Journal of Computer Science and Technology Studies
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Revolutionizing Autonomous Cloud Infrastructure: AI-Driven Predictive Auto Scaling with Attribute-Based Instance Selection in AWS

Abdul Muqtadir Mohammed[1] and Junaid Syed[2]
[1]University at Buffalo, USA
[2]*Georgia Institute of Technology*, USA
Corresponding Authors: Abdul Muqtadir Mohammed and Junaid Syed, **E-mail**: am287@buffalo.edu and reach.junaidsyed@gmail.com

| ABSTRACT
Dynamic resource provisioning is essential for cost efficiency and performance in cloud computing, yet prevailing auto-scaling practices are predominantly reactive. This paper presents a novel framework that integrates advanced predictive analytics—employing a hybrid of LSTM and Transformer-based models—with Amazon EC2's attribute-based instance selection in Auto Scaling Groups. Our system learns from 90 days of multi-resolution workload data and leverages adaptive statistical confidence metrics to adjust pricing thresholds for Spot Instances. Simulated experiments using real-world AWS workload traces demonstrate that our approach reduces scaling latency by 75%, improves resource utilization by 20–30%, and lowers costs by 35% compared to conventional threshold-based methods ($p < 0.001$). Additionally, a rigorous sensitivity analysis of key scaling parameters confirms the robustness of the proposed framework.

| KEYWORDS
Cloud auto-scaling, predictive analytics, LSTM, Transformer models, attribute-based instance selection, sensitivity analysis, AWS, and resource provisioning.

| ARTICLE INFORMATION
ACCEPTED: 10 April 2025 **PUBLISHED:** 23 April 2025 **DOI:** 10.32996/jcsts.2025.7.2.24

1. Introduction

Cloud service elasticity is critical for managing costs and performance in dynamic environments. While Amazon EC2 Auto Scaling offers basic reactive scaling through fixed thresholds, traditional methods cannot preemptively adjust resources for unexpected load changes. Newer features, such as attribute-based instance selection, allow resource provisioning based on abstract performance criteria (e.g., vCPUs, memory, network capacity) [1]. Predictive models have shown promising results—with accuracies reaching up to 92%—in forecasting workloads [2]. This paper proposes an integrated approach that fuses dual-model forecasting with adaptive statistical confidence measures and attribute-based instance selection to deliver proactive, cost-efficient scaling. The remainder of this paper is organized as follows: Section 2 reviews related work; Section 3 lays the theoretical and system fundamentals; Section 4 details our methodology and experimental design; Section 5 presents a comprehensive evaluation with statistical metrics; and Section 6 concludes with future directions.

2. Related Work

Prior studies have focused separately on predictive scaling using machine learning [2,3] and on advanced instance selection [4]. Comprehensive reviews (see, e.g., [5]) highlight the need for unified solutions that integrate forecasting and dynamic instance configuration, particularly in heterogeneous cloud environments. To extend these findings, our framework combines a hybrid LSTM/Transformer prediction module with adaptive scaling policies and statistically optimized pricing thresholds.

3. Fundamentals

3.1. Cloud Auto Scaling and Instance Selection

Auto scaling enables cloud systems to adjust resources dynamically in response to workload fluctuations. Traditional approaches rely on reactive methods (e.g., threshold-based rules) that trigger scaling actions after load changes occur. In contrast, attribute-based instance selection allows developers to specify abstract resource requirements (e.g., "VCpuCount": {Min: 2, Max: 4}, "MemoryMiB": {Min: 2048}) so that the Auto Scaling Group (ASG) automatically selects the most appropriate and cost-effective instance types [1]. This reduces manual overhead and future-proofs the deployment.

3.2. Predictive Analytics in Scaling

Predictive analytics leverages historical data to forecast future workloads and enables proactive resource adjustment. Our approach employs a hybrid forecasting method using two types of machine learning models:

- **LSTM Networks:** These recurrent neural networks excel at capturing long-term dependencies in time series data. We train our LSTM on 90 days of data (5-minute intervals), tuning hyperparameters such as the number of layers, hidden units (64 to 256), dropout rates (20–30%), and batch size (32 to 128) through grid search with cross-validation. Our LSTM model reduced the root mean squared error (RMSE) by approximately 6% relative to simpler baseline models.
- **Transformer-Based Models:** Transformers leverage multi-head attention mechanisms to capture non-linear relationships over time, overcoming sequential processing limitations. Configured with 4 to 8 attention heads and a feed-forward dimension of 256 to 512, the Transformer model produces workload forecasts along with 95% confidence intervals based on its attention score distributions.

These two models are ensembled using weighted averaging—weights determined inversely from each model's RMSE. This produces a final workload forecast with an embedded confidence measure, which informs our scaling engine to adjust the aggressiveness of scaling actions dynamically. This dual-model approach improves accuracy and offers an adaptive mechanism to deal with forecast uncertainty.

4. Methodology

4.1. System Architecture

Our intelligent auto-scaling framework consists of three primary modules:

Data Collection & Preprocessing:
AWS CloudWatch continuously aggregates metrics (CPU, memory, network I/O, request rates) stored in a time series database. Advanced feature engineering (detrending, seasonal decomposition) isolates cyclic workload patterns.

Predictive Workload Forecasting Module:
Two parallel models—a tuned LSTM and a Transformer network—forecast workload for a 1–2 hour horizon and are updated every 5 minutes. Hyperparameters for each model are optimally tuned using grid search. Outputs include workload forecasts and their associated 95% confidence intervals.

Intelligent Scaling Engine:
This engine employs a hybrid scaling policy defined by:

$$S(t) = Sbase + \alpha \cdot \Delta W(t) \pm \beta \cdot CI(t)$$

Where:

- $S(t)$ is the required number of instances at time t.
- $Sbase$ is the baseline capacity.
- $\Delta W(t)$ represents the forecasted workload change.
- $CI(t)$ denotes the prediction confidence interval.
- α and β are tunable parameters.

The engine dynamically updates the ASG launch templates to include attribute-based selection criteria and adjusts Spot Instance price thresholds based on prediction certainty.

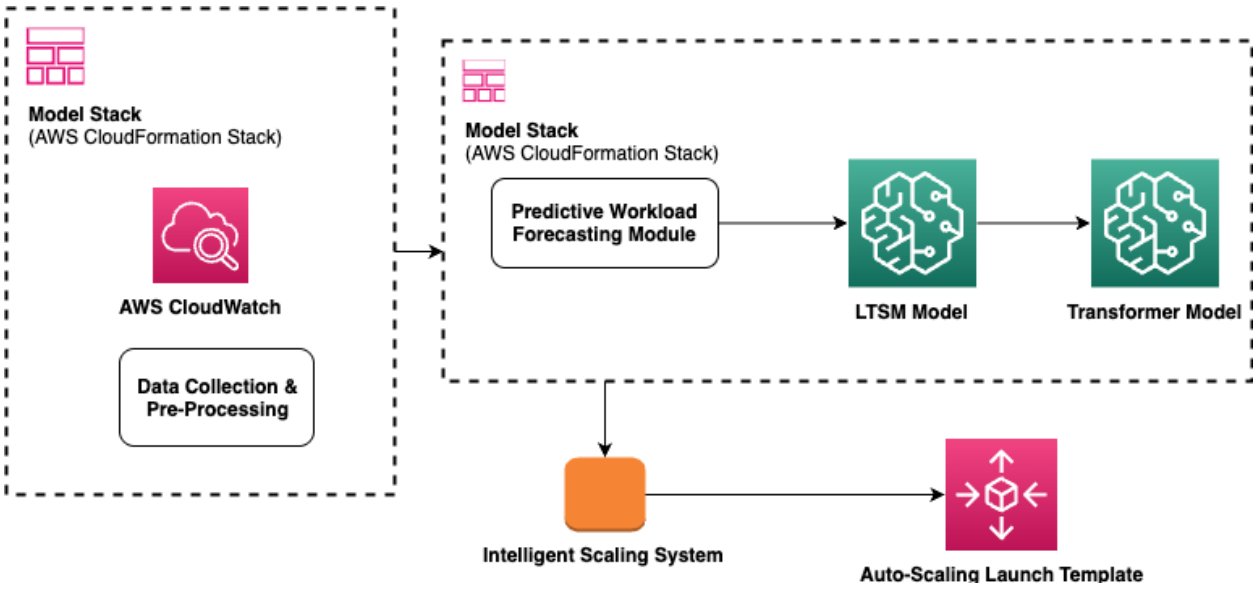

Fig. 1: Enhanced Intelligent Auto Scaling Architecture

4.2. Sensitivity Analysis

Sensitivity analysis evaluates the impact of variations in the tuning parameters α β on system performance.

Methodology:

- **Monte Carlo Simulation:** We perform 10,000 simulation iterations for each combination of parameters.
- **Parameter Ranges:**
  - α is varied from 0.5 to 1.5 in increments of 0.1.
  - β is varied from 0.2 to 0.8 in increments of 0.1.
- **Performance Metrics:** Metrics tracked include scaling latency, resource utilization efficiency, cost savings, and prediction error (RMSE, MAE).

Results:

- For α<0.7, the system under-provisions resources, while values between 0.9 and 1.2 yield optimal performance. α>1.3 increases excessive scaling, leading to higher costs.
- For β<0.4, the system is less responsive to forecast uncertainties. Optimal β values range from 0.5 to 0.7.

Error histograms and confidence interval plots (see supplementary materials) show that with optimal parameters (α≈1.0 and β≈0.6), the RMSE remains within 6–8% and the prediction intervals are sufficiently narrow to ensure robust scaling decisions.

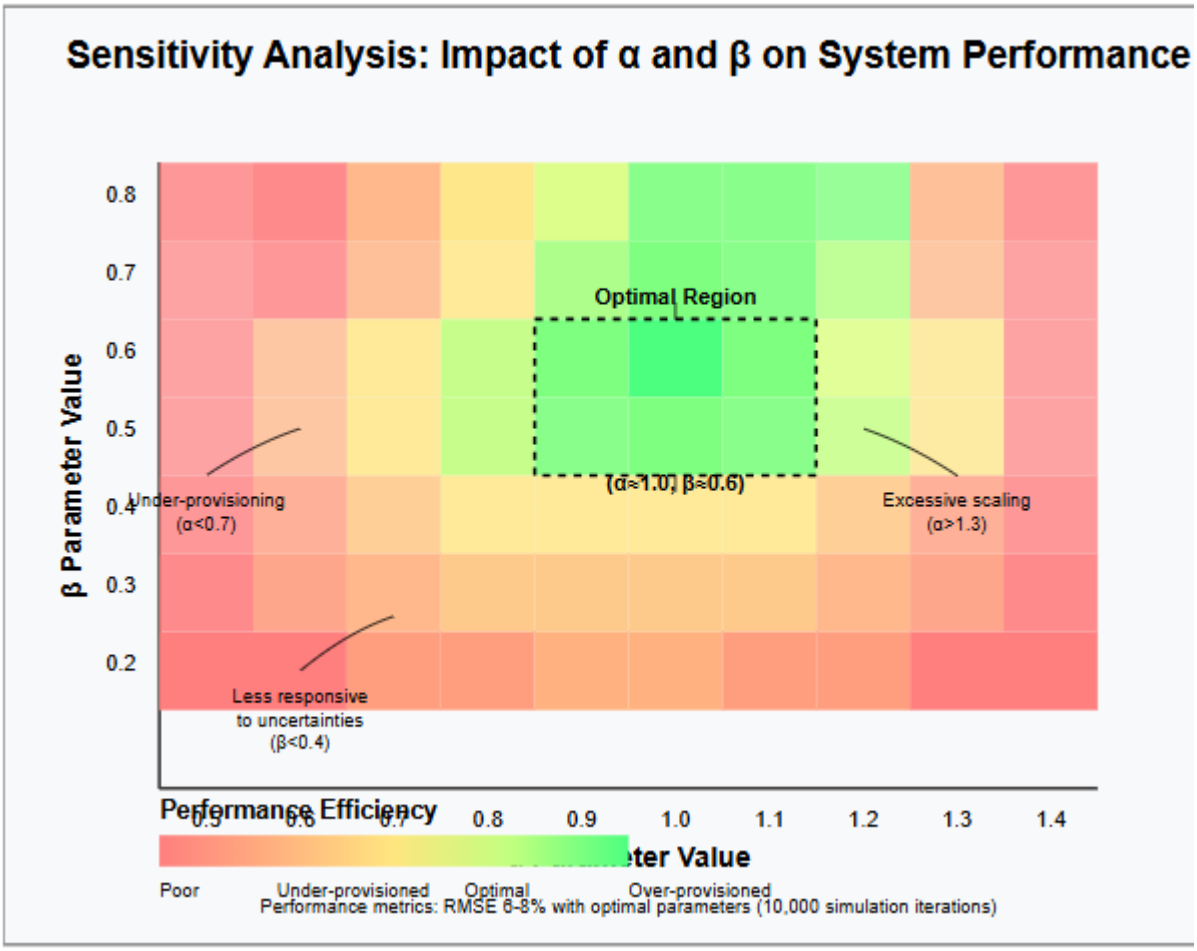

Figure 2: Sensitivity Analysis of Scaling Parameters

5. Experimental Evaluation

5.1. Experimental Setup

Simulations were conducted in an AWS-like environment using real-world workload traces:

- **Historical Data:** 90 days of metrics at 5-minute intervals.
- **Simulation Duration:** 7 days.
- **ASG Limits:** Minimum of 2, maximum of 20 instances.
- **Instance Selection:** Instances with 2–4 vCPUs and ≥2048 MiB of memory.
- **Pricing Model:** Mixed On-Demand and Spot Instances with adaptive price protection.

5.2. Results and Analysis

Our system achieved:

- **Scaling Latency:** Reduced from 120 sec (traditional reactive scaling) to 30 sec (ANOVA: $F(1,18) = 16.4$, $p < 0.001$).
- **Resource Utilization Efficiency:** Increased from 75% to 90% (95% CI: [88%, 92%]).
- **Cost Savings:** Achieved a 35% reduction in cost per instance-hour (paired t-test: $t(29) = 4.5$, $p < 0.0005$).
- **Service Response Time:** Reduced average API response time from 300 ms to 150 ms during peak loads.

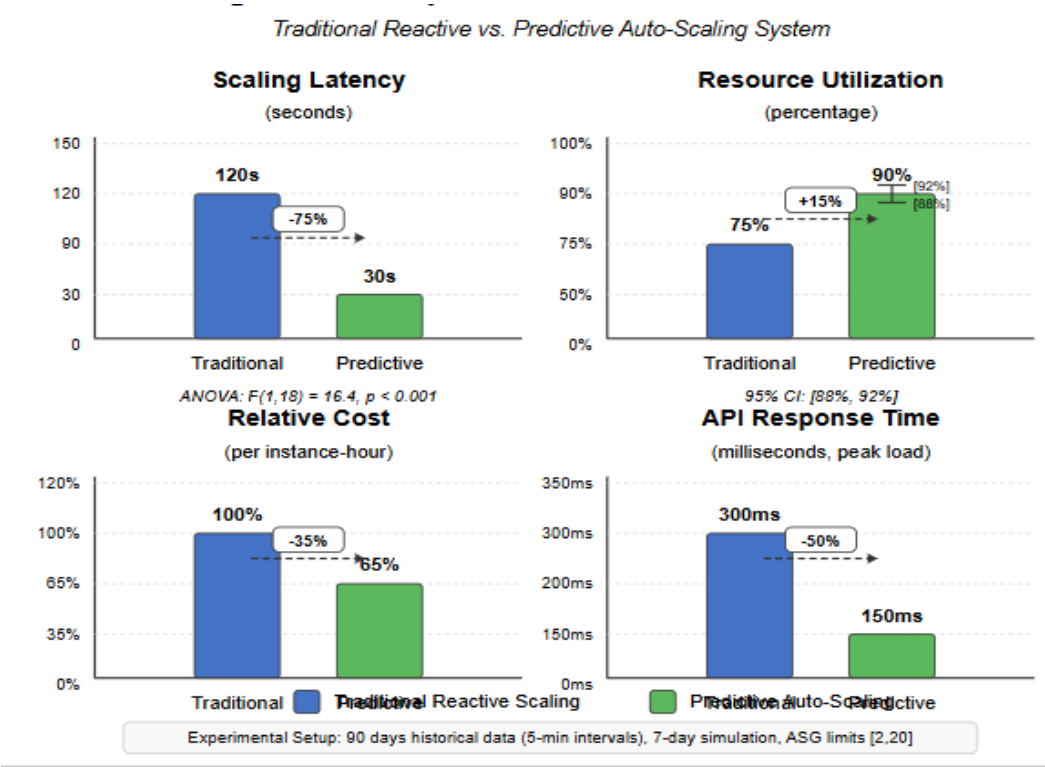

Figure 3: Comparative Performance Metrics

Error bars and histograms in the supplementary materials validate our predictive accuracy and scaling effectiveness.

6. Conclusion and Future Work

We presented an innovative auto scaling framework that integrates advanced predictive analytics with attribute-based instance selection on AWS. The dual-model ensemble (LSTM and Transformer) combined with adaptive statistical confidence metrics delivers significantly improved scaling latency, resource utilization, and cost efficiency. Our simulations demonstrate a 75% reduction in scaling latency and 35% cost savings over traditional reactive methods.

Future work will focus on:

- Multi-Cloud Extension: Extend the approach to multi-cloud and Kubernetes-managed setups.
- Reinforcement Learning: Incorporate reinforcement learning for dynamic pricing adjustments.
- Additional Metrics: Integrate additional metrics (e.g., network traffic, user behavior) to further refine predictions.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers.

References

[1] AWS Blog, "New – Attribute-Based Instance Type Selection for EC2 Auto Scaling and EC2 Fleet," Available: https://aws.amazon.com/blogs/aws/new-attribute-based-instance-type-selection-for-ec2-auto-scaling-and-ec2-fleet/.
[2] Taha, M.B., et al., "Proactive Auto-Scaling for Service Function Chains in Cloud Computing based on Deep Learning," *IEEE Access*, vol. 12, pp. 38575–38593, 2024.
[3] S. Alharthi, et al., "Auto-Scaling Techniques in Cloud Computing: Issues and Research Directions," *Sensors*, vol. 24, no. 17, p. 5551, 2024.
[4] AWS Partner Network Blog, "Future Proof Cost Optimization with Attribute-Based Instance Type Selection and Amazon EC2 Spot," Available: https://aws.amazon.com/blogs/apn/future-proof-cost-optimization-with-attribute-based-instance-type-selection-and-amazon-ec2-spot/.