AIOps – A Systematic Literature Review

Seminar paper

Reiter, Lena, FH Wedel, Wedel, Germany, winf104561@fh-wedel.de

Abstract

It is becoming increasingly complicated for IT operations to manage the vast amounts of data generated by modern IT systems. The use of Artificial Intelligence for IT operations (AIOps) counteracts this problem. This study aims to determine the state of the art of literature on AIOps. Therefore, I systematically evaluate the research findings in a multi-stage process. The results of 15 selected papers are compiled, analyzed, and thematically structured. The state of the studies is discussed with respect to the functions, implementations, tools, benefits, challenges, and trends of AIOps. The paper highlights anomalies and gaps in current literature. The results of this paper provide a synthesized overview for further research.

Keywords: Literature Review, AIOps, Artificial Intelligence for IT operations

Table of Contents

1 Introduction

The increasing drive for digitization in companies and the dynamics of the cloud are presenting IT operations with challenges that traditional management concepts can no longer cope with (Andenmatten 2019; Levin et al. 2019; Li et al. 2020; Masood and Hashmi 2019). Gartner first introduced Artificial Intelligence for IT operations (AIOps) in 2016 to regulate difficulties DevOps could not address (Dang et al. 2019; Shen et al. 2020). AIOps is a promising technology to counteract the increasing complexity in IT management using AI and Big Data (Levin et al. 2019; Notaro et al. 2021; Paradkar 2020). AIOps platforms are defined as software systems that ingest data from a variety of sources to perform comprehensive analysis. The data analyses are used to discover patterns that enable the prediction of incidents (Andenmatten 2019). As a result, problems in IT operations can be identified and resolved more efficiently with less expenditure of time and money (Dang et al. 2019; Gulenko et al. 2020; Levin et al. 2019). There is significant and increasing interest in AIOps (Andenmatten 2019; Gulenko et al. 2020; Masood and Hashmi 2019), but there is a lack of systematic analysis of its wide-ranging and versatile literature. Therefore, I have conducted a comprehensive analysis of the existing research to provide detailed information about the state of the art of literature on AIOPs. The individual facets of AIOps are analyzed, structured, and classified. Similarities, differences, and gaps in studies are pointed out. The literature review can be used as a basis for further research, especially offering broad added value for practical purposes. This paper highlights what to focus on during research to avoid frequent misconceptions and maintain a clear view of the bigger picture. To this end, the following section first describes how the relevant literature is filtered. Second, the results of the selected research are compiled, analyzed, and thematically and categorically classified. This is followed by a discussion of the extracted findings.

2 Literature Review

Following the approach recommended by Tranfield et al. (2003), the implementation was divided into three stages: planning the review, conducting it, and reporting the findings. During the planning process, I established the following criteria for study inclusion and exclusion.

Inclusion:

- The paper is about the use of AI in IT operations
- The concepts of the paper relate in some way to IT management
- The paper deals with the topic of failure management

Exclusion:

- The paper is not written in English or German.

The second phase involves conducting a review. According to Tranfield et al. (2003), a systematic search includes the identification of keywords and a convenient search term. Therefore, I searched for *AIOps* on the *Google Scholar* database. Different keywords emerged for the terms *AI* and *IT operations*.

AI keywords:

- AI
- Artificial intelligence
- Machine learning

IT operations keywords:

- IT
- Failure management
- Data analytics

This resulted in the following search string: *AIOps AND ("AI" OR "artificial intelligence" OR "machine learning" OR "deep learning") AND ("IT" OR "failure management" OR “data analytics”).*
The string was used in six databases: EBSCO Host (Business Source Complete), ScienceDirect, IEEE Xplore, ACM Digital Library, GBV, and Google Scholar, producing a total of 664 articles (cf. Table 1).

Database	Number of query results
EBSCO Host (Business Source Complete)	8
ScienceDirect	5
IEEE Xplore	21
ACM Digital Library	34
Gemeinsamer Bibliotheksverbund (GBV)	9
Google Scholar	587
Σ	664

Table 1. Number of query results about AIOps in different databases (Oct. 2021)

Subsequently, I ran a filtering process consisting of five steps (cf. Figure 1). First, all duplicates of the query results were screened out. After that, all articles whose relevance could be excluded based on the abstract were removed. I used the previously defined criteria to guide this process. Then, all scientifically inapplicable articles were filtered out, followed by all further sources consisting of professional journals. The full texts of the remaining 65 studies were reviewed, and papers that did not meet the established criteria were removed. Following the approach recommended by Webster and Watson (2002), I back reviewed the references of the identified articles. As a result, I included one additional study.

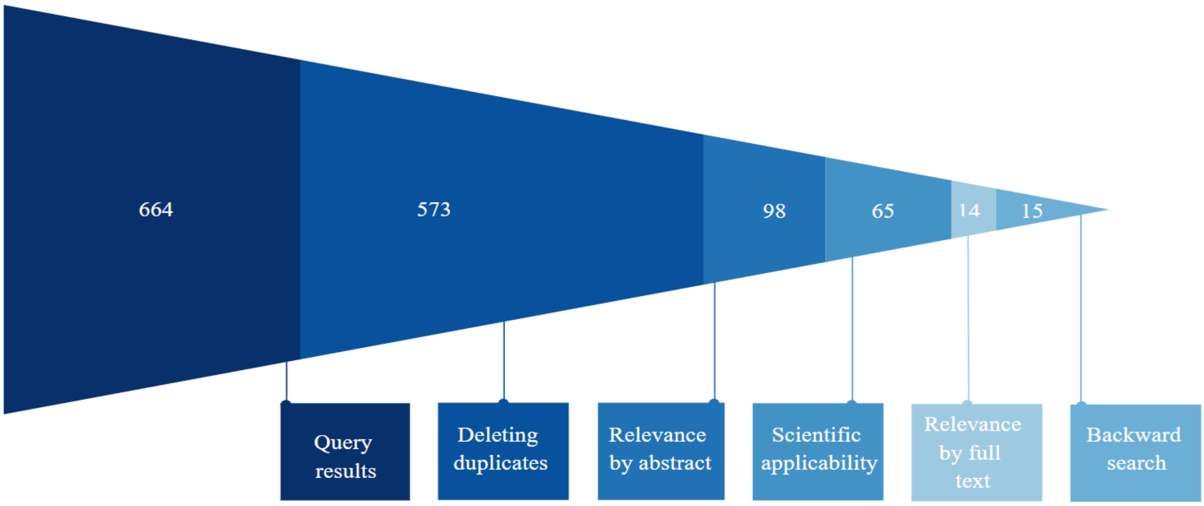

Figure 1. Number of query results on AIOps through the review process (Oct. 2021)

3 Results

The third phase of the literature search, according to Tranfield et al. (2003), consists of evaluating and reporting the results.

Noticeably, most articles are written from a practical perspective (80%). In addition, it is striking that all selected articles were published after 2018. Furthermore, the general number of publications on AIOps is increasing annually (cf. Figure 2). About 75% of all articles on AIOps found in the different databases were published in the last two years (2020–2021).

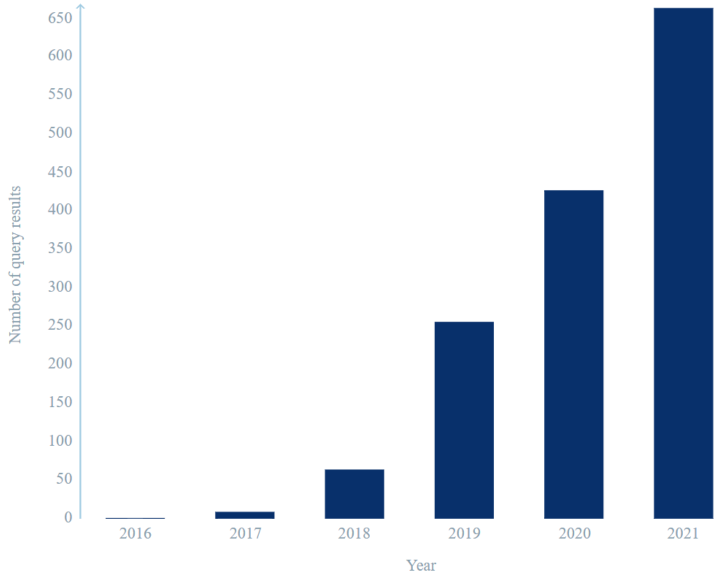

Figure 2. Number of found query results about AIOps (Oct. 2021)

AIOps is a wide-ranging and cross-functional topic. To evaluate the content of the selected studies, I have divided the subject into three main categories and seven subcategories. The first categorization deals with the functions of AIOps, which can be divided into proactive and reactive failure management (Notaro et al. 2021). The second refers to practical adoption, which I thematically separated into implementation and tools. The third part evaluates the findings, which are divided into benefits, challenges, and trends. Following the recommendation of Webster and Watson (2002), I organized the results into a concept matrix (cf. Appendix 1). I then analyzed the content of the articles for each subcategory and summarized it in the following paragraphs.

3.1 Proactive Failure Management

In the literature, proactive failure management is divided into (1) fault prevention and (2) fault prediction (Notaro et al. 2021).

The articles indicate that the prediction of failures in IT operations is mainly used in the context of storage systems. According to Wang and Zhang (2020) and Li et al. (2020), errors in this field can incur enormously high costs and severe damage. The articles point out that due to the large amount of data as well as the complexity of the systems and failure symptoms, errors can be solved most efficiently with AIOps. Wang and Zhang (2020) emphasize that hard disk malfunctions are one of the main reasons for faults in storage systems. They propose an ensemble learning model that uses machine learning to predict errors in hard drives. This allows hard disks to be replaced before they succumb to major damage. Li et al. (2020) describe the use of AIOps in cloud object storage services for fault prediction. The authors explain that the use of random forest-based models deployed with the oversampling technique they presented achieves the most reliable and efficient results.

In the literature, fault prevention means that measures are taken to correct errors before they have an impact (Li et al. 2020). An example of this function of AIOps would be the early replacement of hard disks based on the previously mentioned failure predictions (Wang and Zhang 2020).

3.2 Reactive Failure Management

In the literature, reactive fault management is mainly divided into three functions of AIOps: (1) detection, (2) root cause analysis, and (3) self-healing (Notaro et al. 2021).

A distinction is made between fault and anomaly detection. While fault detection identifies problems, anomaly detection distinguishes between standard and deviating behaviour. Deep learning models are often used in the literature as approaches to define patterns and detect anomalies from a large amount of data (Nedelkoski et al. 2019; Wang et al. 2019). The articles explain that anomaly detection plays a fundamental role in AIOps. Thus, the efficiency of other functions depends on the quality of the anomaly detection. According to Chen et al. (2020) and Levin et al. (2019), fault detection is about identifying those detected anomalies that could lead to problems. The articles state that it is essential to identify the most important anomalies to avoid an unnecessary number of alarms. There are three steps in this filtering process. First, all redundant alarms that can be traced back to the same cause should be aggregated. Furthermore, Chen et al. (2020) recommend an error classification so that fatal faults can be distinguished from less urgent ones. In addition, the definition of the resources to be monitored also plays a role in efficient fault detection. According to the authors, it is advisable to distribute the monitoring to different resource levels, despite the resulting high rate of alarms. The literature also points out that one must take care to avoid filtering out fatal faults during the aggregation process. Chen et al. (2020) refer to a cascading clustering algorithm proposed by He et al. (2018) to identify relevant problems in fault detection. Nevertheless, the literature concludes that an overflow of warnings cannot be completely ruled out despite filtering processes.

In root cause analysis, the aim is to find and analyze the causes and effects of the detected problems. With the help of AI, patterns for the reasons behind the errors can be derived (Andenmatten 2019; Chen et al. 2020; Paradkar 2020; Shen et al. 2020). This allows the identification of certain trends in root cause analysis, which can be used for error prevention or troubleshooting. According to Shen et al. (2020) and Chen et al. (2020), the complexity of this AIOps function can vary. Some causes are more frequent and can be detected quicker by using machine learning, while others are one-time occurrences. The analysis becomes particularly difficult when there are multiple causes for disturbances or when the error is deeply rooted in the system. Chen et al. (2020) and Levin et al. (2019) point out that one must take care to place the problem in context. This includes determining the exact time of the event, identifying the geographic location, classifying the fault in terms of severity, and listing all components affected by the fault. Furthermore, Chen et al. (2020) explain that errors can often be grouped and hierarchically classified. The authors mention causes that induce *parent faults*, which in turn can be assigned to *children faults*. The literature emphasizes that a precise analysis accelerates and facilitates troubleshooting and system recovery. The causes of the detected anomalies can be efficiently analyzed by correlation analysis, expert experience, and statistics (Levin et al. 2019; Chen et al. 2020).

The self-healing function uses AI to resolve a failure and return the system to normal. Chen et al. (2020), Dang et al. (2019), Gulenko et al. (2020), and Shen et al. (2020) describe different types of self-healing. On the one hand, the authors describe remediation where the human initiates a process suggested by the AI to fix a problem. On the other hand, the authors mention a higher level of automation, where the machine rather than the human makes the decision to initiate a specific self-healing action. Gulenko et al. (2020) aim to more precisely describe the degree of problem processing automation in six stages. Shen et al. (2020) point out that there are also faults that require manual error handling, for example because hardware components must be replaced. Moreover, in the literature, the use of AIOps is partly limited to the previously presented functions, leaving out self-healing.

3.3 Implementation

The literature provides various roadmaps and implementation guidance for AIOps. Levin et al. (2019) have summarized the implementation in four stages.

The first phase involves data acquisition. Andenmatten (2019) points out that all data from different departments should be taken into consideration to monitor the system. Shen et al. (2020) categorize the data to be obtained into six distinct groups: metrics, logging, tracing, configuration data, workflow data, and multimedia data. The authors recommend first obtaining an overview of all available data to be able to integrate all relevant data into the AIOps solution.

In the second phase, the collected data is processed, cleaned, and prepared for analysis. The articles consider this stage to be particularly elaborate. Levin et al. (2019) emphasize that with the different tools available, it is important to ensure that the employees using them thoroughly understand their application and compatibility. Adaptability in data processing is another important success factor when implementing an AIOps solution. The analyses should work even with larger amounts of data, different data origin, or other data types (Levin et al. 2019).

The third phase is about defining the features of the data so that statistical and machine learning algorithms can be used. Shen et al. (2020) recommend the *Apache Spark* framework for feature generation.

Levin et al. (2019) state that in the last stage, the identified features are used to implement models for early error handling with artificial intelligence. The fourth phase also includes the visual preparation of the analysis results. The literature points out that the implementation requires experience in working with machine learning models. Each of the four phases poses diverse challenges, which is discussed in more detail in section 3.6.

3.4 Tools

In this section, we will look at the tools covered in the literature. For this, I have mapped the previously listed functions of AIOps with a focus on the tools mentioned by the authors (cf. Table 2) (Sen 2020). Andenmatten (2019) and Levin et al. (2019) emphasize that it is important to ensure that the employees using the tools know their adaptability and compatibility well. The authors affirm that the conclusions of the AI must always correspond to reality, even if changes occur in the system. Gulenko et al. (2020) and Li et al. (2020) consider it essential that the AIOps solutions are interpretable and explainable, even if the performance suffers. In the literature, it is pointed out that the appropriate use of the tools relieves employees of problems that would otherwise be very time-consuming. In this way, working hours can be used more efficiently and in a more targeted manner (Andenmatten 2019).

Tools	Fault Prevention	Fault Prediction	Detection	Root Cause Analysis	Self-Healing
SysTrack (Lakeside)	✓	✓	✓	✓	
StackState		✓	✓	✓	
AIops (Broadcom)			✓	✓	
BigPanda			✓	✓	
Dynatrace			✓	✓	
Optanix			✓	✓	
SL1			✓	✓	✓
Splunk			✓	✓	✓
Zenoss Cloud			✓	✓	✓

Table 1. Assignment of tools to AIOps functions.

3.5 Benefits

Chen et al. (2020) and Masood and Hashmi (2019) emphasize that AIOps reduces the time between fault detection and correction. According to the authors, the handling of errors is also significantly improved, the error rate decreases, and the service-level agreements (SLAs) are shortened. This has several advantages, such as improvements in performance, availability, stability, service quality, engineering productivity, customer satisfaction, and cost reduction (Andenmatten 2019; Dang et al. 2019; Gulenko et al. 2020; Levin et al. 2019).

3.6 Challenges

The challenges cited in the literature can be divided into three categories: data management, human interaction with AIOps, and implementation of artificial intelligence.

First, we will look at the data enabling the analyses and evaluations of the AIOps solution. Dang et al. (2019) and Levin et al. (2019) point out that existing data often is not designed for analytical purposes. When it comes to procuring the data, attention should be paid not only to quantity but also to quality. In addition, Lyu et al. (2021) state that the data can be unbalanced. More data is available in some areas of operations than others, which can lead to an imbalance in the accuracy of the analysis. Andenmatten (2019) state that the proper storage, protection, and retention of data are important factors for the success of AIOps. The literature recommends an iterative approach using snapshots, existing data, and calibrations to gradually gain insights (Levin et al. 2019).

The second category deals with the challenges that arise when humans interact with AIOps. The literature claims that the requirements for employees' skills, mindsets, and areas of activity change with the introduction of AIOps. The employees' manual activities tend in the direction of adaptation and auditing tasks. Moreover, dealing with AI differs from the classic approach of a developer. In machine learning, certain patterns are recognized from existing data sets, which enables the formulation of conclusions for future events; in traditional development, the focus is on concrete cases and distinctions (Andenmatten 2019; Dang et al. 2019). According to Dang et al. (2019), an unrealistic mindset about AI can also lead people to assume AI can solve everything. To counteract this, Gulenko et al. (2020) divide AIOps into different levels of automation. The literature indicates that a high level of automation with opaque decision-making can lead to more difficult interventions for humans when necessary. The authors indicate that it is, therefore, imperative that AIOps provides a transparent and manageable AI solution. Dang et al. (2019) point out that due to the novelty of this technology, there is still a lack of guidelines for developing AIOps solutions in IT operations.

In the third category, I will address the challenges related to the implementation of artificial intelligence in AIOps. According to Dang et al. (2019), the challenges of developing machine learning models for AIOps are different from the difficulties of other machine learning solutions. The article explains that the standard behavior defined in anomaly detection changes naturally over time. Thus, patterns that were previously considered abnormal become standard, for example, because customer behavior has changed over time. Dang et al. (2019) state that such conversions are not considered when implementing the ML models. Still, the reliability of the ML models is critical to the success of an AIOps solution. Therefore, maintainability, adaptability, and verifiability of the AI implementation is critical (Dang et al. 2019; Gulenko et al. 2020).

3.7 Trends

The literature forecasts that the use of AIOps solutions in IT companies will increase significantly in the future (Andenmatten 2019; Gulenko et al. 2020; Masood and Hashmi 2019). The authors assume that AIOps will become increasingly indispensable for companies. Andenmatten (2019) and Gulenko et al. (2020) even claim that IT operations will no longer be able to cope with the challenges of the future without AI. Notaro et al. (2021) point out that there is growing interest of IT management in AIOps, especially in the function of fault detection. Li et al. (2020) show that the focus could shift further toward self-healing when fault detection is applied successfully.

4 Discussion

The articles contain some commonalities, differences, and contradictions. The following is a discussion of the anomalies and gaps in the literature. In addition, the usefulness of this study, as well as the limitations, are described in more detail.

4.1 Striking features in the literature

AIOps was only introduced in 2016 (Dang et al. 2019; Shen et al. 2020). The novelty of AIOps could be the reason for the discrepancies in the articles. Certain terms are not clearly defined and delineated in the context of AIOps. For example, Paradkar (2020) uses fault prediction as an umbrella term for the individual functionalities, while other articles use it to mean only one specific feature of AIOps. Another example is the meaning of the abbreviation AIOps. According to Gulenko et al. (2020), AIOps stands for *AI-supported IT operations*, while Shen et al. (2020) defines it as *Algorithmic IT Operations*. In most literature, however, AIOps stands for *Artificial Intelligence for IT Operations*. Moreover, there is a lack of term delineations in the literature. The functions of AIOps are not officially defined, and they are named differently in the articles. For example, in some articles *remediation* is synonymous with *self-healing*, while other articles use it to mean different functions (Gulenko et al. 2020; Li et al. 2020; Masood and Hashmi 2019; Shen et al. 2019). In terms of content, however, no contradictions are found at the overlapping points. As seen in the literature review results (section 3), many similar findings can be attributed to different studies. In the function of self-healing, there is another striking aspect. Many authors are limited to the other functions of AIOps in reactive failure management (cf. section 3.2). Proactive fault management is also not described comprehensively in the articles. The focus in the literature is mainly on fault detection and root cause analysis. There is a need for additional research on the preventive and self-healing functions of AIOps. Regarding the scientific applicability of the papers, another gap becomes apparent. Restricting the research for AIOps to peer-reviewed articles hardly leads to any results.

4.2 Utility of this study

AIOps was introduced to keep pace with the growing challenges of IT operations (Dang et al. 2019; Shen et al. 2020). Therefore, many articles address how AIOps can be implemented in practice. This often involves the general conversion to AIOps, but other articles also describe how specific functions can be optimized (e.g., Nedelkoski et al. 2019). To classify the different requirements and desires for AIOps, this paper brings together the results of different works. This literature review provides a structuring and classification of the different aspects of AIOps, which enables more targeted research into individual fields. Furthermore, attention is drawn to the individual studies on this subject. This work allows an overview of the different areas of AIOps. It can, therefore, be used as a basis for further research to find information more efficiently. For practical purposes, the literature review provides three benefits. First, the classification of functions and benefits can be used to define the requirements for an AIOps platform. Second, the merging of the implementation steps from the literature can be helpful in the AIOps realization process. Third, the challenges and trends gathered from the literature can be used to work against certain problems in advance and avoid setbacks in AIOps implementation. Moreover, the above-mentioned emerging traps, trends, and elaborated tendencies can prevent deviation from the right path during the implementation. Thus, one can get an overview in advance of the requirements, individual phases, trends, and problems that may arise in the realization.

4.3 Limitations

This study has certain limitations. First, bias cannot be completely ruled out. Thus, some aspects or contradictions may be overlooked. In addition, the filtering process could lead to the exclusion of works that add value in certain subject areas. Further, due to the lack of peer-reviewed journals, the likelihood of having selected an inaccurate article increases. Nevertheless, this study attempts to overcome these limitations and provide a realistic account of the current state of the literature.

5 Conclusion

The literature review provides a holistic overview of the various facets of AIOps. Fifteen papers were selected, and the results of these studies were categorized into functions, practical adoption, and findings. Particularly striking is the increasing interest in the use of AIOps. There is a need for additional studies focused on preventive and self-healing functions. Furthermore, there are gaps in the delineation and definition of terms related to AIOps. This study provides insight into categories of this field and encourages deeper inquiry. AIOps is a developing topic that will likely require further research to prepare IT operations for the future.

		AIOps						
		Functions		Practical Adoption		Findings		
		Proactive Failure Management	Reactive Failure Management	Implementation	Tools	Benefits	Challenges	Trends
Andenmatten	(2019)		✓	✓		✓	✓	✓
Chen et al.	(2020)		✓		✓	✓		
Dang et al.	(2019)		✓			✓	✓	
Gulenko et al.	(2020)		✓			✓	✓	✓
Levin et al.	(2019)		✓	✓		✓	✓	
Li et al.	(2020)	✓						
Lyu et al.	(2021)						✓	
Masood & Hashmi	(2019)				✓	✓	✓	✓
Nedelkoski et al.	(2019)		✓					
Notaro et al.	(2021)	✓	✓					✓
Paradkar	(2020)		✓					
Sen	(2020)				✓			
Shen et al.	(2020)		✓	✓				
Wang & Zhang	(2020)	✓						
Wang et al.	(2019)		✓					

Appendix 1. *Literature Matrix on AIOps*

References

Andenmatten, M. (2019). AIOps – Artificial Intelligence für IT-Operations. *HMD Praxis der Wirtschaftsinformatik, 56*(2), 332–344. https://doi.org/10.1365/s40702-019-00503-y

Chen, Z., Kang, Y., Li, L., Zhang, X., Zhang, H., Xu, H., Zhou, Y., Yang, L., Sun, J., Xu, Z., Dang, Y., Gao, F., Zhao, P., Qiao, B., Lin, Q., Zhang, D., & Lyu, M. R. (2020). Towards Intelligent Incident Management: Why We Need It and How We Make It. *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering.* https://doi.org/10.1145/3368089.3417055

Dang, Y., Lin, Q., & Huang, P. (2019). AIOps: Real-World Challenges and Research Innovations. *2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*. https://doi.org/10.1109/icse-companion.2019.00023

Gulenko, A., Acker, A., Kao, O., & Liu, F. (2020). AI-Governance and Levels of Automation for AIOps-supported System Administration. *2020 29th International Conference on Computer Communications and Networks (ICCCN)*. https://doi.org/10.1109/icccn49398.2020.9209606

He S., Lin Q., Lou J., Zhang H., Lyu M. R., & Zhang D. (2018). Identifying Impactful Service System Problems via Log Analysis. *Proceedings of the 26th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE '18)*. https://doi:10.1145/3236024.3236083.

Levin, A., Garion, S., Kolodner, E. K., Lorenz, D. H., Barabash, K., Kugler, M., & McShane, N. (2019). AIOps for a Cloud Object Storage Service. *2019 IEEE International Congress on Big Data (BigDataCongress)*. https://doi.org/10.1109/bigdatacongress.2019.00036

Li, Y., Jiang, Z. M. J., Li, H., Hassan, A. E., He, C., Huang, R., Zeng, Z., Wang, M., & Chen, P. (2020). Predicting Node Failures in an Ultra-Large-Scale Cloud Computing Platform. *ACM Transactions on Software Engineering and Methodology, 29*(2), 1–24. https://doi.org/10.1145/3385187

Lyu, Y., Li, H., Sayagh, M., Jiang, Z. M. J., & Hassan, A. E. (2021). An Empirical Study of the Impact of Data Splitting Decisions on the Performance of AIOps Solutions. *ACM Transactions on Software Engineering and Methodology*, *30*(4), 1–38. https://doi.org/10.1145/3447876

Masood, A., & Hashmi, A. (2019). AIOps: Predictive Analytics & Machine Learning in Operations. *Cognitive Computing Recipes*, 359–382. https://doi.org/10.1007/978-1-4842-4106-6_7

Nedelkoski, S., Cardoso, J., & Kao, O. (2019). Anomaly Detection and Classification using Distributed Tracing and Deep Learning. *2019 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*. https://doi.org/10.1109/ccgrid.2019.00038

Notaro, P., Cardoso, J., & Gerndt, M. (2021). A Systematic Mapping Study in AIOps. *Service-Oriented Computing – ICSOC 2020 Workshops*, 110–123. https://doi.org/10.1007/978-3-030-76352-7_15

Paradkar, S. (2020). APM to AIOps - Core Transformation. *Global Journal of Enterprise Information System*. https://doi.org/10.18311/gjeis/2020

Sen, A. (2020). DevOps, DevSecOps, AIOPS- Paradigms to IT Operations. *Lecture Notes in Electrical Engineering*, 211–221. https://doi.org/10.1007/978-981-15-7804-5_16

Shen, S., Zhang, J., Huang, D., & Xiao, J. (2020). Evolving from Traditional Systems to AIOps: Design, Implementation and Measurements. *2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*. https://doi.org/10.1109/aeeca49918.2020.9213650

Tranfield, D., Denyer D., & Smart P. (2003). Towards a Methodology for Developing Evidence-informed Management Knowledge by Means of Systematic Review. *British Journal of Management*, *14*, 207–222.

Wang, H., & Zhang, H. (2020). AIOPS Prediction for Hard Drive Failures Based on Stacking Ensemble Model. *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*. https://doi.org/10.1109/ccwc47524.2020.9031232

Wang, J., Jing, Y., Qi, Q., Feng, T., & Liao, J. (2019). ALSR: An Adaptive Label Screening And Relearning Approach For Interval-Oriented Anomaly Detection. *Expert Systems with Applications, 136,* 94–104. https://doi.org/10.1016/j.eswa.2019.06.028

Webster, J., & Watson R. T. (2002). Analyzing the Past to Prepare for the Future: Writing a Literature Review. *MIS Quarterly, 26,* xiii–xxiii.