Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 38 (2019) 233–240

Procedia MANUFACTURING

www.elsevier.com/locate/procedia

29th International Conference on Flexible Automation and Intelligent Manufacturing (FAIM2019), June 24-28, 2019, Limerick, Ireland.

A deep learning approach for anomaly detection with industrial time series data: a refrigerators manufacturing case study

Mattia Carletti[a], Chiara Masiero[b], Alessandro Beghi[a], Gian Antonio Susto[a,*]

[a]*University of Padova, Padova 35122, Italy*
[b]*Statwolf Data Science SRL, Padova 35131, Italy*

Abstract

In refrigerators production, vacuum creation is fundamental to guarantee the correct manufacturing of the product. Before inserting the refrigerant in the refrigerator cabinet, the vacuum is tested through a Pirani gauge that assesses the pressure within the cabinet. Such readings are used to evaluate the vacuum creation process and to verify if leakings are present. In this work, we employ a Deep Learning-based Anomaly Detection approach to associate an Anomaly Score to each pressure profile; this score can be exploited to optimize actions performed by human operators like more detailed inspections or unit exclusion from the downstream production stages. We propose a native time series-based approach based on Deep Learning and compare it with classic ones based on hand-craft features. The proposed approach is designed to be deployed in a Decision Support System for assisting human operators in the following testing operations, helping them in reducing evaluation bias and attention losses that are inevitable in production line environment. Moreover, costs associated with false positives (normally operating units detected as anomalous) and false negatives (undetected anomalies) are considered here to optimize decision making in a cost-reduction perspective. We also describe promising results obtained on real industrial data spanning on a 5-month period and consisting of thousands of tested household units.

Keywords: Anomaly Detection; Decision Support System; Deep Learning; Industry 4.0; Machine Learning; Time series Analysis

* Corresponding author. Tel.: +39-049-8277760.
E-mail address: gianantonio.susto@dei.unipd.it

10.1016/j.promfg.2020.01.031

1. Introduction and main contribution

In refrigerators manufacturing, vacuum creation is a fundamental production stage that has a strong impact on the quality of the resulting product. Before inserting the refrigerant in the refrigerator cabinet, vacuum is tested through a Pirani gauge and pressure measurements are used to evaluate the presence of leakings. However, abnormal behaviors due to leakings often resemble acceptable ones closely, and thus their detection is a challenging task. We aim at spotting pressure profiles whose shape is different from normal one due to issues in the vacuum creation process. Therefore, the problem can be recast in the framework of Anomaly Detection (AD). In this work, we employ Machine Learning (ML)-based Anomaly Detection approaches to detect anomalous behavior in leak testing. ML exploits the availability of historical datasets and provides solutions whose performance are expected to further improve as new data are collected [1]. The situation where production data are monitored on-line and collected for prescriptive analytics is typical in Industry 4.0 scenarios [2] [3] [4]. In this work we show how it can be successfully used to improve quality control for manufactured products in the context of refrigerators production. We recall that AD is an unsupervised task [5], i.e. tagged information on anomalies are generally not available. In general, most AD approaches applied in industrial scenario are based on hand-engineered features whose effective design requires a significant amount of domain knowledge and is far from effortless [6]. Feature design in AD is particularly challenging since features that are considered informative for supervised tasks (like Fault Classification or some Predictive Maintenance use cases [7]) are typically not relevant for AD [8] [9]. Therefore, approaches based on hand-craft features are usually difficult to maintain and scale. Indeed, production process may vary in time. Moreover, it is usually difficult to adapt features to new products. In this work, we propose an unsupervised Deep Learning (DL)-based approach to AD for pressure profiles. We train a Bayesianly Interpretable Neural Network (BINN) to predict the future value of a time series in a self-supervised fashion. Prediction uncertainty, naturally provided by the network, is then used to assess how abnormal the pressure profile is. Thanks to MC dropout [10] (details will be provided in the following Section), training is implemented efficiently and uncertainty estimation can be achieved without the need for additional parameters, whereas ordinary Bayesian Neural Networks (that are usually used to provide output prediction uncertainty) require at least to double the number of parameters [11]. Differently from classic AD techniques, the proposed approach does not require explicit feature design. Moreover, prediction uncertainty is used to associate an anomaly score (also called "health factor" in the literature [12]), i.e. an estimate of how abnormal a piece is, to each leaking test; the anomaly score can be exploited to optimize actions performed by human operators like more detailed inspections or unit exclusion from the downstream production stages. In fact, the AD solution will be deployed in a Decision Support System (DSS) [13] for assisting human operators in the following testing operations, helping them in removing evaluation bias and attention losses that are inevitable in production line environment. Decision making can be optimized by tuning the number of false positives or false negatives in a flexible manner. Indeed, it is possible to reduce the number of normally operating units detected as anomalous (false positives) when a limited workforce is available for further inspection of suspicious pieces. Similarly, the number of undetected anomalies (false negatives) can be reduced when manufacturing high-end products. The contributions of this paper can be summarized as follows:

- To the best of our knowledge, this is one of the first AD approaches applied to consumer goods manufacturing (few other works are present in automotive manufacturing [14] while, as far as we know, this is the first work in white goods manufacturing);
- We propose a DL-based approach for industrial time series that provides a measure of uncertainty that can be exploited in the deployment of a DSS;
- The proposed approach directly deals with time series and requires no feature engineering. This is a clear advantage with respect to classic ML approaches where the design of features is typically a time-consuming procedure, heavily dependent on domain knowledge, and difficult to adapt to process and product changes;
- The proposed approach is tested on real industrial data, corresponding to thousands of tested household units. The comparison with classic ML-based solutions for AD suggests that the proposed approach achieves higher performance in the problem at hand.

The rest of the paper is organized as follows: Section 2 is devoted to present the proposed architecture and to review Bayesian Neural Networks. In Section 3 the considered refrigerator manufacturing case study is described, while experimental findings are reported in Section 4. Finally, in Section 5, conclusive remarks are reported and future research directions are illustrated.

2. Proposed approach

The proposed approach relies on a Bayesian interpretation of Neural Networks [15] [16] [17]. Differently from classic Neural Networks (NNs) [18], where parameters (i.e. weights and biases) are represented by single values, in Bayesian Neural Networks (BNNs) each parameter is associated with a probability distribution. This probabilistic formulation requires a higher number of parameters, but provides confidence intervals for network output, an information that is crucial to improve reliability in many practical applications. In this work, we make use of MC dropout, a technique proposed in [10], to modify a standard Convolutional Neural Network (CNN) so that it can provide output uncertainty estimates. We remark that the resulting network is not Bayesian by itself but is equipped with a natural Bayesian interpretation instead. We remind that in our scenario the aim is to detect abnormal profiles rather than abnormal patterns within time series. We reformulate this problem by defining a self-supervised "proxy task": for each time series, we consider a sliding window of length k and we train the model to predict the $(k+1)$-th sample for each window. Under the assumption that most pressure profiles are normal, we expect to obtain highly confident predictions on values coming from normal examples and greater uncertainty when processing abnormal examples. If we average model uncertainty over all windows extracted from each time series, we can compute an overall model uncertainty measure. Thus, we can consider it as an anomaly score that is then used to discriminate between normal and abnormal pressure profiles, once an appropriate threshold has been defined.

Fig. 1 depicts the proposed approach, whose main steps are: (i) a BINN is trained (blue lines) to reconstruct subsequences taken from time series, under the assumption that most input samples come from normal pressure profiles; (ii) once deployed (red lines), the BINN is used to reconstruct n subsequences $\mathbf{x_i}$ taken from pressure profile $\mathbf{x}$ and the sample variance provided by N forward passes is averaged over all n windows to assign an overall

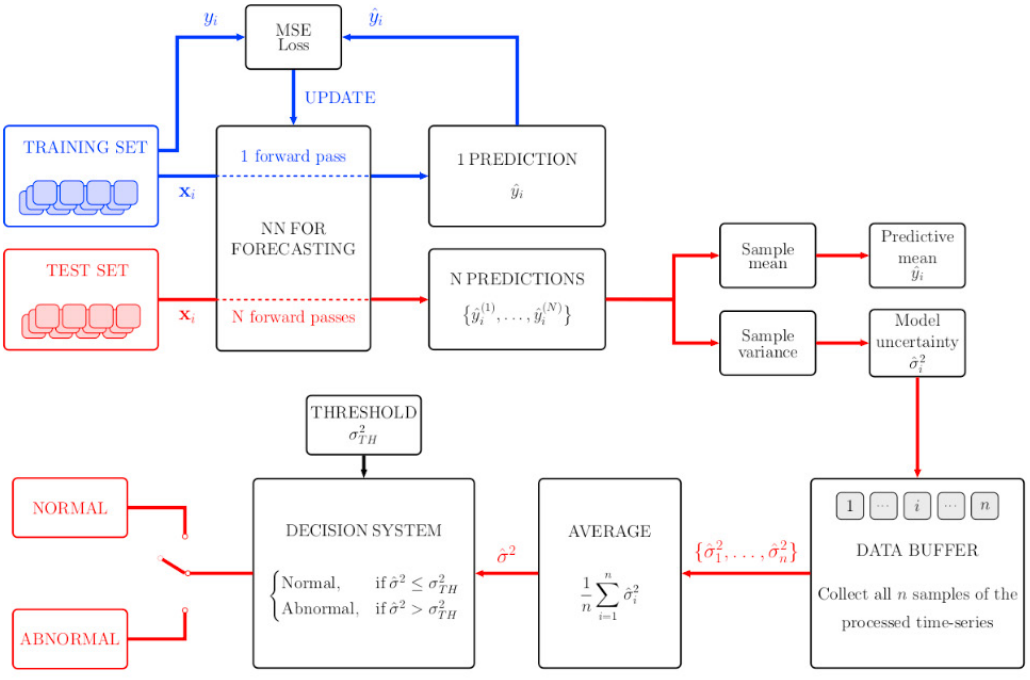

Fig. 1. Proposed approach for detecting abnormal pressure profiles.

anomaly score σ^2 to pressure profile $\mathbf{x}$ (iii) if $\sigma^2 > \sigma_{TH}^2$ the time series $\mathbf{x}$ is deemed abnormal. More details on BINN training and deployment will be given in Section 2.1 and Section 2.2. The threshold can be defined either in a completely unsupervised fashion or, if a training dataset with ground-truth labels is available, based on a trade-off between resulting false positive and false negative samples.

2.1. Review on Bayesian Neural Networks

Let $D = \{\mathbf{x}_i, \mathbf{y}_i\}$ be the set of observed data, where $\mathbf{x}_i \in R^m$ is an input data point, and $\mathbf{y}_i \in R^n$ is the corresponding output. To train a BNN, we first define a prior distribution $p(\mathbf{w})$ over the network parameters $\mathbf{w} \in R^p$. Our aim is updating our prior beliefs about $\mathbf{w}$ considering the observed data. In other words, we are interested in estimating the posterior distribution of the network parameters given the observed data, $\mathrm{p}(\mathbf{w}|\mathbf{x},\mathbf{y}) = p(\mathbf{y}|\mathbf{x},\mathbf{w})p(\mathbf{w})/p(\mathbf{y}|\mathbf{x})$, where $p(\mathbf{y}|\mathbf{x},\mathbf{w})$ is known as likelihood function and $p(\mathbf{y}|\mathbf{x}) = \int p(\mathbf{y}|\mathbf{x},\mathbf{w})p(\mathbf{w})d\mathbf{w}$ as model evidence. Notice that the denominator requires the integration over all possible values of the parameters, making exact computation practically intractable. As a result, we need an approximation of the posterior distribution and thus the problem falls under the scope of approximate Bayesian inference. In recent years, many variational inference techniques have been applied [11] [19] [20] [21] based on the following procedure: (i) define a variational approximation $q(\mathbf{w}|\boldsymbol{\theta})$, parametrized by $\boldsymbol{\theta} \in R^q$, of the true posterior distribution $p(\mathbf{w}|\mathbf{x},\mathbf{y})$; (ii) optimize parameters $\boldsymbol{\theta}$ such that the KL divergence between $q(\mathbf{w}|\boldsymbol{\theta})$ and $p(\mathbf{w}|\,\mathbf{x},\mathbf{y})$ is minimized. This leads to an optimization problem where the cost function to be minimized is

$$\boldsymbol{\theta}^* = argmin_{\boldsymbol{\theta}} \Big[KL\big(q(\mathbf{w}|\boldsymbol{\theta})|\, p(\mathbf{w})\big) - E_{q(\mathbf{w}|\boldsymbol{\theta})}[log\, p(\mathbf{y}|\,\mathbf{x},\mathbf{w})] \Big].$$

$KL(q(\mathbf{w}|\boldsymbol{\theta})|\, p(\mathbf{w}))$ is a complexity term, which prevents the variational distribution to deviate too much from the prior distribution, and $E_{q(\mathbf{w}|\boldsymbol{\theta})}[logp(\mathbf{y}|\mathbf{x},\mathbf{w})]$ is a fit term (also known as reconstruction probability) that encourages the variational distribution to explain the data well.

The main disadvantage when using variational inference techniques is represented by the high computational cost and this motivated the search for cheaper methods to get uncertainty estimates. In [10] the authors propose "a new theoretical framework casting dropout training in deep neural networks as approximate Bayesian inference in deep Gaussian processes". In short, they demonstrate that standard dropout [22] can be interpreted as a variational Bayesian approximation where the variational distribution $q(\mathbf{w})$ is a Bernoulli. The main peculiarity of this technique lies in the fact that dropout is used in the testing phase as well. This can be seen as Monte Carlo sampling from the posterior distribution over the network weights and this is the reason why this method is commonly referred to as MC dropout. More in detail, when the trained network is fed with a new input $\mathbf{x}^*$, we sample the posterior distribution over the weights to obtain the posterior distribution of the output $\mathbf{y}^*$. In practice, this is done by performing N stochastic forward passes through the network and then taking the mean of the output samples as the prediction and the variance as the model uncertainty.

Since a complete dissertation on dropout variational inference is out of the scope of this work, we refer the curious reader to [10] [23]. For a clear comprehension of the following Sections, all we need to know is that MC dropout can be applied to ordinary CNNs with no need to modify the training procedure [24]. Then, the resulting network provides output uncertainty estimates that we will use as anomaly scores to detect abnormal pressure profiles.

2.2. Proposed architecture

Fig. 2 depicts the proposed architecture. The input consists in a window of $k = 8$ contiguous samples from the processed time series and the output (that the network has to predict) is the sample that follows the last sample in the input window. Notice that no feature engineering is needed: the input window contains raw data from the preprocessed time series. Given the small length of the sequences to be processed, we decided to use convolutional layers instead of recurrent layers. This choice has proved to give enough capacity to the network, while maintaining

the number of parameters limited. On top of the two convolutional layers (with 32 and 16 filters of size 2, respectively), we placed two dense layers (with 30 and 1 hidden units, respectively). ReLUs non-linearities are used as activation functions of each layer, except for the last one. In the training phase the network parameters are optimized by means of Adam optimizer [25] to minimize the Mean Square Error between the network predictions and the true outputs. As mentioned in Section 2.1, dropout is used during training and testing as well to induce a Bayesian variational inference interpretation of the model. More in detail, at test time we perform $N = 50$ stochastic forward passes through the network for each test example and we consider the sample mean of the output samples as prediction and the sample variance as an estimate of model uncertainty.

We remind that the goal of the proposed algorithm is to detect abnormal time series and not to detect anomalies within time series. Therefore, we average the sample variance associated to each sliding window within the time series to obtain the anomaly score associated to each pressure profile.

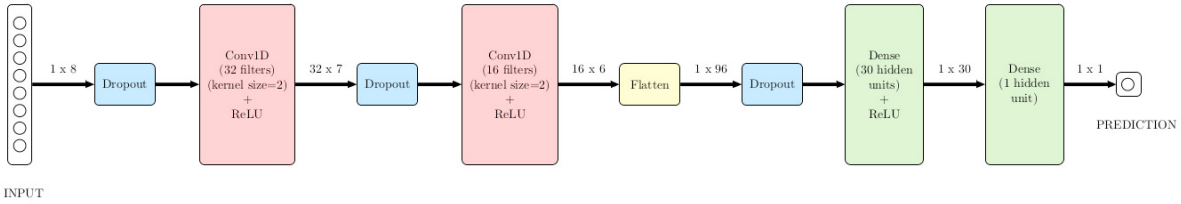

Fig. 2. Proposed BINN (based on MC dropout + CNN) for sequence forecasting.

3. Industrial case study

We tested the proposed approach on a real case study, provided by an industrial partner. Data consist in pressure profiles measured after vacuum creation process for thousands of refrigerators. Preliminarily, we performed a Pareto analysis and we decided to focus on the most frequent refrigerator type, that accounts for more than 10% of manufactured pieces between 2018/03 and 2018/07. Thus, we have 2000 pressure profiles given by pressure measurements acquired by a Pirani gauge, with sampling time $T = 2\ s$. Pressure in the cabinet is measured in the interval between the final stage of the vacuum creation process and the end of the testing phase. This interval consists in three stages: (i) a pump is attached to the cabinet and vacuum creation process begins; (ii) once vacuum creation is complete and the pump is detached, the valve is opened for a few seconds; (iii) the valve is closed and pressure test ends.

Fig. 3 shows equipment used to perform vacuum creation and assess the quality of this process. To enable the evaluation of proposed ML approaches, when the dataset was created, a label was assigned to each time series by a domain expert. If a time series is deemed normal, then it is tagged with 0, otherwise with 1. In the dataset at hand, 15% of the time series are tagged with label 1. As typically happens in industrial case studies, the availability of tagged data can be considered as an exceptional occasion and the presence of labels is not expected in production phase. Thus, we focused on unsupervised techniques for AD.

Fig. 3. Pump carousel used to perform vacuum creation in refrigerators (left); Pirani gauge used to assess vacuum quality (right).

4. Experimental results

4.1. Fully unsupervised scenario

To evaluate the proposed approach, we compared it with three state-of-the-art AD algorithms, namely: Isolation Forest (I-Forest), One-Class SVM (OCSVM) and Principal Component Analysis (PCA). We also considered another DL-based solution where the reconstruction error provided by a fully-connected autoencoder (AE) is used as a measure of anomaly. For each AD algorithm, we considered the implementation available in PyOD Python toolbox [26] (we refer the curious reader to [26] also for references associated with the mentioned algorithms). In the unsupervised scenario, when designing a threshold on prediction variance provided by BINN to distinguish between normal and abnormal profiles, there are no labels we can rely on. However, we notice that the threshold plays the same role as the contamination parameter (i.e. the expected fraction of outliers in the dataset) for the classic algorithms listed above. Since, when no prior information is available, 0.1 is a very common choice for the contamination parameter, we propose to classify a time series as abnormal if its prediction variance is greater than the 90-th percentile of the prediction variance attained by time series in the training stage.

Differently from BINN, all classic AD techniques considered for the comparison, except for the autoencoder, are not designed to deal directly with time series. Thus, we propose a straightforward feature extraction procedure: (i) for each time series, we split it in 7 windows (of length 10) and (ii) for each window we compute minimum value, maximum value, average value, median value, standard deviation and median absolute deviation. The resulting design matrix has 42 columns. We performed 10 random iterations and computed mean and standard deviation for precision, recall and F1 score. The experimental results shown in Tab. 1 suggest that our approach exhibits the best performance according to all the metrics considered. One may argue that the creation of more informative features, for the task at hand, may improve the performance of classic AD algorithms. However, if we assume that the design of elaborate, domain-specific features is not the preferable solution, due to scalability and maintenance issues, the choice of straightforward statistics such as the proposed ones seems very natural. Indeed, hand-craft features based on very specific domain knowledge are usually hard to adapt to process or product changes. To assess the impact of feature design, we also applied the above-mentioned algorithms to time series samples directly, without an intermediate feature extraction stage. We remark that this naive approach was considered only to assess the quality of the features proposed above, and it is not advisable in scenarios where longer time series are involved. On the contrary, BINN can be adapted to different time series durations by tuning number and length of considered windows. Tab. 1 shows that the proposed approach still outperforms the others. This highlights once again the great advantage of a solution that can handle directly time series data. In addition, the fact that some algorithms achieve better results with raw data suggests that the extraction of informative features that are not heavily dependent on domain knowledge is a very challenging task.

Tab. 1. Performance of proposed BINN approach in its completely unsupervised version versus off-the-shelf AD algorithms.

	Raw data			Features		
Algorithm	F1 score	Precision	Recall	F1 score	Precision	Recall
BINN	**0.743 ± 0.013**	**0.857 ± 0.018**	**0.655 ± 0.015**	-	-	-
PCA	0.612 ± 0.054	0.760 ± 0.054	0.514 ± 0.059	0.573 ± 0.035	0.725 ± 0.048	0.478 ± 0.051
OCSVM	0.584 ± 0.046	0.727 ± 0.030	0.492 ± 0.063	0.540 ± 0.039	0.653 ± 0.047	0.464 ± 0.048
AE	0.562 ± 0.061	0.692 ± 0.074	0.476 ± 0.064	0.551 ± 0.037	0.681 ± 0.036	0.466 ± 0.052
I-Forest	0.534 ± 0.056	0.682 ± 0.048	0.441 ± 0.063	0.524 ± 0.037	0.653 ± 0.061	0.441 ± 0.043

4.2. Impact and tuning of decision function threshold

In this Section we analyze the performance of BINN as a function of the threshold in the decision function and make a comparison with the techniques introduced in Section 4.1. The aim of such an analysis is to prove that the higher performance of our approach is not due to an ad-hoc selection of the threshold. Thus, we consider available

labels provided by a domain expert to define an optimal threshold in the decision function. Following the considerations given in [27], we assess optimality through the precision-recall plot which is more informative than ROC when dealing with imbalanced datasets. Ideally, the best performing algorithm would be located at the upper right corner, where both precision and recall values are equal to 1. Fig. 4 shows that even when the best threshold is selected for each algorithm, BINN still outperforms all classic AD techniques.

Since the proposed approach is designed to be integrated into a Decision Support System (DSS), it is possible to tune the threshold on prediction variance based on resulting false positive and false negative rates. Indeed, for lots of high-end products it may be advisable to reduce false negatives as much as possible; on the other hand, when reduced workforce is available, it may be useful to reduce the number of false positives to make human operators focus on a small number of pieces that are defective with high probability.

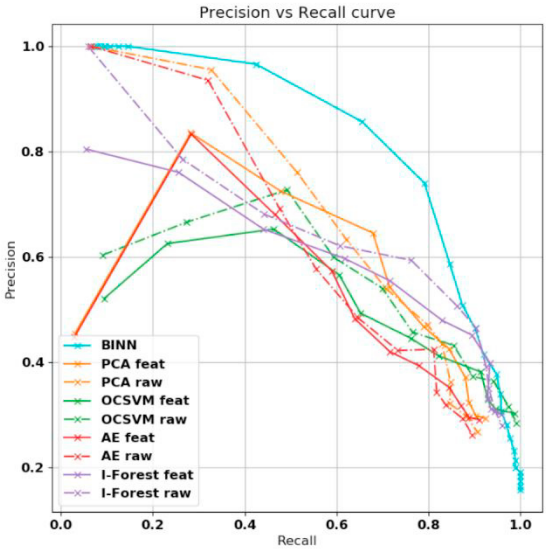

Fig. 4. Precision-recall plots. Each cross represents a precision-recall pair corresponding to a specific value of the threshold.

5. Conclusions and research directions

In this work we propose an approach to Anomaly Detection in refrigerators manufacturing based on a Bayesianly Interpretable Neural Network. To the best of our knowledge, this is one of the first Anomaly Detection approaches applied to consumer goods manufacturing.

The proposed approach requires no feature engineering because it can deal directly with time series. Thus, it is simpler to deploy than classic Machine Learning-based solutions for Anomaly Detection. For each analyzed time series, we compute a measure of uncertainty that can be exploited inside a Decision Support System to drive corrective actions; in particular, it can be used to focus first on most anomalous units when limited workforce is available. On the other hand, only units whose evaluated uncertainty is very low may be considered acceptable when high-end products manufacturing is involved.

Finally, we performed tests on real industrial data corresponding to thousands of tested refrigerators household units. The comparison with other Anomaly Detection techniques (that have been proven to be effective in other manufacturing areas [5] [8] [9]) suggests that the proposed approach achieves higher performance for the problem at hand.

We foresee at least two relevant future research directions: (i) we could use uncertainty associated to the task of predicting future values for single windows to highlight local abnormal behaviors that may be difficult to spot when looking at the entire time series; (ii) we can make the proposed approach adaptive with respect to new products and changes in the production process by means of adaptive thresholds and/or condition-based model retraining.

Acknowledgements

This work has been supported by the Regione Veneto project PreMANI (MANIFATTURA PREDITTIVA: progettazione, sviluppo e implementazione di soluzioni di Digital Manufacturing per la previsione della Qualità e la Manutenzione Intelligente) and the Regione Veneto project ADAPT4.0 (Svilupppo di un sistema adattativo per il riconoscimento di guasti nell'Industria 4.0). We would also like to thank Nvidia Corporation for supporting our research with a Nvidia Titan V GPU.

References

[1] S. M. Erfani, S. Rajasegarar, S. Karunasekera e C. Leckie, «High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning,» *Pattern Recognition,* vol. 58, pp. 121-134, 2016.

[2] J. Moyne, J. Samantaray e M. Armacost, «Big data capabilities applied to semiconductor manufacturing advanced process control,» *IEEE transactions on semiconductor manufacturing,* vol. 29, pp. p283--291, 2016.

[3] G. A. Susto, A. B. Johnston, P. G. O'Hara e S. McLoone, «Virtual metrology enabled early stage prediction for enhanced control of multi-stage fabrication processes,» in *Automation Science and Engineering (CASE), 2013 IEEE International Conference on*, 2013.

[4] G. A. Susto, A. Schirru, S. Pampuri, D. Pagano, S. McLoone e A. Beghi, «A predictive maintenance system for integral type faults based on support vector machines: An application to ion implantation,» in *IEEE Conference on Automation Science and Engineering (CASE), 2013*.

[5] B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. Cho e H. Chen, «Deep autoencoding gaussian mixture model for unsupervised anomaly detection,» 2018.

[6] Y. Wang, X. Ma e P. Qian, «Wind turbine fault detection and identification through PCA-based optimal variable selection,» *IEEE Transactions on Sustainable Energy,* 2018.

[7] G. A. Susto, S. McLoone, D. Pagano, A. Schirru, S. Pampuri e A. Beghi, «Prediction of integral type failures in semiconductor manufacturing through classification methods,» in *Emerging Technologies & Factory Automation (ETFA), 2013 IEEE 18th Conference on*.

[8] L. Puggini e S. McLoone, «An enhanced variable selection and Isolation Forest based methodology for anomaly detection with OES data,» *Engineering Applications of Artificial Intelligence,* vol. 67, pp. 126-135, 2018.

[9] A. Beghi, L. Cecchinato, C. Corazzol, M. Rampazzo, F. Simmini e G. A. Susto, «A one-class svm based tool for machine learning novelty detection in hvac chiller systems,» *IFAC Proceedings Volumes,* vol. 47, pp. 1953-1958, 2014.

[10] Y. Gal e Z. Ghahramani, «Dropout as a Bayesian approximation: Representing model uncertainty in deep learning,» in ICML 2016.

[11] C. Blundell, J. Cornebise, K. Kavukcuoglu e D. Wierstra, «Weight Uncertainty in Neural Networks,» *Proceedings of the 32 nd International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37*.

[12] G. A. Susto, A. Schirru, S. Pampuri, A. Beghi e G. De Nicolao, «A hidden-Gamma model-based filtering and prediction approach for monotonic health factors in manufacturing,» *Control Engineering Practice,* vol. 74, pp. 84-94, 2018.

[13] Z. Guo, E. Ngai, C. Yang e X. Liang, «An RFID-based intelligent decision support system architecture for production monitoring and scheduling in a distributed manufacturing environment,» *International journal of production economics,* vol. 159, pp. 16-28, 2015.

[14] A. Theissler, «Detecting known and unknown faults in automotive systems using ensemble-based anomaly detection,» *Knowledge-Based Systems,* vol. 123, pp. 163-173, 2017.

[15] J. S. Denker e Y. Lecun, «Transforming neural-net output levels to probability distributions,» in *NIPS*, 1991.

[16] J. F. G. Freitas, «Bayesian methods for neural networks,» 2003.

[17] R. M. Neal, Bayesian learning for neural networks, vol. 118, Springer Science & Business Media, 2012.

[18] I. Goodfellow, Y. Bengio, A. Courville e Y. Bengio, Deep learning, vol. 1, MIT press Cambridge, 2016.

[19] A. Graves, «Practical variational inference for neural networks,» in *Advances in neural information processing systems*, 2011.

[20] J. Paisley, D. Blei e M. Jordan, «Variational Bayesian inference with stochastic search,» *arXiv preprint arXiv:1206.6430,* 2012.

[21] D. P. Kingma e M. Welling, «Auto-encoding variational bayes,» *2nd International Conference on Learning Representations, 2014.*

[22] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever e R. Salakhutdinov, «Dropout: a simple way to prevent neural networks from overfitting,» *The Journal of Machine Learning Research,* vol. 15, pp. 1929-1958, 2014.

[23] Y. Gal, «Uncertainty in deep learning,» *PhD thesis, University of Cambridge,* 2016.

[24] Y. Gal e Z. Ghahramani, «Bayesian convolutional neural networks with Bernoulli approximate variational inference,» *arXiv preprint arXiv:1506.02158,* 2015.

[25] D. P. Kingma e J. Ba, «Adam: A method for stochastic optimization,» *arXiv preprint arXiv:1412.6980,* 2014.

[26] Y. Zhao, Z. Nasrullah e Z. Li, «PyOD: A Python Toolbox for Scalable Outlier Detection,» *arXiv preprint arXiv:1901.01588,* 2019.

[27] T. Saito e M. Rehmsmeier, «The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets,» *PloS one,* vol. 10, p. e0118432, 2015.