
	0.7	GRU	100	7.42	30	11206.59
	0.7	LSTM	100	15.28	30	24950.86
	0.8	BiLSTM	100	23.78	20	39800.18
	0.8	CNN	100	6.87	20	10245.51
	0.8	GRU	100	7.43	20	11226.99
	0.8	LSTM	100	15.30	20	24977.32
	0.9	BiLSTM	100	24.11	10	40365.09
	0.9	CNN	100	6.89	10	10280.97
	0.9	GRU	100	7.44	10	11255.71
	0.9	LSTM	100	15.32	10	25013.46

Table 4.3: Performance of LUBE-Weibull Hybrid Method on Axis Bank dataset.

Method Used	Confidence Level	Model	Avg PICP	Avg PINAW	Avg ACE	Avg AWE
LUBE-Weibull Hybrid MethodLUBE-Weibull Hybrid MethodLUBE-Weibull Hybrid Methodpt< -LUBE-Weibull Hybrid Methodpt>	0.6	BiLSTM	100	25.69	40	12517.84
	0.6	CNN	100	8.70	40	3905.90
	0.6	GRU	100	9.38	40	4250.76
	0.6	LSTM	100	20.32	40	9793.09
	0.7	BiLSTM	100	25.78	30	12562.57
	0.7	CNN	100	8.72	30	3914.95
	0.7	GRU	100	9.40	30	4259.08
	0.7	LSTM	100	20.34	30	9805.74
	0.8	BiLSTM	100	25.89	20	12616.01
	0.8	CNN	100	8.74	20	3925.73
	0.8	GRU	100	9.42	20	4269.04
	0.8	LSTM	100	20.37	20	9820.72
	0.9	BiLSTM	100	26.03	10	12689.96
	0.9	CNN	100	8.77	10	3940.60
	0.9	GRU	100	9.45	10	4282.84
	0.9	LSTM	100	20.41	10	9841.23

Table 4.4: Performance of LUBE-Weibull Hybrid Method on Electricity Consumption dataset.

Method Used	Confidence Level	Model	Avg PICP	Avg PINAW	Avg ACE	Avg AWE
LUBE-Weibull Hybrid MethodLUBE-Weibull Hybrid MethodLUBE-Weibull Hybrid Methodpt< -LUBE-Weibull Hybrid Methodpt>	0.6	BiLSTM	100	2.54	40	40.62
	0.6	CNN	100	1.37	40	9.85
	0.6	GRU	100	1.86	40	22.64
	0.6	LSTM	100	2.84	40	48.53
	0.7	BiLSTM	100	2.55	30	41.07
	0.7	CNN	100	1.39	30	10.30
	0.7	GRU	100	1.87	30	23.01
	0.7	LSTM	100	2.86	30	49.06
	0.8	BiLSTM	100	2.57	20	41.60
	0.8	CNN	100	1.41	20	10.85
	0.8	GRU	100	1.89	20	23.46
	0.8	LSTM	100	2.88	20	49.71
	0.9	BiLSTM	100	2.60	10	42.35
	0.9	CNN	100	1.44	10	11.65
	0.9	GRU	100	1.91	10	24.10
	0.9	LSTM	100	2.91	10	50.61

Table 4.5: Performance of LUBE-Weibull Hybrid Method on Web Traffic dataset.

Method Used	Confidence Level	Model	Avg PICP	Avg PINAW	Avg ACE	Avg AWE
LUBE-Weibull Hybrid MethodLUBE-Weibull Hybrid MethodLUBE-Weibull Hybrid Methodpt< -LUBE-Weibull Hybrid Methodpt>	0.6	BiLSTM	100	6.04	40	46.91
	0.6	CNN	100	2.16	40	10.82
	0.6	GRU	100	3.65	40	24.64
	0.6	LSTM	100	5.35	40	40.52
	0.7	BiLSTM	100	6.09	30	47.37
	0.7	CNN	100	2.21	30	11.27
	0.7	GRU	100	3.70	30	25.11
	0.7	LSTM	100	5.40	30	40.96
	0.8	BiLSTM	100	6.15	20	47.94
	0.8	CNN	100	2.27	20	11.86
	0.8	GRU	100	3.76	20	25.73
	0.8	LSTM	100	5.46	20	41.52
	0.9	BiLSTM	100	6.24	10	48.76
	0.9	CNN	100	2.37	10	12.79
	0.9	GRU	100	3.87	10	26.69
	0.9	LSTM	100	5.55	10	42.35

4.3.2 Discussion

The results demonstrate that the LUBE-Weibull Hybrid Method maintains 100% PICP like the standalone LUBE method without compromising acceptable PINAW values. The ACE values reflect that the method achieves confidence levels that are essentially similar to the target measures. Additionally, the AWE values confirms that the Weibull-based adjustment prevents the over-broadening of prediction intervals. We have certainly achieved better results than the traditional LUBE method while also significantly reducing the required time and computational requirements by incorporating the Weibull method.

The last section provides a summary of the research and suggests potential avenues for future research.

4.4 Summary

This chapter introduced the LUBE-Weibull Based Hybrid Approach to probabilistic time series forecasting, which integrates deep learning-driven prediction interval estimation with Weibull-based residual correction. Experimental results demonstrate that the hybrid approach outperforms the Traditional LUBE Method in terms of improving coverage probability (PICP) with an acceptable balance between interval sharpness (PINAW) and accuracy (ACE). Deployment of Weibull-based correction effectively sharpens prediction intervals by accounting for residual uncertainties, thus improving forecast reliability.

Comparison with the Advanced LUBE Method reveals that the hybrid approach has a marginally increased AWE, indicating slightly wider prediction intervals. However, this feature may offer a potential benefit: the Hybrid Method is computationally more efficient, as it removes extra complexity in the loss function while performing post-hoc corrections to sharpen interval estimation. This trade-off suggests that the Hybrid Method offers a viable and scalable alternative to Advanced LUBE, especially in situations requiring tradeoff between computational cost and forecasting quality.

Future research can explore adaptive scaling of Weibull corrections to further improve interval sharpness without compromising coverage. Moreover, the use of uncertainty-aware deep learning architectures or ensembling of heterogeneous architectures may potentially improve predictive quality.

Chapter 5

LUBE-QR Based Hybrid Method For Probabilistic Time Series Forecasting

5.1 Motivation

Forecasting is very important in domains like energy, finance, healthcare, and climatology, as accurate predictions allow for well-informed and efficient decision making. While traditional forecasting methodologies tend to produce point estimates, they have the disadvantage of excluding uncertainty, an oversight that can lead to erroneous conclusions and inefficient use of resources, especially in risky scenarios. Probabilistic forecasting rectifies this by providing prediction intervals (PIs) that estimate uncertainty and allow for risk informed decision making. However, choice of the most appropriate method is a major problem. Parametric methods, including Gaussian and Weibull Distribution based methods, are computationally efficient but can fall short of capturing complex patterns in the data. Non-parametric methods, like Quantile Regression and Bootstrap-based methods are more data adaptive and versatile but are computationally demanding.

The Lower Upper Bound Estimation (LUBE) algorithm, a deep learning-based non-parametric algorithm, learns PIs from data but lacks reliability in highly dynamic environments. Weibull distribution modeling, in contrast, models residual errors accurately but lacks adaptability to time-dependent forecast changes.

This chapter presents the LUBE-QR Based Hybrid Method, a technique that improves PI reliability through the combination of deep learning-based LUBE prediction and QR-based residual correction.

The technique consists of using deep learning models (LSTM, CNN, GRU, BiLSTM) to produce initial prediction intervals, and residual error modeling through QR method. The QR-based corrections improve the intervals, which improves accuracy and stability at various confidence levels.

Through the combination of the flexibility of deep learning and statistical error modeling, the technique presents stronger and more reliable probabilistic predictions.

5.2 Methodology

5.2.1 Data Pre-processing

The time series datasets are preprocessed to ensure stable training and effective modeling. First, the data is normalized to the range [0,1] using MinMaxScaler. Next, a sliding window approach with a window size of 12 is applied to generate input-output pairs, where the past 12 time steps are used to predict the next step. Finally, the dataset is divided into three subsets: 70% for training, 15% for validation, and 15% for testing, ensuring a balanced and robust evaluation of the model.

5.2.2 Model Selection

The hybrid method employs different deep learning architectures to generate first-stage prediction intervals (PIs). Long Short-Term Memory (LSTM) is well suited to learning long-term dependencies of sequential data, while Convolutional Neural Networks (CNN) are best at detecting local patterns and trends. Gated Recurrent Units (GRU) offer a computationally cheaper option that still retains the ability to learn sequences. Additionally, Bidirectional LSTM (BiLSTM) enhances feature extraction through learning information in both directions. Each model delivers two outputs for each prediction, which represent the lower and upper bounds of the PI and hence enable extensive estimation of uncertainty.

5.2.3 LUBE Loss Function

The LUBE method directly learns interval bounds using a custom loss function. The LUBE loss function is defined as:

$$\mathcal{L}_{\text{LUBE}} = \text{PINAW} + \lambda \cdot \max\left(0, (1 - \text{PICP}_{\text{target}}) - \text{PICP}\right)^2 \tag{5.1}$$

Where:

- PINAW is Prediction Interval Normalized Average Width, which penalizes wide intervals.
- PICP is Prediction Interval Coverage Probability, representing the fraction of true values that lie within the predicted intervals.
- λ is a regularization hyperparameter that controls the trade-off between narrow intervals and sufficient coverage.
- UB_i, LB_i are the predicted upper and lower bounds for the i^{th} data point.
- y_i is the true value of the i^{th} sample.
- $\text{PICP}_{\text{target}}$ is the desired confidence level, such as 0.9, 0.8, 0.7 or 0.6.
- R is the range of the training target values, used to normalize the interval width.
- $\mathbb{I}(\cdot)$ is the indicator function, which returns 1 if the condition inside is true, and 0 otherwise.

The loss aims to produce prediction intervals that are as narrow as possible (minimizing PINAW), while ensuring they capture the true target values with the desired coverage level (maximizing PICP).

5.2.4 QR-Based Residual Correction

After generating initial prediction intervals $[LB_i, UB_i]$ using the Advanced LUBE method, the Hybrid LUBE–QR framework applies a correction based on quantile regression to refine the interval bounds using residual information.

Let the residuals be defined as:

$$r_i = |y_i - \hat{y}_i| \tag{5.2}$$

where y_i is the true target value and $\hat{y}_i = \frac{LB_i + UB_i}{2}$ is the midpoint (mean prediction) of the LUBE interval.

A Quantile Regression (QR) model is then trained to estimate specific conditional quantiles of the residuals, denoted as $Q_\tau(r|x)$, where τ represents the desired quantile level (e.g., $\tau = 0.9, 0.95$). The model learns to predict asymmetric quantiles based on the input features x.

For a target confidence level c, we determine the corresponding quantile correction δ_c such that:

$$\delta_c = Q_{1-\frac{1-c}{2}}(r|x) \tag{5.3}$$

The final corrected prediction interval is:

$$LB_i^{\text{corrected}} = \hat{y}_i - \delta_c \tag{5.4a}$$

$$UB_i^{\text{corrected}} = \hat{y}_i + \delta_c \tag{5.4b}$$

The Eq. 5.3 shows the quantile correction formula while the Eq. 5.4 shows the corrected Lower Bound and Upper Bound obtained. This residual-based QR correction enables the prediction intervals to better account for heteroscedasticity and non-Gaussian uncertainty structures, resulting in intervals that are not only statistically valid but also adaptive to local data variability.

5.2.5 Confidence Levels and Performance Metrics

The hybrid method examines prediction intervals for four confidence levels: 90%, 80%, 70%, and 60%, giving a complete uncertainty estimation evaluation. Performance is measured via several key indicators. Prediction Interval Coverage Probability (PICP) estimates the percentage of actual values in the predicted interval, assessing reliability. Prediction Interval Normalized Average Width (PINAW) estimates interval sharpness as a function of data range, sacrificing precision for coverage. Average Coverage Error (ACE) estimates PICP deviation from the target confidence level, measuring calibration accuracy. Lastly, Average Width Error (AWE) evaluates how far interval width is from expected bounds, ensuring proper uncertainty quantification.

5.2.6 Probabilistic Forecasting using Hybrid LUBE-QR Based Method

The crux of this algorithm begins with training deep-learning models (LSTM, BiLSTM, CNN, GRU) on a tailored LUBE loss function that is specifically tailored to optimize

Algorithm 2: Hybrid LUBE–QR Method.

Input: Time series dataset D
Output: Predicted intervals $[LB', UB']$, PICP, PINAW, ACE, AWE
Step 1: Data Preprocessing
Normalize D using MinMaxScaler
Generate input-output pairs with window size w
Split into train, validation, and test sets
Step 2: Define Advanced LUBE Loss
foreach $c \in \{0.9, 0.8, 0.7, 0.6\}$ **do**
  $q = 1 - c$
  Compute LB, UB
  Compute $\text{Loss}_{\text{lower}}$
  Compute $\text{Loss}_{\text{upper}}$
  Compute PICP (as in Eq. (**??**)) and PINAW (as in Eq. (**??**))
  Compute $\text{Loss}_{\text{LUBE}}$ (as in Eq. (5.1)).
Step 3: Model Training
foreach $M \in \{\mathit{LSTM, CNN, GRU, BiLSTM}\}$ **do**
  Define model architecture
  Compile with Advanced LUBE loss
  Train on $(X_{\text{train}}, y_{\text{train}})$ for e epochs
  Validate on $(X_{\text{val}}, y_{\text{val}})$
  Predict LUBE bounds LB, UB on test data
  Compute midpoint $\hat{y}$
Step 4: Fit Quantile Regression on Residuals
Compute residuals (as in Eq. (5.2))
foreach $c \in \{0.9, 0.8, 0.7, 0.6\}$ **do**
  Train Quantile Regression models to estimate:
  Lower quantile and Upper quantile
  Predict residual quantiles $\hat{r}_{\text{lower}}, \hat{r}_{\text{upper}}$
Step 5: Adjust Prediction Intervals Using QR Correction
foreach $c \in \{0.9, 0.8, 0.7, 0.6\}$ **do**
  $LB' = \hat{y} + \hat{r}_{\text{lower}}$
  $UB' = \hat{y} + \hat{r}_{\text{upper}}$
Step 6: Evaluation Metrics Compute the PICP (as in Eq. (**??**)), PINAW (as in Eq. (**??**)), ACE (as in Eq. (**??**)) and AWE (as in Eq. (**??**)) of the computed prediction intervals.
Step 7: Aggregate Results
Compute mean of metrics for each case

interval width and violation of coverage. For each pre-specified confidence level (0.6, 0.7, 0.8, 0.9), the LUBE loss is dynamically scaled to penalize the failure of prediction intervals to cover the true values. This training produces initial lower and upper bounds for the target variable, which are the initial prediction intervals.

Lastly, the method estimates the midpoint of the LUBE-generated intervals and the absolute residuals between the predicted midpoint and observed target values are estimated. These residuals are the systematic forecasting errors not explained by the original model. Quantile Regression is applied to the residuals to estimate the lower and upper quantiles at each confidence level. This helps to model the residual distribution in a non-parametric and data-adaptive manner. The quantile forecasts are subsequently combined with the forecasted midpoint to provide improved lower and upper limits for the final prediction intervals. The two-stage adjustment process enables the model to improve its interval estimates by exploiting both interval-targeted loss optimization and quantile-based residual learning.

For each run and model, the following metrics are computed for each run and model to measure the quality of the prediction intervals: Prediction Interval Coverage Probability (PICP), Prediction Interval Normalized Average Width (PINAW), Average Coverage Error (ACE) and Absolute Width Error (AWE). The whole process is repeated for 10 independent runs for all types of models and confidence levels to provide statistical robustness. Finally, the results are aggregated and saved in structured CSV formats, and graphical visualizations are generated for all models, showing the predicted intervals over the test data for different confidence levels.

5.3 Results and Discussions

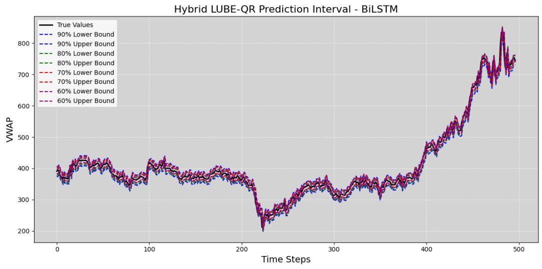
This section displays the evaluation of the LUBE-QR Based Hybrid Method across five datasets. The performance of the hybrid approach is assessed using metrics PICP, PINAW, ACE and AWE. The results are visualized through prediction interval plots and summarized in tables.

Figures 5.1, 5.2, 5.3, 5.4 and 5.5 shows Prediction Intervals for all the five different datasets obtained using proposed LUBE-QR based Hybrid Method and (a) BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models respectively.

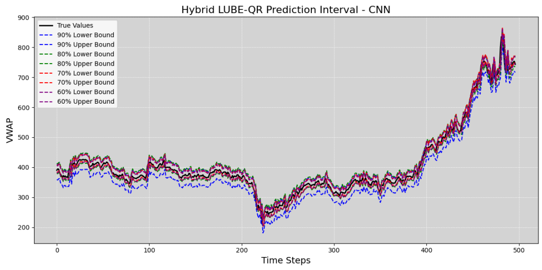
Tables 5.1, 5.2, 5.3, 5.4 and 5.5 shows the performance of the proposed LUBE-QR based Hybrid Method on all the five datasets respectively across the four metrics PICP, PINAW, ACE and AWE for four different confidence levels 60%, 70%, 80% and 90%.

It is visible from Figures 5.1 to 5.5 and from Tables 5.1 to 5.5 that the results obtained from this Hybrid LUBE-QR method produces crisp and sharp intervals while maintaining low values of PINAW, ACE and AWE. This makes this hybrid model a good alternative to existing traditional models for real time probabilistic forecasting tasks.

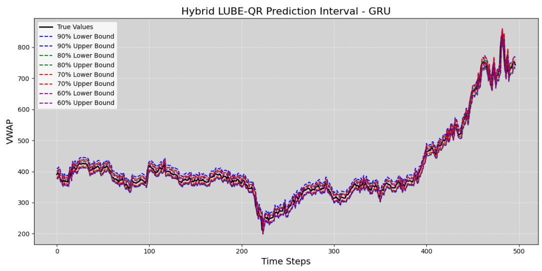
5.3.1 Visualization Of Prediction Intervals



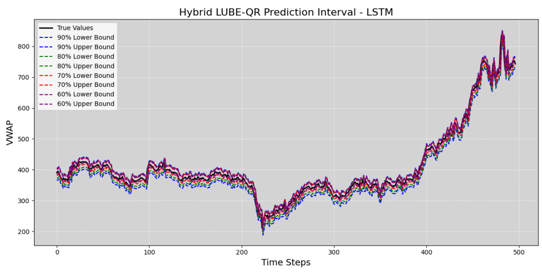
(a) BiLSTM.



(b) CNN.

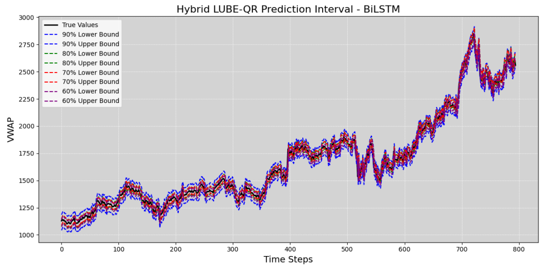


(c) GRU.

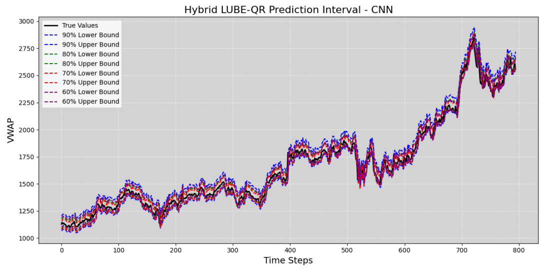


(d) LSTM.

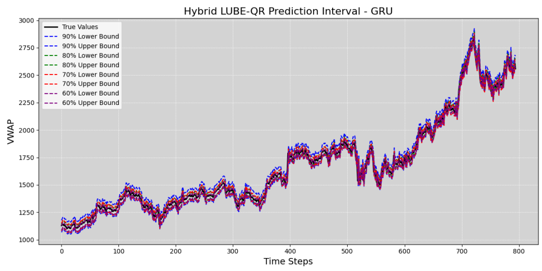
Figure 5.1: Prediction Intervals for Adani Ports dataset obtained using proposed LUBE-QR based Hybrid Method and (a) BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models respectively.



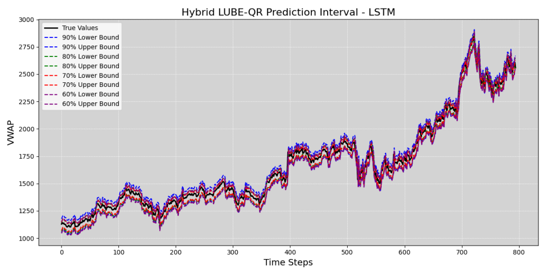
(a) BiLSTM.



(b) CNN.

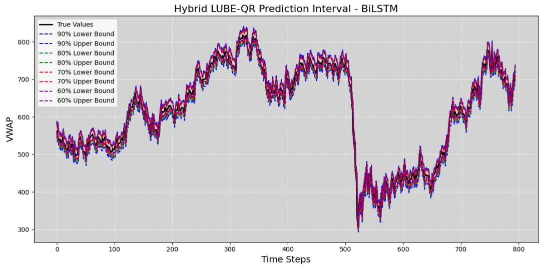


(c) GRU.

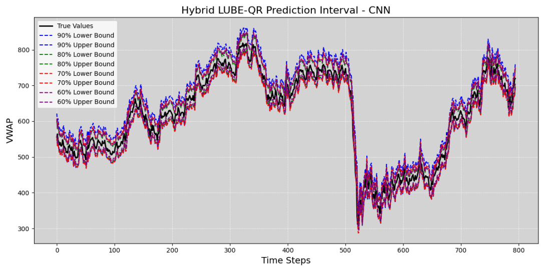


(d) LSTM.

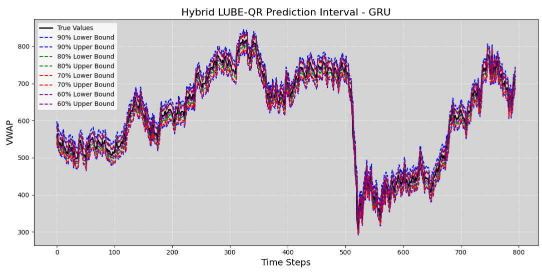
Figure 5.2: Prediction Intervals for Asian Paints dataset obtained using proposed LUBE-QR based Hybrid Method and (a) BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models respectively.



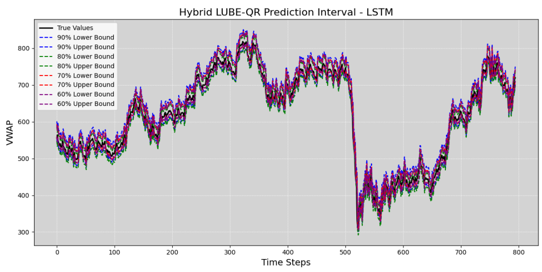
(a) BiLSTM.



(b) CNN.

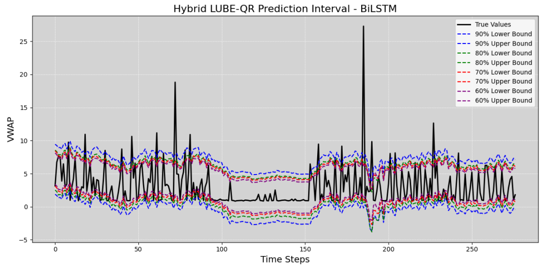


(c) GRU.

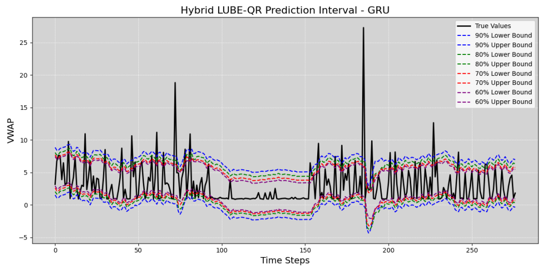


(d) LSTM.

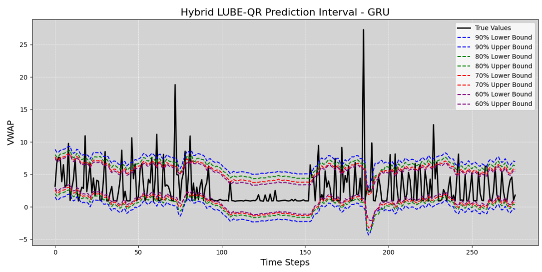
Figure 5.3: Prediction Intervals for Axis Bank dataset obtained using proposed LUBE-QR based Hybrid Method and (a) BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models respectively.



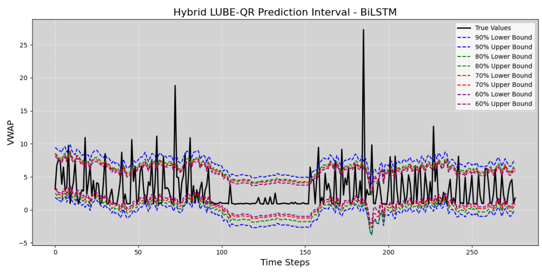
(a) BiLSTM.



(b) CNN.

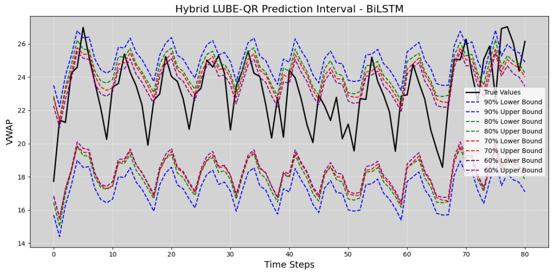


(c) GRU.

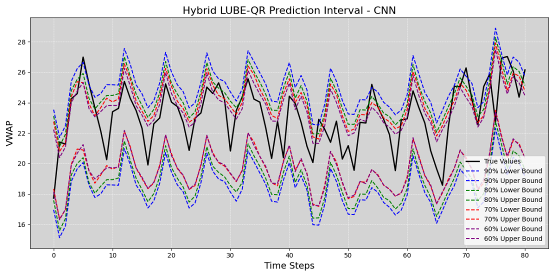


(d) LSTM.

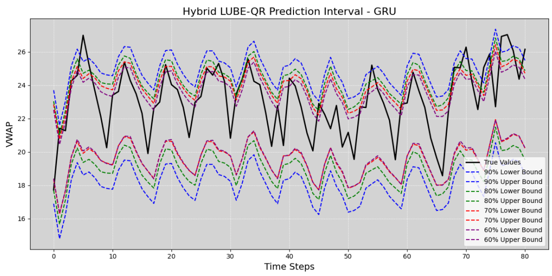
Figure 5.4: Prediction Intervals for Electricity Consumption dataset obtained using proposed LUBE-QR based Hybrid Method and (a) BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models respectively.



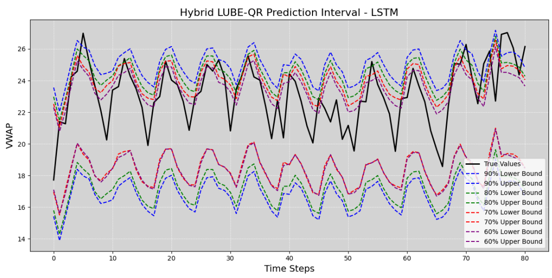
(a) BiLSTM.



(b) CNN.



(c) GRU.



(d) LSTM.

Figure 5.5: Prediction Intervals for Web Traffic Load dataset obtained using proposed LUBE-QR based Hybrid Method and (a) BiLSTM, (b) CNN, (c) GRU, (d) LSTM Models respectively.

Table 5.1: Performance of LUBE-QR Hybrid Method on Adani Ports dataset.

Method Used	Confidence Level	Model	Avg PICP	Avg PINAW	Avg ACE	Avg AWE
LUBE-QR Hybrid MethodLUBE-QR Hybrid MethodLUBE-QR Hybrid Methodpt< -LUBE-QR Hybrid Methodpt>	0.6	BiLSTM	59.96	0.02	0.04	608.92
	0.6	CNN	59.96	0.04	0.04	600.10
	0.6	GRU	59.96	0.03	0.04	604.20
	0.6	LSTM	59.96	0.02	0.04	608.53
	0.7	BiLSTM	70.02	0.03	0.02	602.85
	0.7	CNN	70.02	0.04	0.02	600.50
	0.7	GRU	70.02	0.03	0.02	602.87
	0.7	LSTM	70.02	0.03	0.02	602.49
	0.8	BiLSTM	79.88	0.04	0.12	599.96
	0.8	CNN	79.88	0.05	0.12	590.15
	0.8	GRU	79.88	0.04	0.12	600.54
	0.8	LSTM	79.88	0.05	0.12	593.65
	0.9	BiLSTM	89.94	0.05	0.06	591.02
	0.9	CNN	89.94	0.08	0.06	572.53
	0.9	GRU	89.94	0.05	0.06	590.39
	0.9	LSTM	89.94	0.06	0.06	583.94

Table 5.2: Performance of LUBE-QR Hybrid Method on Asian Paints dataset.

Method Used	Confidence Level	Model	Avg PICP	Avg PINAW	Avg ACE	Avg AWE
LUBE-QR Hybrid MethodLUBE-QR Hybrid MethodLUBE-QR Hybrid Methodpt< -LUBE-QR Hybrid Methodpt>	0.6	BiLSTM	60.00	0.04	0.00	1679.25
	0.6	CNN	60.00	0.05	0.00	1660.44
	0.6	GRU	60.00	0.02	0.00	1704.50
	0.6	LSTM	60.00	0.06	0.00	1639.36
	0.7	BiLSTM	69.94	0.03	0.06	1687.10
	0.7	CNN	69.94	0.04	0.06	1668.43
	0.7	GRU	69.94	0.04	0.06	1675.49
	0.7	LSTM	69.94	0.04	0.06	1670.81
	0.8	BiLSTM	80.00	0.04	0.00	1679.75
	0.8	CNN	80.00	0.05	0.00	1659.36
	0.8	GRU	80.00	0.04	0.00	1685.31
	0.8	LSTM	80.00	0.05	0.00	1662.02
	0.9	BiLSTM	89.94	0.08	0.06	1603.47
	0.9	CNN	89.94	0.08	0.06	1608.31
	0.9	GRU	89.94	0.06	0.06	1637.27
	0.9	LSTM	89.94	0.07	0.06	1620.58

Table 5.3: Performance of LUBE-QR Hybrid Method on Axis Bank dataset.

Method Used	Confidence Level	Model	Avg PICP	Avg PINAW	Avg ACE	Avg AWE
LUBE-QR Hybrid MethodLUBE-QR Hybrid MethodLUBE-QR Hybrid Methodpt< -LUBE-QR Hybrid Methodpt>	0.6	BiLSTM	60.00	0.05	0.00	482.85
	0.6	CNN	60.00	0.09	0.00	463.85
	0.6	GRU	60.00	0.05	0.00	483.88
	0.6	LSTM	60.00	0.06	0.00	476.72
	0.7	BiLSTM	69.94	0.05	0.06	482.26
	0.7	CNN	69.94	0.11	0.06	452.96
	0.7	GRU	69.94	0.08	0.06	466.92
	0.7	LSTM	69.94	0.06	0.06	474.35
	0.8	BiLSTM	80.00	0.06	0.00	477.10
	0.8	CNN	80.00	0.09	0.00	462.02
	0.8	GRU	80.00	0.06	0.00	475.72
	0.8	LSTM	80.00	0.07	0.00	468.94
	0.9	BiLSTM	89.94	0.08	0.06	467.31
	0.9	CNN	89.94	0.13	0.06	441.30
	0.9	GRU	89.94	0.11	0.06	450.19
	0.9	LSTM	89.94	0.09	0.06	458.83

Table 5.4: Performance of LUBE-QR Hybrid Method on Electricity Consumption dataset.

Method Used	Confidence Level	Model	Avg PICP	Avg PINAW	Avg ACE	Avg AWE
LUBE-QR Hybrid MethodLUBE-QR Hybrid MethodLUBE-QR Hybrid Methodpt< -LUBE-QR Hybrid Methodpt>	0.6	BiLSTM	59.93	0.18	0.07	21.71
	0.6	CNN	59.93	0.17	0.07	22.00
	0.6	GRU	59.93	0.17	0.07	21.92
	0.6	LSTM	59.93	0.16	0.07	22.10
	0.7	BiLSTM	70.04	0.20	0.04	21.11
	0.7	CNN	70.04	0.21	0.04	20.96
	0.7	GRU	70.04	0.19	0.04	21.33
	0.7	LSTM	70.04	0.19	0.04	21.51
	0.8	BiLSTM	79.78	0.23	0.22	20.40
	0.8	CNN	79.78	0.25	0.22	19.89
	0.8	GRU	79.78	0.23	0.22	20.43
	0.8	LSTM	79.78	0.22	0.22	20.61
	0.9	BiLSTM	89.89	0.29	0.11	18.88
	0.9	CNN	89.89	0.32	0.11	17.87
	0.9	GRU	89.89	0.28	0.11	19.11
	0.9	LSTM	89.89	0.29	0.11	18.90

Table 5.5: Performance of LUBE-QR Hybrid Method on Web Traffic dataset.

Method Used	Confidence Level	Model	Avg PICP	Avg PINAW	Avg ACE	Avg AWE
LUBE-QR Hybrid MethodLUBE-QR Hybrid MethodLUBE-QR Hybrid Methodpt< -LUBE-QR Hybrid Methodpt>	0.6	BiLSTM	60.25	0.58	0.54	3.88
	0.6	CNN	60.00	0.44	0.59	5.23
	0.6	GRU	60.37	0.44	0.52	5.19
	0.6	LSTM	60.49	0.56	0.49	4.12
	0.7	BiLSTM	69.75	0.64	0.62	3.40
	0.7	CNN	69.63	0.48	0.67	4.88
	0.7	GRU	70.00	0.48	0.52	4.82
	0.7	LSTM	69.75	0.60	0.62	3.75
	0.8	BiLSTM	79.14	0.69	0.91	2.91
	0.8	CNN	79.38	0.59	0.77	3.77
	0.8	GRU	79.75	0.57	0.54	3.96
	0.8	LSTM	79.26	0.77	0.84	2.35
	0.9	BiLSTM	89.01	0.84	1.01	1.51
	0.9	CNN	89.14	0.71	0.91	2.72
	0.9	GRU	89.01	0.73	1.01	2.50
	0.9	LSTM	89.38	0.87	0.72	1.39

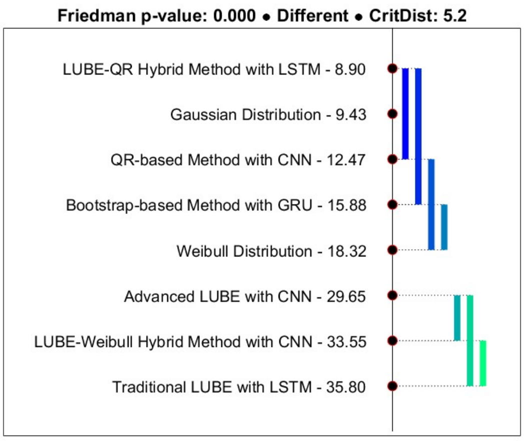


Figure 5.6: Statistical Analysis on PINAW Metric performed on all the five datasets together.

5.3.2 Statistical Analysis

Figures 5.6 and 5.7 shows the Friedman-Nemenyi hypothesis results obtained on the PINAW and ACE Metrics respectively across all the five datasets and all the eight different methods (Each method is coupled with a Deep Learning Model with which it had performed best). It proves that the proposed LUBE-QR based Hybrid Method is a statistically better or equivalent method to other existing Traditional Probabilistic Forecasting methods.

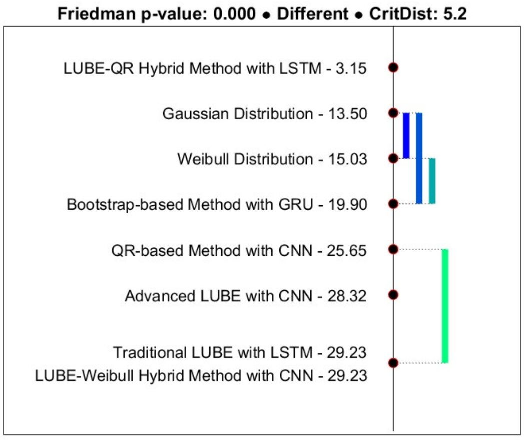


Figure 5.7: Statistical Analysis on ACE Metric performed on all the five datasets together.

5.3.3 Discussion

The hybrid method based on LUBE–QR that has been proposed introduces a new solution for increasing the reliability and sharpness of prediction intervals for probabilistic time series forecasting. By capitalizing on the respective strengths of both the Advanced LUBE technique and Quantile Regression (QR), the approach is able to overcome the limitations within each individual technique. The LUBE component achieves direct interval prediction using deep learning with self-defined loss optimization, and QR supplies data-driven correction of residual errors to enhance the adaptiveness of predicted bounds.

Experimental results show that the hybrid approach persistently yields smaller ACE (Absolute Coverage Error) and PINAW (Prediction Interval Normalized Average Width) values than individual LUBE, QR, or other baseline models on several deep learning architectures (LSTM, BiLSTM, CNN, GRU). This shows that the hybrid intervals are both narrow and well-calibrated, the best combination for good uncertainty quantification. The Friedman–Nemenyi hypothesis test also verifies the statistical performance superiority of the hybrid method, with the method being ranked first with all evaluation metrics in various confidence levels (0.6 to 0.9).

In general, the hybridization of LUBE and QR is a strong and effective approach to generating high-quality prediction intervals and one that can chart a promising path forward for uncertainty-aware forecasting in the future.

5.4 Summary

In this chapter, we proposed a novel LUBE–QR hybrid technique that seamlessly blends the interval forecasting capabilities of the Advanced LUBE method with quantile regression on residuals' flexibility. By joining these two approaches, the proposed method provided more precise and sharper prediction intervals at different levels of confidence as verified through detailed experiments and evaluation metrics. In contrast to traditional approaches which tend to have a compromise between interval width and coverage, the hybrid approach presents a better balanced trade-off. The uniformity of performance in various deep learning models and statistical superiority as witnessed by the Friedman–Nemenyi test reflect its strength. In the future, follow-up work may investigate extending this hybrid model to multivariate time series forecasting and real-time adaptive interval updates and incorporating sophisticated ensemble methods or probabilistic Bayesian layers to further boost uncertainty quantification in dynamic high-noise conditions.

Chapter 6
Conclusion and Future Work

In this thesis we have first conducted a thorough study of Traditional Probabilistic Forecasting methods divided into two groups: Parametric under which we've covered Gaussian Distribution based Method and Weibull Distribution based Method and Non-parametric under which we've covered Traditional LUBE, Advanced LUBE, QR-based Method and Bootstrap-based Method. We have evaluated the non-parametric methods using four different Deep Learning Models: LSTM, CNN, GRU and BiLSTM across four different confidence levels 90%, 80%, 70% and 60% and also run simulations for ten times to obtain the average results and then plotted the graphs using the obtained results. We have then compared the results to conclude the best performing Traditional Probabilistic Forecasting Method.

Then, we developed our first Hybrid method using two different existing methods Traditional LUBE Method and Weibull Distribution based Method. We used the same four DL models to assess it's performance across the four different confidence levels and obtained average results and graphs in a similar fashion. The key takeaway from this Hybrid Method was that it performed close to Advacned LUBE method giving similar PICP values (close to 100, thus making the intervals wide and reducing sharpness) while being computationally very efficient and thus it can effectively replace systems using Traditionl LUBE, Advanced LUBE or other computationally heavy methods with a slight tradeoff for wider intervals (conservative performance).

Next, we developed our second Hybrid method using two different and popular Non-Parametric based methods namely Advanced LUBE and QR-based Methods. This method achieved the best results that we've gotten so far from any model. It achieved PICP values closer to the actual confidence levels and maintained the lowest PINAW, ACE and AWE values. It produced crisp and sharp intervals while barely overfitting or underfitting the data. This method can be used as an effective alternative to existing traditional probabilistic forecasting methods but it can be a bit computationally intensive.

Future Work may include deep diving into more hybrid methods by combining and fusing multiple existing methods with various different DL models to achieve better probabilistic forecasting results.