

MUTUAL BENEFIT IN AI AND CONTAINERS ORCHESTRATION INTEGRATION

Lytovchenko Vladyslav

graduate of the Faculty of Computer Science
Kharkiv National University of Radio Electronics, Ukraine

Scientific adviser: Olga Gubaryeva

PhD, Associate Professor of the Foreign languages Department
Kharkiv National University of Radio Electronics, Ukraine

In today's software development industry, container orchestration and artificial intelligence play an essential role in making applications more efficient and scalable. Container orchestration allows you to efficiently deploy and manage applications using containers, while AI opens up new horizons for automation and data analysis. The interaction of these two areas can lead to synergy, performance improvement and optimization.

One of the main challenges in container orchestration is scaling and managing large numbers of containers [1]. Companies face challenges in scheduling and placing containers on hosts as well as in load balancing and optimal use of resources. Another challenge is managing and optimizing network connections between containers. The use of artificial intelligence can help solve these problems by automating decision making and streamlining container orchestrators.

Artificial intelligence can help automatically scale applications based on load analysis and resource requirement prediction [2]. It can also be used to optimize the distribution of containers on hosts, load balancing and anomalies detection in application performance. Moreover, we can orchestrate artificial intelligence to enhance its work and improve the efficiency of calculations.

The use of artificial intelligence in container orchestration has a number of examples [3]. In practice, artificial intelligence (AI) can help with container orchestration in several ways. Here are some examples:

- Autoscaling. AI can be applied for automatic scaling of containerized applications based on load changes. Machine learning algorithms can analyze historical application load data and predict future load. The AI can automatically adjust the number of containers, scaling the application up or down for optimal performance according to these predictions.
- Resource Optimization. AI can help optimize the use of resources in container orchestration. AI can analyze the characteristics and load of each host, as well as the requirements and priorities of applications, and suggest the optimal placement of containers, given the available resources.
- Perform load balancing. AI can assist in automatic load balancing between containers. Machine learning algorithms can analyze the current load of each container and optimize load balancing to equalize resource usage. AI can also optimize the performance of each container, taking into account application preferences and priorities.
- Automatically detect and respond to changes. AI can help automatically detect changes in the container environment and take appropriate action. For example, AI can analyze container performance and availability metrics and, if a problem is detected, automatically restart the container, move it to another host, or take other actions to keep the application running smoothly.

On the other hand, containerization and orchestration can also be used to improve AI performance. By orchestrating and deploying AI models and algorithms in containers, we can achieve high flexibility and scalability. Orchestration tools such as Kubernetes allow you to scale your infrastructure automatically in order to provide fast and responsive access to computing resources for processing data and executing complex AI algorithms.

This offers a number of benefits. First, it makes it easier to develop and test AI models, as containers provide an isolated environment in which users can run different versions of models and their components. This allows developers to rapidly prototype and deploy new models.

Secondly, containerization and orchestration allow AI applications to scale efficiently. The use of orchestrators allows you to scale lots of containers automatically depending on the load. As a result, it can ensure high system performance and responsiveness. This is especially useful when processing large amounts of data or when working with distributed computing.

In addition, containerization and orchestration make it easier to deploy AI solutions across multiple environments. Containerized AI models can be easily transferred between different cloud platforms or local servers. This positively impacts flexibility and mobility of AI applications, which is especially important when used in different environments or when developing multi-platform solutions.

Applying AI to improve container orchestration can be achieved by integrating tools such as Kubernetes, TensorFlow, and Prometheus [4]. For example, Kubernetes can be used to manage and automatically orchestrate containerized applications including some AI applications, while artificial intelligence implemented with TensorFlow can analyze load data and predict the required resources to optimally scale applications. At the same time, Prometheus can collect performance and resource usage metrics that should be used by artificial intelligence to make decisions about load balancing or anomaly detection and remediation. This interaction of tools and artificial intelligence allows you to create smart and autonomous systems that can adapt to changing conditions and optimize the performance of containerized applications.

References:

1. Kanjilal J. How to tackle container orchestration challenges | TechTarget. IT Operations. URL: <https://www.techtarget.com/searchitoperations/tip/How-to-tackle-container-orchestration-challenges> (date of access: 18.05.2023).
2. Zhong, Zhiheng & Xu, Minxian & Rodríguez, María & Xu, Cheng-Zhong & Buyya, Rajkumar. (2022). Machine Learning-based Orchestration of Containers: A Taxonomy and Future Directions. ACM Computing Surveys. 54. 10.1145/3510415.
3. DevOps Use Cases for AI-Assisted Kubernetes. DevOps.com. URL: <https://devops.com/devops-use-cases-for-ai-assisted-kubernetes/> (date of access: 18.05.2023).
4. Deploying a TensorFlow Model to Kubernetes. Medium. URL: <https://medium.com/geekculture/deploying-a-tensorflow-model-to-kubernetes-444b36672207> (date of access: 18.05.2023).