序列进行建模难以精确刻画系统多线程和并发的复杂情形,可能需要大量的系统知识辅助离线建模过程,构建故障根因诊断模型可能需要日志文本中包含一些关联信息,如请求 ID、线程 ID 等,具有一定的适用范围限制;基于机器学习的方法可以方便输出与故障敏感的日志集合或输出故障类型,对日志的内容和系统知识的假设较少,能够有效地从不同的日志类型中提取较为全面的特征并构建模型,适用事务型日志和操作型日志;缺点在于十分依赖日志特征的选取和处理,需要离线训练阶段以及大量带标签的训练数据,故障定位粒度较粗,辅助故障诊断的能力弱于基于关联推断的方法.同时,机器学习模型和深度学习模型往往难以理解,对运维人员修复故障的辅助能力不足.

6 总结与未来展望

6.1 总 结

以第 1 节所述基于日志数据的分布式系统故障诊断研究框架为指导,对日志处理与特征提取技术、异常检测、故障预测及故障根因诊断技术的相关工作进行总结分析(见表 6),并得出如下结论:(1) 日志模板挖掘方法是日志处理与特征提取技术的主流,且该方法通常用于异常检测和故障根因诊断;(2) 日志特征提取方法,特别是基于规则的结构化信息提取方法通常与基于日志数据的故障预测组合;(3) 基于日志数据的故障根因诊断技术是当前的研究热点,相关研究工作应用了多种的日志模板挖掘方法和日志特征提取方法.

Table 6 Summary of log-based failure diagnosis approaches

表 6 基于日志数据的故障诊断技术相关文献总结

		日志处理与特征提取					
		日志模板挖掘方法			日志特征提取方法		
		基于静态代码分析的日志模板挖掘方法	基于频繁项集挖掘的日志模板挖掘方法	基于聚类的日志模板挖掘方法	基于自然语言处理的日志特征提取方法	基于规则的结构化信息提取方法	基于统计模型的日志特征提取方法
基于日志数据的异常检测	基于图模型的异常检测方法	–	[15,22]	[28,30,40]	[50–52]	[60]	–
	基于概率分析的异常检测方法	–	–	[27,32,33]	[49]	–	[67]
	基于机器学习的异常检测方法	[8,9] [10,11]	–	[28,73,74]	[56]	[59]	–
基于日志数据的故障预测	基于概率分析的故障预测方法	–	–	[40]	–	[62,76–78]	–
	基于机器学习的故障预测方法	–	–	–	[81,82]	[79,80,83]	–
基于日志数据的故障根因诊断	基于关联推断的故障根因诊断	[7,10,12] [13,84,89]	[21–23]	[28,30–32,37]	–	[58,60,63,64]	[32]
	基于机器学习的故障根因诊断	[8,92]	[17,20,95]	[26,28,33,36, 43,73,93]	[52–54,56]	[59,61,65,66]	[33]

进一步地,通过对日志处理与特征提取技术、异常检测、故障预测及故障根因诊断所使用的重要技术进行总结分析(见表 7),得出如下结论:(1) 机器学习和统计分析是基于日志数据的分布式软件系统故障诊断最常用的关键技术,在 4 个子技术中均有所应用;(2) 关联概率分析和有向图构建与计算是构建异常检测、故障预测、故障根因诊断模型的重要手段;(3) 日志处理与特征提取和基于日志数据的故障根因诊断中应用的技术最多.

Table 7 Summary of log-based failure diagnosis techniques

表 7 基于日志数据的故障诊断技术手段总结

	机器学习	静态代码分析	自然语言处理	统计分析	关联概率分析	频繁项集挖掘	有向图构建与计算
日志处理与特征提取	√	√	√	√	–	√	–
基于日志数据的异常检测	√	–	–	√	√	–	√
基于日志数据的故障预测	√	–	–	√	√	√	–
基于日志数据的故障根因诊断	√	√	√	√	√	–	√

6.2 未来展望

本文就日志处理与特征提取、基于日志数据的异常检测、基于日志数据的故障预测和基于日志数据的故障根因诊断 4 个关键技术,对基于日志数据的故障诊断研究领域的相关研究工作进行了综述和分析.虽然现有研究工作已经取得了一定的成果和进展,但该领域仍存在许多关键问题值得深入探讨和研究.

1. 故障诊断任务驱动的日志嵌入表示方法

现有日志处理与特征提取技术存在两个关键问题:(1) 该技术以简化日志表示、降低日志复杂性为根本目的而非提升故障诊断效率.例如,日志模板的本质是对海量日志的抽象,以模板的形式简化复杂的日志文本;日志特征提取方法则从日志文本中摄取少量的简单信息,如词频信息、特殊标识符信息等.这些方法提取的信息或特征可能与故障诊断任务无关,同时导致关键故障信息丢失.(2) 该技术难以做到精准和普适.例如,日志中的常量和变量交织分布情况可能及其复杂,以概率统计为基础的日志模板挖掘方法往往难以完全正确生成日志模板;而日志特征提取方法则依赖于日志文本的内容和结构,对于不同系统的日志而言,其提取的特征质量千差万别.为解决上述问题,未来可能从全新的角度思考日志处理与特征提取技术,借鉴基于深度学习的自然语言处理技术中的词嵌入表示(word embedding)方法,以故障诊断任务为目标构建或训练日志的嵌入表示方法.

2. 面向复杂系统的精准的故障预测模型与方法

现有基于日志数据的故障预测技术的相关研究工作较为薄弱,主要存在如下几个关键问题:(1) 现有故障预测模型多以机器学习算法为基础,方法较为简单,预测粒度粗,无法支持复杂的甚至是多故障集中爆发情况的预测;(2) 现有故障预测模型多以单日志序列为训练集,无法支持分布式系统中故障在多组件中蔓延和传播情况下的故障预测;(3) 现有故障预测方法往往仅利用了日志数据的少量文本或统计特征,在故障预测精确度方面仍有很大的提升空间.因此,未来可能以多组件协作的复杂分布式软件系统为目标,以时间序列分析技术、机器学习技术、图计算技术等为基本手段,构建面向复杂系统的精准故障预测模型.

3. 在线自学习、自更新异常检测、故障预测及故障根因诊断模型与方法

在当今 DevOps 的快速开发迭代环境中,系统更新极为频繁,随之带来的是日志的频繁更新与改变.这种频繁的系统更新要求故障诊断模型具备快速在线训练、更新和适应的能力,以保障故障诊断模型的可用性和有效性.然而,现有异常检测、故障预测及故障根因诊断模型均采用离线训练、在线使用的模式.这种模式仅支持离线重训练(re-training),速度慢、效率低,无法适应频繁的系统更新.因此,未来需要研究异常检测、故障预测及故障根因诊断模型的在线训练、更新方法与策略,故障诊断方法的自适应扩展及集成技术,实现异常检测、故障预测及故障根因诊断模型的在线动态快速自训练、自更新、自学习及自适应.

References:

[1] Elliot S. DevOps and the cost of downtime: Fortune 1000 best practice metrics quantified. International Data Corporation (IDC), 2014.

[2] The 10 biggest cloud outages of 2018. 2018. https://www.crn.com/slide-shows/cloud/the-10-biggest-cloud-outages-of-2018

[3] https://yq.aliyun.com/articles/603866/, 2018.

[4] https://www.ithome.com/html/it/372627, 2018.

[5] Market guide for AIOps platforms. 2016. https://www.gartner.com/doc/3892967/market-guide-aiops-platforms

[6] What is AIOps. 2018. https://www.bmc.com/blogs/what-is-aiops

[7] Zhao X, Zhang Y, Lion D, *et al*. Lprof: A non-intrusive request flow profiler for distributed systems. In: Proc. of the 11th Symp. on Operating Systems Design and Implementation. USENIX, 2014. 629–644.

[8] Xu W, Huang L, Fox A, *et al*. Detecting large-scale system problems by mining console logs. In: Proc. of the Int'l Conf. on Machine Learning. IEEE, 2009. 117–132.

[9] Xu W, Huang L, Fox A, *et al*. Mining console logs for large-scale system problem detection. In: Proc. of the Workshop on Tackling Computer Systems Problems with Machine Learning Techniques. San Diego: IEEE, 2008. 4.

[10] Xu W, Huang L, Fox A, *et al*. Online system problem detection by mining patterns of console logs. In: Proc. of the 9th Int'l Conf. on Data Mining. Miami: IEEE, 2009. 588–597.

[11] Xu W. System problem detection by mining console logs [Ph.D. Thesis]. Berkeley: University of California, 2010.

[12] Yuan D, Mai H, Xiong W, *et al*. SherLog: Error diagnosis by connecting clues from run-time logs. In: Proc. of the 15th Edition on Architectural Support for Programming Languages and Operating Systems. ACM, 2010. 143–154.

[13] Ghanbari S, Hashemi AB, Amza C. Stage-aware anomaly detection through tracking log points. In: Proc. of the 15th Int'l Middleware Conf. ACM, 2014. 253–264.

[14] Vaarandi R. A breadth-first algorithm for mining frequent patterns from event logs. In: Proc. of the Int'l Conf. on Intelligence in Communication Systems. IEEE, 2004. 293–308.

[15] Nandi A, Mandal A, Atreja S, *et al*. Anomaly detection using program control flow graph mining from execution logs. In: Proc. of the Int'l Conf. on Knowledge Discovery and Data Mining. ACM, 2016. 215–224.

[16] Makanju AAO, Zincir-Heywood AN, Milios EE. Clustering event logs using iterative partitioning. In: Proc. of the 15th Int'l Conf. on Knowledge Discovery and Data Mining. ACM, 2009. 1255–1264.

[17] Reidemeister T, Jiang M, Ward PAS. Mining unstructured log files for recurrent fault diagnosis. In: Proc. of the Int'l Symp. on Integrated Network Management. DBLP, 2011. 377–384.

[18] Vaarandi R. A data clustering algorithm for mining patterns from event logs. In: Proc. of the Int'l Conf. on IT Operations & Management. IEEE, 2003. 119–126.

[19] Vaarandi R, Pihelgas M. LogCluster: A data clustering and pattern mining algorithm for event logs. In: Proc. of the Int'l Conf.on Network and Service Management. IEEE, 2015. 1–7.

[20] Reidemeister T, Munawar MA, Ward PAS. Identifying symptoms of recurrent faults in log files of distributed information systems. In: Proc. of the Network Operations and Management Symp. IEEE, 2010. 187–194.

[21] Tak BC, Tao S, Yang L, *et al*. LOGAN: Problem diagnosis in the cloud using log-based reference models. In: Proc. of the Int'l Conf. on Cloud Engineering. IEEE, 2016. 62–67.

[22] Babenko A, Mariani L, Pastore F. AVA: Automated interpretation of dynamically detected anomalies. In: Proc. of the Int'l Symp. on Software Testing and Analysis. IEEE, 2009. 237–248.

[23] Mariani L, Pastore F. Automated identification of failure causes in system logs. In: Proc. of the Int'l Symp. on Software Reliability Engineering. IEEE, 2008. 117–126.

[24] Gainaru A, Cappello F, Trausan-Matu S, *et al*. Event log mining tool for large scale HPC systems. In: Proc. of the European Conf. on Parallel Processing. Berlin, Heidelberg. Springer-Verlag, 2011. 52–64.

[25] Vaarandi R. Mining event logs with SLCT and loghound. In: Proc. of the Network Operations and Management Symp. IEEE, 2008. 1071–1074.

[26] Lin Q, Zhang H, Lou J G, *et al*. Log clustering-based problem identification for online service systems. In: Proc. of the 38th Int'l Conf. on Software Engineering Companion. ACM, 2016. 102–111.

[27] Lou JG, Fu Q, Yang S, *et al*. Mining invariants from console logs for system problem detection. In: Proc. of the ATC. USENIX, 2010. 231–244.

[28] Fu Q, Lou JG, Lin Q, *et al*. Contextual analysis of program logs for understanding system behaviors. In: Proc. of the Int'l Conf. on Mining Software Repositories. IEEE, 2013. 397–400.

[29] Ding R, Fu Q, Lou JG, *et al*. Healing online service systems via mining historical issue repositories. In: Proc. of the Int'l Conf. on Automated Software Engineering. IEEE, 2012. 318–321.

[30] Fu Q, Lou JG, Wang Y, *et al*. Execution anomaly detection in distributed systems through unstructured log analysis. In: Proc. of the Int'l Conf. on Data Mining. IEEE, 2009. 149–158.

[31] Lou JG, Fu Q, Wang Y, *et al*. Mining dependency in distributed systems through unstructured logs analysis. ACM SIGOPS Operating Systems Review, 2010,44(1):91–96.

[32] Chen C, Singh N, Yajnik S. Log analytics for dependable enterprise telephony. In: Proc. of the 9th European Dependable Computing Conf. IEEE, 2012. 94–101.

[33] Du S, Cao J. Behavioral anomaly detection approach based on log monitoring. In: Proc. of the Int'l Conf. on Behavioral, Economic and Socio-cultural Computing. IEEE, 2015.

[34] Li T, Liang F, Ma S, *et al*. An integrated framework on mining logs files for computing system management. In: Proc. of the Int'l Conf. on Knowledge Discovery and Data Mining. Chicago: ACM, 2005.

[35] Aharon M, Barash G, Cohen I, *et al*. One graph is worth a thousand logs: Uncovering hidden structures in massive system event logs. In: Proc. of the Joint European Conf. on Machine Learning and Knowledge Discovery in Databases. Berlin, Heidelberg: Springer-Verlag, 2009. 227–243.

[36] Jia T, Li Y, Tang H, *et al*. An approach to pinpointing bug-induced failure in logs of open cloud platforms. In: Proc. of the Int'l Conf. on Cloud Computing. IEEE, 2016. 294–302.

[37] Debnath B, Solaimani M, Gulzar MAG, *et al*. LogLens: A real-time log analysis system. In: Proc. of the 38th Int'l Conf. on Distributed Computing Systems (ICDCS). IEEE, 2018. 1052–1062.

[38] He P, Zhu J, He S, *et al*. Towards automated log parsing for large-scale log data analysis. IEEE Trans. on Dependable & Secure Computin, 2017,(99):1.

[39] He P, Zhu J, Zheng Z, *et al*. Drain: An online log parsing approach with fixed depth tree. In: Proc. of the Int'l Conf. on Web Services. IEEE, 2017. 33–40.

[40] Watanabe Y, Otsuka H, Sonoda M, Kikuchi S, Matsumoto Y. Online failure prediction in cloud datacenters by real-time message pattern learning. In: Proc. of the Int'l Conf. on Cloud Computing Technology and Science (CloudCom). IEEE, 2012. 504–511.

[41] Jiang ZM, Hassan AE, Hamann G, *et al*. An automated approach for abstracting execution logs to execution events. Journal of Software Maintenance & Evolution Research & Practice, 2008,20(4):249–267.

[42] Jiang ZM, Hassan AE, Flora P, *et al*. Abstracting execution logs to execution events for enterprise applications. In: Proc. of the 8th Int'l Conf. on Quality Software. IEEE, 2008. 181–186.

[43] Du M, Li F, Zheng G, *et al*. Deeplog: Anomaly detection and diagnosis from system logs through deep learning. In: Proc. of the 2017 Conf. on Computer and Communications Security. ACM, 2017. 1285–1298.

[44] Zhu KQ, Fisher K, Walker D. Incremental learning of system log formats. ACM SIGOPS Operating Systems Review, 2010,44(1): 85–90.

[45] Hamooni H, Debnath B, Xu J, *et al*. Logmine: Fast pattern recognition for log analytics. In: Proc. of the 25th Int'l Conf. on Information and Knowledge Management. ACM, 2016. 1573–1582.

[46] Mizutani M. Incremental mining of system log format. In: Proc. of the Int'l Conf. on Services Computing. IEEE, 2013. 595–602.

[47] Tang L, Li T, Perng CS. LogSig: Generating system events from raw textual logs. In: Proc. of the 20th ACM Int'l Conf. on Information and Knowledge Management. ACM, 2011. 785–794.

[48] Du M, Li F. Spell: Streaming parsing of system event logs. In: Proc. of the 16th Int'l Conf. on Data Mining. IEEE, 2016. 859–864.

[49] Oliner AJ, Aiken A. Online detection of multi-component interactions in production systems. In: Proc. of the IEEE/IFIP Int'l Conf. on Dependable Systems and Networks. Hong Kong: IEEE, 2011. 49–60.

[50] Oliner AJ, Aiken A, Stearley J. Alert detection in system logs. In: Proc.of the Int'l Conf. on Data Mining. IEEE, 2008. 959–964.

[51] Stearley J, Oliner AJ. Bad words: Finding faults in spirit's syslogs. In: Proc. of the Int'l Symp. on CLUSTER. IEEE, 2008. 765–770.

[52] Juvonen A, Hamalainen T. An efficient network log anomaly detection system using random projection dimensionality reduction. In: Proc. of the 6th Int'l Conf. on New Technologies, Mobility and Security. IEEE, 2014. 1–5.

[53] Sipola T, Juvonen A, Lehtonen J. Anomaly detection from network logs using diffusion maps. In: Proc. of the IFIP Advances in Information & Communication Technology, IEEE, 2011,363:172–181.

[54] Sipola T, Juvonen A, Lehtonen J. Dimensionality reduction framework for detecting anomalies from network logs. In: Proc. of the Engineering Intelligent Systems. 2012.

[55] Bertero C, Roy M, Sauvanaud C, *et al*. Experience report: Log mining using natural language processing and application to anomaly detection. In: Proc. of the 28th Int'l Symp. on Software Reliability Engineering (ISSRE). IEEE, 2017. 351–360.

[56] Brown A, Tuor A, Hutchinson B, *et al*. Recurrent neural network attention mechanisms for interpretable system log anomaly detection. arXiv Preprint arXiv:1803.04967, 2018.

[57] Chow M, Meisner D, Flinn J, *et al*. The mystery machine: End-to-end performance analysis of large-scale internet services. In: Proc. of the 11th Symp. on Operating Systems Design and Implementation. USENIX, 2014. 217–231.

[58] Chuah E, Jhumka A, Narasimhamurthy S, *et al*. Linking resource usage anomalies with system failures from cluster log data. In: Proc. of the 32nd Int'l Symp. on Reliable Distributed Systems. IEEE, 2013. 111–120.

[59] Yen TF, Oprea A, Onarlioglu K, *et al*. Beehive: Large-scale log analysis for detecting suspicious activity in enterprise networks. In: Proc. of the Int'l Conf. on Computer Security Applications. IEEE, 2013. 199–208.

[60] Yu X, Joshi P, Xu J, *et al*. CloudSeer: Workflow monitoring of cloud infrastructures via interleaved logs. SIGOPS Operating Systems Review, 2016,50(2):489–502.

[61] Davidsen B, Kristensen E. Pinpoint: Problem determination in large, dynamic Internet services. In: Proc. of the Int'l Conf. on Dependable Systems and Networks. IEEE, 2002. 595–604.

[62] Liang Y, Zhang Y, Sivasubramaniam A, *et al*. BlueGene/L failure analysis and prediction models. In: Proc. of the Int'l Conf. on Dependable Systems and Networks. IEEE, 2006. 425–434.

[63] Beschastnikh I, Brun Y, Ernst MD, *et al*. Inferring models of concurrent systems from logs of their behavior with CSight. In: Proc. of the 36th Int'l Conf. on Software Engineering. ACM, 2014. 468–479.

[64] Chuah E, Kuo S, Hiew P, *et al*. Diagnosing the root-causes of failures from cluster log files. In: Proc. of the Int'l Conf. on High Performance Computing. IEEE, 2010. 1–10.

[65] Nagaraj K, Killian C, Neville J. Structured comparative analysis of systems logs to diagnose performance problems. In: Proc. of the 9th USENIX Conf. on Networked Systems Design and Implementation. USENIX, 2012. 26.

[66] Mi HB, Wang HM, Zhou YF, *et al*. Localizing root causes of performance anomalies in cloud computing systems by analyzing request trace logs. Science China Information Sciences, 2012,55(12):2757–2773.

[67] Lim C, Singh N, Yajnik S. A log mining approach to failure analysis of enterprise telephony systems. In: Proc. of the Int'l Conf. on Dependable Systems and Networks. IEEE, 2008. 398–403.

[68] Yan X, Zhou W, Gao Y, *et al*. PADM: Page rank-based anomaly detection method of log sequences by graph computing. In: Proc. of the Int'l Conf. on Cloud Computing Technology and Science. IEEE, 2014. 700–703.

[69] Oliner AJ, Kulkarni AV, Aiken A. Using correlated surprise to infer shared influence. In Proc. of the IEEE/IFIP Int'l Conf. on Dependable Systems and Networks. Chicago: IEEE, 2010. 191–200.

[70] Rao X, Wang HM, Chen ZB, *et al*. Detecting faults by tracing companion states in cloud computing systems. Chinese Journal of Computers, 2012,35(5):856–870 (in Chinese with English abstract).

[71] Rao X, Tian Q, *et al*. Sub-sequence feature vector-based massive system log anomaly detection in cloud computing systems. In: Proc. of the National Conf. of Information Storage. 2012 (in Chinese with English abstract).

[72] Vinayakumar R, Soman KP, Poornachandran P. Long short-term memory-based operation log anomaly detection. In: Proc. of the Int'l Conf. on Advances in Computing, Communications and Informatics. IEEE, 2017. 236–242.

[73] Lu S, Wei X, Li Y, *et al*. Detecting anomaly in big data system logs using convolutional neural network. In: Proc. of the 16th Int'l Conf. on Dependable, Autonomic and Secure Computing. IEEE, 2018. 151–158.

[74] Meng WB, Liu Y, *et al*. LogAnomaly: Unsupervised detection of sequential and quantitative anomalies in unstructured logs. In: Proc. of the Int'l Joint Conf. on Artificial Intelligence. IEEE, 2019.

[75] Cui Y, Sun Y, Hu J, *et al*. A convolutional auto-encoder method for anomaly detection on system logs. In: Proc. of the Int'l Conf. on Systems, Man and Cybernetics (SMC). IEEE, 2018. 3057–3062.

[76] Gainaru A, Cappello F, Fullop J, *et al*. Adaptive event prediction strategy with dynamic time window for large-scale HPC systems. In: Proc. of the Managing Large-scale Systems Via the Analysis of System Logs & the Application of Machine Learning Techniques. ACM, 2011. 1–8.

[77] Lan Z, Gu J, Zheng Z, *et al*. A study of dynamic meta-learning for failure prediction in large-scale systems. Journal of Parallel & Distributed Computing, 2010,70(6):630–643.

[78] Navarro JM, Parada GHA, Duenas JC. System failure prediction through rare-events elastic-net logistic regression. In: Proc. of the Int'l Conf. on Artificial Intelligence, Modelling and Simulation. IEEE, 2014. 120–125.

[79] Shalan A, Zulkernine M. Runtime prediction of failure modes from system error logs. In: Proc. of the Int'l Conf. on Engineering of Complex Computer Systems. IEEE, 2013. 232–241.

[80] Yu L, Zheng Z, Lan Z, *et al*. Practical online failure prediction for Blue Gene/P: Period-based vs. event-driven. In: Proc. of the Int'l Conf. on Dependable Systems and Networks Workshops. IEEE, 2011. 259–264.

[81] Fronza I, Sillitti A, Succi G, *et al*. Failure prediction based on log files using random indexing and support vector machines. Journal of Systems & Software, 2013,86(1):2–11.

[82] Fulp EW, Fink GA, Haack JN. Predicting computer system failures using support vector machines. In: Proc. of the 1st USENIX Workshop on the Analysis of System Logs. San Diego: USENIX, 2008.

[83] Sahoo RK, Oliner AJ, Rish I, *et al*. Critical event prediction for proactive management in large-scale computer clusters. In: Proc. of the ACM Int'l Conf. on Knowledge Discovery and Data Mining. Washington: ACM, 2003. 426–435.

[84] Zhang Y, Makarov S, Ren X, *et al*. Pensieve: Non-intrusive failure reproduction for distributed systems using the event chaining approach. In: Proc. of the 26th Symp. on Operating Systems Principles. ACM, 2017. 19–33.

[85] Yamanishi K, Maruyama Y. Dynamic syslog mining for network failure monitoring. In: Proc. of the Int'l Conf. on Knowledge Discovery and Data Mining. ACM, 2005. 499–508.

[86] Liang Y, Zhang Y, Sivasubramaniam A, *et al*. Filtering failure logs for a bluegene/l prototype. In: Proc. of the Int'l Conf. on Dependable Systems and Networks. IEEE, 2005. 476–485.

[87] Shang W, Nagappan M, Hassan AE, *et al*. Understanding log lines using development knowledge. In: Proc. of the Int'l Conf. on Software Maintenance and Evolution. IEEE, 2014. 21–30.

[88] Nguyen H, Dean DJ, Kc K, *et al*. Insight: In-situ online service failure path inference in production computing infrastructures. In: Proc. of the Annual Technical Conf. USENIX, 2014. 269–280.

[89] Zhao X, Rodrigues K, Luo Y, *et al*. Non-intrusive performance profiling for entire software stacks based on the flow reconstruction principle. In: Proc. of the 12th Symp. on Operating Systems Design and Implementation. USENIX, 2016. 603–618.

[90] Tan J, Pan X, Kavulya S, *et al*. SALSA: Analyzing logs as state machines. WASL, 2008,8:6.

[91] Rao X. Research on log-based trust management in large-scale distributed software system [Ph.D. Thesis]. Changsha: National University of Defense Technology, 2011 (in Chinese with English abstract).

[92] Kc K, Gu X. ELT: Efficient log-based troubleshooting system for cloud computing infrastructures. In: Proc. of the Int'l Symp. on Reliable Distributed Systems. IEEE, 2011. 11–20.

[93] Lo D, Cheng H, Han J, *et al*. Classification of software behaviors for failure detection: A discriminative pattern mining approach. In: Proc. of the Int'l Conf. on Knowledge Discovery and Data Mining. ACM, 2009. 557–566.

[94] Mi H, Wang H, Yin G, *et al*. Performance problems diagnosis in cloud computing systems by mining request trace logs. In: Proc. of Network Operations and Management Symp. IEEE, 2012. 893–899.

[95] Xu J, Chen P, Yang L, *et al*. LogDC: Problem diagnosis for declartively-deployed cloud applications with log. In: Proc. of the 14th Int'l Conf. on e-Business Engineering (ICEBE). IEEE, 2017. 282–287.

附中文参考文献:

[70] 饶翔,王怀民,陈振邦,等.云计算系统中基于伴随状态追踪的故障检测机制.计算机学报,2012,35(5):856–870.

[71] 饶翔,田青,朱鸿宇,等.云计算系统中基于子序列特征向量的海量日志异常检测方法.见:全国信息存储技术学术会议.2012.

[91] 饶翔.基于日志的大规模分布式软件系统可信保障技术研究[博士学位论文].长沙:国防科学技术大学,2011.

贾统(1993－),男,博士,主要研究领域为分布式计算,智能运维.

吴中海(1968－),男,博士,教授,博士生导师,CCF 杰出会员,主要研究领域为大数据技术,系统安全,嵌入式软件.

李影(1975－),女,博士,教授,博士生导师,CCF 高级会员,主要研究领域为分布式计算,可信计算.