

AIOps: Real-World Challenges and Research Innovations

Yingnong Dang
Microsoft Azure
Redmond, WA, U.S.A.
yidang@microsoft.com

Qingwei Lin
Microsoft Research
Beijing, China
qlin@microsoft.com

Peng Huang
Johns Hopkins University
Baltimore, MD, U.S.A
huang@cs.jhu.edu

Abstract—AIOps is about empowering software and service engineers (e.g., developers, program managers, support engineers, site reliability engineers) to efficiently and effectively build and operate online services and applications at scale with artificial intelligence (AI) and machine learning (ML) techniques. AIOps can help improve service quality and customer satisfaction, boost engineering productivity, and reduce operational cost. In this technical briefing, we first summarize the real-world challenges in building AIOps solutions based on our practice and experience in Microsoft. We then propose a roadmap of AIOps related research directions, and share a few successful AIOps solutions we have built for Microsoft service products.

Keywords—AIOps, DevOps, Software Analytics

I. WHY AIOPS?

Software industry has been transformed from delivering boxed products to releasing services (including online services and applications). Accordingly, the way services are built and released is different from traditional boxed products, which brings up the importance of operational efficacy for services. DevOps [4], a method for facilitating continuous development and release of services, has been widely adopted. With the proliferation of cloud computing, the scale and complexity of services have increased dramatically. The ever-increasing scale and complexity of services pose significant challenges to software and service engineers on efficiently and effectively building and operating services with DevOps. In this context, the term AIOps came out from Gartner [1] to address the DevOps challenges with AI.

There is no widely agreed-upon definition of AIOps yet. In general, AIOps is about empowering software and service engineers to efficiently and effectively build and operate services that are easy to support and maintain by using artificial intelligence and machine learning techniques. The value of AIOps can be significant: ensuring high service quality and customer satisfaction, boosting engineering productivity, and reducing operational cost.

II. OUR VISION OF AIOPS

We envision that AIOps will help achieve the following three goals, as shown in Figure 1.

High service intelligence. An AIOps-powered service will have timely awareness of changes from multiple aspects, e.g., quality degradation, cost increase, workload bump, etc. An AIOps-powered service may also predict its future status based on its historical behaviors, workload patterns, and underlying

infrastructure activities, etc. Such self-awareness and predictability will further trigger self-adaption or auto-healing behaviors of a service, with low human intervention.

High customer satisfaction. A service with built-in intelligence can understand customer usage behavior and take proactive actions to improve customer satisfaction. For example, a service can automatically recommend tuning suggestions to a customer for her to obtain best performance (e.g., adjusting configuration, redundancy level, resource allocations); a service may also know that a customer is suffering from a service quality issue and proactively engage with the customer and provide a solution or workaround, instead of reactively responding to customer complaints through human support.

High engineering productivity. Software and service engineers have powerful tools to effectively and efficiently build and operate services through the whole lifecycle of services. Engineers and operators are relieved of tedious tasks like (1) manually collecting information from various sources for investigating an issue; (2) fixing repeated issues. Engineers and operations are also powered by AI/ML techniques to learn the patterns of system behaviors, predict the future of the service behaviors and customer activities for making necessary architecture changes and service adaption strategy changes, etc.

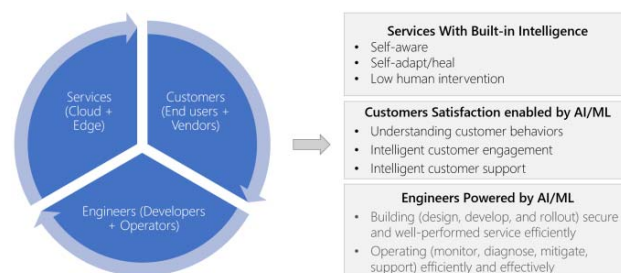


Figure 1: Our Vision of AIOps

III. REAL-WORLD CHALLENGES

The software industry is still at the early stage of innovating and adopting AIOps solutions. On the one hand, the community just started to realize the importance of AIOps. As IDC predicted [2], by 2024, 60% of firms will have adopted ML/AI analytics for DevOps, accelerating software delivery and improving quality, security, and compliance via data integration, auto triggers, and predictive ALM (Agile Lifecycle Management). On the other hand, building AIOps solutions and adopting them in real-world settings are still challenging today from both

technical and non-technical perspectives. Based on our practice and experience in Microsoft, we summarize the major challenges of building AIOps solution as follows.

A. Gaps in innovation methodologies and mindset

Gap in innovation methodologies. Building AIOps solutions requires holistic thinking and sufficient understanding about the whole problem space, from business value and constraints, data, models, to system and process integration considerations, etc. Today, there lacks innovation methodologies that can guide people in different disciplines (e.g., business stake holders, engineers, data scientists) to build AIOps solutions.

Difficulty of the mindset shift. The essential methodology of AIOps solutions is to learn from history for predicting future and to identify patterns from large amount of data. Such mindset is substantially different from the traditional engineering mindset (e.g., digging into individual cases by looking at bug reproducing steps and detailed logs, which is inefficient or even infeasible in large-scale service scenarios). Meanwhile, there is a strong AI-solves-everything mindset, which is not a realistic expectation.

B. Engineering changes needed to support AIOps

Traditional engineering best practices do not fit the needs. Building AIOps solutions needs significant engineering efforts. AIOps-oriented engineering is still at a very early stage, and the best practice/principles/design patterns are not well established in the industry yet. For example, the AIOps engineering principles should include data/label quality monitoring and assurances, continuous model-quality validation, and actionability of insights.

The data quality and quantity available today do not serve the needs of AIOps solutions. Although major cloud services today collect terabytes and even petabytes of telemetry data every day/month, there still lacks representative and high-quality data for building AIOps solutions. A continuous improvement of data quality and quantity is necessary. The method of instrumentation and collection of telemetry also needs to be re-visited (e.g., principled instrumentation for AIOps solutions instead of ad-hoc logging for debugging a few issues).

C. Difficulty on building ML models for AIOps

Building ML/AI model for AIOps solutions has unique challenges that are not always seen in other ML/AI scenarios. The challenges for building supervised machine learning model for AIOps include: no clear ground truth labels or huge manual efforts to obtain high quality ones (extremely imbalance, too small amount, high degree of noise, etc.)[6], complex dependencies/relations among components/services[7], complicated feature engineering effort due to the high complexity of cloud service behaviors, continuous model update and online learning, and the risk of service interruptions caused by misbehaving ML models.

In many AIOps scenarios, due to the difficulty of obtaining label data, only unsupervised or semi-supervised machine learning models is feasible. For example, detecting anomalous behavior of services [8]. It is difficult to have enough labels to learn “what is abnormal” of a service, because almost every

service is ever evolving with the change of customer needs and underlying infrastructure changes. The difficulty of building high-quality unsupervised models lies in the complexity of the internal logic of services and the huge volume of the telemetry data that needs to be analyzed.

IV. RESEARCH INNOVATIONS ON AIOps

AIOps can be viewed as a cross-disciplinary research and innovation area. We believe there is a long way to go for the industry to achieve our AIOps vision. We will focus on the technical innovations that are needed to achieve our AIOps vision. Meanwhile, AIOps related research is not entirely new. For example, many of the research works on software analytics [3] can be viewed as AIOps innovations.

A. Cross-disciplinary research

AIOps innovations involve research areas including (but not limited to) system design, software engineering, big data, artificial intelligence, machine learning, distributed computing, and information visualization. One example is that, system researchers need to work with machine learning experts to build services with self-awareness and auto-adaptation [5].

B. Close collaboration between academia and industry

AIOps innovations call for a close partnership between academia and industry. The real pain of software and service engineers needs to be well understood. The running behaviors of real-world services need to be researched. While the proliferation of open-source software enables easy access to source code for the research community, it is far from enough for AIOps innovations.

V. OUTLINE OF TECHNICAL BRIEFING

In this technical briefing, we will present our position in AIOps with great details: (1) talking about the motivation and emerging importance of AIOps; (2) describing the real-world challenges of building AIOps solutions based on our experience in Microsoft; (3) introducing a set of sample AIOps solutions that have successfully benefited Microsoft service products; (4) sharing some learnings from our AIOps practice.

REFERENCES

- [1] “Everything you need to know about AIOps”, from <https://www.moogsoft.com/resources/aiops/guide/everything-aiops/> (retrieved as of Feb. 12, 2019)
- [2] IDC FutureScape, “Worldwide CIO Agenda 2019 Predictions”, doc #US44390218, October 2018
- [3] D. Zhang, S. Han, et. Al., “Software Analytics in Practice”, IEEE Software, 2013
- [4] G. Kim, P. Debois, et al., “The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations”, IT Revolution Press, Oct. 2016
- [5] P. Huang, C. Guo, et. Al., “Capturing and Enhancing In Situ System Observability for Failure Detection”, In Proceedings of OSDI 2018
- [6] Y. Xu, K. Sui, et. Al., “Improving Service Availability of Cloud Systems by Predicting Disk Error”, in Proceedings of USNIX ATC 2018
- [7] Q. Lin, K. Hsieh, et. Al., “Predicting Node Failure in Cloud Service Systems”, In proceedings of FSE 2018
- [8] Q. Lin, J. Lou, et. Al., “iDice: Problem Identification for Emerging Issues”, In Proceeding of ICSE 2016