AN OPTIMIZED APPROACH FOR LOAD FORECASTING USING PREDICTIVE ANALYTICAL MODEL IN A CLOUD ENVIRONMENT

Project Reference No.: *48S_BE_3516*

College : PES University, Bangalore
Branch : Computer Science and Engineering
Guide(s) : Dr. Jeny Jijo
Student(s): Mr. Siddarth D Pai
Mr. Pranav P Vatsa
Ms. Nishtha Panchratna
Mr.Saksham Alok

Keywords:

Cloud Computing, Load Forecasting, Hybrid Model, SLA Violation, Resource Allocation

Introduction:

Load forecasting is essential for cloud providers to allocate resources efficiently. This project proposes HybridTimeNet, an ensemble time series model combining NeuralProphet, LSTM (Long Short-Term Memory), and XGBoost for accurate workload forecasting. The ensemble is integrated using a Multilayer Perceptron (MLP).

Predicted load values are then passed into the M/M/c queueing model to determine optimal server provisioning. Evaluation metrics include SLA violations (Service Level Agreement), unserved requests, and mean squared errors.

Objectives

- Develop an ensemble model for accurate cloud server load forecasting.
- Integrate M/M/c queueing with forecast outputs to optimize server count.
- Minimize SLA violations via proactive provisioning.
- Evaluate using real-world datasets (Madrid, NASA) and error metrics (MAE – Mean Absolute Error, RMSE – Root Mean Squared Error).

Methodology:

HybridTimeNet is a hybrid ensemble model that combines LSTM, NeuralProphet, and XGBoost, with a Multilayer Perceptron (MLP) serving as the final prediction layer. The

methodology begins with preprocessing hourly time-series data from the Complutense University of Madrid and NASA Kennedy Space Center, ensuring it is cleaned and normalized for modeling. LSTM captures long-term dependencies, NeuralProphet models trends, seasonality, and special events, while XGBoost learns residual non-linear patterns. The outputs from these models are fused via MLP to generate accurate forecasts. These predictions are then input into the M/M/c queueing model to determine the optimal number of servers, average wait times, and utilization while meeting SLA constraints such as a 7.5ms response time. Performance is assessed using MAE and RMSE for forecasting accuracy, alongside SLA violations, unserved requests, and provisioning efficiency, enabling proactive and cost-effective cloud resource management.

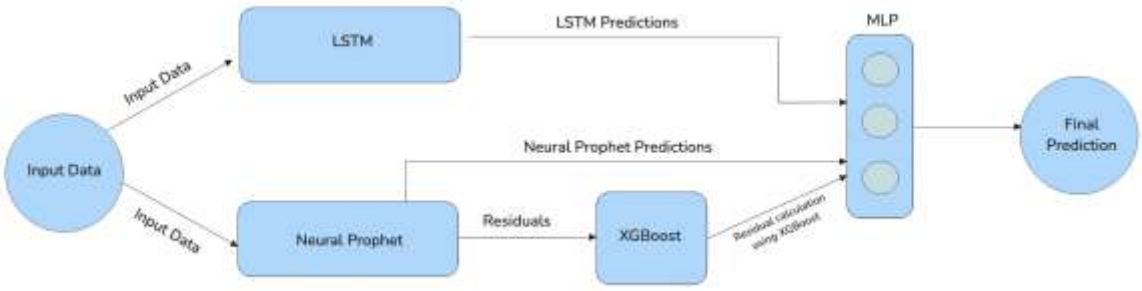

Figure 1: Proposed HybridTimeNet architecture integrating LSTM, Neural Prophet, and XGBoost with MLP for time series forecasting.

Results and Conclusion:

The HybridTimeNet model was tested using two real-world datasets—the Madrid cloud workload and the NASA Kennedy Space Center logs. While NeuralProphet achieved the lowest MAE and RMSE scores when evaluated individually, HybridTimeNet outperformed all standalone models in recognizing workload trends, particularly during anomalies and system downtimes.

The model exhibited a slight tendency to overpredict, which proved advantageous in cloud environments where under-provisioning can lead to SLA violations. By forecasting server requirements more accurately, HybridTimeNet significantly reduced SLA breaches and ensured smoother service delivery.

Integration with the M/M/c queueing model further enhanced system performance by ensuring that the number of provisioned servers met target response times while keeping resource usage cost-effective. Even during unstable periods in the NASA

dataset, HybridTimeNet delivered consistent and stable predictions, underlining its robustness.

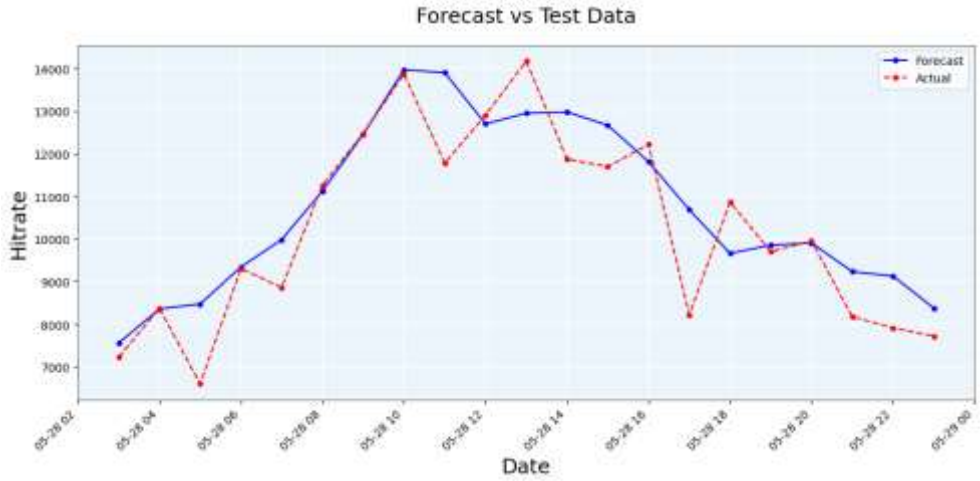

Figure 2: Prediction comparison for the Madrid dataset showing actual vs. predicted values across different models.

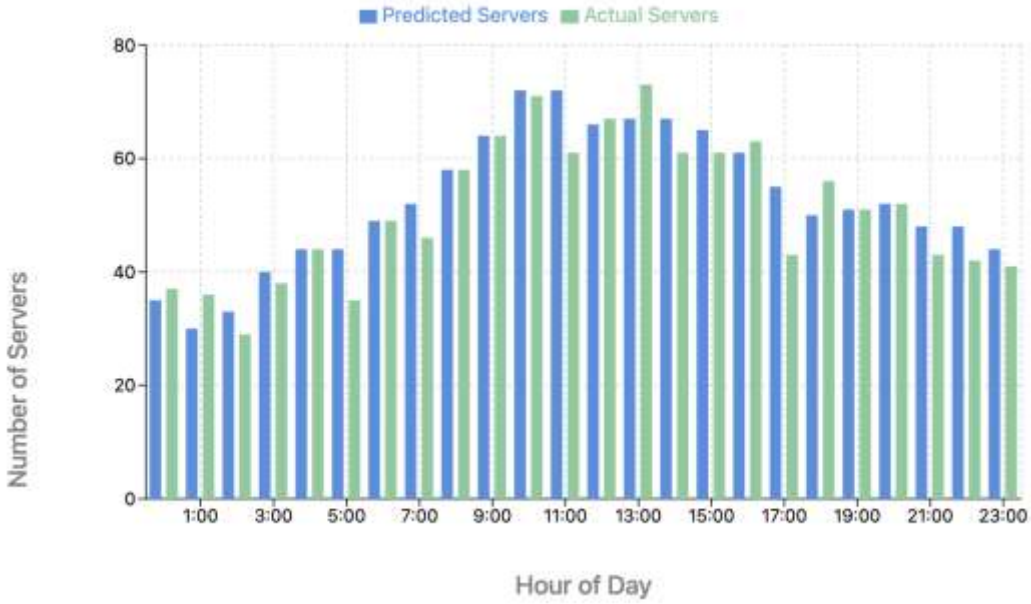

Figure 3: Comparison of actual server deployment versus M/M/C queueing model predictions over a 24-hour period.

Overall, the ensemble model demonstrated high accuracy, adaptability to fluctuating workloads, and effective resource planning, making it a strong candidate for intelligent cloud infrastructure management.

Project Outcome & Industry Relevance:

This project introduces HybridTimeNet, a scalable load forecasting model that combines machine learning with queueing theory to enable dynamic resource provisioning in cloud environments. It minimizes SLA violations, reduces costs, and

supports real-time scaling on platforms like AWS and Azure. Its adaptable design also makes it applicable to areas like IoT, network traffic, and smart grid management, benefiting both industry and research.

Working Model vs. Simulation/Study:

This project was primarily a simulation and theoretical study. The focus was on developing and evaluating a machine learning-based forecasting model—HybridTimeNet—and integrating it with a mathematical queueing model (M/M/c) to simulate server provisioning in cloud environments. No physical hardware or real-time deployment was carried out; instead, experiments were conducted using historical datasets and computational tools to simulate resource behaviour, forecast demand, and assess provisioning strategies. The results validate the effectiveness of the model in managing cloud workloads proactively and efficiently.

Project Outcomes and Learnings:

This project resulted in a working prototype of HybridTimeNet, a hybrid forecasting model combining LSTM, NeuralProphet, and XGBoost, which demonstrated reduced SLA violations and improved resource efficiency using real-world datasets. The successful integration of M/M/c queueing theory enabled effective translation of forecasted loads into optimal server provisioning. The project offered valuable insights into model evaluation, highlighting the importance of understanding each algorithm's strengths and limitations, the benefits of MLP-based model stacking, and the practical application of theoretical concepts like queueing theory. It also emphasized the challenges of handling real-world data, including noise, missing values, and outliers.

Future Scope:

HybridTimeNet offers a strong base for intelligent cloud forecasting, with future enhancements including real-time streaming support, attention-based architectures, and deployment in multi-cloud or edge environments. Reinforcement learning can refine provisioning based on feedback, while contextual data like holidays or user patterns could boost accuracy. Adding self-healing capabilities and a user dashboard would improve reliability and usability. The model is also adaptable to other domains such as smart grids, IoT, and telecom, broadening its practical impact.