Benchmarking Deep Anomaly Detection on Real Process Data of a Continuous Distillation Process

Aparna Muraleedharana, Fabian Hartungb,c, Dennis Wagnerb, Marius Kloftb, Jakob Burgera\*

aTechnical University Munich (TUM) Campus Straubing 94315, Deutschland

bRPTU Kaiserslautern, Kaiserslautern 67663, Deutschland

cBASF SE, Ludwigshafen 67063, Deutschland

burger@tum.de

Abstract

Our work delves into anomaly detection (AD) within the chemical industry, which is vital for maximizing product yields and ensuring operational safety. Over the past decade, many AD methods based on deep machine learning have appeared [1], and they are usually developed, assessed, and compared using artificial process data from the Tennessee-Eastman Process (TEP) [3]. Real chemical processes may exhibit distinct anomalies and dynamics, potentially undermining the effectiveness of methods tailored solely to TEP data. In response, the present work presents findings on AD using deep learning for a continuously operated mini plant in our lab. The dataset, spanning several weeks of operation and encompassing fault-free and faulty operations, serves as the test set for 22 literature methods for AD with deep learning. We compare the performances with prior evaluations on TEP data [2].

**Keywords**: Machine Learning, Anomaly Detection, Distillation, Data Generation

* 1. Introduction

Anomaly detection (AD) in the chemical industry is crucial for ensuring safety, maintaining quality standards, reducing costs, complying with regulations, optimizing processes, and, most importantly, preventing potential disasters or environmental damage. In the present work, we focus on AD in continuous chemical processes. Continuous chemical processes are integral to the industry for ensuring efficiency and consistent, high-quality output in a dynamic market landscape. These processes generate time series data, enabling real-time monitoring, predictive maintenance, and the detection of anomalies, thereby facilitating optimized operations and ensuring product quality and safety. For the last three decades, the Tennessee Eastman Process (TEP) has been the benchmark dataset for time-series anomaly detection, including previous works on anomaly detection in chemical processes [4]. Recently, we have published [3] an extensive comparison of 27 anomaly detection methods from the literature for unsupervised AD using a synthetic TEP dataset [32]. The TEP was created and published by Downs and Vogel in 1993 [5] and represents a simulated chemical process rather than an actual industrial plant; therefore, the respective dataset is a synthetic multivariate time series that does not cover the full spectrum of possible operating scenarios or abnormalities encountered in real-time industrial settings. Since it cannot be guaranteed to simulate every effect of an occurring anomaly and its impact on the whole plant system, we wanted to examine possible performance differences of the models between synthetic TEP data and real-world data.

Therefore, the key challenge is that the availability of chemical data in the public domain could be more extensive. The present work aims to improve this situation and generate experimental data from a continuous distillation plant in TUM Campus Straubing. The experimental data generated is a time series dataset with and without anomalies. Using this data, we tested 22 of the literature mentioned above methods on AD and compared their performance with respective results for the synthetic TEP data.

* 1. Methods
		1. Experimental Data Generation

The continuous distillation plant is a mini plant with a capacity of 5t/y feed (Figure 1). Its core is a steel column (operating pressure 1.5 bar). It is controlled and monitored using a LabVIEW system, the primary data collection tool. Multivariate time series data from 17 sensors was collected. The distillation column has one pressure controller, two flow controllers, two level controllers, and two temperature controllers. Six control loops are implemented in LabVIEW, enabling control and operation of the plant. The control strategy within the LabVIEW system offers adaptability for easy adjustments as required.

Figure 1: Setup of the continuous distillation plant (steel) for experiments with water

Experiments conducted with a single component water generated time series data spanning 30 days. Choosing water as the sole component streamlined the process by removing the separation aspect and directed the focus toward the overall material balance. Using water also leads to increased safety, enabling the remote and autonomous operation of the plant. Experimental data with and without anomalies were generated and labeled. The labeled data was further used to train, test, and validate anomaly detection methods.

* + 1. Anomaly Detection

Advancements in time-series anomaly detection have prominently featured the evolution of unsupervised deep learning methods, notably with the application of sophisticated neural network architectures like recurrent neural networks (RNNs)[6], Long Short-Term Memory Networks (LSTMs)[7], and Variational Autoencoders (VAEs)[8] and various Generative Adversarial Network (GAN)[12] adaptations and operate independently without the need for labeled anomaly instances. Notably, RNNs and LSTMs, lauded for their sequential learning capabilities, excel in capturing extensive temporal dependencies, effectively pinpointing subtle anomalies. Conversely, an AE decodes and reconstructs input data, enabling the identification of deviations by comparing reconstructed sequences to the learned normal patterns. Furthermore, GANs create synthetic data instances and assess their authenticity against the original distribution, proficiently detecting anomalies by discerning disparities between generated and authentic data representations.

These unsupervised deep learning models [9] are trained on patterns representing standard behavior within time series data, allowing them to internalize fundamental structures and representations of regular sequences. Consequently, anomalies reflecting deviations from the learned norms are effectively detected within the data. In our prior work [3], we extensively evaluated and compared 27 unsupervised deep anomaly detection techniques found in contemporary literature for time-series anomaly detection. This thorough assessment gauged their detection accuracies utilizing the synthetic TEP dataset.

Our evaluation involved a comparative analysis of reconstruction-based methods[26], forecasting-based [13], generative methods [8], and hybrid approaches. Detecting anomalies within a time series involves generating anomaly scores[14] for each time step, with detection occurring if the score surpasses a predefined threshold. Commonly used evaluation metrics such as the F1-score [2] and the area under the precision-recall curve (AUPRC) [2] aid us in comprehensively assessing the performance of anomaly detection methods. The F1-score considers true positives, false negatives, and false positives, offering insights into precision and recall. Precision signifies the accuracy of identified anomalies among all flagged instances, while recall measures the model's capability to detect genuine anomalies. The AUPRC provides an overall evaluation of the model's performance across varied thresholds, offering insights into precision for different recall levels. Understanding the associated costs of missed anomalies (false negatives) versus falsely detected anomalies (false positives) in specific use cases is critical, necessitating threshold customization for optimal performance in real-world applications.

The findings [2] indicated that, on average, reconstruction-based methods exhibited superior performance, followed by generative methods, with forecasting-based models demonstrating the least effectiveness in anomaly detection. To substantiate these observations using authentic process data, 22 models underwent training, testing, and validation phases utilizing experimental data derived from a continuous distillation plant operating with water. All methods were trained equally with a training data set of five runs with about 1000 time steps each. The starting phase of the plant was in all data sets removed, and all training data were free of anomalies and normalized. For a better comparison, the hyperparameters were not fixed but optimized with cross-validation, as in our previous evaluation[2] limited to a maximum of 24 hours, to adjust them to this new data set. For this cross-validation, we split the training set into folds of the same size. The training excluded one of these folds, on which we validated the trained model. This process was repeated by switching the folds to have every single one as a validation fold one time. Therefore, the worst metrics scores for one of those validation folds marked the final score of the used hyperparameters. The ones with the best final score were chosen by repeating this process with different values and combinations of the possible hyperparameters. Afterward, all the optimal trained models were evaluated on a test data set containing ten other runs than the training set with 1,000-11,000 time steps each. These data contain anomalies of varying length and type. Again, for this evaluation, we compared the methods regarding their overall F1-Score and the AUPRC and ranked the average performance of their achieved ranking in both.

* 1. Results

Table 1: Results of the evaluation of 22 AD methods on real process data with water.
\* The combined ranking obtained with TEP process data is given in brackets.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Method** | **Method Type** | **F1-Score** | **F1-Score ranking** | **AUPRC** | **AUPRC Ranking** | **Combined Ranking \*** |
| MTAD\_GAT[15] | Hybrid | 0.9106 | 1 | 0.9442 | 1 | 1 (21) |
| LSTM\_AE\_OCSVM[16] | Hybrid | 0.6460 | 2 | 0.7435 | 2 | 2 (16) |
| THOC[17] | Hybrid | 0.6223 | 3 | 0.6702 | 4 | 3 (22) |
| TADGAN[18] | Generative GAN based | 0.5893 | 4 | 0.6723 | 3 | 4 (18) |
| LSTM\_VAE\_GAN[19] | Generative VAE based | 0.5101 | 5 | 0.6445 | 5 | 5 (15) |
| MSCRED[20] | Reconstruction based | 0.4519 | 6 | 0.6004 | 6 | 6 (5) |
| LSTM\_AE[21] | Reconstruction based | 0.3484 | 7 | 0.4900 | 7 | 7 (4) |
| DONUT[22] | Generative VAE based | 0.2464 | 9 | 0.4111 | 8 | 8 (6) |
| TKN\_AE[23] | Reconstruction based | 0.2580 | 8 | 0.3831 | 9 | 9 (7) |
| BEATGAN[12] | Generative GAN based | 0.2232 | 10 | 0.3368 | 14 | 10 (1) |
| DENSE\_AE[24] | Reconstruction based | 0.1964 | 15 | 0.3586 | 10 | 11 (3) |
| GMM\_VAE[25] | Generative VAE based | 0.2084 | 11 | 0.3353 | 15 | 12 (13) |
| UNTRAINED\_AE[26] | Reconstruction based | 0.1988 | 14 | 0.3440 | 12 | 13 (10) |
| LSTM\_MAX\_AE[14] | Reconstruction based | 0.2007 | 13 | 0.3386 | 13 | 14 (14) |
| USAD[24] | Reconstruction based | 0.1926 | 16 | 0.3583 | 11 | 15 (12) |
| OMNI[27] | Generative VAE based | 0.2010 | 12 | 0.3276 | 18 | 16 (8) |
| STGAT MAD[28] | Reconstruction based | 0.1770 | 19 | 0.3297 | 16 | 17 (19) |
| LSTM VAE Park[8] | Generative VAE based | 0.1864 | 17 | 0.3190 | 19 | 18 (11) |
| MADGAN[29] | Generative GAN based | 0.1681 | 20 | 0.3296 | 17 | 19 (20) |
| **Method** | **Method Type** | **F1-Score** | **F1-Score ranking** | **AUPRC** | **AUPRC Ranking** | **Combined Ranking \*** |
| SIS\_VAE[30] | Generative VAE based | 0.1803 | 18 | 0.3077 | 20 | 20 (9) |
| TCN\_AE[31] | Reconstruction based | 0.1387 | 21 | 0.3011 | 21 | 21 (2) |
| GENAD[20] | Reconstruction based | 0.0812 | 22 | 0.2543 | 22 | 22 (17) |

Table 1 displays the results obtained from the experiments with water. The methods are evaluated using the F1 score, AUPRC score, and a combined ranking based on these two scores with their average placement and score differences. MTAD\_GAT, LSTM\_AE\_OCSVM, and THOC perform the best among all the methods, while SIS\_VAE, TCN\_AE, and GENAD have the lowest rankings. F1 and AUPRC scores of the methods are lower with the experimental data set than with the TEP data set. The rankings of these methods [2] with the TEP dataset exhibit significant divergence compared to the results obtained from the experimental process data. For the experimental data set, it was found that the hybrid methods perform best. We conject that this is the case because the data amount is limited and combined method architecture is better capable of training the data logic.

* 1. Conclusions

Hybrid-based deep learning anomaly detection (AD) techniques outperformed others when handling experimental data from the continuous distillation column with a single-component water system. Conversely, despite excelling with TEP data, reconstruction, and generative methods did not demonstrate comparable success with the experimental dataset. We conclude that improving and comparing AD methods using synthetic datasets (like the TEP data) is insufficient to yield high-performance methods for real plants.

All tested AD methods yielded notably low F1 and AUPRC scores on the experimental dataset, suggesting room for improvement by acquiring more process data from the continuous distillation mini-plant. Devising experiments aimed explicitly at generating anomalies that are not easily detectable by a human expert is also a crucial step for more effective training of these methods. It remains a pivotal, challenging aspect for future work. Future work should provide more publicly available experimental and synthetic data from physical modeling [11] and machine learning. We also plan to measure experimental data for the azeotropic distillation of n-butanol and water in future work.

References

1. N. M. Nor et al., 2020, Reviews in Chemical Engineering 36.4, 513–553, DOI: 10.1515/revce-2017-0069

2. F. Hartung et al., 2023, Chemie Ingenieur Technik, in press, DOI: 10.1002/cite.202200238

3. J. J. Downs and E. F. Vogel, 1993, Computers & Chemical Engineering 17.3, 245–255, DOI: 10.1016/0098-1354(93)80018-I.

4. Madakyaru, M., Kini, K.R, 2022, Int. j. inf. tecnol. 14, 3001–3010, DOI:

 10.1007/s41870-022-01046-0

5. J. J. Downs, E. F. Vogel, 1993, Comput. Chem. Eng., 17 (3), 245–255. DOI: https://doi.org/10.1016/0098-1354(93)80018-I

6. J.J.Hopfield, 1982, PNAS, 79 (8) 2554-2558. DOI: https://doi.org/10.1073/pnas.79.8.2554

7. S.Hochreiter, J.Schmidhuber, 1997, Neural computation, MIT Press, 9(8), 1735–1780. DOI: https://doi.org/10.1162/neco.1997.9.8.1735

8. M. Soelch, et al., 2016, arXiv:1602.07109. DOI: https://doi.org/10.48550/arXiv.1602.07109

9. L. Ruff, et al., 2018, Proc. Mach. Learn. Res. 2018, 80, 4393–4402.

10. Lyzlova et al., 1979, Russian Journal of Physical Chemistry (Leningrad) 52.3 : 551-555.

11. E. Forte, F. Jirasek, M. Bortz, J. Burger, J. Vrabec, and H. Hasse, 2019, Chemie Ingenieur Technik, 91.3, 201–214. DOI: 10.1002/cite.201800056.

12. B. Zhou et al., 2019, in Proc. of the 28th Int. Joint Conf. on Artificial Intelligence, 4433–4439, DOI:https://doi.org/10.24963/ijcai.2019/616

13. P. Malhotra, L. Vig, G. Shroff, P. Agarwal, 2015, in Proc. of the 23rd European Symposium on Artificial Neural Networks, ES2015-56.

14. A. H. Mirza, S. Cosan, 2018, in Proc. of the 26th Signal Processing and Communications Applications Conf., IEEE, Piscataway,1–4. DOI: https://doi.org/10.1109/SIU.2018.8404689

15. H. Zhao et al., 2020, in Proc. of the 20th Int. Conf. on Data Mining, IEEE, Piscataway, NJ 2020, 841–850. DOI: https://doi.org/10.1109/ICDM50108.2020.00093

16. M. S. Elsayed et al., 2020, in Proc. of the 16th Symposium on QoS and Security for Wireless and Mobile Networks, 37–45. DOI: https://doi.org/10.1145/3416013.3426457

17. L. Shen, Z. Li, J. T. Kwok, 2020, in Proc. of the 34th Int. Conf. on Neural Information Processing Systems, Curran Associates, Red Hook, 13016–13026, Article 1092.

18. A. Geiger, et al., 2020, in Proc. of the 8th Int. Conf. on Big Data, IEEE, Piscataway, 33–43. DOI: https://doi.org/10.1109/BigData50022.2020.9378139

19. Z. Niu, K. Yu, X. Wu, 2020, Sensors, 20 (13), 3738. DOI: https://doi.org/10.3390/s20133738

20. X. Hua et al., 2022, arXiv:2202.04250, 2022.DOI: https://doi.org/10.48550/arXiv.2202.04250

21. C. Zhang et al., 2019, in Proc. of the 33th AAAI Conf. on Artificial Intelligence, AAAI Press, 1409–1416. DOI: https://doi.org/10.1609/aaai.-v33i01.33011409

22. H. Xu et al., 2018, in Proc. of the 27th Int. World Wide Web Conf. DOI: https://doi.org/10.1145/3178876.3185996

23. Jiehui Xu et al.,2022, ICLR, https://arxiv.org/abs/2110.02642

24. J. Audibert et al., 2020, in Proc. of the 26th Int. Conf. on Knowledge Discovery & Data Mining, 3395–3404. DOI: https://doi.org/10.1145/3394486.3403392

25. Y. Guo, W. Liao, Q. Wang, L. Yu, T. Ji, P. Li, 2018, Proc. Mach. Learn., 95, 97–112.

26. S. Kim et al., 2022, in Proc. of the 36thAAAI Conf. on Artificial Intelligence, 7194–7201. DOI: https://doi.org/10.1609/aaai.v36i7.20680

27. Y. Su et al., 2019, in Proc. of the 25th Int. Conf. on Knowledge Discovery & Data Mining, 2828–2837. DOI: https://doi.org/10.1145/3292500.3330672

28. J. Zhan et al., 2022, in Proc. of the 47th Int. Conf. on Acoustics, Speech, and Signal Processing, IEEE, 3568–3572. DOI: https://doi.org/10.1109/ICASSP43922.2022.9747274

29. D. Li et al., 2019, in Proc. of the 28th Int. Conf. on Artificial Neural Networks, Springer, 703–716. DOI: https://doi.org/10.1007/978-3-030-30490-4\_56

30. L. Li, J. Yan, H. Wang, Y. Jin, 2021, IEEE Trans. Neural Networks Learn.Syst., 32 (3), 1177–1191. DOI: https://doi.org/10.1109/TNNLS.2020.2980749

31. M. Thill, W. Konen, T. Baeck, 2020, in Proc. of the 9th Int. Conf. on Bioinspired Methods and Their Applications, 161–173. DOI: https://doi.org/10.1007/978-3-030-63710-1\_13

32. A. C. Rieth et al., 2017, Additional Tennessee Eastman Process Simulation Data for Anomaly Detection Evaluation, Harvard Dataverse, 2017. DOI: https://doi.org/10.7910/DVN/6C3JR1

33. Ferre, A., Voggenreiter, J., Tönges, Y. et al., 2021, MTZ Worldw 82, 26–31, DOI: https://doi.org/10.1007/s38313-021-0639-x