**Résumé de l'article : *Self-Supervised Anomaly Detection from Distributed Traces***

**Détails de l’article**

* **Titre** : Self-Supervised Anomaly Detection from Distributed Traces
* **Auteurs** : Jasmin Bogatinovski, Sasho Nedelkoski, Jorge Cardoso, Odej Kao
* **Année de publication** : 2020
* **Journal/Conférence** : 13th IEEE/ACM International Conference on Utility and Cloud Computing (UCC)

**1. Résumé**

Cet article propose une méthode d'apprentissage auto-supervisée pour détecter les anomalies dans les systèmes distribués, en utilisant des données de traçage distribuées. La méthode repose sur une tâche d'apprentissage appelée ***Masked Span Prediction*** (MSP) qui prédit des événements manquants dans un ensemble de traces tout en exploitant les relations contextuelles dans les données. La technique surpasse les approches basées sur les modèles LSTM dans la détection d'anomalies, en particulier pour les longues séquences de traces, grâce à l'utilisation de mécanismes d'attention.

**2. Domaine traité**

La recherche se concentre sur la **détection d'anomalies dans les systèmes distribués**, avec un accent particulier sur l'utilisation des **traces distribuées** comme source principale de données.

**3. Problème résolu**

* **Problématique scientifique** : Les approches traditionnelles de détection d'anomalies dans les systèmes distribués ne gèrent pas efficacement les longues traces ou les interactions complexes entre services.
* **Lien avec le thème** : Dans un cadre **DevOps**, où la surveillance distribuée est essentielle, cette méthode permet d'améliorer la détection précoce des anomalies et la résilience des systèmes basés sur des microservices.
* **Question de recherche** : Comment détecter efficacement des anomalies dans des systèmes distribués à l'aide de techniques d'apprentissage non supervisées capables de gérer des données bruyantes et des séquences longues ?

**4. Intérêt d’avoir une solution pour le domaine d’étude**

Une telle solution améliore la capacité à surveiller les systèmes complexes basés sur des architectures de microservices, offrant une détection précoce des anomalies avant qu'elles n'entraînent des défaillances majeures. Cela s'aligne avec les principes DevOps en facilitant une réponse proactive et en augmentant la fiabilité des applications.

**5. Approche adoptée**

L’approche repose sur :

* L’utilisation des traces distribuées comme source principale d’information.
* Une tâche d'apprentissage appelée MSP, qui masque aléatoirement des événements dans une trace pour prédire leur position à partir de leur contexte.
* Une architecture basée sur des mécanismes d’attention (**self-attention**) pour capturer les relations globales et locales au sein des traces.

**6. Solution proposée**

* La solution comprend une architecture de type ***encoder-decoder*** basée sur des mécanismes d’attention.
* Le modèle est entraîné de manière auto-supervisée à partir de traces normales uniquement, sans besoin d’étiquetage manuel.
* Un score d'anomalie est calculé en comparant les prédictions du modèle avec les valeurs réelles des événements dans les traces.

**7. Discussion**

* **Points positifs** :
  + Gère efficacement les longues traces et les données complexes.
  + Supérieur aux modèles basés sur les LSTM pour la détection d'anomalies dans des traces distribuées.
  + Robuste face à des permutations mineures dans les traces normales.
* **Manquements** :
  + Limité aux données de traçage ; pourrait bénéficier d'une intégration avec d'autres sources (métriques, logs).
  + Nécessite une infrastructure pour capturer et traiter les traces distribuées.

**8. Intérêt pour le problème**

Cette approche est directement pertinente pour la **surveillance distribuée des applications en microservices** dans un cadre DevOps. Elle propose une méthode robuste et évolutive pour détecter les anomalies, essentielle à la gestion proactive des performances et à la résilience des systèmes distribués.

**9. Travaux à regarder**

Voici quelques références de l'article qui cadrent avec le thème :

1. **Du, M., Li, F., Zheng, G., & Srikumar, V. (2017). "Deeplog: Anomaly detection and diagnosis from system logs through deep learning." Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security.**
2. **Nedelkoski, S., Cardoso, J., & Kao, O. (2019). "Anomaly detection and classification using distributed tracing and deep learning." Proceedings of the 2019 IEEE/ACM International Symposium on Cluster, Cloud, and Grid Computing.**
3. **Gulenko, A., Schmidt, F., Acker, A., Wallschläger, M., Kao, O., & Liu, F. (2018). "Detecting anomalous behavior of black-box services modeled with distance-based online clustering." 2018 IEEE 11th International Conference on Cloud Computing.**
4. **Brownlee, J. ; How to Grid Search Hyperparameters for Deep Learning Models in Python with Keras. Machine Learning Mastery. Available online:**[**https://machinelearningmastery.com/grid-search-hyperparameters-deep-learning-models-python-keras/**](https://machinelearningmastery.com/grid-search-hyperparameters-deep-learning-models-python-keras/)**(accessed on 1 July 2023).**
5. **Xu, S.; Liu, H.; Duan, L.; Wu, W. An improved LOF outlier detection algorithm. In Proceedings of the 2021 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 28–30 June 2021; pp. 113–117. [Google Scholar]**
6. **Labs, G. Grafana Observability Survey 2023. Available online: https://grafana.com/observability-survey-2023/ (accessed on 4 May 2023).**
7. Sridharan, C. Distributed Systems Observability; O’Reilly Media, Inc.: Sebastopol, CA, USA, 2018. [Google Scholar]
8. **Rosay, A.; Riou, K.; Carlier, F.; Leroux, P. Multi-layer perceptron for network intrusion detection: From a study on two recent data sets to deployment on automotive processor. *Ann. Telecommun.* 2022, *77*, 371–394. [**[**Google Scholar**](https://scholar.google.com/scholar_lookup?title=Multi-layer+perceptron+for+network+intrusion+detection:+From+a+study+on+two+recent+data+sets+to+deployment+on+automotive+processor&author=Rosay,+A.&author=Riou,+K.&author=Carlier,+F.&author=Leroux,+P.&publication_year=2022&journal=Ann.+Telecommun.&volume=77&pages=371%E2%80%93394&doi=10.1007/s12243-021-00852-0)**] [[CrossRef](https://doi.org/10.1007/s12243-021-00852-0" \t "_blank)]**
9. Zang, X.; Chen, W.; Zou, J.; Zhou, S.; Lisong, H.; Ruigang, L. A fault diagnosis method for microservices based on multi-factor self-adaptive heartbeat detection algorithm. In Proceedings of the 2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2), Beijing, China, 20–22 October 2018; pp. 1–6. [[**Google Scholar**](https://scholar.google.com/scholar_lookup?title=A+fault+diagnosis+method+for+microservices+based+on+multi-factor+self-adaptive+heartbeat+detection+algorithm&conference=Proceedings+of+the+2018+2nd+IEEE+Conference+on+Energy+Internet+and+Energy+System+Integration+(EI2)&author=Zang,+X.&author=Chen,+W.&author=Zou,+J.&author=Zhou,+S.&author=Lisong,+H.&author=Ruigang,+L.&publication_year=2018&pages=1%E2%80%936)]
10. Meng, L.; Ji, F.; Sun, Y.; Wang, T. Detecting anomalies in microservices with execution trace comparison. *Future Gener. Comput. Syst.* **2021**, *116*, 291–301. [[**Google Scholar**](https://scholar.google.com/scholar_lookup?title=Detecting+anomalies+in+microservices+with+execution+trace+comparison&author=Meng,+L.&author=Ji,+F.&author=Sun,+Y.&author=Wang,+T.&publication_year=2021&journal=Future+Gener.+Comput.+Syst.&volume=116&pages=291%E2%80%93301&doi=10.1016/j.future.2020.10.040)] [**[CrossRef](https://doi.org/10.1016/j.future.2020.10.040" \t "_blank)**]
11. Jin, M.; Lv, A.; Zhu, Y.; Wen, Z.; Zhong, Y.; Zhao, Z.; Wu, J.; Li, H.; He, H.; Chen, F. An anomaly detection algorithm for microservice architecture based on robust principal component analysis. *IEEE Access* **2020**, *8*, 226397–226408. [[**Google Scholar**](https://scholar.google.com/scholar_lookup?title=An+anomaly+detection+algorithm+for+microservice+architecture+based+on+robust+principal+component+analysis&author=Jin,+M.&author=Lv,+A.&author=Zhu,+Y.&author=Wen,+Z.&author=Zhong,+Y.&author=Zhao,+Z.&author=Wu,+J.&author=Li,+H.&author=He,+H.&author=Chen,+F.&publication_year=2020&journal=IEEE+Access&volume=8&pages=226397%E2%80%93226408&doi=10.1109/ACCESS.2020.3044610)] [**[CrossRef](https://doi.org/10.1109/ACCESS.2020.3044610" \t "_blank)**]
12. **Liu, P.; Xu, H.; Ouyang, Q.; Jiao, R.; Chen, Z.; Zhang, S.; Yang, J.; Mo, L.; Zeng, J.; Xue, W.; et al. Unsupervised detection of microservice trace anomalies through service-level deep bayesian networks. In Proceedings of the 2020 IEEE 31st International Symposium on Software Reliability Engineering (ISSRE), Coimbra, Portugal, 12–15 October 2020; pp. 48–58. [**[**Google Scholar**](https://scholar.google.com/scholar_lookup?title=Unsupervised+detection+of+microservice+trace+anomalies+through+service-level+deep+bayesian+networks&conference=Proceedings+of+the+2020+IEEE+31st+International+Symposium+on+Software+Reliability+Engineering+(ISSRE)&author=Liu,+P.&author=Xu,+H.&author=Ouyang,+Q.&author=Jiao,+R.&author=Chen,+Z.&author=Zhang,+S.&author=Yang,+J.&author=Mo,+L.&author=Zeng,+J.&author=Xue,+W.&publication_year=2020&pages=48%E2%80%9358)**]**
13. Brown, A.; Tuor, A.; Hutchinson, B.; Nichols, N. Recurrent neural network attention mechanisms for interpretable system log anomaly detection. In Proceedings of the First Workshop on Machine Learning for Computing Systems, Tempe, AZ, USA, 12 June 2018; pp. 1–8. [[**Google Scholar**](https://scholar.google.com/scholar_lookup?title=Recurrent+neural+network+attention+mechanisms+for+interpretable+system+log+anomaly+detection&conference=Proceedings+of+the+First+Workshop+on+Machine+Learning+for+Computing+Systems&author=Brown,+A.&author=Tuor,+A.&author=Hutchinson,+B.&author=Nichols,+N.&publication_year=2018&pages=1%E2%80%938)]
14. Lewis, J.; Fowler, M. Microservices: A Definition of This New Architectural Term. 2014. Available online: [**https://martinfowler.com/articles/microservices.html.**](https://martinfowler.com/articles/microservices.html.) (accessed on 4 May 2023).
15. Newman, S. *Building Microservices*; O’Reilly Media, Inc.: Sebastopol, CA, USA, 2021. [[**Google Scholar**](https://scholar.google.com/scholar_lookup?title=Building+Microservices&author=Newman,+S.&publication_year=2021)].