




From Univariate to Multivariate Time Series Anomaly Detection with Non-Local Information

Julien Audibert^{1,2(✉)}, Sébastien Marti², Frédéric Guyard³,
and Maria A. Zuluaga¹ 

¹ EURECOM, Sophia Antipolis, France
{audibert.julien,maria.zuluaga}@eurecom.fr

² Orange, Sophia Antipolis, France
sebastien.marti@orange.com

³ Orange Labs, Sophia Antipolis, France
frederic.guyard@orange.com

Abstract. Deep neural networks (DNNs) are attractive alternatives to more traditional methods for time series anomaly detection thanks to their capacity to automatically learn discriminative features. Despite their demonstrated power, different works have suggested that introducing engineered features in the time series can further improve the performance. In this work, we present a feature engineering strategy to transform univariate time series into a multivariate one by introducing non-local information in the augmented data. In this way, we aim to address an intrinsic limitation of the features learned by DNNs, which is they rely on local information only. We study the performance of our combination compared to each individual method and show that our method achieves better performance without increasing computational time on a set of 250 univariate time series proposed by the University of California, Riverside at the 2021 KDDCup competition.

Keywords: Anomaly detection · Time series · Feature engineering · Non-local information

1 Introduction

A time series is a set of measured values that model and represent the behavior of a process over time. Time series are used in a wide range of fields such as healthcare [8], industrial control systems [2], and finance [15]. Detecting behavior or patterns that do not match the expected behavior of previously visualized data is a critical task and an active research discipline called time series anomaly detection [3, 5]. Numerous methods to address this problem have been developed in recent years including statistical, machine learning and deep neural networks (DNNs) methods.

The performance of machine learning algorithms is correlated to the quality of the extracted features [14]. Feature engineering for augmenting time series

data is usually done by bringing external but correlated information as an extra variate to the time series. This, however, requires domain knowledge about the measured process. Another strategy is to create local features on the time series, such as moving averages or local maximum and minimum. Both strategies, as they are manual, are not very efficient, time consuming and require high domain knowledge expertise [7]. In theory, DNNs have emerged as a promising alternative given their demonstrated capacity to automatically learn local features, thus addressing the limitations of more conventional statistical and machine learning methods. Despite their demonstrated power to learn such local features, it has been shown that feature engineering can accelerate and improve the learning performance of DNNs [4].

In this work, we propose a novel feature engineering strategy to augment time series data in the context of anomaly detection using DNNs. Our goal is two-fold. First, we aim to transform univariate time series into multi-variate time series to improve DNNs performance. Second, we aim to use a feature engineering strategy that introduces non-local information into the time series, which DNNs are not able to learn. To achieve this, we propose to use a data structure called Matrix-Profile as a generic non-trivial feature. Matrix-Profile allows to extract non-local features corresponding to the similarity among the sub-sequences of a time series. The main contributions of this paper are:

- We propose an approach that transforms univariate time series into multivariate by using a feature engineering strategy that introduces non-local information to improve the performance of DNNs.
- We study and analyze the performance of this approach and of each method separately using the KDDCup 2021 dataset consisting of 250 univariate time series.

The rest of this paper is organized as follows. Section 2 briefly reviews other works on feature engineering for anomaly detection in time series. The Sect. 3 presents the transformation of univariate time series into multivariate one and the methods which constitute our framework. Section 4 describe the experiments and demonstrate the performance of our approach. The paper concludes with some discussion and perspectives in Sect. 5.

2 Related Works

Different studies have raised the importance of feature engineering for the detection of anomalies and the superiority of multivariate models in time series. A first study conducted by Carta *et al.* [4] shows that in network anomaly detection, the introduction of new features is essential to improve the performance of state-of-the-art solutions. Fesht *et al.* [7] compare the performance of manual and automatic feature engineering methods on drinking-water quality anomaly detection. The study concludes that automatic feature engineering methods obtain better performances in terms of F1-score. Ouyand *et al.* [11] shows that feature extraction is one of the essential keys for machine learning and proposes a method called

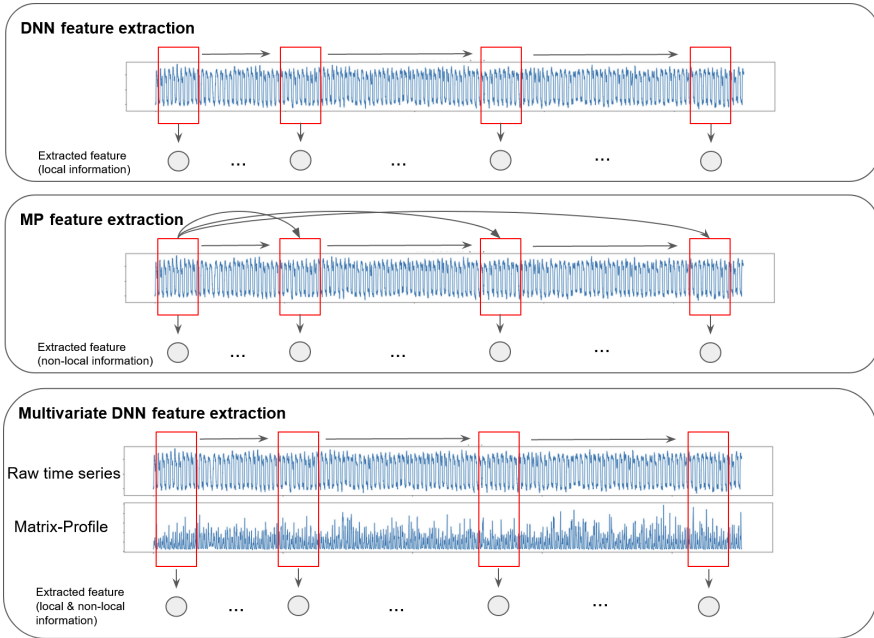


Fig. 1. Top: DNN automatic feature learning and extraction is limited to a local neighborhood, which is typically represented by the input window information. Middle: the matrix profile algorithm relies on non-local features, which are obtained by comparing every window of the time series. Bottom: the proposed strategy brings non-local feature information to a DNN by transforming the original univariate time series into a multivariate one by combining the raw time series and the non-local information obtained with matrix profile.

hierarchical time series feature extraction used for supervised binary classification. Finally, in [1], the authors conclude that multivariate models provided a more precise and accurate forecast with smaller confidence intervals and better measures of accuracy. Thus, studies have demonstrated the importance of feature engineering to improve anomaly detection models as well as the performance of multivariate methods compared to univariate ones on time series. Motivated by these ideas, our work aims to investigate how feature engineering using non-local information to achieve variate augmentation in time series can improve the performance of anomaly detection DNN models in univariate time series.

3 From Univariate to Multivariate Time Series

To take advantage of the performance of multivariate methods of anomaly detection on univariate time series it is necessary to transform the univariate time series into multivariate one. This can be achieved by adding external information to the time series, which requires specific domain knowledge. Our strategy,

instead, transforms the univariate time series into a multivariate one, without any further information than the original time series, and is generic in that no specific knowledge on what the time series represents is required.

Our strategy consists in building another time series (i.e. another variate) by extracting non-local information from the raw time series, which DNN approaches fail to obtain as they typically operate in local neighborhood. To this end, we make use of the Matrix-Profile (MP) [16, 17], a data structure for time series analysis. The proposed strategy is illustrated in Fig. 1.

The Matrix profile estimates the minimal distance between all sub-sequences of a time series. Thus, the Matrix-Profile value for a given sub-sequence is the minimum pairwise Euclidean distance to all other sub-sequences of the time series. A low value in the matrix profile indicates that this sub-sequence has at least one relatively similar sub-sequence located somewhere in the original series. In [9], it is shown that a high value indicates that the original series must have an abnormal sub-sequence. Therefore the matrix profile can be used as an anomaly score, with a high value indicating an anomaly.

In our approach, we propose to use the anomaly score obtained by Matrix-Profile over a given time series and merge it point-by-point with the original data. This can be thus seen as a data augmentation procedure using non-local information from the same signal.

As the new time series is just a multivariate time series, any given anomaly detection method can be used to identify anomalous points in it. In this work, we investigate three different estimation model-based techniques [3] as base anomaly detection methods. Among these category of methods, the auto-encoder [13] is among the most commonly used. An auto-encoder (AE) is an artificial neural network combining an encoder E and a decoder D . The encoder part takes the input window W and maps it into a set of latent variables Z , whereas the decoder maps the latent variables Z back into the input space as a reconstruction \widehat{W} . The difference between the original input vector W and the reconstruction \widehat{W} is called the reconstruction error. Thus, the training objective aims to minimize this error. Auto-encoder-based anomaly detection uses the reconstruction error as the anomaly score. Time windows with a high score are considered to be anomalies [6].

Alongside the AE, we consider a more complex approach based on a Variational AutoEncoder (VAE) coupled with a recurrent neural network, the Long Short-Term Memory Variational Auto-Encoders (LSTM-VAE) [12]. In the LSTM-VAE, the feed forward network of the VAE is replaced by a Long Short-Term Memory (LSTM), which allows to model the temporal dependencies. As in the AE, the input data is projected in a latent space. However, differently from the AE, this representation is then used to estimate an output distribution and not to simply reconstruct a sample. An anomaly is detected when the log-likelihood is below a threshold.

The third estimation model-based method we consider is denoted UnSupervised Anomaly Detection (USAD) [2]. USAD is composed of three elements: an encoder network and two decoder networks. The three elements are connected into an architecture composed of two auto-encoders sharing the same

encoder network within a two-phase adversarial training framework. The adversarial training allows to overcome the intrinsic limitations of AEs by training a model capable of identifying when the input data does not contain an anomaly and thus perform a good reconstruction. At the same time, the AE architecture allows to gain stability during adversarial training of the two decoders.

The architecture is trained in two phases. First, the two AEs are trained to learn to reconstruct the normal input windows. Secondly, the two AEs are trained in an adversarial way, where the first one seeks to fool the second one, while this latter one aims to learn when the data is real (coming directly from the input) or reconstructed (coming from the other autoencoder). As with the base AE, the anomaly score is obtained as the difference between the input data and the data reconstructed by the concatenated autoencoders.

4 Experiments and Results

This section first describes the datasets used and the experimental setup used in our work. Then, we study the performance of our proposed approach and compare it against other techniques.

4.1 Datasets

In our experiments we use 250 univariate time series proposed by the University of California, Riverside at the 2021 KDDCup competition, consisting of univariate time series from many different fields. The 250 time series are composed of a training part containing data considered as normal and a test part containing one anomaly. The time series range from 6680 points for the smallest to 900000 points for the largest. The length of the training set represents on average 31% of the total length of the time series (i.e. a training on the first 31% points of the time series and a test on the next 69% points) with a minimum length of 2.5% and a maximum of 76.9%. All the time series are min-max normalized.

4.2 Experimental Setup

We use the percentage of correctly labeled series to assess the performance of our method. A time series is considered to be correctly predicted when the index of the point labeled as anomalous is included in a window of 100 points around the true anomaly.

We compare our method against the matrix-profile (MP), the auto-encoder (AE), the LSTM-VAE and USAD without the transformation of the time series. We compute the performance of the three anomaly detection methods AE, LSTM-VAE and USAD on a transformed univariate time series obtained using only non-local information, i.e. with Matrix-profile (MP-AE, MP-LSTM-VAE and MP-USAD). We assess both the AE, LSTM-VAE and USAD's performance using the proposed multivariate transformation, consisting of the original raw time series and the series obtained with MP, respectively (TS+MP)-AE,

Table 1. Hyper-parameter settings of the different methods

Method	Parameters
MP	<i>window_size = 100, discords = True</i>
AE	<i>window_size = 100, latent_dimension = 10, Epochs = 100</i>
LSTM-VAE	<i>window_size = 100, Epochs = 100</i>
USAD	<i>window_size = 100, latent_dimension = 10, Epochs = 100</i>

Table 2. Methods performance and computational time.

Method	Performance	Train and Test time (s $\times 10^3$)
Matrix-Profile	0.416	1.47
AE	0.236	22.00
LSTM-VAE	0.198	85.31
USAD	0.276	29.00
MP-AE	0.292	22.16
MP-LSTM-VAE	0.344	84.30
MP-USAD	0.404	29.10
(TS+MA)-AE	0.148	22.38
(TS+MA)-LSTM-VAE	0.134	85.43
(TS+MA)-USAD	0.176	29.12
(TS+MP)-AE	0.536	22.50
(TS+MP)-LSTM-VAE	0.446	85.83
(TS+MP)-USAD	0.488	29.28

(TS+MP)-LSTM-VAE and (TS+MP)-USAD. To validate the relevance of the use of non-local information in the transformation of the time series, we also consider an identical combination with a local feature engineering strategy. In particular, in our experiments we use the moving average (MA), respectively (TS+MA)-AE, (TS+MA)-LSTM-VAE and (TS+MA)-USAD).

Implementation. We implement the AE using Pytorch and we used publicly available implementations for MP[1]¹, LSTM-VAE² and USAD³. Table 1 details the hyper-parameter setup used for each method. Where a parameter is not specified, it indicated that we used those set by default in the original implementation

All experiments are performed on a machine equipped with an Intel(R) Xeon(R) CPU E5-2699 v4 @ 2.20 GHz and 270 GB RAM, in a docker container

¹ <https://stumpy.readthedocs.io>.

² <https://github.com/TimyadNyda/Variational-Lstm-Autoencoder>.

³ <https://github.com/robustml-eurecom/usad>.

running CentOS 7 version 3.10.0 with access to an NVIDIA GeForce GTX 1080 Ti 11 GB GPU.

4.3 Results

Table 2 presents the results obtained by the different methods in terms of performance accuracy and computational times. Interestingly, we observe that the performance of DNN-based methods on univariate time series is very low and largely surpassed by the more conventional approach, the matrix profile. However, once the same techniques use the proposed data transformation strategy, we observe an important boost in their performance. The Auto-Encoder and the LSTM-VAE score almost 2.3 times higher when the combination of the matrix profile and real data is used as input instead of the original data. Similarly, USAD’s performance increases by 1.8 times when the matrix profile and raw time series combination is used compared to its performance using only the raw time series.

Nevertheless, we observe that the non-local transformation alone is not enough to boost the performance of DNN methods. For instance, if the input consists only of the univariate time series transformed using the matrix profile, while there is some increased performance, this one is milder than when using a multivariate time series. This confirms that DNN methods perform better in a multivariate setup for anomaly detection.

Regarding the use of local features, i.e. the moving average, we observed that adding it does not allow USAD, LSTM-VAE and AE to increase their performance. Indeed, the combination of raw time series and moving average degrades the performance of AE and USAD by about 0.1 and the performance of LSTM-VAE by about 0.06. This suggests that any local features that might be discriminative can be extracted by the DNNs and introducing new manually crafted ones may be detrimental.

Finally, as it is expected, the computational time of DNN-based methods is much longer than the MP. However, what is interesting in our findings is that the computational time of DNN methods is very little impacted when the dimension of the time series increases. In fact, the AE’s computational time goes from 21993 s in the fastest univariate configuration to 22491 s in the multivariate case. This means an increase of only 2.2% on computational time for a gain in performance of 230%.

5 Discussion and Conclusions

In this paper, we propose an approach to augment univariate time series using a feature engineering strategy that introduces non-local information in the generation of an additional variate to the series. In this way, we expect to address a limitation of DNNs, as they are not conceived to learn automatically non-local features. We achieve automatic non-local feature extraction by relying on the

Matrix-Profile, a method that computes the minimum pairwise Euclidean distance of all subsequences of the time series, and combining its output with the original time series.

We used data from the KDDcup 2021 competition containing 250 univariate time series to study the performance of our method. The performance analysis highlighted the relevance of transforming the univariate time series using the proposed feature engineering and data augmentation strategy. Our results show that introducing non-local information to augment the dimension of the series improves the performance of DNN methods. For instance, by using a very simple method, such as an autoencoder, we were able to obtain a gain in performance of 230%, without significantly increasing the computational time. As such, our preliminary results suggest that non-local information represents an important source of additional information that can increase performance of DNN methods.

While our approach focuses on the particular case of transforming uni- to multivariate time series, this idea could be used to augment time series, which are multivariate at origin, as a way to introduce non-local information.

In this work, we used three methods of anomaly detection based on Deep Neural Networks in combination with Matrix profile. The good performance on a simple auto-encoder, a recurrent network such as LSTM-VAE and USAD, a state-of-the-art neural network, suggest that our combination could generalize to other DNN methods. Therefore, future works should explore other feature engineering techniques that can provide non-local information, as well as other multivariate DNN anomaly detection methods.

Finally, our findings are consistent with one of the results of the time series prediction competition, the M4 challenge [10], which highlighted the predictive power of ensemble approaches combining learning-based with more conventional statistical methods. Due to the great success of DNN methods in the recent years, it is now often the case that more traditional methods are overseen. Our results suggest that the use of hybrid approaches should be further explored.

References

1. Aboagye-Sarfo, P., Mai, Q., Sanfilippo, F.M., Preen, D.B., Stewart, L.M., Fatovich, D.M.: A comparison of multivariate and univariate time series approaches to modelling and forecasting emergency department demand in western australia. *J. Biomed. Inform.* **57**, 62–73 (2015)
2. Audibert, J., Michiardi, P., Guyard, F., Marti, S., Zuluaga, M.A.: USAD: unsupervised anomaly detection on multivariate time series. In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2020*, pp. 3395–3404. Association for Computing Machinery, New York (2020)
3. Blázquez-García, A., Conde, A., Mori, U., Lozano, J.A.: A review on outlier/anomaly detection in time series data. *ACM Comput. Surv. (CSUR)* **54**(3), 1–33 (2021)
4. Carta, S., Podda, A.S., Reforgiato Recupero, D.R., Saia, R.: A local feature engineering strategy to improve network anomaly detection. *Future Internet* **12**(10), 177 (2020)

5. Domingues, R., Filippone, M., Michiardi, P., Zouaoui, J.: A comparative evaluation of outlier detection algorithms: experiments and analyses. *Pattern Recogn.* **74**, 406–421 (2018)
6. Fan, C., Xiao, F., Zhao, Y., Wang, J.: Analytical investigation of autoencoder-based methods for unsupervised anomaly detection in building energy data. *Appl. Energy* **211**, 1123–1135 (2018)
7. Fehst, V., La, H.C., Nghiem, T.D., Mayer, B.E., Englert, P., Fiebig, K.H.: Automatic vs. manual feature engineering for anomaly detection of drinking-water quality. In: *Proceedings of the Genetic and Evolutionary Computation Conference Companion, GECCO 2018*, pp. 5–6. Association for Computing Machinery, New York (2018)
8. Kale, D.C., et al.: An examination of multivariate time series hashing with applications to health care. In: *2014 IEEE International Conference on Data Mining*, pp. 260–269 (2014)
9. Linardi, M., Zhu, Y., Palpanas, T., Keogh, E.J.: Matrix profile goes mad: variable-length motif and discord discovery in data series. *Data Min. Knowl. Disc.* **34**, 1022–1071 (2020)
10. Makridakis, S., Spiliotis, E., Assimakopoulos, V.: The M4 competition: 100,000 time series and 61 forecasting methods. *Int. J. Forecast.* **36**(1), 54–74 (2020)
11. Ouyang, Z., Sun, X., Yue, D.: Hierarchical time series feature extraction for power consumption anomaly detection. In: Li, K., Xue, Y., Cui, S., Niu, Q., Yang, Z., Luk, P. (eds.) *LSMS/ICSEE -2017. CCIS*, vol. 763, pp. 267–275. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-6364-0_27
12. Park, D., Hoshi, Y., Kemp, C.C.: A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder. *IEEE Robot. Autom. Lett.* **3**(3), 1544–1551 (2018)
13. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning Internal Representations by Error Propagation, pp. 318–362. MIT Press, Cambridge (1986)
14. Soni, A.N.: Feature extraction methods for time series functions using machine learning. *Int. J. Innov. Res. Sci. Eng. Technol.* **7**(8), 8661–8665 (2018)
15. Theodossiou, P.T.: Predicting shifts in the mean of a multivariate time series process: an application in predicting business failures. *J. Am. Stat. Assoc.* **88**(422), 441–449 (1993)
16. Yeh, C.M., et al.: Matrix profile I: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. In: *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pp. 1317–1322 (2016)
17. Yeh, C.C.M., Kavantzas, N., Keogh, E.: Matrix profile VI: meaningful multidimensional motif discovery. In: *2017 IEEE International Conference on Data Mining (ICDM)*, pp. 565–574. IEEE (2017)