Model Performance Report

1. Executive Summary

This report evaluates the performance of three forecasting models — XGBoost, LSTM, and Prophet — developed for sales forecasting. Each model was assessed using regression metrics, and performance comparisons were drawn to recommend the best-suited model for deployment. The LSTM model delivered the lowest RMSE and MSE, making it the most accurate, while XGBoost offered faster inference and easier deployment. Prophet underperformed but remains useful for baseline forecasting.

2. Modeling Objective Recap

The objective is to predict daily sales for each store-family combination. Metrics used for model evaluation include RMSE, MAE, MAPE, and R^2 Score. These metrics were chosen for their ability to capture forecast accuracy, scale sensitivity, and explainability in a business context.

3. Model Descriptions

3.0 Baseline: Linear Regression

A simple regression model used as a baseline for comparison. Selected a subset of engineered features ('sales_lag_7', 'rolling_mean_7', 'onpromotion', 'is_holiday') and trained using ordinary least squares. While fast and interpretable, it could not capture complex temporal or nonlinear patterns. It served as a benchmark to assess the value of more sophisticated models.

3.1 XGBoost

A tree-based ensemble boosting model known for its speed and performance. It handles missing values well and is highly interpretable, but lacks temporal context.

3.2 LSTM

A neural network capable of modeling long-term dependencies in time series data. Input sequences were reshaped and standardized. The model captured patterns well, though it required significant training time.

3.3 Prophet

A univariate time series model developed by Meta, known for handling seasonality and holidays automatically. It is intuitive but limited in handling complex exogenous features.

4. Model Performance Comparison

The table below summarizes the key metrics for each model.

Model	RMSE	MSE	R^2 Score	Training Time
Linear Regression	422.5	178504.86	0.90	1 sec
XGBoost	385.49	148600.95	0.93	25 sec
LSTM	379.1	143766.2	0.93	41 min
Prophet	1130	12700	0.14	30 sec

5. Error Analysis

The LSTM model outperformed others during promotional periods and holiday spikes. XGBoost was slightly less accurate during volatile demand periods but performed well overall. Prophet consistently underpredicted during holidays and promotions, which hurt its accuracy on key business dates.

6. Final Model Justification

LSTM was selected due to its superior accuracy (lowest RMSE and highest R^2 Score). However, XGBoost remains a viable alternative for real-time applications due to its speed and simplicity. Prophet, while interpretable, was not competitive in this dataset. No ensemble was used, but combining LSTM and XGBoost may be explored in the future.

Linear Regression performed poorly on high-variance dates such as holidays and promotions. It often failed to capture nonlinear interactions, leading to consistent underfitting in volatile periods. However, its low complexity and fast runtime made it a useful reference.

7. Limitations & Next Steps

Limitations include missing or incomplete holiday/event data and the lack of additional economic indicators. Future work includes integrating more external data sources, improving feature engineering, experimenting with hybrid models, and fine-tuning LSTM for edge deployment.

8. Appendix

• **XGBoost Parameters:** max_depth=6, learning_rate=0.1, n_estimators=200, subsample=0.8, colsample_bytree=0.8

• **LSTM Parameters:** Bidirectional LSTM (1 layer, 64 units), embedding_dim=10, dense_units=128, dropout=0.2, optimizer=adam, loss=mse, batch_size=32, epochs=50

• **Prophet Parameters:** n_changepoints=10, seasonality_mode='multiplicative', yearly_seasonality=True, holidays included, added regressors=['onpromotion', 'is_weekend', 'is_holiday', 'promo_last_7_days', 'sales_lag_7', 'rolling_mean_7']