COL851 Course Project

Kartik and Brian

October 13, 2025

Contents

1 Introduction

1.1 Student details

Kartik Meena, 2021CS50618; Brian Sajeev Kattikat, 2021CS50609

1.2 Project Objectives

The primary goals of this project are:

1. To deploy and evaluate the zero-shot performance of the Chronos LLM for PM forecasting on datasets from **Gurgaon (City A)** and **Patna (City B)**.

2. To perform a parametric study on the effect of **context length** (2, 4, 8, 10, 14 days) and **forecasting horizon** (4, 8, 12, 24, 48 hours) on accuracy (RMSE).

3. To benchmark different feasible **pre-trained model variants** (`chronos-t5-mini`, `chronos-t5-small`, `chronos-bolt-small`) while adhering to the constraint of **laptop CPU** computation.

4. To identify the overall best-performing configuration (model, context, and horizon) based on the combined **Average RMSE** for both cities.

2 Methodology

2.1 Data Description and Pre-processing

The time series data comprises PM concentration records for Gurgaon (City A) and Patna (City B). For this experiment, only two columns were utilized: **Column 1 (Time Index)** and **Column 5 (PM concentration)**. The frequency of the data is assumed to be hourly. The data was partitioned into historical context windows and future prediction windows for rolling-window evaluation, using the **Root Mean Squared Error (RMSE)** as the accuracy metric.

2.2 Chronos Model and Constraints

The **Amazon Chronos** model was used in its **Zero-Shot** mode. Due to the **laptop CPU constraint**, three model variants were tested to identify the best trade-off between size and performance:

- `amazon/chronos-t5-mini`
- `amazon/chronos-t5-small`
- `amazon/chronos-bolt-small`

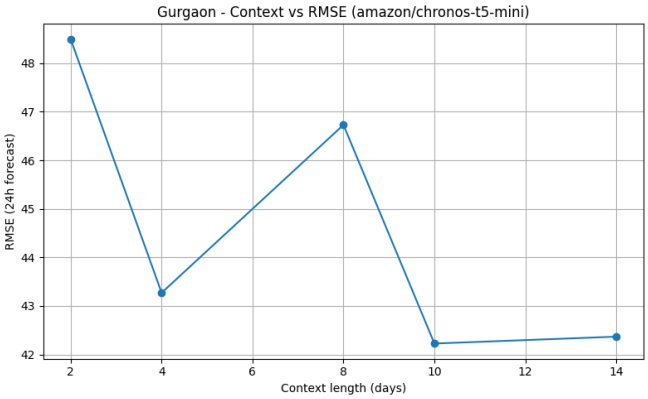
3 Results

3.1 Performance Summary on Gurgaon (City A)

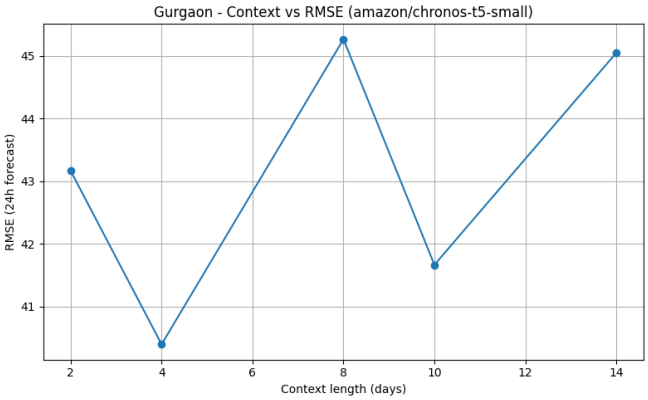
For Gurgaon (City A), the overall lowest RMSE was achieved at the shortest horizon by the `chronos-bolt-small` model. The feasibility check allowed for testing three variants: `chronos-t5-mini`, `chronos-t5-small`, and `chronos-bolt-small`.

Table 1: Summary of Best Zero-Shot Performance on Gurgaon Data (City A)

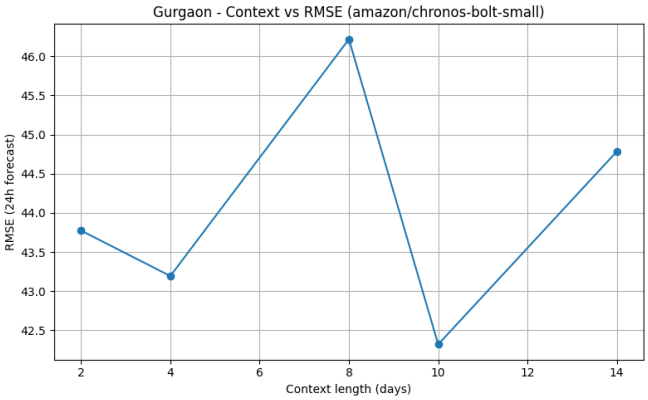
Model	Best Ctx (days)	Min RMSE (24h)	Best Hrz (hrs)	Min RMSE (10d Ctx
`t5-mini`	10	42.23	4	26.42
`t5-small`	4	**40.40**	4	26.34
`bolt-small`	10	42.32	4	**25.74**



(a) `chronos-t5-mini` Context

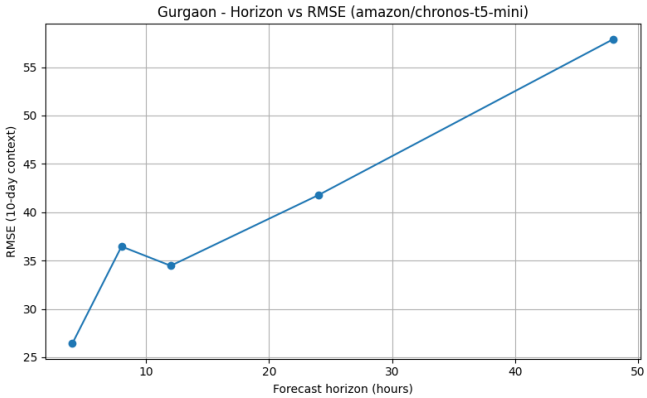


(b) `chronos-t5-small` Context

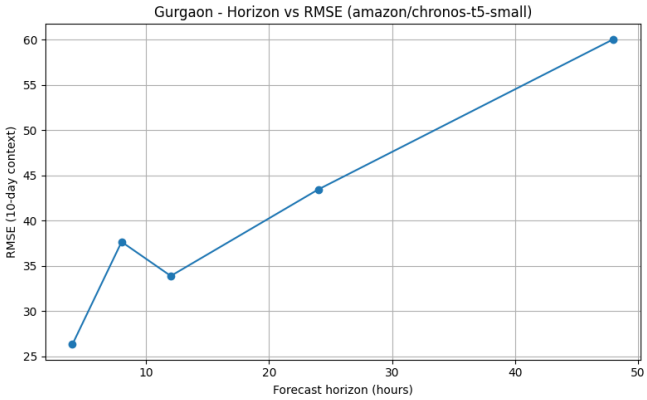


(c) `chronos-bolt-small` Context

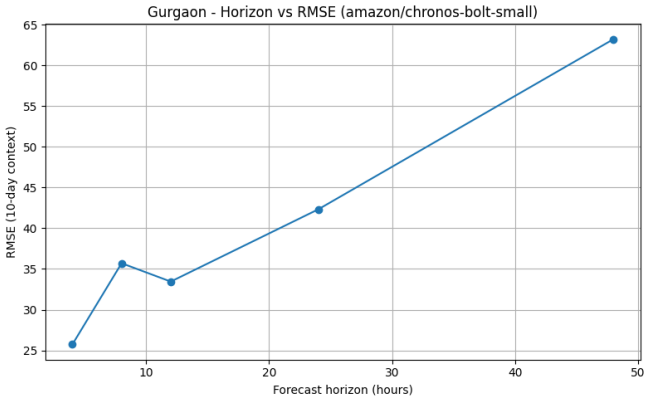
Figure 1: Average RMSE for PM forecasting (24h horizon) with varying context lengths (Gurgaon).



(a) `chronos-t5-mini` Horizon



(b) `chronos-t5-small` Horizon



(c) `chronos-bolt-small` Horizon

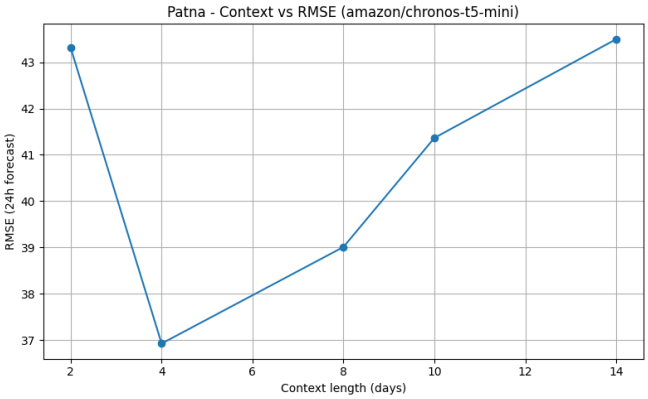
Figure 2: Average RMSE for PM forecasting (10-day context) with varying forecasting horizons (Gurgaon).

3.2 Performance Summary on Patna (City B)

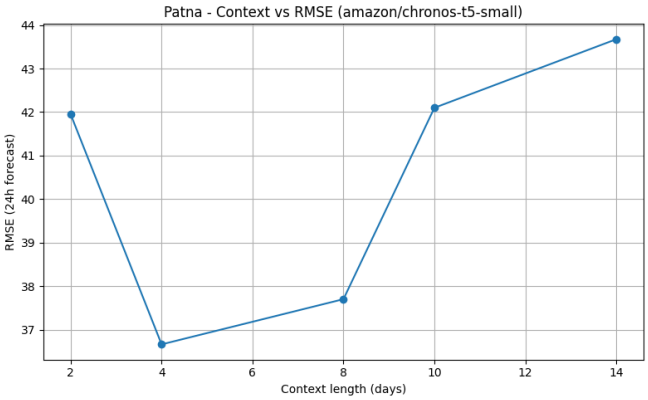
For Patna (City B), the `chronos-bolt-small` model also delivered the lowest RMSE, achieving slightly better overall performance than in Gurgaon.

Table 2: Summary of Best Zero-Shot Performance on Patna Data (City B)

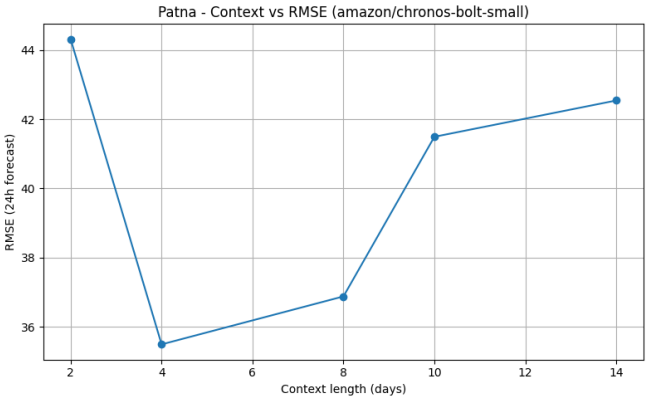
Model	Best Ctx (days)	Min RMSE (24h)	Best Hrz (hrs)	Min RMSE (10d Ctx
`t5-mini`	4	36.92	4	25.78
`t5-small`	4	36.67	4	25.09
`bolt-small`	4	**35.49**	4	**24.88**



(a) `chronos-t5-mini` Context

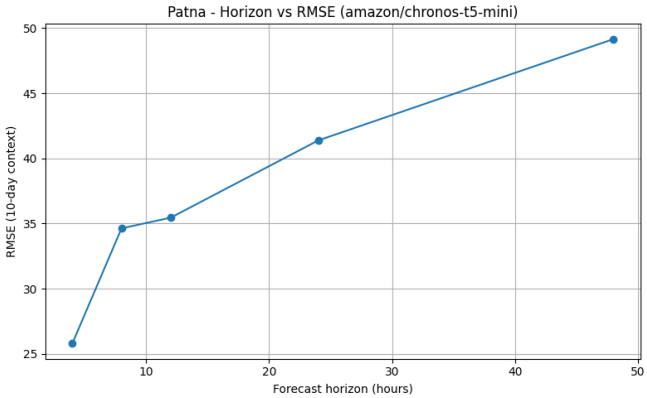


(b) `chronos-t5-small` Context

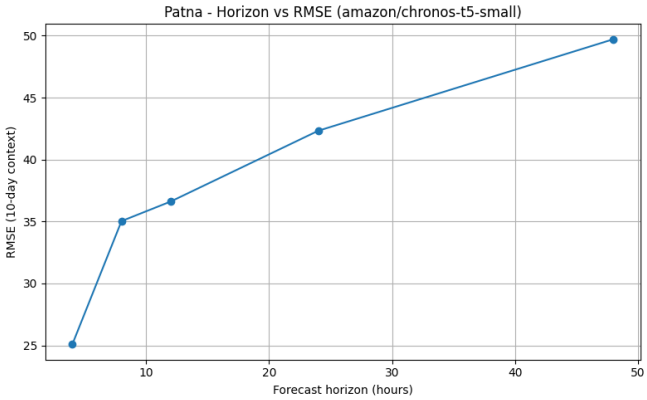


(c) `chronos-bolt-small` Context

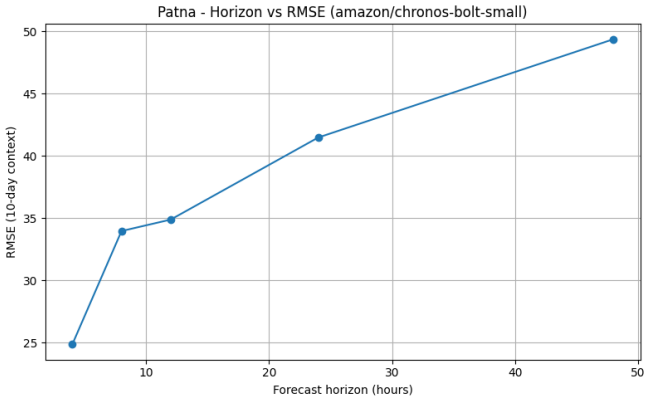
Figure 3: Average RMSE for PM forecasting (24h horizon) with varying context lengths (Patna).



(a) `chronos-t5-mini` Horizon



(b) `chronos-t5-small` Horizon



(c) `chronos-bolt-small` Horizon

Figure 4: Average RMSE for PM forecasting (10-day context) with varying forecasting horizons (Patna).

4 Discussion

4.1 Optimal Configuration: Best Average RMSE for Both Cities

The comprehensive analysis across both cities reveals a clear superior zero-shot configuration:

- **Best Model Variant:** **`amazon/chronos-bolt-small`** (Min RMSE ≈ 24.88)
- **Best Forecasting Horizon:** **4 hours**
- **Best Context Length:** **4 days**

The **`chronos-bolt-small`** model consistently delivered the lowest RMSE in the most predictable scenario (4-hour horizon) for both cities, slightly outperforming the T5-based variants. The optimal context length, which minimizes RMSE for the challenging 24-hour horizon, was **4 days** for Patna, and either 4 or 10 days for Gurgaon depending on the model. This suggests that for highly non-linear data like PM, a relatively short context (around 4 days) is often sufficient to capture relevant daily/short-term patterns without adding unnecessary noise or computational load.

4.2 Comparison between Cities

- **Overall Accuracy:** Patna (City B) achieved a slightly lower overall minimum RMSE ($\approx$ **24.88**) compared to Gurgaon (City A, $\approx$ **25.74**). This indicates that the PM time series in Patna may exhibit slightly clearer or more stable patterns that the pre-trained Chronos model can identify in a zero-shot setting.
- **Context Length Preference:** Patna consistently favored a **4-day** context across all models for the 24-hour prediction task, while Gurgaon's best context was either 4 or 10 days. This implies that Patna's relevant forecasting signals may be concentrated in the immediate past, whereas Gurgaon's patterns might occasionally benefit from capturing weekly or 10-day cycles.
- **Horizon Sensitivity:** Both cities demonstrated high sensitivity to the forecasting horizon. RMSE values increased steeply as the horizon moved from 4 hours to 48 hours, highlighting the inherent difficulty of zero-shot, univariate forecasting for volatile air quality data in the absence of co-variates.

5 Performance and Resource Analysis

5.1 Measurement Setup

Performance metrics were collected using `psutil` and Prometheus during zero-shot Chronos forecasts on CPU. Each configuration measured:

- Average and 95th-percentile inference latency
- Throughput (samples/sec)
- CPU utilization (%)

- Memory utilization (%)

Experiments varied context length (2–14 days) and horizon (4–48 h) across three models.

5.2 Observed Performance Trends

- **Latency:** Increased with longer context and horizon. For Gurgaon (`t5-mini`), average latency rose from 0.35 s (2 days) to 0.60 s (14 days).
- **Throughput:** Inversely related to latency; `chronos-bolt-small` reached ≈ 37–42 samples/s.
- **CPU Utilization:** Averaged 55–65 % across all settings, peaking near 76 % for longest contexts.

5.3 Quantitative Highlights

Table 3: Performance Summary (10-day context, 24-hour horizon, CPU)

Model	City	Latency (s)	Throughput	CPU (%)	MEM (%)	RMSE
`t5-mini`	Gurgaon	0.50	1.98	56.7	91.2	115.75
`t5-small`	Gurgaon	0.79	1.26	57.5	92.6	116.88
`bolt-small`	Gurgaon	0.027	37.04	61.1	92.8	262.82
`t5-mini`	Patna	0.50	1.99	58.1	91.6	163.17
`t5-small`	Patna	0.78	1.28	57.4	92.3	157.03
`bolt-small`	Patna	0.025	40.73	55.0	92.9	117.11

5.4 Visualizations

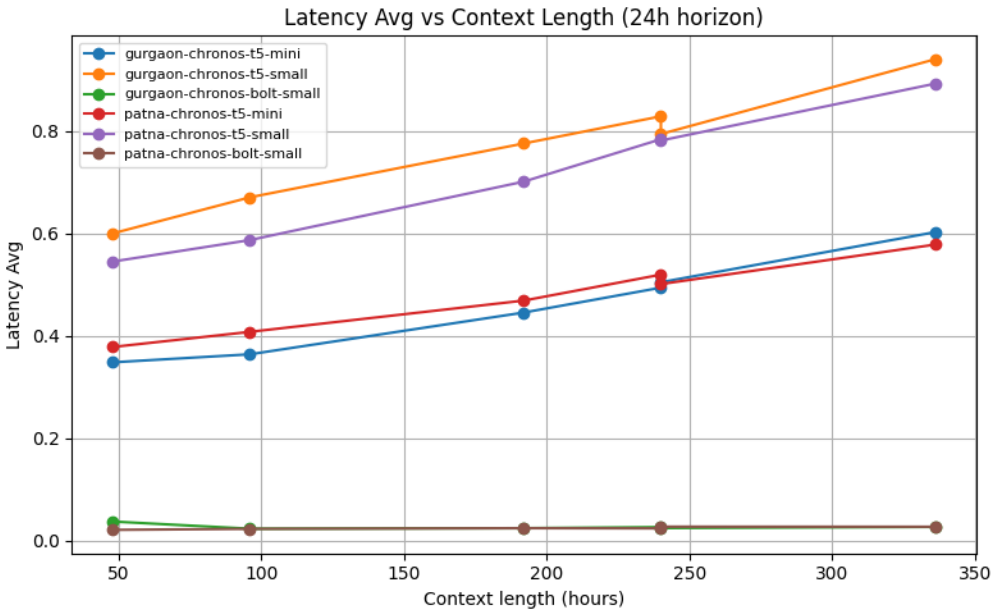


Figure 5: Average latency vs. context length.

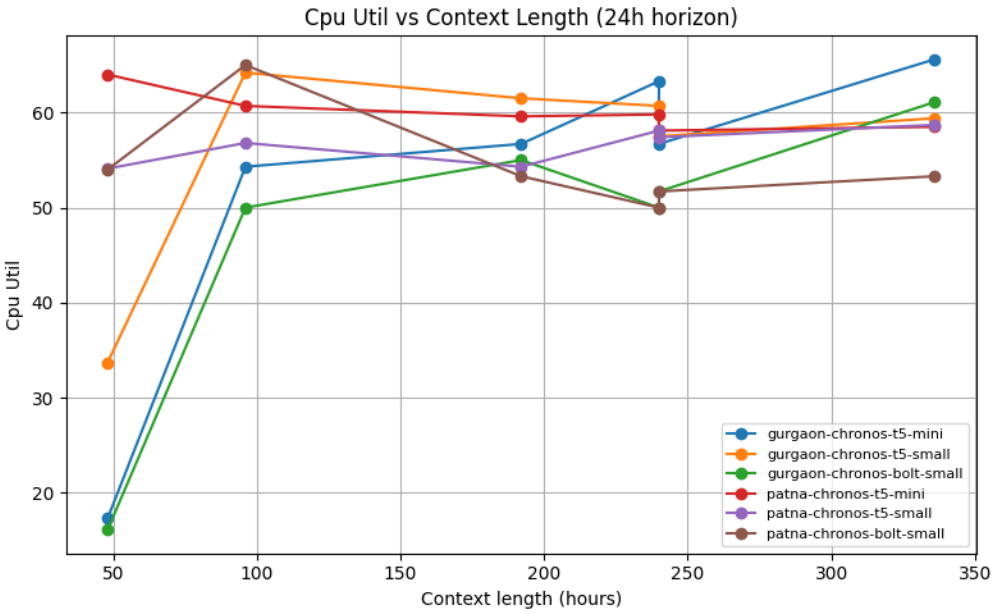


Figure 6: CPU utilization vs. context length.