Studies On Multi - Modal Fake News Detection

Research Project

Supriya Saha
(122CS0101)

Supervision of
Prof. Shyamapada Mukherjee

Department of Computer Science and Engineering
NIT Rourkela-769008, India

October 21, 2025

Outline

Introduction

- Misinformation spreads quickly on social media using both text and images which creates a multimodal detection challenge.
- Traditional models treat text and images separately or fuse them weakly, leading to poor semantic alignment and limited interpretability.
- SpotFake uses BERT (text) + VGG19 (image) with simple fusion but this limits its ability to handle complex, real-world misinformation.
- Recent advances improve this by:
  - Contrastive Learning (CLIP-style): Aligns text–image embeddings for stronger cross-modal understanding.
  - Cross-Modal Attention: Focuses on the most informative tokens or regions.
  - Explainability (Grad-CAM, SHAP): Displays decision reasoning for transparency.

Literature Review

Table 1: Recent Literature on Multi - Modal Fake News Detection.

Reference	Model	
Abbas Khosravi et al. (2011), IEEE Transactions on Neural Networks [?]	Neural Networks	
Cameron Cornell et al. (2024), Int. Journal of Forecasting [?]	Neural Networks	Pro
Yan Xu et al. (2024), Computers and Industrial Engineering [?]	Quantile Regression	
Sourav Kumar Purohit and S. Panigrahi (2024), Information Sciences [?]	Hybrid (Statistical + DL)	Pro
Härdle et al. (2003), Int. Statistical Review [?]	Statistical	

Literature Survey(Cont.)

- According to literature review these were the most frequently used probabilistic forecasting methods:

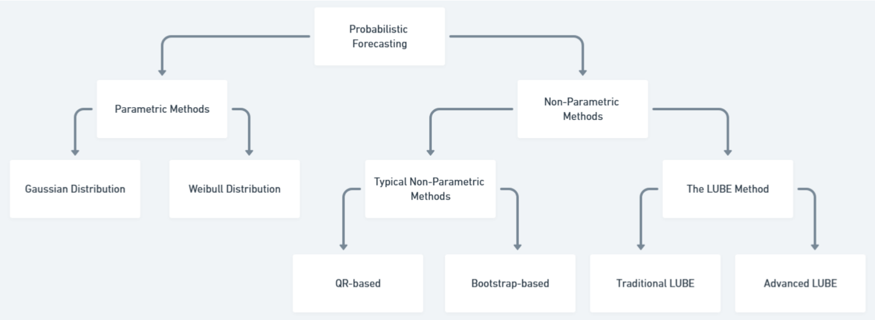

Figure 1: Classification of Traditional Probabilistic Forecasting Methods.

Key Insights from Recent Literature

- Handling uncertainty through interval predictions has become a standard for time series forecasting tasks.
- Combining statistical and deep learning techniques enhances accuracy and flexibility in forecasting.
- Metrics like PICP (Prediction Interval Coverage Probability), PINAW (Prediction Interval Normalized Average Width), ACE (Absolute Coverage Error) and AWE (Average Width Error) are widely adopted for comprehensive performance assessment.

Key Insights from Recent Literature (Cont.)

- **Research Gap:** No comprehensive studies comparing the parametric and non-parametric methods and their performance on a variety of datasets under different confidence levels are conducted. Also, there is a lack of hybrid methods.
- **Contribution:** Examined the performance of six traditional probabilistic forecasting methods across a variety of datasets using different evaluation metrics to assess their performance. Also developed two hybrid methods that out-performs the traditional methods.

Objectives

- Develop a BERT + ResNet50 multimodal model (compare with VGG19).
- Apply CLIP-style contrastive pre-training for text–image alignment.
- Introduce cross-modal attention for selective, context-aware fusion.
- Integrate explainability tools — Grad-CAM, SHAP and attention heatmaps for interpretable predictions.

Dataset Overview

Table 2: Summary of Datasets Used.

Dataset Name	Length of Dataset
Nifty50 Dataset (2000–2021)	
Adani Ports	3322
Asian Paints	5306
Axis Bank	5306
Web Traffic	550
Electricity Consumption	1858

Evaluation Metrics

- **PICP (Prediction Interval Coverage Probability):** Proportion of true values within prediction intervals.
- **PINAW (Prediction Interval Normalized Average Width):** Width of intervals (normalized).
- **ACE (Absolute Coverage Error):** Absolute difference between the actual coverage (PICP) and the desired confidence level.
- **AWE (Average Width Error:)** Average deviation of the widths of the prediction intervals from an expected or desired width.

Methodology

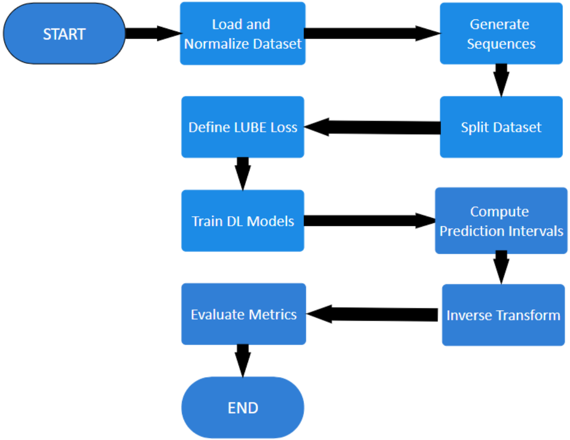

Figure 2: Traditional LUBE Method Flowchart.

Methodology (Cont.)

Algorithm 1 Traditional LUBE 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 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
  Compute $\text{Loss}_{\text{LUBE}}$

Step 3: Model Training foreach $M \in \{LSTM, CNN, GRU, BiLSTM\}$ **do**
  Define model architecture
  Compile with LUBE loss
  Train on $(X_{\text{train}}, y_{\text{train}})$ for e epochs
  Validate on $(X_{\text{val}}, y_{\text{val}})$
  Save LB, UB for test data

Step 4: Evaluation Metrics Compute the PINAW, ACE and AWE of the computed prediction intervals.
Step 5: Aggregate Results Compute mean of metrics for all models

Methodology (Cont.)

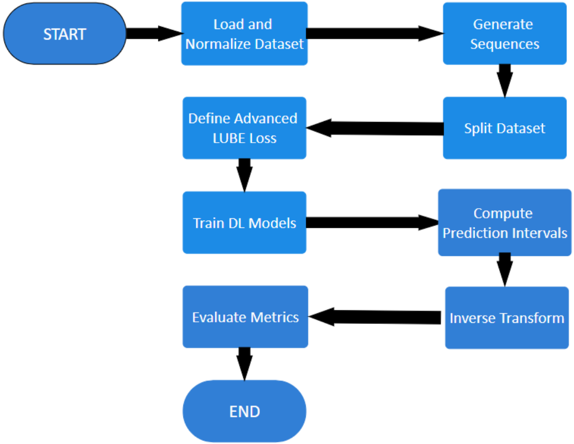

Figure 3: Advanced LUBE Method Flowchart.

Methodology (Cont.)

Algorithm 2 Advanced LUBE 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 and PINAW
Compute $\text{Loss}_{\text{LUBE}}$

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 $(X_{\text{val}}, y_{\text{val}})$
Save LB, UB for test data

Step 4: Evaluation Metrics Compute the ACE and AWE of the computed prediction intervals.
Step 5: Aggregate Results Compute mean of metrics for all models and confidence levels

Methodology (Cont.)

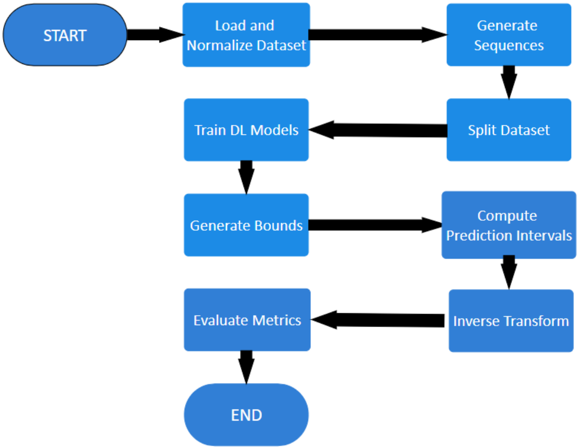

Figure 4: Quantile Regression based Method Flowchart.

Methodology (Cont.)

Algorithm 3 Quantile Regression (QR) based 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 QR Loss Function
foreach $c \in \{0.9, 0.8, 0.7, 0.6\}$ **do**
Compute q_{lower} and q_{upper}
Compute LB, UB
Compute $Loss_{\text{lower}}$ and $Loss_{\text{upper}}$
Compute PICP and PINAW
Compute $Loss_{\text{QR}}$
Step 3: Model Training
foreach $M \in \{LSTM, CNN, GRU, BiLSTM\}$ **do**
Define model architecture
Compile with QR-based loss
Train on $(X_{\text{train}}, y_{\text{train}})$ for e epochs
Validate on $(X_{\text{val}}, y_{\text{val}})$
Save LB, UB for test data
Step 4: Evaluation Metrics Compute ACE and AWE of the computed prediction intervals.
Step 5: Aggregate Results Compute mean of metrics for all models and confidence levels

Methodology (Cont.)

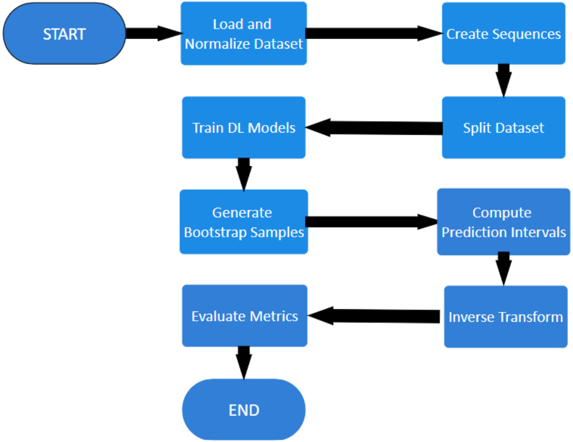

Figure 5: Bootstrap based Method Flowchart.

Methodology (Cont.)

Algorithm 4 Bootstrap based 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: Model Training
  foreach *model* $M \in \{LSTM, CNN, GRU, BiLSTM\}$ **do**
    Train M on $(X_{\text{train}}, y_{\text{train}})$ for e epochs using MSE loss
Step 3: Bootstrap Prediction Intervals
  foreach *confidence level* $c \in \{0.9, 0.8, 0.7, 0.6\}$ **do**
    foreach *trained model* M **do**
      Generate B bootstrap samples from X_{test}
       Predict y_{pred} on each sample using M
       For each y^i_{test}, compute LB_i and UB_i as empirical quantiles at q_{lower} and q_{upper}
Step 4: Evaluation Metrics Compute PICP, PINAW, ACE and AWE for each model and confidence level
Step 5: Aggregate Results Compute mean of all metrics across all runs

Methodology (Cont.)

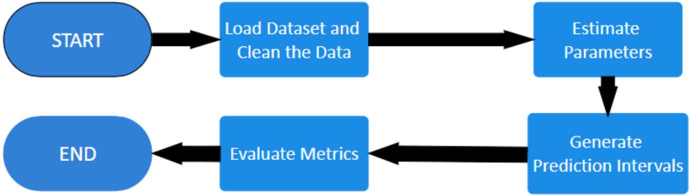

Figure 6: Gaussian Distribution based Method Flowchart.

Methodology (Cont.)

Algorithm 5 Gaussian Distribution based Method.

Input: Time series dataset D with target variable y
Output: Predicted intervals $[LB, UB]$, PICP, PINAW, ACE, AWE
Step 1: Data Preprocessing
Remove missing values to obtain cleaned target y
Step 2: Define Confidence Levels
$C = \{0.9, 0.8, 0.7, 0.6\}$
Step 3: Estimate Distribution Parameters
Compute mean μ and standard deviation σ of y
Step 4: Generate Prediction Intervals
foreach $c \in C$ **do**
  Compute z_c
  Compute LB and UB
Step 5: Evaluation Metrics Compute the PICP, PINAW, ACE and AWE of the computed prediction intervals.
Step 6: Aggregate Results Compute mean of all metrics across all runs

Methodology (Cont.)

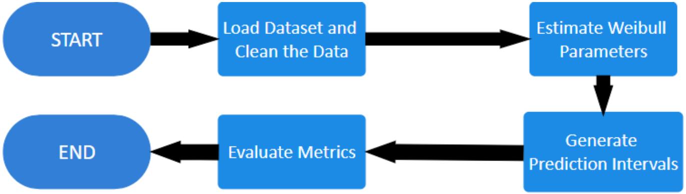

Figure 7: Weibull Distribution based Method Flowchart.

Methodology (Cont.)

Algorithm 6 Weibull Distribution based Method.

Input: Time series dataset D with target variable y
Output: Predicted intervals $[LB, UB]$, PICP, PINAW, ACE, AWE
Step 1: Data Preprocessing
Remove missing values to obtain cleaned target y
Step 2: Define Confidence Levels
$C = \{0.9, 0.8, 0.7, 0.6\}$
Step 3: Estimate Weibull Parameters
Fit Weibull distribution to y using Maximum Likelihood Estimation (MLE)
Obtain shape κ, scale λ, and location θ (fixed to 0)
Step 4: Generate Prediction Intervals
foreach $c \in C$ **do**
  Compute α
    Compute LB and UB

Step 5: Evaluation Metrics Compute the PICP, PINAW, ACE and AWE of the computed prediction intervals.
Step 6: Aggregate Results Compute mean of all metrics across all runs

Methodology (Cont.)

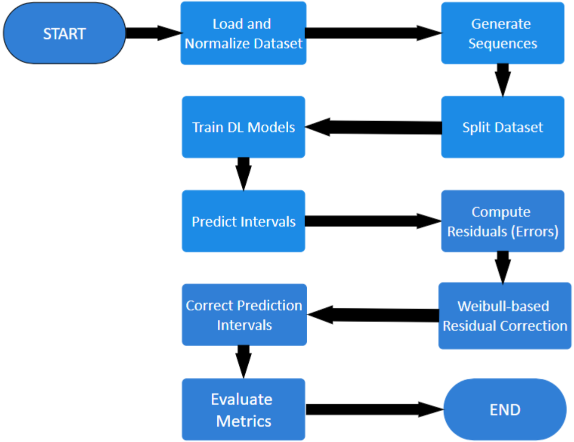

Figure 8: LUBE-Weibull based Hybrid Method Flowchart.

Methodology (Cont.)

Algorithm 7 Proposed 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 and PINAW
      Compute $\text{Loss}_{\text{LUBE}}$
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
Step 6: Evaluation Metrics Compute PICP, PINAW, ACE and AWE of the computed prediction intervals.
Step 7: Aggregate Results Compute mean of all metrics across all runs

Methodology (Cont.)

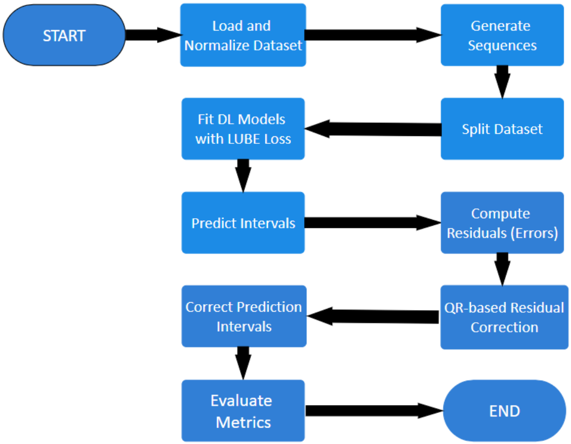

Figure 9: LUBE-QR based Hybrid Method Flowchart.

Methodology (Cont.)

Algorithm 8 Proposed 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 $\text{Loss}_{\text{lower}}$ and $\text{Loss}_{\text{upper}}$
    Compute PICP and PINAW, Compute $\text{Loss}_{\text{LUBE}}$.
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 LUBE bounds LB, UB on test data, Compute midpoint $\hat{y}$
Step 4: Fit Quantile Regression on Residuals
  Compute residuals
  foreach $c \in \{0.9, 0.8, 0.7, 0.6\}$ **do**
    Train Quantile Regression models to estimate Lower quantile and Upper quantile and 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, PINAW, ACE and AWE of the computed prediction intervals.
Step 7: Aggregate Results Compute mean of all metrics across all runs

Experimental Setup

- All the experiments are conducted on Acer Nitro V Laptop containing i7 12th Gen Processor, 32 GB DDR4 RAM and RTX 3070 Ti Graphics Card.
- Jupyter Notebook was used to run the simulations using the Python programming language (Python 3.11.5) within the Anaconda Navigator on Windows 11 OS version 23H2.

Simulation Results

Table 3: Performance Comparison of Non-Parametric Methods on PICP Metric.

Dataset	Method Used	Confidence Levels			
		0.6	**0.7**	**0.8**	**0.9**
Adani Ports	Traditional LUBE with LSTM	100	100	100	100
	Advanced LUBE with CNN	100	100	100	100
	QR-based with CNN	79.07	86.96	86.74	95.73
	Bootstrap-based with GRU	58.95	68.43	78.29	88.05
Asian Paints	Traditional LUBE with LSTM	100	100	100	100
	Advanced LUBE with CNN	100	100	100	100
	QR-based with CNN	81.00	95.28	94.19	98.68
	Bootstrap-based with GRU	58.70	69.01	78.91	88.70
Electricity Consumption	Traditional LUBE with LSTM	100	100	100	100
	Advanced LUBE with CNN	99.68	99.67	99.67	99.69
	QR-based with CNN	56.94	73.78	82.73	87.52
	Bootstrap-based with GRU	19.50	28.20	31.76	41.40
Web Traffic	Traditional LUBE with LSTM	100	100	100	100
	Advanced LUBE with CNN	99.96	99.92	99.41	99.19
	QR-based with CNN	15.12	18.41	24.39	19.51
	Bootstrap-based with GRU	20.12	26.82	35.24	42.31

Simulation Results (Cont.)

Table 4: Performance Comparison of Parametric Methods on PICP Metric.

Dataset	Method Used	Confidence Levels			
		0.6	0.7	0.8	0.9
Adani Ports	Gaussian Distribution	55.75	71.19	88.32	91.36
	Weibull Distribution	56.08	65.77	86.66	92.50
Asian Paints	Gaussian Distribution	63.38	83.58	86.78	90.93
	Weibull Distribution	56.26	68.69	86.28	93.35
Electricity Consumption	Gaussian Distribution	79.60	83.36	87.41	94.30
	Weibull Distribution	64.10	80.30	86.54	96.77
Web Traffic	Gaussian Distribution	62.54	71.27	78.73	89.45
	Weibull Distribution	70	77.45	84.54	92.90

Simulation Results (Cont.)

Table 5: Performance Comparison of Non-Parametric Methods on PINAW Metric.

Dataset	Method Used	Confidence Levels			
		0.6	**0.7**	**0.8**	**0.9**
Adani Ports	Traditional LUBE with LSTM	4.94	4.80	6.85	5.10
	Advanced LUBE with CNN	3.08	3.05	3.14	3.08
	QR with CNN	0.10	0.11	0.13	0.19
	Bootstrap with GRU	0.17	0.28	0.46	0.68
Asian Paints	Traditional LUBE with LSTM	6.34	6.39	6.48	5.87
	Advanced LUBE with CNN	4.87	5.12	5.04	5.24
	QR with CNN	0.13	0.20	0.21	0.32
	Bootstrap with GRU	0.36	0.49	0.64	0.79
Electricity Consumption	Traditional LUBE with LSTM	2.24	2.68	2.75	1.97
	Advanced LUBE with CNN	0.99	0.99	0.98	1.01
	QR with CNN	0.16	0.20	0.24	0.30
	Bootstrap with GRU	0.07	0.10	0.11	0.13
Web Traffic	Traditional LUBE with LSTM	3.77	2.87	4.02	3.75
	Advanced LUBE with CNN	1.84	1.78	1.79	1.68
	QR with CNN	0.25	0.33	0.40	0.49
	Bootstrap with GRU	0.20	0.25	0.30	0.34

Simulation Results (Cont.)

Table 6: Performance Comparison of Parametric Methods on PINAW Metric.

Dataset	Method Used	Confidence Levels			
		0.6	**0.7**	**0.8**	**0.9**
Adani Ports	Gaussian Distribution	0.09	0.12	0.14	0.19
	Weibull Distribution	0.27	0.33	0.40	0.51
Asian Paints	Gaussian Distribution	0.34	0.42	0.52	0.66
	Weibull Distribution	0.31	0.38	0.48	0.62
Electricity Consumption	Gaussian Distribution	0.003	0.004	0.005	0.006
	Weibull Distribution	0.18	0.22	0.27	0.37
Web Traffic	Gaussian Distribution	0.008	0.009	0.10	0.15
	Weibull Distribution	0.39	0.48	0.60	0.76

Simulation Results (Cont.)

Table 7: Performance Comparison of Non-Parametric Methods on ACE Metric.

Dataset	Method Used	Confidence Levels			
		0.6	0.7	0.8	0.9
Adani Ports	Traditional LUBE with LSTM	40	30	20	10
	Advanced LUBE with CNN	40	30	20	10
	QR with CNN	21.86	16.96	12.12	5.75
	Bootstrap with GRU	2.17	1.65	1.71	1.95
Asian Paints	Traditional LUBE with LSTM	40	30	20	10
	Advanced LUBE with CNN	40	30	20	10
	QR with CNN	23.66	25.28	16.91	8.68
	Bootstrap with GRU	1.80	1.35	1.35	1.30
Electricity Consumption	Traditional LUBE with LSTM	40	30	20	10
	Advanced LUBE with CNN	39.68	29.67	19.67	9.69
	QR with CNN	10.44	4.11	5.94	5.66
	Bootstrap with GRU	40.50	41.80	48.24	48.60
Web Traffic	Traditional LUBE with LSTM	40	30	20	10
	Advanced LUBE with CNN	39.96	28.92	19.41	9.19
	QR with CNN	44.87	51.58	55.60	70.48
	Bootstrap with GRU	39.87	43.17	44.75	47.68

Simulation Results (Cont.)

Table 8: Performance Comparison of Parametric Methods on ACE Metric.

Dataset	**Method Used**	**Confidence Levels**			
		0.6	**0.7**	**0.8**	**0.9**
Adani Ports	Gaussian Distribution	4.25	1.19	8.32	1.36
	Weibull Distribution	3.92	4.23	6.66	2.50
Asian Paints	Gaussian Distribution	3.38	13.58	6.78	0.93
	Weibull Distribution	3.74	1.31	6.28	3.35
Electricity Consumption	Gaussian Distribution	19.60	13.36	7.41	4.30
	Weibull Distribution	4.10	10.30	6.54	6.77
Web Traffic	Gaussian Distribution	2.54	1.27	1.27	0.55
	Weibull Distribution	10	7.45	4.54	2.90

Simulation Results(Cont.)

Table 9: Performance Comparison of Non-Parametric Methods on AWE Metric.

Dataset	Method Used	Confidence Levels			
		0.6	**0.7**	**0.8**	**0.9**
Adani Ports	Traditional LUBE with LSTM	2460.54	2369.84	3652.41	2557.74
	Advanced LUBE with CNN	1297.77	1278.93	1333.73	1297.10
	QR with CNN	562.45	551.85	539.85	507.37
	Bootstrap with GRU	519.90	451.96	337.02	198.01
Asian Paints	Traditional LUBE with LSTM	9338.41	9424.85	9571.73	8519.34
	Advanced LUBE with CNN	6758.33	7205.28	7062.18	7401.71
	QR with CNN	1518.25	1402.63	1380.19	1185.35
	Bootstrap with GRU	1111.23	885.19	627.15	373.43
Electricity Consumption	Traditional LUBE with LSTM	32.70	44.36	46.31	25.68
	Advanced LUBE with CNN	0.90	0.65	1.08	0.92
	QR with CNN	22.03	21.14	19.87	18.44
	Bootstrap with GRU	24.39	23.69	23.39	22.91
Web Traffic	Traditional LUBE with LSTM	25.84	17.50	28.19	25.63
	Advanced LUBE with CNN	7.85	7.31	7.37	6.38
	QR with CNN	6.90	6.21	5.51	4.67
	Bootstrap with GRU	7.44	6.96	6.48	6.07

Simulation Results (Cont.)

Table 10: Performance Comparison of Parametric Methods on AWE Metric.

Dataset	Method Used	Confidence Levels			
		0.6	**0.7**	**0.8**	**0.9**
Adani Ports	Gaussian Distribution	163.11	200.87	248.38	318.79
	Weibull Distribution	323.55	395.97	485.05	612.33
Asian Paints	Gaussian Distribution	904.70	1114.12	1377.61	1768.14
	Weibull Distribution	1558.04	1923.42	2388.31	3092.62
Electricity Consumption	Gaussian Distribution	2.94	3.62	4.48	5.74
	Weibull Distribution	4.71	5.86	7.36	9.73
Web Traffic	Gaussian Distribution	2.21	2.72	3.37	4.32
	Weibull Distribution	5.25	6.48	8.02	10.31

Simulation Results (Cont.)

Table 11: Performance Comparison of Proposed Hybrid Methods on PICP Metric.

Dataset	Method Used	Confidence Levels			
		0.6	**0.7**	**0.8**	**0.9**
Adani Ports	LUBE-Weibull Hybrid Method with CNN	100	100	100	100
	LUBE-QR Hybrid Method with LSTM	59.96	70.02	79.88	89.94
Asian Paints	LUBE-Weibull Hybrid Method with CNN	100	100	100	100
	LUBE-QR Hybrid Method with LSTM	60	69.94	80	89.94
Electricity Consumption	LUBE-Weibull Hybrid Method with CNN	100	100	100	100
	LUBE-QR Hybrid Method with LSTM	59.93	70.04	79.78	89.89
Web Traffic	LUBE-Weibull Hybrid Method with CNN	100	100	100	100
	LUBE-QR Hybrid Method with LSTM	60.49	69.75	79.26	89.38

Simulation Results (Cont.)

Table 12: Performance Comparison of Proposed Hybrid Methods on PINAW Metric.

Dataset	Method Used	Confidence Levels			
		0.6	**0.7**	**0.8**	**0.9**
Adani Ports	LUBE-Weibull Hybrid Method with CNN	4.15	4.16	4.17	4.18
	LUBE-QR Hybrid Method with LSTM	0.02	0.03	0.05	0.06
Asian Paints	LUBE-Weibull Hybrid Method with CNN	6.84	6.85	6.87	6.89
	LUBE-QR Hybrid Method with LSTM	0.06	0.04	0.05	0.07
Electricity Consumption	LUBE-Weibull Hybrid Method with CNN	1.37	1.39	1.41	1.44
	LUBE-QR Hybrid Method with LSTM	0.16	0.19	0.22	0.29
Web Traffic	LUBE-Weibull Hybrid Method with CNN	2.16	2.21	2.27	2.37
	LUBE-QR Hybrid Method with LSTM	0.56	0.60	0.77	0.87

Simulation Results (Cont.)

Table 13: Performance Comparison of Proposed Hybrid Methods on ACE Metric.

Dataset	Method Used	Confidence Levels			
		0.6	**0.7**	**0.8**	**0.9**
Adani Ports	LUBE-Weibull Hybrid Method with CNN	40	30	20	10
	LUBE-QR Hybrid Method with LSTM	0.04	0.02	0.12	0.06
Asian Paints	LUBE-Weibull Hybrid Method with CNN	40	30	20	10
	LUBE-QR Hybrid Method with LSTM	0.00	0.06	0.00	0.06
Electricity Consumption	LUBE-Weibull Hybrid Method with CNN	40	30	20	10
	LUBE-QR Hybrid Method with LSTM	0.07	0.04	0.22	0.11
Web Traffic	LUBE-Weibull Hybrid Method with CNN	40	30	20	10
	LUBE-QR Hybrid Method with LSTM	0.49	0.62	0.84	0.72

Simulation Results (Cont.)

Table 14: Performance Comparison of Proposed Hybrid Methods on AWE Metric.

Dataset	**Method Used**	**Confidence Levels**			
		0.6	**0.7**	**0.8**	**0.9**
Adani Ports	LUBE-Weibull Hybrid Method with CNN	1961.88	1968.03	1975.35	1985.47
	LUBE-QR Hybrid Method with LSTM	608.53	602.49	593.65	583.94
Asian Paints	LUBE-Weibull Hybrid Method with CNN	10198.06	10219.74	10245.51	10280.97
	LUBE-QR Hybrid Method with LSTM	1639.36	1670.81	1662.02	1620.58
Electricity Consumption	LUBE-Weibull Hybrid Method with CNN	9.85	10.30	10.85	11.65
	LUBE-QR Hybrid Method with LSTM	22.10	21.51	20.61	18.90
Web Traffic	LUBE-Weibull Hybrid Method with CNN	10.82	11.27	11.86	12.79
	LUBE-QR Hybrid Method with LSTM	4.12	3.75	2.34	1.39

Graphical Analysis

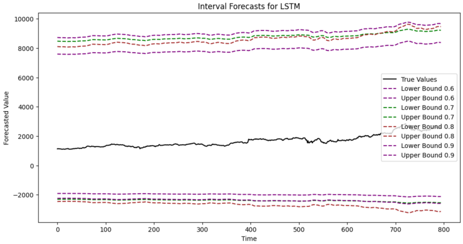

Figure 10: Prediction Intervals for Asian Paints dataset obtained using Traditional LUBE method and LSTM model.

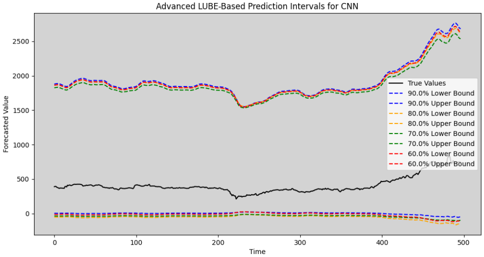

Figure 11: Prediction Intervals for Adani Ports dataset obtained using Advanced LUBE method and CNN model.

Graphical Analysis (Cont.)

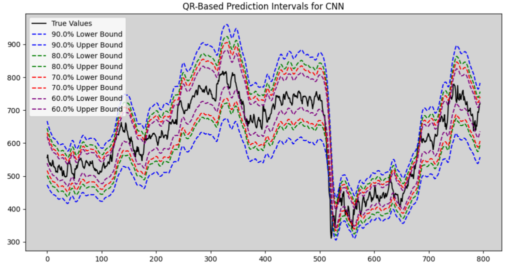

Figure 12: Prediction Intervals for Axis Bank dataset obtained using QR-based method and CNN model.

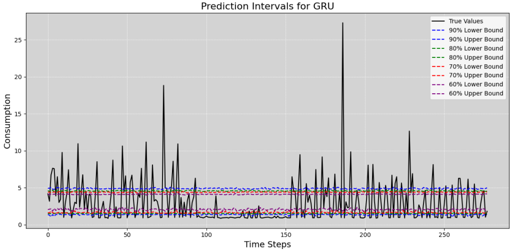

Figure 13: Prediction Intervals for Electricity Consumption dataset obtained using Bootstrap-based method and GRU model.

Graphical Analysis (Cont.)

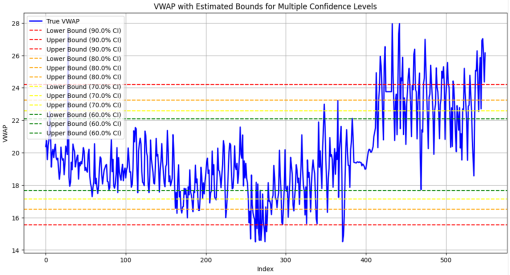

Figure 14: Prediction Intervals for Web Traffic dataset obtained using Gaussian Distribution based method.

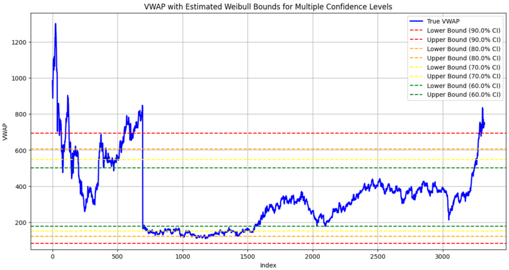

Figure 15: Prediction Intervals for Adani Ports dataset obtained using Weibull Distribution based method.

Graphical Analysis (Cont.)

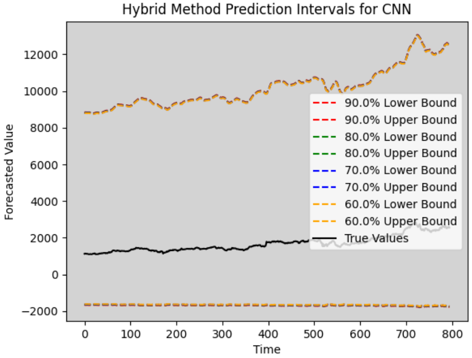

Figure 16: Prediction Intervals for Asian Paints dataset obtained using LUBE-Weibull based hybrid method with CNN model.

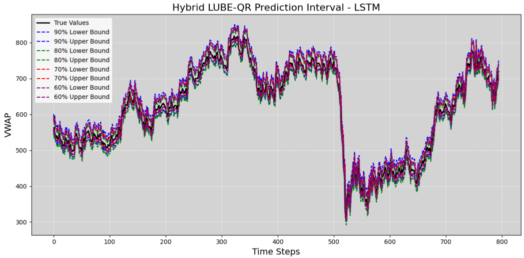

Figure 17: Prediction Intervals for Axis Bank dataset obtained using LUBE-QR based hybrid method and LSTM model.

Statistical Analysis

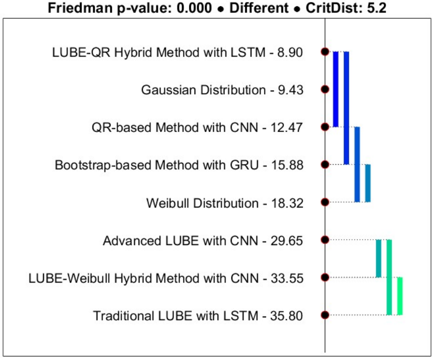

Figure 18: Friedman-Nemenyi Hypothesis Test on PINAW Metric performed on all the five datasets together.

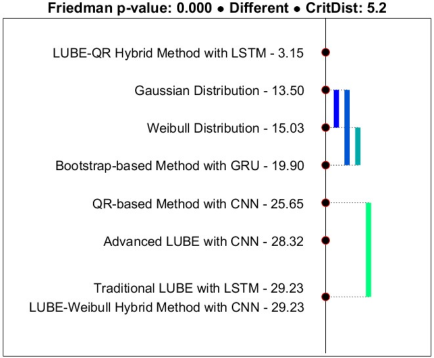

Figure 19: Friedman-Nemenyi Hypothesis Test on ACE Metric performed on all the five datasets together.

Statistical Analysis (Cont.)

- Friedman-Nemenyi hypothesis results (Figs. 18 and 19) show PINAW and ACE metric comparisons across five datasets and eight forecasting methods (each paired with its best-performing DL model).
- The proposed LUBE–QR based Hybrid Method is found to be statistically better than other forecasting methods assessed.

Conclusion

- The Advanced LUBE method gave 100% PICP values consistently across all the datasets while maintaining low PINAW and AWE Values. These results show that this method is a highly reliable method with a drawback of wider intervals (i.e conservative interval widths).
- Among the parametric methods, Gaussian Distribution showed promising results but it is limited by its assumption of normality and fixed width intervals, making it unsuitable for complex, non-stationary time series with dynamic patterns.

Conclusion (Cont.)

- The proposed LUBE-Weibull hybrid method is a computationally efficient alternative to computationally expensive methods. It achieved consistent results at par with Advanced LUBE method at PICP metric while being slightly worse on PINAW and AWE metric values. However because of it's efficiency, it can be a good alternative for situations that demands low computational forecasting.
- The proposed LUBE-QR hybrid method performed the best out of all the methods assessed. It achieved desirable PICP values across each dataset and confidence levels while having the least PINAW, ACE and AWE values. It was also determined to be the best method statistically by Friedman-Nemenyi Hypothesis test.

Future Work

- 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.
- Extend current methods to handle multiple correlated time series jointly with shared uncertainty modeling.

References

Thank You