RNN-based Early Cyber-Attack Detection for the Tennessee Eastman Process

Pavel Filonov [1] **Fedor Kitashov** [1] **Andrey Lavrentyev** [1]

Abstract

An RNN-based forecasting approach is used to early detect anomalies in industrial multivariate time series data from a simulated Tennessee Eastman Process (TEP) with many cyber-attacks. This work continues a previously proposed LSTM-based approach to the fault detection in simpler data. It is considered necessary to adapt the RNN network to deal with data containing stochastic, stationary, transitive and a rich variety of anomalous behaviours. There is particular focus on early detection with special NAB-metric. A comparison with the DPCA approach is provided. The generated data set is made publicly available.

1. Introduction

Modern Industrial Control Systems (ICS) deals with multivariate time series data of technological processes: sensors and controls signals. Comprising a cyber components, ICSs are a target of cyber-attacks (for example (Lee et al., 2014)), that can modify sensor and controls values, or the parameters of control logic (set points). Such cyber-attacks can be detected as an anomalies in technological signals. This raises the issue of early anomaly detection.

Different approaches have been proposed to detect anomalies in industrial data. Anomalies can arise for different reasons, besides cyber-attacks: equipment malfunctions, human errors, analogous signals interruptions, etc. Here we provide only a short overview of such approaches: RNN-based (Nanduri et al., 2016), LSTM-based forecasting (Filonov et al., 2016; Malhotra et al., 2015) and encoder-decoder (Malhotra et al., 2016), clustering based (Kiss et al., 2015), PCA, DPCA, FDA, DFDA, CVA, PLS (Chiang et al., 2001), one-class SVM and segmentation (Marti et al., 2015), change point detection (Matteson & James, 2013), process invariants (Adepu & Mathur, 2016).

One of the main problems with the verification of proposed approaches is the lack of available industrial datasets with labelling of normal and anomalous behaviour as well as the absence of rich anomalous behaviour examples. Finding data from real objects under cyber-attacks is problematic because these are quite unique incidences and industry vendors do not want to share such data. Experimenting with attacks on real *test* objects is not a solution because it is very costly. One of a possibility for generating anomalous behaviour is data augmentation as in (Yadav et al., 2016). Another possibility is to use a mathematical model of a cyber-physical system for both physics and control dynamics and simulate multiple realistic cyber-attacks. In our previous work (Filonov et al., 2016) we used this approach with a gasoil heating loop process (GHL) (GHL, 2016) implemented with the Modelica tool. The generated data is quite rich but it lacks of some stochastic properties and reflects a rather simple control logic.

In the current paper we use the well-known TEP model (Downs & Vogel, 1993; Ricker, 2013) which allows rich and realistic datasets to be generated. Cyber-attack simulation using TEP was proposed in (Krotofil, 2014) and implemented in the Matlab/Simulink tool and .NET code. We used our own implementation of the TEP model completely in Python code which allowed us to simulate a lot of cyber-attacks and generate datasets as well as a streaming data.

To detect anomalies in TEP data we further developed the RNN-based forecasting approach that we used for GHL data. TEP data requires the RNN network to be adapted in order to deal with stochasticity, stationary and transitive behaviours. We also focused more on early detection and for this purpose used Numenta Anomaly Benchmark (NAB) metric (Lavin & Ahmad, 2015). We provide a comparison with the fault detection approach traditionally used for TEP based on DPCA (Chiang et al., 2001), and which we combined here with the NAB-metric.

2. Dataset Description

The TEP model is represented in Figure 1. It was simulated at different normal modes and under cyber-attacks. The generated datasets characteristics are represented in Table 1.

[1] Kaspersky Lab, Moscow, Russian Federation. Correspondence to: Andrey Lavrentyev <Andrey.Lavrentyev@kaspersky.com>.

ICML 2017 Time Series Workshop, Sydney, Australia, 2017.

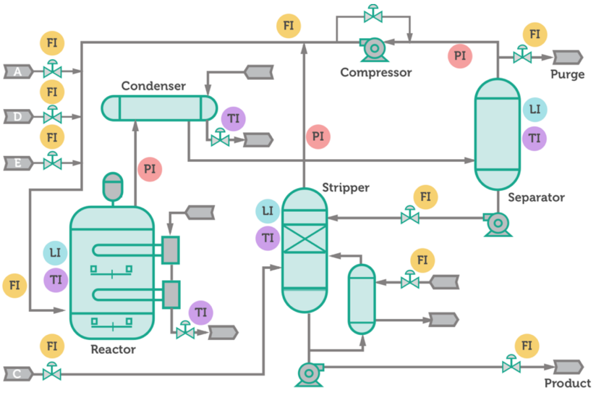

Figure 1. Tennessee Eastman Process

We generated a training dataset with 201 single-mode and 336 transient-mode samples and a test dataset with 142 MEAS/MV/SP attacks samples (TEP, 2017). Each sample is a multivariate time series of dimension 59. Besides samples for 7 single modes of TEP operation we generated samples for 28 transient modes via 4 variants of SP changes for each single mode: decreasing by 2% catalyst C purge, changing product mix by 10%, decreasing product rate by 15%, decreasing reactor pressure by $1-2\%$. Indicators of attacks in the test dataset are equal to 1.0 at the intervals of corresponding attacks (to MEAS, MV or SP). There were three kinds of attacks used at the MEAS and MV: a) Integrity: changing a value to something different from that simulated by the TEP-model, b) DoS (denial of service): at some point a value of a variable is frozen for the duration of an attack, c) Noise: add nose to value.

An attack on an industrial plant can very quickly reach a critical situation where further model simulation becomes impossible and the plant operation must be stopped. In order not to make the task of detection too simple, we tuned the attack intervals so that the plant could return to a level of stable operation after an attack, and proposed four series of attacks.

3. RNN-based Anomaly Detection

We use RNN-based forecasting model. Anomaly detection is made on the base of MSE (mean square error) between prediction and observation.

3.1. Pre- and Post- Processing

Input data is normalized (parameters are calculates based on the training dataset).

Prediction square error is summarised and smoothed with EMWA. Smoothing factor α is calculated using the size of input window w as $\alpha = 1 - \exp\left(-\frac{\ln 2}{w}\right)$.

59	**Time Series Dimension**
41	sensors (MEAS - measurements)
12	controls (MV - manipulated variables)
1	MEAS attack indicator
1	MV attack indicator
1	SP (set point) attack indicator
3	special variables (state, product rate, hourly cost)
	Plant Modes
7	normal modes
28	transient modes
	Attack Types
	DoS (value is frozen)
	Integrity (value is changed)
	Noise (value + noise)
	Attack Series (#, Type, MV/MEAS/SP, duration)
	#21: Integrity: MEAS ”reactor temperature”, 0.012-0.027 h
	#22: DoS: MV ”Stripper liquid product flow”, MEAS ”Stripper level”, MEAS ”Stripper underfow”, 5.663-25.019 h
	#23: DoS: MV ”D feed flow”, 10 h
	#24: Noise: MV ”C feed flow”, MV ”Purge flow”, MEAS ”Stripper underfow”, MV ”Stripper steam flow”, 7.727 - 71.291 h
1000	**Points per Hour**
	Training set (duration hours)
201	samples with one normal mode (120 h)
336	samples with transient mode (120 h)
	Test set (duration hours)
142	samples with attacks ($\leq$ 120 h, till broken)

Table 1. TEP dataset characteristics

The minimal detection threshold value is calculated as 0.999 quantile from the smoothed error in the training dataset.

3.2. RNN Architecture and Training

To cope with the TEP dataset, we adopted the previously used LSTM architecture for the GHL dataset in a way that is represented in Table 2.

For both datasets we use stacked RNN with 2 hidden layers, each with 64 cells. The input window is equal to the prediction window. ReLU as an activation function for hidden layers and linear activation function for the output layer are used.

To train RNN we use MSE loss-function and the RMSProp algorithm. Learning step equals 0.001. Number of epochs

Dataset	Cell	Layer	Memory	Dropout	Window
GHL	LSTM	2x64	stateful	0.1	120
TEP	GRU	2x64	stateless	no	100

Table 2. RNN architecture for GHL and TEP datasets

equals 100. Average time of one training epoch is 70 seconds with batch size = 2048 and hardware Tesla P40, Intel Xeon CPU E5-2650 v4 2.20GHz. The resulting dependency of loss-functions vs epoch for training and validation datasets is represented in Figure 2.

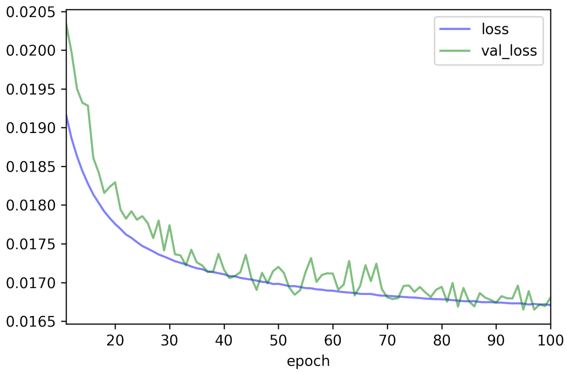

Figure 2. Loss-function value vs epoch number for training (loss) and validation (val-loss) datasets

Examples of trained RNN model prediction for a single mode normal behaviour sample is represented in Figure 3, for a transient mode sample in Figure 4, and for an MEAS attack sample in Figure 5.

3.3. Quality Metric

To compare the results of different anomaly detection approaches we selected the NAB-metric that scores in range $s \in [-1.0, 1.0]$ ($s = 1.0$ if detection is at the anomaly beginning, $s = 0.0$ if detection is at the end of anomaly window, $s \in (-1.0, 0.0)$ if detection is not too far from the end of anomaly window, $s = -1.0$ otherwise). Table 3 shows standard profile weights (Lavin & Ahmad, 2015) for TP, TN, FP, FN for the NAB-metric.

	Positive	Negative
True	1.0	1.0
False	0.11	1.0

Table 3. Standard profile weights for the NAB-metric

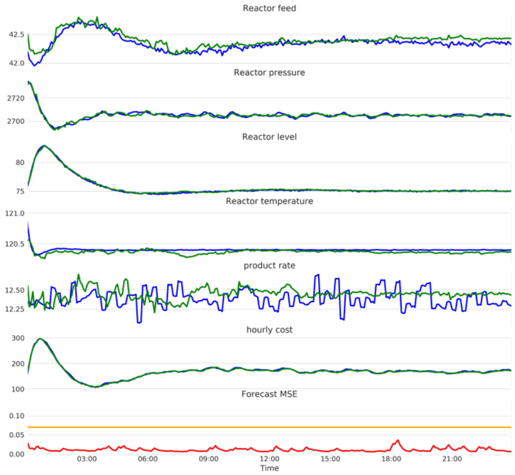

Figure 3. Example of RNN prediction (green) for a single mode normal behaviour sample

Experimenting with different kinds of attacks on the TEP we observed that the anomaly window is not necessary equal to the attack interval. Quite often the consequence of an attack, which is also anomalous behaviour, continues after the attack has stopped. So, selecting a correct anomaly window for the NAB metric is quite a tricky process. To average this out we use an anomaly window equal to twice the attack interval.

The RNN-based detector was tested under different detection thresholds. Several cyber-attacks datasets were concatenated in one.

3.4. Comparison with DPCA

Working with the GHL dataset we found that the most successful alternative to the LSTM-based approach is PCA. Here we compare our RNN-based approach with dynamic PCA (DPCA).

DPCA parameters are: time window size - 10; space dimension - 590; number of main components - 19 (Kaiser rule $\lambda > 1.0$).

With DPCA we were only able to train separate models for each TEP single operation mode. For transient mode we faced with many false positives (FP) detection with DPCA. So, we ignored that cases and calculated scores for DPCA as an average of the scores for each single mode (m):

$$\overline{\text{DPCA}} = \sum_{m=0}^{6} \text{DPCA}_{(m)}$$

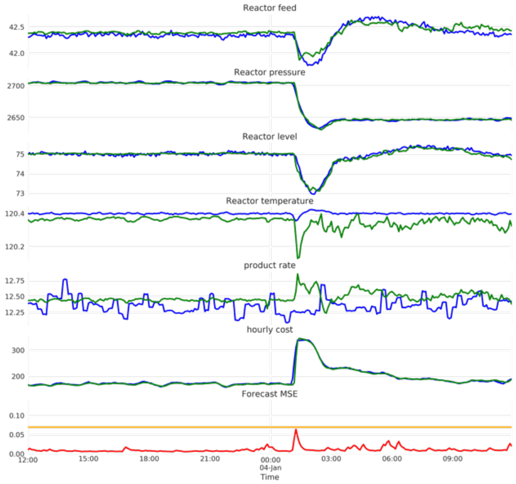

Figure 4. Example of RNN prediction (green) for a transient mode normal behaviour sample

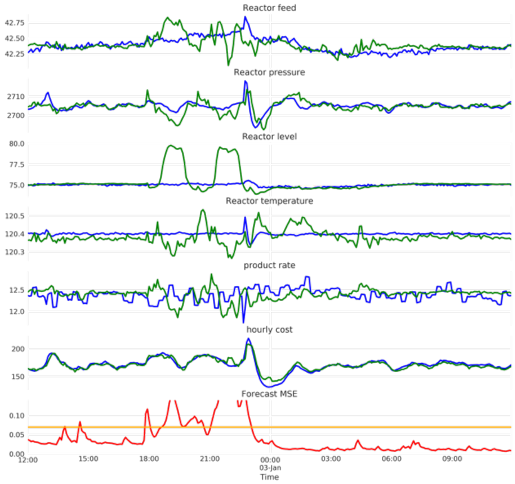

Figure 5. Example of RNN prediction (green) for an MEAS-attack sample

We tested RNN and DPCA on the TEP dataset using the NAB-metric. Anomalies detections results are shown in Table 4.

Method (attacks series)	**NAB-score**
Ideal detector	1.000
RNN (all)	0.373
$\overline{\text{DPCA}}$ (all)	0.086
RNN (except #23)	0.803
$\overline{\text{DPCA}}$ (except #23)	0.649

Table 4. RNN vs DPCA NAB-scores

We connect the decrease in the RNN and DPCA detection score in the NAB-metric for attacks on MV #23 (”D-feed flow DoS”) with the TEP physics, i.e. the consequences of control changes taking place for quite a long time after an attack.

4. Conclusion

The RNN-based approach with GRU stateless cells and without dropout is capable of effectively dealing with stochasticity, stationarity, transient and anomalous behaviour in a realistic TEP dataset. The NAB-metric makes it possible to validate the model for early detection. A comparison with DPCA shows that the RNN-based approach has better scores for MEAS and SP attacks. Attacks on MV are detected with RNN with some delay, which we explain by the longer anomaly window of the consequences of such attacks. We also found that DPCA model can be trained only for a separate single mode, and for a transient mode DPCA gives many false positives (FP). From a practical point of view of industrial anomaly detection application, it is more convenient to have one trained model for all kinds of plant modes, what we achieved only with RNN approach.

The generated TEP datasets with normal and anomalous behaviour caused by cyber-attacks are made publicly available.

Acknowledgements

The authors are sincerely grateful to Konstantin Kiselev for the implementation of the TEP model on Python and Artem Vorontsov for useful discussions.

This work was supported by the Kaspersky Lab.

References

Gasoil Heating Loop dataset, 2016. URL https://kas.pr/ics-research/dataset_ghl_1.

Tennessee Eastman Process with cyber-attacks dataset, 2017. URL https://kas.pr/ics-research/dataset_tep_59.

Adepu, Sridhar and Mathur, Aditya P. Detecting multipoint attacks in a water treatment system using intermittent control actions. In *Proceedings of the Singapore Cyber-Security Conference (SG-CRC)*, volume 14, pp.

59–74, 2016. URL http://ebooks.iospress.nl/publication/42053.

Chiang, L H, Russell, E L, and Braatz, R D. Fault detection and diagnosis in industrial systems. *Measurement Science and Technology*, 12(10): 1745, 2001. URL http://stacks.iop.org/0957-0233/12/i=10/a=706.

Downs, J and Vogel, E. A plant-wide industrial process control problem. *Computers & chemical engineering*, 17(3):245–255, 1993.

Filonov, P, Lavrentyev, A, and Vorontsov, A. Multivariate industrial time series with cyber-attack simulation: Fault detection using an lstm-based predictive data model. *NIPS 2016 Time Series Workshop papers*, 2016. URL http://arxiv.org/abs/1612.06676.

Kiss, Istvan, Haller, Piroska, and Berea, Adela. Denial of service attack detection in case of Tennessee Eastman Challenge Process. *Procedia Technology*, 19:835 – 841, 2015. 8th International Conference Interdisciplinarity in Engineering, INTER-ENG 2014, 9-10 October 2014, Tirgu Mures, Romania.

Krotofil, M. Damn vulnerable chemical process, 2014. URL http://github.com/satejnik/DVCP-TE.

Lavin, A and Ahmad, Subutai. Evaluating real-time anomaly detection algorithms - the Numenta Anomaly Benchmark. *CoRR*, abs/1510.03336, 2015. URL http://arxiv.org/abs/1510.03336.

Lee, Robert M., Assante, Michael J., and Conway, Tim. ICS CP/PE (Cyber-to-Physical or Process Effects) case study paper - German Steel Mill Cyber Attack, Dec 2014. URL https://ics.sans.org/media/ICS-CPPE-case-Study-2-German-Steelworks_Facility.pdf. ICS Defense Use Case (DUC).

Malhotra, Pankaj, Vig, Lovekesh, Shroff, Gautam, and Agarwal, Puneet. Long Short Term Memory networks for Anomaly Detection in time series. In *23rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning.*, April 2015.

Malhotra, Pankaj, Ramakrishnan, Anusha, Anand, Gaurangi, Vig, Lovekesh, Agarwal, Puneet, and Shroff, Gautam. LSTM-based encoder-decoder for multi-sensor anomaly detection. *CoRR*, abs/1607.00148, 2016. URL http://arxiv.org/abs/1607.00148.

Marti, Luis, Sanchez-Pi, Nayat, Molina, Jose Manuel, and Garcia, Ana Cristina Bicharra. Anomaly detection based on sensor data in petroleum industry applications. *Sensors*, 15(2):2774, 2015.

Matteson, David S and James, Nicholas A. A nonparametric approach for multiple change point analysis of multivariate data. *Journal of the American Statistical Association*, 109(505): 0:334–345, 2013. URL https://arxiv.org/abs/1306.4933v2.

Nanduri, Anvardh, Candidate, M S, and Sherry, Lance. Anomaly detection in aircraft data using recurrent neural networks (rnn). 2016.

Ricker, N Lawrence. Tennessee Eastman Challenge Archive, May 2013. URL http://depts.washington.edu/control/LARRY/TE/download.html.

Yadav, Mohit, Malhotra, Pankaj, Vig, Lovekesh, Sriram, K., and Shroff, Gautam. ODE - augmented training improves anomaly detection in sensor data from machines. *CoRR*, abs/1605.01534, 2016. URL http://arxiv.org/abs/1605.01534.