Artificial Intelligence and Digital Technology

Article

Automation and Life Cycle Management Optimization of Large-Scale Machine Learning Platforms

Yixian Jiang[1,*]

[1] Information Networking Institute, Carnegie Mellon University, Pittsburgh, PA, 15213, USA
* Correspondence: Yixian Jiang, Information Networking Institute, Carnegie Mellon University, Pittsburgh, PA, 15213, USA

Abstract: With the continuous deepening of intelligent technology, machine learning technology has been adopted in many fields, making the management and maintenance of large machine learning systems particularly complex. Automated operations and optimization of the entire system lifecycle have become the core components for improving operational efficiency and reducing maintenance costs. This study aims to examine the architecture design and component functions of large-scale machine learning systems, and analyze the challenges encountered in current automation implementation, resource allocation, parameter optimization, and system maintenance, and propose corresponding improvement measures. These measures include the refinement of processes, intelligent management of resources, establishment of an automated model evaluation system, and the creation of an intelligent operation and maintenance system. These suggestions will help improve the operational performance and management level of the system, and create more efficient and scalable machine learning application platforms for various enterprises.

Keywords: large-scale machine learning platform; automated management; life cycle management; resource optimization; intelligent operation and maintenance

Published: 23 May 2025

1. Introduction

Driven by the rapid advancement of big data and intelligent technology, machine learning technology has become the engine of innovation in various industries. Machine learning systems are responsible for processing massive amounts of data and training advanced models, and their operational efficiency directly determines the speed and quality of model development. With the continuous expansion of these systems, the traditional mode of relying on manual operation and management can no longer keep up with the development needs of modern machine learning tasks. The optimization of automated operations and lifecycle management has become the core approach to improving system efficiency and reducing operational costs. Although some large-scale machine learning systems have achieved some results in automated management, there are still problems such as process blockage, unreasonable resource allocation, and lack of direction in hyperparameter adjustment. This study aims to examine the current status of automation and lifecycle management in large-scale machine learning systems, analyze the challenges they face, and provide targeted improvement solutions to help achieve intelligent and efficient system management [1].

2. Basic Architecture of Large-Scale Machine Learning Platforms

The large-scale platform architecture of machine learning is the fundamental framework for ensuring efficient operation of machine learning tasks, covering key modules such as data storage layer, computing layer, task scheduling layer, and service layer, as shown in Figure 1. The main function of the data storage layer is to meet the storage and

management needs of large amounts of data, ensuring efficient data reading and writing. The computing layer is responsible for executing the computation process of model training and prediction, usually built on distributed computing architectures (such as Spark, TensorFlow, PyTorch, etc.), and completes complex computing tasks through the collaboration of numerous nodes. The task scheduling layer is responsible for coordinating various machine learning tasks, such as data cleaning, model construction, parameter optimization, etc., to ensure that tasks proceed in sequence and handle dependencies between tasks. The service layer serves as a bridge between the platform and external systems, providing application programming interfaces and service channels to enhance the platform's scalability and adaptability [2].

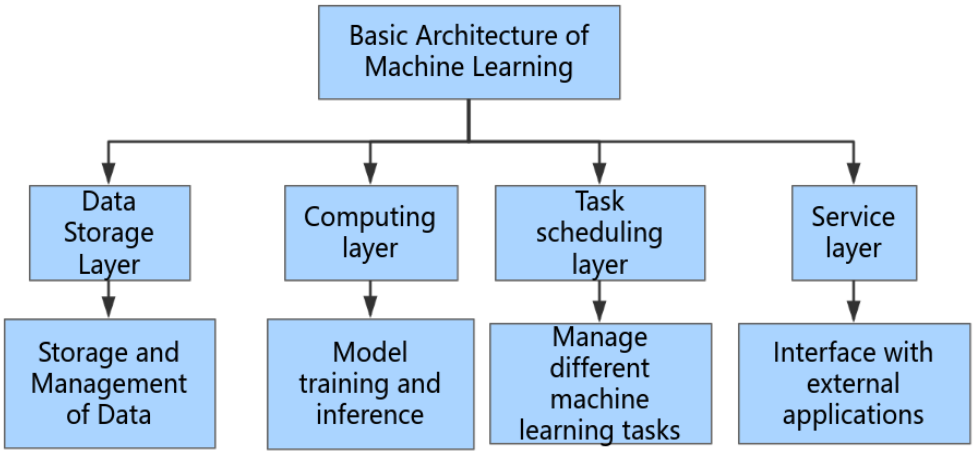

Figure 1. Core Framework of Machine Learning.

3. The Current Status of Automation and Lifecycle Management in Large-Scale Machine Learning Platforms

3.1. Poor Connection of Automated Processes

On numerous large-scale machine learning platforms, automated operational processes play an important role in improving job efficiency and reducing labor costs. However, many platforms still face many challenges in integrating automated processes. The automation links between independent modules have not been unified and standardized, and there is a long lag in the conversion between data preprocessing and model training, which makes it difficult to achieve rapid and effective integration from data storage to model training. In addition, automation tools cannot completely replace manual operations in specific stages. In response to unexpected situations or the need to adjust model parameters, due to the lack of a coordinated management and monitoring mechanism, some automation tasks cannot report faults in a timely manner during the execution phase, which causes delays in the entire workflow and slows down overall work efficiency. The automation platform lacks the feature of self-adjustment, and in the face of unexpected abnormal situations, the system is not equipped with appropriate processing strategies [3]. The lack of intelligent feedback systems in various stages of the automation process makes it difficult to effectively address issues in a timely manner, which has a negative impact on the overall operational efficiency of the platform.

3.2. Inefficient Dilemma of Resource Utilization

In large-scale machine learning systems, how to efficiently utilize resources is a key factor in determining system performance and cost control. There is still potential for improvement in the efficiency of resource allocation and management. Although most systems rely on distributed computing architectures to meet large-scale data processing and computing needs, resource management still relies on fixed configuration schemes and

lacks the ability to dynamically adjust on demand. This leads to some computing nodes facing resource overload during peak task periods, while failing to effectively utilize these resources during low task periods, resulting in unnecessary resource waste. The utilization efficiency of storage resources also needs to be improved. Data storage is generally based on distributed file systems, where data is scattered across numerous nodes and storage devices, which invisibly increases the latency of data reading and transmission. With the growth of data scale, the efficiency and access speed of data management have become bottlenecks that constrain system performance [4]. Especially in application scenarios that require frequent data exchange and real-time model inference, the mismatch between storage and computing resources further increases the burden on the system.

3.3. Blind Hyperparameter Tuning without Scientific Strategy Guidance

During the training phase of machine learning models, optimizing hyperparameters is the core step in improving model performance. At present, many large-scale machine learning systems are facing challenges in hyperparameter optimization, such as purposelessness and lack of systematic optimization strategies. The optimization of hyperparameters generally adopts methods such as trial and error or grid scanning. Although these methods are easy to operate, they come with high computational costs, especially when dealing with highly complex models and massive datasets, where time and resource consumption are particularly significant. Lack of precise optimization solutions often leads to inefficient repetitive experiments, resulting in unnecessary consumption of a large amount of computing resources. At present, many platforms are not equipped with intelligent hyperparameter optimization tools, which cannot flexibly adjust optimization strategies based on the unique properties and specific task requirements of different models. Although some cutting-edge techniques, such as Bayesian optimization and genetic algorithms, have been proposed to improve optimization processes, their application has not been widely promoted, and they often have compatibility issues with platform automation tools, affecting their integration and practical application effectiveness [5].

3.4. High Complexity of Operation and Maintenance

When operating a massive machine learning platform, system maintenance and management is an extremely complex and arduous task. The personnel responsible for system maintenance must face thousands of computing units, storage systems, and data transmission processes, which makes monitoring the system's condition and performance extremely difficult. The conventional maintenance method relies too much on manual operation and intervention, which is difficult to meet the requirements of large platforms for rapid response and high availability. With the continuous growth of platform scale, the burden of manual maintenance is increasing, and maintenance personnel have to spend huge time and energy to solve system failures, overcome performance limitations, and coordinate task allocation, all of which seriously affect the stability and long-term development of the platform. The fault detection and repair function of the system is not agile enough. In massive machine learning systems, numerous components and complex external correlations can trigger a series of failures due to individual component errors, ultimately leading to the paralysis of the entire system. Conventional operation and maintenance methods and processes are often insufficient to quickly identify hidden dangers, and after a fault occurs, the repair steps are lengthy and time-consuming, which poses a serious threat to the stability of the system.

4. Optimization Strategies for Automation and Lifecycle Management of Large-Scale Machine Learning Platforms

4.1. Process Lean Reshaping

In machine learning systems, the key objective of pursuing process lean is to eliminate unnecessary steps, improve the efficiency of each step, achieve optimal resource utilization, and reduce time and economic costs. Process lean focuses on simplifying each operational step to the extreme, ensuring that each link operates in an optimized mode, and eliminating all forms of resource waste and efficiency bottlenecks. The essence of process lean is the deep optimization of existing process execution efficiency, while also involving the redesign and integration of various task units to enhance the collaborative efficiency of the entire system.

In the operation of machine learning systems, process lean requires a reassessment of processes at multiple stages, including data collection, cleaning, preprocessing, model training, and inference. Traditional linear processes face issues such as repetitive operations, frequent manual interventions, and ambiguous task dependencies, all of which can have a negative impact on the overall efficiency of the process. By reconstructing the process, unnecessary steps can be reduced, process automation can be achieved, synchronization between computation and data processing can be enhanced, and efficient linkage between various modules can be ensured.

Taking the model training process as an example, there are multiple models that need to be trained in parallel on the platform, and the training time of each model depends on the data input and model complexity. We can lean the process by introducing mathematical modeling and optimization algorithms. There are n models that need to be trained, each with a training time of ti. The training process of the models is parallel, and the computational resource constraint is R. Assuming that each model occupies resources of ri, the total training time can be minimized by optimizing the resource allocation of the models. By optimizing algorithms such as minimizing the objective function:

$$T = \max_{1 \le i \le n} \left(\frac{ti}{ri} \right) \tag{1}$$

After fine optimization of the process, the efficiency of training has been improved, and the management of the entire lifecycle has been deeply optimized. By compressing the implementation duration of each stage, the overall response rate of the platform has been enhanced, and the idle rate of system resources has been reduced. The refined management of this process has improved the automation level of the platform, making the entire cycle of machine learning more efficient and accurate.

4.2. Fine Grained Management of Intelligent Resources

How to scientifically allocate and efficiently utilize resources in building large-scale machine learning systems has become the core link in maintaining system performance and economy. By adopting intelligent resource management strategies and relying on algorithms, data analysis, and predictive algorithms, we are able to flexibly and dynamically adjust and optimize computing resources, storage resources, and network resources. This strategy is significantly different from the fixed resource allocation in the past. Intelligent resource management dynamically adjusts the resource allocation plan by monitoring the system's operating load, task nature, and resource consumption in real-time, ensuring that all tasks run in the optimal environment and improving the performance indicators and resource utilization efficiency of the entire platform.

When multiple machine learning tasks are executed concurrently on the platform, and these tasks have different resource requirements, traditional management methods still allocate resources according to preset configurations and rules. In contrast, intelligent resource management can adjust resource allocation in real-time based on the actual needs of tasks, demonstrating higher flexibility. If task A requires a large amount of computing

resources and consumes resources quickly, while task B only performs lightweight inference calculations, the system can adjust the resource priority based on the differences in task types. The following is a hypothetical resource scheduling scenario (Table 1):

Table 1. Resource Scheduling.

Task ID	Resource requirements (CPU core)	Task type	Allocate resources (CPU core)	Resource consumption rate (unit time)	Actual usage of resources (CPU core)
Task A	16	model training	16	1.5	15
Task B	4	model reasoning	4	0.5	3
Task C	8	data processing	6	1.2	5

In the above table, the resource requirements for task A and task B are 16 and 4 CPU cores, respectively. Task A consumes resources at a higher rate due to model training, while task B has lower resource consumption for inference tasks. Through intelligent resource refinement management, the system dynamically adjusts the resource allocation of task A and task B, adjusts according to the actual needs of the tasks, optimizes overall resource utilization, and avoids resource waste.

After careful resource optimization management, the platform's resource utilization efficiency has increased, the processing time of computing tasks has decreased, and operating costs have also been reduced. In machine learning platforms that handle massive amounts of data, the efficiency of automation and resource lifecycle control is enhanced through intelligent and refined resource management, ensuring the platform's adaptability to changing and complex workloads.

4.3. Building an Automated Model Evaluation Framework to Screen the Optimal Version

Faced with the increasing trend of model types and their complexity, relying on manual model screening methods can no longer meet the urgent need for efficient selection of top model versions. Establishing an automated model evaluation system is particularly crucial. The system can automatically evaluate and compare numerous model versions based on established evaluation criteria and parameters, and select the most outstanding model version. This system reduces manual involvement and improves the efficiency and accuracy of evaluation, ensuring that the platform can lock in the most superior performance model in the shortest possible time and fully tap into the potential of the model in practical application scenarios.

Typically, automated model evaluation systems involve multiple steps such as dataset segmentation, model training, performance evaluation, and model version control. By implementing consistent and standardized evaluation of numerous model versions, the system can accurately identify the performance advantages and disadvantages between different versions, assisting developers in quickly locating the most ideal model version in a large-scale model library. In a certain classification task, we hope to evaluate the performance of the model through indicators such as accuracy, precision, and recall. We can define an evaluation function $f(M)$ to measure the performance of model M on a given dataset:

$$f(M)=\alpha \cdot \text{accuracy}(M)+\beta \cdot \text{precision}(M)+\gamma \cdot \text{recall}(M) \quad (2)$$

In formula (2), α, β, and γ represent the weights of accuracy, precision, and recall, respectively. By evaluating different model versions M_1, M_2, ..., M_n as described above, we can calculate the comprehensive evaluation value of each model and select the model with the highest evaluation value as the optimal version. In a certain experiment, we evaluated three models and obtained the following evaluation results (Table 2):

Table 2. Model Evaluation Results.

Model version	Accuracy rate	Accuracy rate	Recall rate	Comprehensive evaluation value
Model M_1	0.85	0.80	0.75	0.795
Model M_2	0.88	0.83	0.78	0.825
Model M_3	0.84	0.79	0.76	0.792

After calculating the overall ratings of each model, we determine that Model M_2 is the most outstanding version, enhancing the operational efficiency of the system and ensuring the stability and reliability of the model in practical environments. The introduction of an automated evaluation system has shortened the time required for model development and improved the overall efficiency of machine learning project implementation, providing a more robust and efficient service guarantee for the platform.

4.4. Building an Intelligent Operations Center

Faced with the continuous increase in scale and difficulty of machine learning tasks, traditional operational methods are no longer sufficient to meet the platform's requirements for near-zero downtime, rapid response, and automatic recovery capabilities. The center has adopted cutting-edge AI technology, automated tools, and in-depth data analysis to build a comprehensive and dynamically updated operation and maintenance management system, ensuring that the platform can maintain stable and efficient operation even in changing and complex production environments.

By continuously tracking key performance parameters of the system, including processor utilization, information transmission rate, and storage space conditions, the intelligent operation and maintenance center can actively detect potential fault points and performance obstacles in the system. This platform utilizes advanced artificial intelligence algorithms to predict fluctuations in system load and automatically perform resource scheduling optimization operations, effectively preventing the risk of system crashes or performance degradation. For example, in the process of processing large amounts of data, the intelligent operation and maintenance center can track the resource consumption status of each computing node in real time, automatically identify and adjust those overloaded nodes, achieve load balancing, and prevent individual nodes from causing system bottlenecks.

Thanks to the operation of the intelligent operation and maintenance center, the level of automated operation and maintenance has been significantly improved, and the full lifecycle management of the system has been further optimized. This intelligent operation and maintenance method not only enhances the stability and reliability of the system, reduces the frequency of manual intervention, but also ensures the sustainable and efficient operation of the platform under high-intensity machine learning tasks.

5. Conclusion

This article explores the current application status of automation and full lifecycle management on large-scale machine learning platforms, and proposes practical and feasible improvement measures for the identified problems. By finely restructuring the homework process, the platform can more efficiently respond to various stages of work and prevent resource loss caused by process disconnection. The intelligent resource management mechanism helps to ensure the full utilization of computing resources while reducing cost expenditures and improving the flexibility and efficiency of resource allocation. Building an automated model evaluation system and intelligent operation and maintenance center will further consolidate the robustness and scalability of the platform. Adopting these suggestions can not only significantly improve the overall performance of the platform, but also bring more intelligent and efficient response methods to the management of machine learning tasks. With technological advancements, large-scale machine learning platforms in the future will tend towards higher levels of automation and intelligence, providing stronger technological support for business and research fields.

References

1. C. Ma, M. Jaggi, F. E. Curtis, N. Srebro, and M. Takáč, "An accelerated communication-efficient primal-dual optimization framework for structured machine learning," Optim. Methods Softw., vol. 36, no. 1, pp. 20–44, 2019, doi: 10.1080/10556788.2019.1650361.
2. K. Swanson, P. Walther, J. Leitz, S. Mukherjee, J. C. Wu, R. V. Shivnaraine, and J. Zou, "ADMET-AI: A machine learning ADMET platform for evaluation of large-scale chemical libraries," Bioinformatics, vol. 40, no. 7, p. btae416, Jul. 2024, doi: 10.1093/bioinformatics/btae416.
3. X. Zhang, S. Jiang, X. Wang, K. Duan, Y. Xiao, D. Xu, and P. O. De Pablos, "Promoting sales of knowledge products on knowledge payment platforms: A large-scale study with a machine learning approach," J. Innov. Knowl., vol. 9, no. 3, p. 100497, 2024, doi: 10.1016/j.jik.2024.100497.
4. M. Wang, W. Fu, X. He, S. Hao, and X. Wu, "A survey on large-scale machine learning," IEEE Trans. Knowl. Data Eng., vol. 34, no. 6, pp. 2574–2594, Jun. 1, 2022, doi: 10.1109/TKDE.2020.3015777.
5. J. Oukili, J. Kumar, J. Burren, S. Cochran, M. Bubner, D. Nasyrov, and B. Farmani, "Large-scale industrial deployment of machine learning workflows for seismic data processing," First Break, vol. 41, no. 12, pp. 57–64, 2023, doi: 10.3997/1365-2397.fb2023101.