Machine learning and deep learning approaches for multivariate time series prediction and anomaly detection

Thill, M.

Chapter 8

Conclusion and Outlook

Time series anomaly detection is an intriguing and also challenging topic that will likely not lose any of its relevance in the future. Today, accurate anomaly detection algorithms are already critical in system health monitoring or predictive maintenance (PdM) applications or in detecting intrusions into organizational networks. The need for reliable anomaly detection methods is expected to increase in the coming years. Particularly in the industrial context, in the course of ongoing digitization, it will become necessary to analyze growing volumes of data in an automated manner using sophisticated and efficient algorithms.

During the work on this thesis, we had the opportunity to look into the broad domain of time series analysis and anomaly detection and add several contributions to this field. Our focus was on so-called *unsupervised* machine learning (ML) approaches, and we could introduce several novel algorithms with state-of-the-art performance. In unsupervised anomaly detection problems, a model attempts to learn the normal underlying behavior of a system without an external supervisor. The model's understanding of normality is then used to detect abnormal (anomalous) events. Unsupervised learning tasks are usually considered harder than their supervised counterparts since no target function exists, which defines the notion of normal. Instead, the algorithm has to learn to separate normal from anomalous behavior. The reason why unsupervised learning methods are used for anomaly detection is that labeled data are usually rather sparse. In the following, we will conclude this thesis with a short summary, by answering the research questions formulated at the beginning of the thesis and by discussing open questions, and giving an outlook on possible future work.

8.1 Discussion

We have presented four different unsupervised anomaly detection algorithms (some of them in several variants) in the course of this work: SORAD, DWT-MLEAD, LSTM-AD and, TCN-AE. SORAD learns a simple regression model to predict future values of a time series and simultaneously estimate the prediction errors' mean and variance. It operates fully online (or using small batches, or completely offline) and is up-and-running after a very short transient phase. Overall, due to its capabilities to adapt to new concepts, SORAD could outperform other state-of-the-art algorithms on Yahoo's Webscope S5 benchmark, which contains many non-stationary time series. Also, DWT-MLEAD can be either used in

online or offline settings. The main idea behind DWT-MLEAD is to examine a time series at different time scales using the discrete wavelet transform (DWT) to detect short-range and longer-range anomalies. For long-range anomalies, DWT-MLEAD performs better than SORAD. While SORAD and DWT-MLEAD were mostly tested on relatively short time series with less than 10 000 points, our deep learning approaches LSTM-AD and TCN-AE were applied to time series with length 100 000 or more. LSTM-AD uses a stack of recurrent long short-term memory (LSTM) neural networks. Similarly to SORAD, it is trained to predict normal time series behavior. However, LSTM-AD is extended in many aspects, which allows it to learn complex temporal patterns (such as in ECG data). For complex quasi-periodic time series, we found that the window-based error correction method introduced in Chapter 6 is crucial to improving the overall detection accuracy of LSTM-AD. While we did not test it yet, this method might also be beneficial for some of the other algorithms. It could also be integrated into the actual learning procedure (currently, it is an extra module applied after computing the prediction errors). LSTM-AD is trained offline and cannot adapt to new concepts in non-stationary time series. However, we found that it can be used for time series with weak non-stationary behavior, such as baseline wandering or signal quality changes.

For TCN-AE, we revisited the idea of analyzing time series at different time scales. This is achieved with a convolutional neural network architecture based on so-called dilated convolutions (which have their origin in DWT). TCN-AE is a reconstruction-based algorithm with a novel autoencoder architecture that can also be applied to unpredictable time series. It is possible to increase TCN-AE's receptive field exponentially with only a linear increase in the number of trainable weights. Due to the exponentially increasing receptive field in TCN-AE, it is the only algorithm that can learn long-term correlations in time series. Like LSTM-AD, TCN-AE can also work with time series that exhibit some baseline wandering or changes in the signal quality/noise. It significantly outperforms other state-of-the-art DL architectures in terms of precision & recall, computation time, and the number of trainable weights on the challenging ECG benchmark and MGAB (both introduced in Section 2.3).

Not surprisingly, our DL models LSTM-AD and TCN-AE require more training data than the other approaches, mainly due to the large number of trainable parameters. Hence, they are more likely to perform well when trained on longer time series with several ten-thousand points but not when trained on small data sets with less than a few thousand points. On the other hand, all algorithms can deal with very long time series and have no restrictions in this regard.

The main ingredient for DWT-MLEAD and TCN-AE to reliably detect short- and long-range anomalies simultaneously are their hierarchical temporal architecture and their capability to analyze time series at different time resolutions. The general idea in both approaches is that anomalies might become more apparent on some time scales than others. DWT-MLEAD realizes this with a decimating DWT. In practice, the DWT of a signal is computed by passing it through a series of filters and subsampling layers. The parameters of the filters are not trainable and depend on the choice of the mother wavelet. Similarly, TCN-AE uses dilated convolutional layers (which have their origin in DWTs), however, with

trainable filters. In both approaches, the temporal receptive field can easily be doubled by adding a new layer.

Theoretically, all our algorithms can process multivariate time series. However, in this work, DWT-MLEAD was only tested on univariate time series and would require some minor modifications for higher dimensions.

	Characteristics						
	Small Data Sets	Multivariate Time Series	Analysis at Different Time Scales	Long-term Correlations	Unpredictable Time Series	Weak Non-stationary Behavior	Strong Non-stationary Behavior
ADVec [173]	√					√	√
DNN-AE [46, 58]		√			√	√	
DWT-MLEAD (Ch. 5)	√	(√)	√		√	√	√
LSTM-AD (Ch. 6)		√	(√)	(√)		√	
LSTM-ED [108]		√	(√)		√	√	
NuPIC [160]	√	√				√	√
SORAD (Ch. 4)	√	√				√	√
TCN-AE (Ch. 7)		√	√	√	√	√	

Table 8.1: This table summarizes the suitability of various time series anomaly detection algorithms, given different time series characteristics. A checkmark (in parentheses) indicates that an algorithm can (partially) handle time series with the specified characteristic.

Based on the time series characteristics described in Section 2.4, we tried to indicate in Table 8.1 for all algorithms used in this work, which ones might be suited for which characteristics. For example, we expect that DL algorithms are generally less well suited for data sets that are relatively small (less than a few thousand points). Most algorithms investigated in this work should be applicable to multivariate time series or to time series with weak non-stationary behavior. Algorithms that can run entirely online, such as SORAD, generally tend to perform better than offline algorithms on time series with strong non-stationary behavior.

In order to learn long-term correlations, a long temporal memory is required. Algorithms, such as LSTM-ED or DNN-AE, designed to encode and reconstruct short sub-sequences, do not scale well for increasing sub-sequence lengths and usually cannot learn long-term correlations very well. On the other hand, TCN-AE exhibits a much longer memory than recurrent architectures (e.g., LSTM or GRU) with the same capacity (network size) and does not suffer from vanishing/exploding gradients. Hence, TCN-AE could be a reasonable choice if long-term correlations in time series have to be learned. As described before, SORAD, NuPIC, and LSTM-AD will likely not perform well on unpredictable (in the sense of forecastable) time series since all three approaches rely on single-step or multi-step ahead prediction.

8.2 Conclusions

In Sec. 1.1.3, we posed research questions that motivated our work in the following. At this point, we would like to revisit these questions and highlight the contributions that emerged in their context.

Q1: Is it possible to successfully train and apply novel unsupervised machine learning models for anomaly detection and do they advance the state of the art?

A1: Although it is challenging to analyze time series in an unsupervised fashion, especially when considering the data's temporal nature, we can answer this question mostly positively. Overall, we introduced four different anomaly detection algorithms for time series, which can be used in different contexts and show competitive performance when benchmarked against other state-of-the-art algorithms: The online SORAD could significantly outperform other algorithms [173, 160] on non-stationary time series with short-term anomalies. Our second online algorithm, DWT-MLEAD, is able to analyze time series on different time scales and detect short-term and longer-term anomalies. It is better than other algorithms [173, 160] when applied to benchmarks with anomalies being diverse in their time scales. LSTM-AD is a DL model with several enhancements that obtains state-of-the-art results on quasi-periodic time series such as ECG signals or the Mackey-Glass Anomaly Benchmark (MGAB). TCN-AE is a novel reconstruction-based DL approach that analyzes time series at different time scales and can learn long-term relationships. For a real-world anomaly detection task for ECG time series, TCN-AE outperforms other strong (DL) anomaly detection algorithms [160, 46, 108] with respect to detection accuracy, model size, and computation time, significantly improving the state of the art. All algorithms presented in this work are *unsupervised* and do not require any ground truth labels for the training process. Only for evaluation purposes, the anomaly labels are partly required, for example, to configure the anomaly threshold of all algorithms in a way that allows a fair comparison (e.g., by achieving equal accuracy, EAC). Furthermore, none of the algorithms requires solely normal data in order to learn the regular behavior of a time series (up to 5% of the data was anomalous in the examples used in our experiments), demonstrating the algorithms noise resilience. Contrary to other state-of-the-art algorithms, all of our algorithms are applicable to time series with two or more dimensions. However, due to the lack of suitable benchmarking data, we did not experiment with high-dimensional time series.

Q2: Given certain characteristics of the time series data, can we advise which algorithm is most suited for detecting anomalies?

A2: To answer this research question, we investigated two aspects: (1) What are general characteristics/properties of time series which are important in the context of anomaly detection and for choosing suitable algorithms? (2) How do existing state-of-the-art algorithms deal with these properties and can we find better approaches to analyze time series with different characteristics? Based on the different problems that we studied, we found several answers to both questions in the individual chapters, which are discussed on a higher level considering the overall context in Section 8.1.

In summary, we identified several recurring characteristics appearing in time series. Each proposed algorithm addresses one or more of these characteristics. However, there is no universally applicable algorithm. All algorithms have their justification for different problems. An important decision criterion for the choice of the algorithm is the data size. Deep learning approaches (TCN-AE, LSTM-AD) tend to be less well suited for small data sets. Another factor for many problems is to analyze a time series and to detect anomalies across different frequency scales. The capability to learn long-term dependencies is necessary for many types of problems and requires that the model has a large temporal receptive field (TCN-AE) or an efficient temporal memory (LSTM-AD). Online and offline algorithms should be able to process time series with weak forms of non-stationarity (e.g. baseline wandering, changes in signal quality) which are ubiquitous in real-world problems. For strong non-stationary behavior, algorithms with online adaptability (DWT-MLEAD, SORAD) are needed that are stable on the one side and, on the other side, can learn new concepts in the data. Finally, the dimension of a time series plays an important role in the choice of the algorithm. Some algorithms are better suited for multivariate time series than others. Although already mentioned (and although not directly related to the nature of a time series), we want to emphasize that the available time series data usually does not permit to train supervised learning models in practice, due to the sparse amount of labeled data. Hence, all our algorithms are trained in an unsupervised fashion.

Q3: How can online learning methods be successfully used for anomaly detection in time series or data streams?
A3: We investigated this research question mainly in Chapter 4 and Chapter 5. Note that while all our models can be run online on new data after training, only SORAD and DWT-MLEAD are online adaptable; hence, we do not have to train them offline. Instead, they only require a very short transient phase to be ready for use, and they can continually adapt to new concepts in the data. SORAD uses the recursive least squares (RLS) algorithm to learn its prediction model and estimates the mean and (co-) variance of the prediction errors online. Additionally, it is possible to add a certain amount of forgetting to the RLS model and to the estimation of the error distribution, which enables the online adaptability of the algorithm. Although the regression model is linear, it can be augmented with non-linear features, such as polynomials or radial basis functions (RBFs). DWT-MLEAD algorithm can compute the (causal and decimating) discrete wavelet transform entirely online. For individual frequency scales of the current DWT, the algorithm estimates a mean vector and covariance matrix in an online fashion. It is also possible to add forgetting to DWT-MLEAD so that the algorithm can adapt itself in non-stationary environments. The online adaptability of SORAD and DWT-MLEAD appears to be beneficial for time series with concept changes or drifts, as our experiments showed. Furthermore, both algorithms did not show any signs of numerical instabilities or other diverging behavior.

LSTM-AD and TCN-AE are not online-adaptable in the sense that they continue adjusting their weights after the (offline) training. However, both algorithms are capable of dealing with non-stationary artifacts such as baseline wandering.

So far it is an open question to us whether DL models can be trained online. Today, most online learning algorithms are designed to learn shallow models with convex optimization (e.g., linear least squares) techniques. In this work we did not investigate to which extent DL models, with highly non-convex objective functions, can be trained in an online fashion. Hence, this research question cannot yet be answered conclusively and has much potential for future investigations.

8.3 Future Work

Although we explored some challenges mentioned in the introduction and present novel approaches to time series anomaly detection, many challenges beyond this work remain that research could address in future work. Apart from the more detailed discussions in the previous chapters on possible further work on individual problems and improvements to particular algorithms, we see some more general points which are worth investigating in the future:

Need for Better Benchmarks Although several benchmarks for time series anomaly detection are publicly available and we could also design a challenging synthetic benchmark based on Mackey-Glass time series and construct a complex benchmark based on ECG data, we have the impression that many of the current data sets do not adequately reflect the reality that will prevail in the future. It will become increasingly difficult to assess algorithms' performance regarding their practicability for real-world problems based on the current benchmarks. Many currently popular benchmarks contain only relatively short (several thousand points) and only one-dimensional time series. However, applications today collect thousands of high-dimensional data points in a short time. Especially for benchmarking DL approaches, many current benchmarks are insufficient due to their somewhat limited size. Other issues are triviality (i.e., simple thresholding methods can detect some anomalies), insertion of unrealistic synthetic anomalies into normal data, or the translation of traditional classification problems into anomaly detection problems by simply re-labeling minority classes as anomalies. Another common problem is that not always guidelines exist on how to measure an algorithm's performance. For the same benchmark, researchers assess algorithms' performance in very different ways and report performance metrics that are incomparable to other works. For these reasons, we believe there is a need for better benchmarks in time series anomaly detection, which are relevant to practice and widely accepted among the research community and allow a fair and thorough evaluation of different approaches. These new benchmarks also require standardized rule sets that clearly state how to assess the performance of an algorithm.

Interpretability Another issue that will likely get more relevant in the context of (time series) anomaly detection is the *interpretability* of models. Interpretability is concerned with the problem of getting some explanation or understanding for the decisions made by our

models. In the context of anomaly detection, interpretability is linked to the question of whether an algorithm can also identify the underlying cause of a detected anomaly. Usually, an anomaly is associated with a problem in the monitored system or process. Hence, in many real-world applications, it is not sufficient to solely detect anomalous behavior. To fix the problem and possibly prevent further harm, it is also critical to identify the actual source of the problem and find a way to handle this problem. In some cases, human experts might be able to analyze the anomalous pattern and recognize the cause. However, especially in high-dimensional & high-frequency time series, where anomalies might occur only in a small subset of the high-dimensional space, this becomes increasingly difficult. In other cases, it might be of interest to understand an algorithm's decision to reduce algorithmic bias (e.g., caused by biased training data) or to justify individual decisions. While researchers have invested much effort in developing anomaly detection algorithms in recent years, interpretable approaches have not received as much attention yet. Overall, we see much potential in this area.

Integrating Expert Feedback All the anomaly detection algorithms presented in this work and many approaches described in the literature are trained in an unsupervised fashion. The main reason for this is that the available labeled data is sparse, and the few labeled instances are needed for tuning and validation purposes. Unsurprisingly, many anomaly detection models have a relatively low performance when initially deployed. That is, the models produce many false alarms or overlook real anomalous behavior. One idea to improve the overall prediction quality could be to integrate expert feedback into the model during operation since human experts (such as machine operators) are commonly involved in anomaly detection tasks and have to check and acknowledge alarms. With such a human-in-the-loop (HITL) approach, the model could gradually adapt itself and improve its predictions. However, one has to keep in mind that also expert feedback will mostly be sparse and has to be handled efficiently so that the improvement will become apparent in a reasonable time. Although some work exists on how to incorporate expert feedback into anomaly detection [32, 138, 136], it is a mostly open research question how such additional information can be utilized efficiently, apart from simple adjustments of the anomaly threshold.