

An automated approach for selecting the most suitable AWS EC2 instance for software projects prior to deployment

Kumarage K.S.D.^{1*}, Lakshan W.D.D.² and Hewaratna A.I.³

¹Department of Computing and Information Systems,
Faculty of Computing, Sabaragamuwa University of Sri Lanka

²Department of Software Engineering, Faculty of Computing,
Sabaragamuwa University of Sri Lanka

³AI and Technology, Boolean Labs

*ksdilrukshi@std.appsc.sab.ac.lk

Cloud computing is now a staple of a modern software development with provisioning of scalable and on-demand resources. Amazon Web Services (AWS), the cloud services leader, offers great variety of types of the Elastic Compute Cloud (EC2) instances, and it makes the decision about the adequate instance to be chosen before deployment a difficult task. A poor decision will result in a reduction in performance, redundant costs, and slow deployment times. Available tools like AWS Compute Optimizer and Instance Type Finder are based on CloudWatch telemetry which means that applications must be deployed first and then it can recommend something, which in turn adds additional cost and delays. This study suggests the pre-deployment EC2 instance recommendation framework which is independent of cloud-generated telemetry. The proposed system analyzes local workload behavior using a hybrid prediction approach that combines machine learning with rule-based reasoning. System-level and application-level profiling tools are used to collect performance metrics from representative workloads, including CPU-intensive, memory-intensive, I/O-intensive, and mixed workloads. The collected metrics, such as CPU usage, memory consumption, disk throughput, and network activity, are preprocessed and transformed into structured feature vectors. In parallel, an EC2 instance specification dataset is constructed using official AWS documentation. A supervised XGBoost classifier is then applied to map workloads to the most suitable EC2 instance family, with initial labels generated through rule-based feature matching. After identifying the instance family, a secondary rule-based decision layer selects the specific instance type based on vCPU requirements, memory demand, network performance, and EBS usage patterns. To improve transparency and user understanding, a Retrieval-Augmented Generation (RAG) module retrieves relevant AWS documentation to support each recommendation.

Keywords: *Cloud computing; AWS EC2; Instance selection; Workload profiling; Automation*