An Industrial Manufacturing Dataset together with Anomaly Detection Results integrated in an Open & Stand Alone Sharing Platform for Sustainable Replication

Gerold Hoelzl, Jonas Zausinger, Matthias Kranz
Chair of Embedded Systems
University of Passau
Passau, Germany
email: first.last@uni-passau.de

Bastian Fleischmann, Sebastian Soller
Almanara Research GmbH
Ruhstorf, Germany
email: first.last@almanara-research.de

***Abstract*—We aim at systems that make sense out of occurring anomalies to autonomously learn to predict and detect possible occurring machine drifts, failures and deviations, and the corresponding errors in the machines and products itself. To assess our prediction and classification methods, we collected data from a fully automated industrial machinery including 3 internal sensors in a large-scale dataset (> 87000 manufactured pieces with 39 different product types, in a timespan of nearly 7 months). We present the scenario and describe the collected data and the sensors. We describe the machine data and the corresponding errors, and present a generic tool that allows visualization, scripting, etc., especially when datasets have to be shared, as it gives an insight into the complexity of the data and the algorithms and make experiments as described in the paper reproducible. We argue to be currently in a replication crisis in data analysis that makes it close to impossible to replicate empirical findings due to the lack of the availability of the underlaying data and the implemented algorithms. We reached a point where we need to question if the results can be believed and how the datasets for evaluation are designed and recorded. To support an inevitable fundamental change towards the full openness of published results in collected data and the used algorithmic processing with minimum effort, we present and make publicly available (i) a large-scale dataset for IoT (Internet of Things) based predictive maintenance in an industrial setting combined with (ii) artificial intelligence algorithms used by our group, elaborated on the dataset embedded in (iii) a general tool to foster easily sharing of both for replicating results.**

Keywords*—*sensor based manufacturing dataset; industry; machine learning; anomaly detection; defect detection; industry 4.0; data sharing; toolset for result replication.

I. INTRODUCTION

Industry 4.0 has become an important topic for researchers in the industrial domain. With the availability of sensors, controllers and communication networks, a vast amount of data can be collected to improve aspects of the industrial production process [1]. Depending on the data, it can be utilized for various application scenarios adapted from techniques used in e.g. in human activity recognition architectures [2][3][4]. Important applications are transparency of the production process, a highly customizable and dynamic production process and smart manufacturing using machine learning [1][5].

One aspect of smart manufacturing is predictive maintenance and machine fault detection [6][7]. Machine learning in combination with sensor data collected beforehand is used to predict the health of the machinery or detect deviations from the normal state. When detecting a deviation from the normal state a technician can be notified to take suitable action. To this end, potential damages can be reduced by suggesting maintenance beforehand or detecting defects when they occur. For this, anomaly detection is successfully applied by researchers in the industrial domain [8][9]. An example for the application of fault detection is the early detection of machine defects by observing the vibration of machine parts using specifically placed vibration sensors [10][11].

In the industrial setting, anomaly detection is often applied in areas where the machine executes similar steps for prolonged periods of time. Deviations from the normal operation are expected to be induced machine issues. However, another important goal of the advancements in industrial production is a highly dynamic production that adjusts itself at any given time. Different products are produced interchangeably as the machinery adapts the operation mode according to the desired final product. Therefore, the operation mode and the notion of normal behavior can also rapidly change. This poses new challenges for fault detection. Changing a product type can be falsely identified as a defect. Likewise, types produced in small volume can be identified as anomalous, as they are insufficiently represented in the training data. Additionally, comparing results for such an industrial process is challenging due to the high variability of the process. Often each research group collects their own data - some of which may not be publicly available - using a custom set of specifically placed sensors to perform their experiments. This makes replicating and comparing results, and as a consequence, improving the methods more challenging.

To this end, we present a dataset gathered from a highly dynamic real-world industrial process and intended to be used for fault detection, using already available internal sensors that are part of the machine by default to increase the technical applicability, in this paper. These sensors collect internal information on the movement and electrical current of machine parts. We provide intrinsic sensor measurements of a CNC (Computerized Numerical Control) machine that is part of a larger production line. There, the produced product type, and configuration of the machine changes on the fly. The dataset spans over a time period of nearly 7 months and contains the production of 87650 workpieces from 39 different types.

These product types share similar basic traits, but can differ in characteristics such as size, design, and the presence of certain traits. In addition to the sensor data, we also collect the occurrence of machine events that are labeled by workers as a ground truth. Together with the dataset we introduce a tool to work with the data. This tool aims to facilitate the usage of the data by providing a simple playground for experimenting.

We show first results from using product type-aware anomaly detection to detect machine faults by performing anomaly detection both globally and in the context of the product type using well-known anomaly detection techniques. These results serve as a baseline for future work to make machine fault detection in a highly dynamic environment more robust.

The remaining paper is structured as follows. Section II describes the collected Dataset used for the Evaluation of the Anomaly Detection Algorithms and our developed Sharing Platform. In Section III we present the Evaluation of the Anomaly Detection Algorithms in our Application Case. Challenges and Future Developments are highlighted in Section IV. The paper is closed with Section V, that summarizes and recaps the achievements and contributions. Section VI links to the online sources where the collected Dataset and the developed Tools are available for download.

II. Dataset

A. Data

We obtain our dataset from an ongoing real-world industrial CNC production line. This production line operates fully automated and produces a variety of different products depending on customer demand and ad hoc supply. Further, the production is performed in a mixed fashion. Products within a configured set of possible product types are produced in a nearly arbitrary order and quantity. Fig. 1 shows the total number of products within the configured set of possible product types for the day, while the products in the configured set are produced in arbitrary order. The product types can differ on properties such as size, design, and weight. Therefore, the processing of the workpiece is adjusted depending on the desired result.

For this dataset we use various internal sensors to observe a single CNC machine that is part of this production line. These sensors are part of the machine's standard equipment. By using the internal sensors, the transferability of results to similar production lines is increased due to reduced requirements for the sensor setup. An overview of the measured machine properties is shown in Table I. We observe the speed and electric current of the milling spindle and the electric current of the servomotor. The electric current of the servomotor also has multiple channels for the current in each direction. We collect the data for this dataset over a period of nearly 7 months, spanning from November 2020 to May 2021. In that time frame, a total of 87650 workpieces from 39 different product types are observed during production.

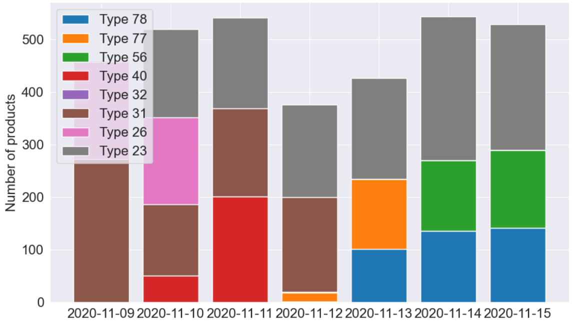


Fig. 1. Number of manufactured products per product type during a seven-day period. Value denotes the total number of products for this day, as configured product types are produced interchangeably in a nearly arbitrary order.

TABLE I: Overview of the observed features in the dataset

Feature	Channels	Samples	Time Unit	Sampling Rate	HDF5 File
Spindle Speed	1	87650	ms	7,8125Hz	spindle.h5
Spindle Current	1	87650	ms	7,8125Hz	spindlemeter.h5
Electric Motor Current	2	87650	ms	7,8125Hz	servometer.h5

The workpieces are processed in a sequential order and the production can differ depending on the product type. Therefore, the measurements are segmented into time series for each individual product. A measurement starts when a new unprocessed workpiece enters the CNC machine. Once the product is finished and exits the machine, the measurement is stopped. In between this period, we collect data of the aforementioned properties with each sensor aiming to measure their respective property every 128 milliseconds. On average we measure around 1100 time steps per sensor channel during the production of a single workpiece. An example for the resulting time series is shown in Fig. 2. There, the rough shape, and value range of the time series for a single product during normal production is illustrated.

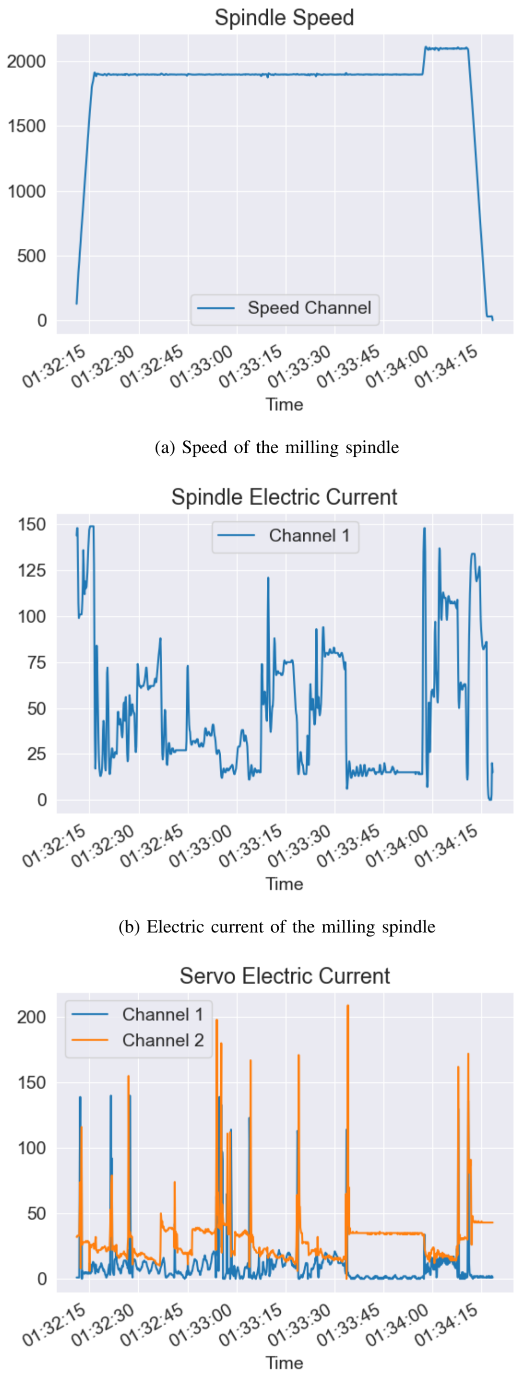
The results of the measurements are written into CSV (Comma-Separated Values) files, as shown in Fig. 3. Each property is in a separate file as the sensors collect the data independent of each other. The rows of the CSV files are structured as follows:

<timestamp>,<channel 1>,<channel 2>,. . . ,<channel n>

The first column is the time at which the measurement point was collected. The following columns are the values of the respective channels at that time.

Each product type has an individual program that defines how the production process is performed. Therefore, the measured values can deviate, when comparing the processing of different product types. Distinct types can differ in properties like duration and shape of the time series. So, the data collection system also queries what product is produced by the machine. With this we can attribute each segment to a distinct product type.

To make our data accessible and to work with it we convert each individual CSV file to a *pandas* Data Frame. The structure of a Data Frame corresponds to the structure of the respective CSV file. The columns of the Data Frame



(a) Speed of the milling spindle

(b) Electric current of the milling spindle

(c) Electric current of the servomotor

Fig. 2. Measured values of the three observed properties during the processing of a single workpiece.

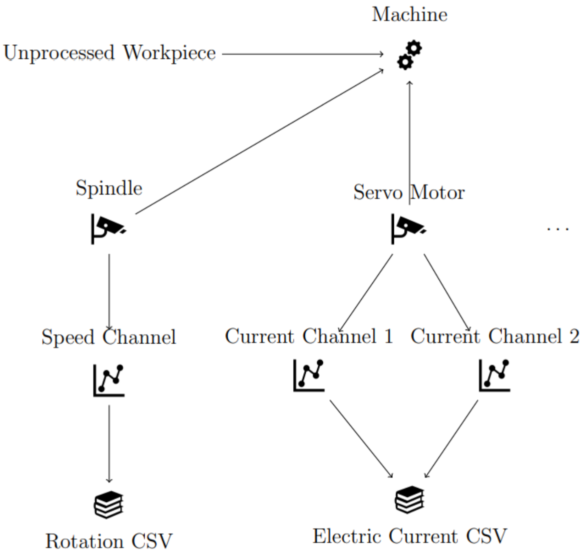


Fig. 3. Sensor collection setup.

are the channels of the property, while the index is the time of measurement. Since we have many CSV files, we collect the Data Frames in HDF5 (Hierarchical Data Format version 5) files. Each property is stored in an individual HDF5 file. The names of the files for the respective properties are shown in Table I. The hierarchical structure of the HDF5 files themselves looks as follows:

/<property>/t<product type>/<measurement>

The ID of the product type, that corresponds to the measurement, is encoded within the hierarchy as additional information.

B. Ground Truth

The ground truth data consists of machine defects and production issues during the time we collect the measurements. It is in the *Events.xlsx* file and contains labels for events when the machine is down or transitions into a state that requires human intervention or repair. Machine downtimes are only included when the machine is turned on and has enough available material. In our dataset, we have 1033 instances of such events.

The entries in the ground truth have three timestamps. The first timestamp - *date* - stands for the time the event is detected by a worker or a monitoring system. The monitoring system detects events when the production time of a workpiece surpasses a certain threshold. The other two timestamps denote the time period of the event, with the *start* and *end* timestamps. These are inserted by the workers once they perform a checkup or fix the machine. The events are non-overlapping and only one event should occur at a time. During this period either no products exit the production line, or production runs at a limited capacity.

Each entry has a label for the type of event that occurs. The labels are inserted by the workers after they resolve the issue. We have five broader groups of events denoted by numbers: Critical-(1), Major-(2), Minor-(3), Organizational-(4) and Unknown-(5).

Critical events have a severe impact on the production and the machine. They usually require the replacement of machine parts and not responding timely can cause even

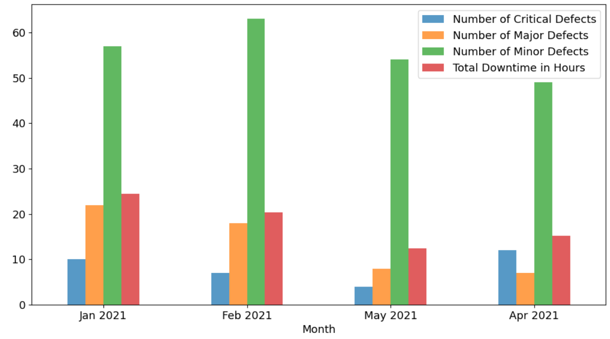


Fig. 4. Number of defects for the three most severe events and total downtime due to these three types of events in the first four months of 2021.

more damage. Critical events are usually when an important machine part breaks. Major events are less severe, but still have a high impact on the operationality of the machine and might require intervention of trained personnel. They are usually problems with the internals of the machine. Minor issues mostly interrupt the production but can be fixed with little effort and technical expertise. Common cases for minor issues are jammed workpieces or loose parts, defects on minor parts of the machine and incorrect operation of the machine. Organizational events are intentional disruptions of the production, such as performing changes on the machine. Unknown events are issues with an unknown source that could not be linked to a certain machine in the production line.

Fig. 4 shows the occurrence of critical, major and minor events in the first four months of 2021, along with the total downtime in hours due to these events. Minor events are the most frequent types of events, while critical and major events occur more rarely. In most cases, critical defects also occur less often than major defects. In total one day to half a day of production time is lost due to these types of defects every month.

The groups have a set of issues that are assigned to them. Each kind of issue is denoted by a categorical number that uniquely identifies them. Following is a list of the groups and the issues that are assigned to them: *Critical*: 1, 2; *Major*: 3, 4; *Minor*: 5, 6, 7, 8, 9, 10; *Organizational*: 11, 12; *Unknown*: 13. The ground truth contains all detected issues of these types that occur during the time of our measurements. It includes issues that originate from normal machine operation as well as defects that originate from external factors, such as human interference.

C. Data Access and Distribution

We developed a browser tool to facilitate access to and experimentation with the data by allowing users to visualize datasets and perform and reproduce data processing steps. The purpose of this tool is to reduce the burden of entry of working with this data by providing the ability to quickly run small tests, view the code of the algorithms along with the visualizations of the data, and thus facilitate the step to own experiments/applications with the data. The goal of this tool is to provide both the data and the code for the experiments in the same environment.

The tool can be populated with custom algorithms in Python code and custom data. It allows experiments to be executed on the data and then displays the visualization of the data and results.

The experiments are comprised of data processing steps combined into a pipeline. For each pipeline step, the users can define how data and intermediate results are to be displayed and visualized by customizing the executed code. For this purpose, the users are provided with a web interface in which they can add and edit scripts for each pipeline step containing the algorithms and the definitions of the visualizations. After the execution of the pipeline, the results and defined visualizations are displayed on the web interface.

The web interface consists of a starting page for an initial overview of the dataset, separate tabs to view and edit each individual data processing steps and the functionality to execute the data processing pipeline. Furthermore, the resulting visualizations and output of the data processing steps are displayed in the tab of the respective data processing step once the pipeline is executed along with the respective code. In addition, users can create additional data processing steps, edit and delete the existing ones and define how final and intermediate results are to be displayed for each step. Therefore, the tool provides a plug and play playground to adjust the data processing for further work. Further, the code for the data processing steps can be extracted to a different environment once a playground is no longer required, as it is contained as python scripts within the too data.

The created visualizations together with data and data processing steps can be passed on to other users so that they can quickly execute the already preset application and thus immediately execute the Algorithms and view the results.

In Fig. 5, the usage flow and the concept behind the application is shown. Researchers from the publishing side (Group 1) can make their data and scripts available in a way the data can be easily accessed, viewed, and executed experiments can be replicated in a local environment only requiring python and relevant libraries for the data processing. This allows other researchers (Group 2) to work with the data, inspect results and perform further work on the data that can yet again be made available.

We bundle this tool together with our raw dataset to make the data more accessible, enable replication and facilitate future work.

D. Data Quality

With the design of the sensor collection and label collection setup we aim to ensure the quality of the provided data on a level that correctly reflects the industrial process, but also inherits challenges of the real-world processes. This might even include unknown errors upcoming algorithmic solutions have to cope with.

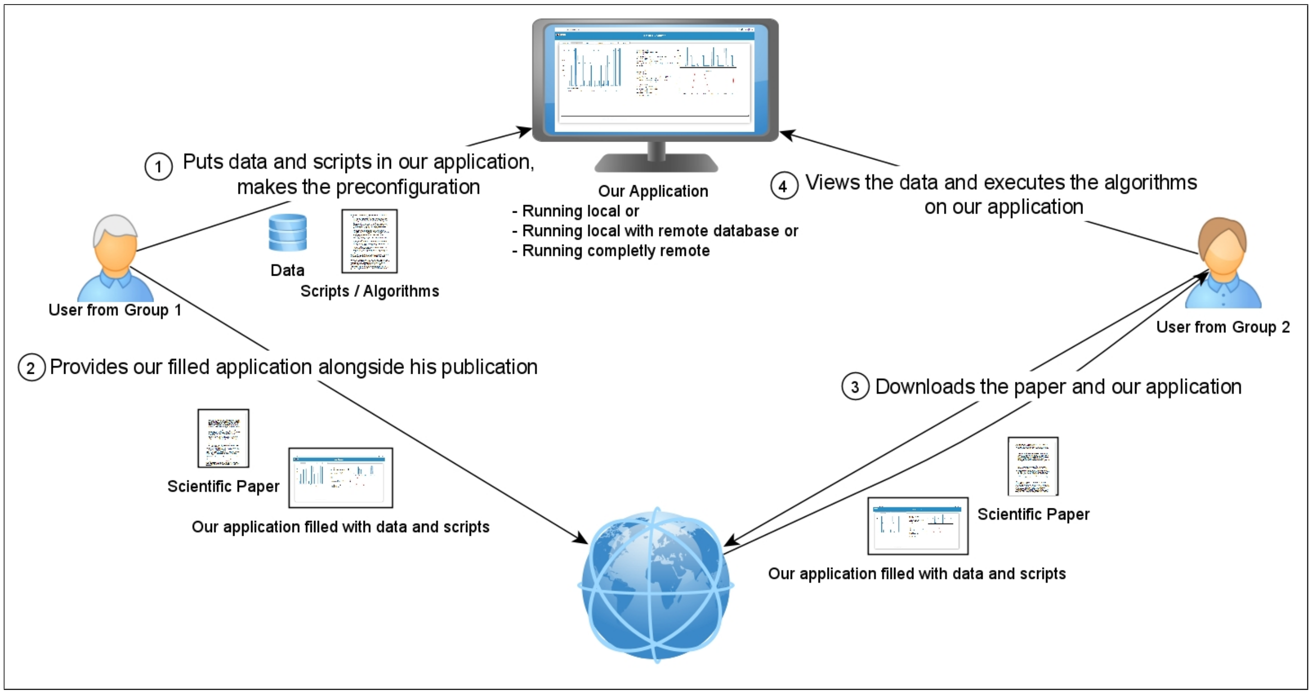

1 Puts data and scripts in our application, makes the preconfiguration
Our Application
- Running local or
- Running local with remote database or
- Running completly remote
4 Views the data and executes the algorithms on our application
Data
Scripts / Algorithms
User from Group 1
User from Group 2
2 Provides our filled application alongside his publication
3 Downloads the paper and our application
Scientific Paper
Our application filled with data and scripts
Scientific Paper
Our application filled with data and scripts

Fig. 5. Schematic of the application workflow and deployment.

To collect the dataset, we made sure that workers are trained and experienced in labelling events that happen during production. The manual labelling of the data by workers during the production is an already long-time established process and the workers are used to it. Therefore, potential mistakes are reduced as workers are already accustomed to the system. We validated the semantic correctness of the sensor and label data by performing multiple experiments [12][13][14][15] on the data to predict upcoming errors in the production line.

The challenging and unique part of the dataset on the other hand is, that we provide a dataset that faithfully represents the available data in an ongoing industrial production under real-world circumstances. This can possibly conflict with the goal of providing a ”clean” dataset where any unexplainable data is filtered out but offers the advantage of still containing each little and possible latent piece of data. We accept minor limitations regarding the quality (i.e. sensor failures) of the data that remain in the dataset on purpose. We see this as a tremendous advantage compared to clean, sometimes even artificial datasets, as our approach leads to more robust algorithms that must deal with imperfect circumstances in a non-perfect world. In addition to sensor failures, sensor noise and further inaccuracies can be present. It can occur that the completion of a product is recognized too early or too late. In the latter case, measurements of other products are attributed to a single product. As the properties are measured independently of each other, the time sensors values are sampled can deviate between different properties. While we aim to collect measurements every 128 millisecond, the actual period between sensor measurements is variable. Reasons are latency, limited processing power and throughput of the industrial network.

The ground truth consists of events that could be detected as they had an impact on the production. Therefore, labels exist for the most important events that occur, but labels are not all encompassing. As the labels are generated by observing the production instead of generating them from the sensor signal, anomalies found in the sensor signals that have no perceivable long-term impact on the production remain unlabeled. Labels for anomalies like a temporary drop-off in the production speed are unavailable. These events would only be labelled indirectly if they result in a noticeable production issue later. As it is a non-isolated running system there are also several external factors that can induce anomalies unrelated to defects. For example, the machine can be stopped or slowed down to perform trials and visual checkups. Such events are also included in the ground truth and might not have early indicators. Therefore, events prior indicators or anomalies can exist in the ground truth. As previously noted, there also exist instances of labels with an unknown source when the workers were unable to identify the defect or were absent during the occurrence of the defect. This also means it is unknown if the labels are relevant to defects of the machine.

III. Application Case - Evaluation

A major objective to achieve with this data is the detection of machine faults and defects during production. Observing the ongoing stream of sensor data, a machine problem should be reliably detected either when it occurs or in advance to take possible countermeasures and reduce damages. On the other hand, false positives - due to the dynamic production - should be minimized as appropriate responses require capacities from trained technicians. Therefore - following our previous work on similar machines [12][13][16] - we execute a baseline experiment for machine fault detection with this dataset using anomaly detection methods.

For this baseline approach we perform anomaly detection on the level of products. As samples we use all sensor data obtained during the processing of an individual item. Each sample consists of multiple time series that form the feature vector. The number of time steps in these time series can be very high and variable. Therefore, we first reduce the size of the time series to a fixed length. Piecewise Aggregate Approximation [17] is applied to transform each time series into a time series with 100 time steps. We then combine all time series to a single feature vector that is used as representation for anomaly detection. From the set of events, we aim to detect events from the critical and major groups. These issues have the biggest impact on the production and the machine. Therefore, detecting these issues has the highest priority for us. Other less critical events are ignored for this experiment.

As the objective of the fault detection is to detect problems in the ongoing production, we setup the anomaly detection to reflect an on-line application. We use Holdout Cross Validation to tune the algorithms. Therefore, the training, validation and test sets only contain data that is measured in the respective time frames. During the test stage, we also use both the training and validation set to train the anomaly detection models.

Before training the anomaly detection models, we first clean the training set by removing all samples that were measured in a time frame of 4 hours before a critical or major event. Then we use the cleaned training set to train a model for the global production context. This model should detect deviations from previously seen production across all different product types. To also take deviations in the context of the produced product type into account, we subdivide the training set by the product types. Each resulting subset only contains measurements of a single product type and is also used to train models. So, we also build models that evaluate the deviation of measurements compared to their respective peer group with the same product type. A deviation from other measurements of the same product type should be detected by these models. As machine learning techniques for the models we used: Isolation Forest [18], One-Class Support Vector Machine (SVM) [19], Autoencoder [20] and Variational Autoencoder (VAE) [21], k-Nearest Neighbors (KNN) [22], Minimum Covariance Determinant (MCD) [23], and Histogram-based Outlier Score (HBOS) [24].

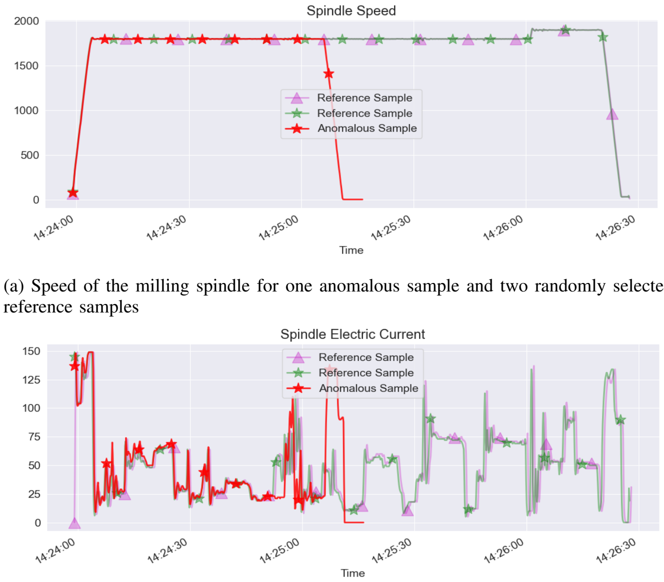
For a new measurement anomaly detection is performed with both the global model and the model of the respective product type. The state of the machine during the measurement is then deemed anomalous if both models detect it as an anomaly. Otherwise, it is considered normal.

To evaluate the performance of the anomaly detection in such a scenario we calculate the precision, recall and the scores using the scoring method by Lavin et al. [25]. We use a window of 4 hours before the actual event for all metrics, as at the time of the event there is already an impact on the machine and in the best-case events should be detected before they have an impact. Therefore, all detections within that window are considered true positives. For the method by Lavin et al. [25] we also calculate the scores for all three proposed profiles, giving different weights to false positives, false negatives, and true positives. The standard profile of this method is also used as the metric during parameter tuning.

TABLE II: RESULTS OF THE ANOMALY DETECTION EXPERIMENTS.

Method	Recall	Precision	Standard[25]	Low FN[25]	Low FP[25]
One-Class SVM	86.6	41.5	59.89	68.85	53.03
Isolation Forest	86.6	44.1	61.31	69.80	54.90
Autoencoder	86.6	41.2	60.23	69.08	53.71
VAE	86.6	40.6	60.04	68.96	53.34
KNN	30.0	50.0	24.25	26.17	23.51
MCD	53.3	23.8	33.66	40.24	25.52
hbos	83.3	37.8	58.16	66.58	51.54

We show the results using the baseline approach to detect machine defects in Table II. In this table, we show the recall, precision, and scores of the aforementioned scoring method with the three default profiles for the corresponding machine-learning method. One-Class SVM, Isolation Forest, Autoencoder and Variational Autoencoder already perform quite well and can detect or find early indicators for 86% of the detects during the test period. Isolation Forest performs slightly better than the others as it has fewer detections without corresponding labels. In Fig. 6, we show the first anomaly detected by the four aforementioned methods during the test period. This detection is compared to two other randomly selected reference samples of the same product types by overlapping and aligning them by their start. Visually the anomalous measurement is distinct from the other measurements. The curve of the electric current of the anomalous sample starts to deviate 45 seconds into the measurements, while the movement drops off after 60 seconds. The production then stops too early shortly after that and no more data is received, while the production in the reference samples continues normally. This anomaly also coincides with a critical defect that occurs at the same time. In terms of precision the scores are lower. There we have a precision of 40% in most cases, except for KNN where the recall is also very low. Generally, the detection of false positives is a challenge for all methods. All methods have a better recall than precision. This also reflects in the other scoring metrics. The scores are generally lower when putting an emphasis on few false positives. Therefore, a relatively high recall is already achievable and thus the considered types of events are detectable. On the other hand, detections that cannot be attributed to the considered events exist and a higher precision would be desirable. One thing to note in that context, is that only critical and major labels created by factory workers are used for evaluation. However, anomalies could also exist outside of these labels, as there are also other types of events and the labels are not created by analyzing the sensor signals themselves but by observing the production of the machine. Therefore, the false positives are only the context of the available labels for critical and major events. As a baseline, we manage to achieve a recall of up to 86% and a precision of up to 40% to 50% on major and critical events by performing anomaly detection in the global context and the context of the concrete product type.



(a) Speed of the milling spindle for one anomalous sample and two randomly selected reference samples

(b) Electric current of the milling spindle for one anomalous sample and two randomly selected reference samples

Fig. 6. Speed and electric current of the spindle during the first anomaly detected by the One-Class SVM in the test set compared to two randomly selected reference samples at other times. The start of the reference samples is aligned with the start of the anomalous sample to visualize differences.

IV. Challenges with Dataset/Future

The experiment and results in this paper are the foundation for future work and intended to be a basis for assessing new approaches and experiments. Therefore, there is potential to refine the approach or find new approaches that perform better in terms of the raw scores (e.g., by using different representations of the sensor data or machine learning techniques). On the other hand, we only use the most critical events as target labels. While we can already achieve decent results in terms of recall, the performance in terms of precision lower. This suggests there could be other events, in addition to the considered ones, that could be identified reliably. Consequently, exploring the recognizability of different event types is another aspect to look at.

During our experiments and first tests for an online application, collaborating with technical experts by communicating the occurrence of anomalies posed a big challenge. Usually, the occurrence of an anomaly alone is insufficient information for them. On the other hand, the technical experts usually have extensive knowledge about the machine on a technical level. Communicating an explanation in how the sensor data deviates or the expected type of problem that will occur could help to resolve machine problems more efficiently. So, after performing anomaly detection, explaining the detection, or linking it to a concrete type of problem would be another goal. Connected to explaining the detection, is the automatic labelling of the defects. Currently workers must manually label downtimes of the machine when they occur. This means when workers are not present during the downtime or lack training, information can be lost about the cause. As the labelling is also performed manually, there is always the potential that mistakes can occur. This can potentially hinder performing special measures against the systematic occurrence of certain kinds of defects. When certain problems arise regularly or in a high frequency, more throughout inspection and maintenance needs to be performed to eliminate the cause. Therefore, another aspect to improve the uptime of the machinery would be to use historic information from the dataset and create an automatic classification system to also label the downtimes automatically. Lastly, the labels in this dataset only capture the potential effect of anomalies and the labels are not directly linked to samples. Anomalies in the sensor data are not labeled. This poses a challenge when evaluating new algorithms as commonly used metrics are only applicable to a limited extent. In our experiment we dealt with this by using time windows around the events. However, this requires a parameter that influences semantics of the anomaly detection. As labels for anomalies in sensor data are often unavailable in real-world industrial processes, exploring non-parametric methods to evaluate anomaly detection with fuzzy labels can help to improve the applicability in industrial settings.

Currently our approach is based on raw sensor data streams and the hypothesis that an unexpected event, named anomaly, happens in the near future, and with a causal relation to a critical event. Given this systematic, our approach can be generalized to cases, where a given signal characteristic and its future outcome is known, but it's unclear when the signal characteristics itself began the diverge from the expected one, finally resulting in an unexpected or unwanted system behavior.

V. Conclusion

Detecting failures and defects of industrial machines by observing deviations from the normal operations is an important aspect to increase the availability of machinery and the efficiency of industrial production lines. By providing information about possible (upcoming) defects to machine operators and technical personnel actions can be taken to avoid or mitigate possible damages. One challenging scenario, that becomes more important as industrial production advances, is the detection of failures in a highly dynamic production, where the production can rapidly change depending on the requirements for the desired product. In this paper, we present a real-world dataset collected from a single machine that is part of a dynamic production line. In this production, line configured product types can be produced interchangeably depending on demand. The dataset consists of intrinsic sensor data, collected by internal sensors that are part of the default equipment of the machine. These sensors measure the movement and electric current of different machine parts. In addition to the sensor data, we also provide the occurrences of observed machine downtime - manually labelled by workers - as a form of ground truth. Along with the dataset, we also provide a tool to access and work more easily with the data by providing a playground to test new approaches. We show initial results of an approach that uses a majority vote between anomaly detection in a global context and in the context of the concrete product type to detect machine defects. There, we achieve a recall up to 86% and a precision of around 40% to 50%. This shows that, while being able to detect already a high number of defects, the precision should still be improved for technicians to effectively use the information. These results serve as a baseline for future work and improvements to detect defects more reliably in a dynamic real-world process. Further, we also outline additional challenges - aside from detecting the machine defects - that operators of the machine have when interacting with the machine to gain insight on the machine, increase uptime and take correct measures. This dataset could also help to tackle these challenges and further improve industrial production.

VI. Availability of the Dataset and Tools

The full dataset, the scripts, and the sharing platform for this paper are made publicly available at https://www.hasisaurus.at/DataSet.html. When using our Dataset please cite this work and/or one of [12][13][14][15][16].

References

[1] G. Aceto, V. Persico, and A. Pescapé, "A survey on information and communication technologies for industry 4.0: State-of-the-art, taxonomies, perspectives, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3467–3501, 2019.

[2] G. Hoelzl, "A personalised body motion sensitive training system based on auditive feedback," in *Proceedings of the* 1^{st} *Annual International ICST Conference on Mobile Computing, Applications, and Services (MobiCASE09)*, ser. Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, T. Phan, A. Montanari, and P. Zerfos, Eds., vol. 35, ICST. San Diego, California, USA: Springer, October 26-29 2009, ISBN: 978-3-642-12606-2.

[3] F. Huppert, G. Hoelzl, and M. Kranz, "Guidecopter - a precise drone-based haptic guidance interface for blind or visually impaired people," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, ser. CHI '21. New York, NY, USA: Association for Computing Machinery, 2021.

[4] A. Adel, M. A. Seif, G. Hoelzl, M. Kranz, S. Abdennadher, and I. S. M. Khalil, "Rendering 3D virtual objects in mid-air using controlled magnetic fields," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, Canada*. IEEE/RSJ, 2017, pp. 349 – 356.

[5] D. Preuveneers and E. Ilie-Zudor, "The intelligent industry of the future: A survey on emerging trends, research challenges and opportunities in industry 4.0," *Journal of Ambient Intelligence and Smart Environments*, vol. 9, no. 3, pp. 287–298, 2017.

[6] A. Angelopoulos *et al.*, "Tackling faults in the industry 4.0 era—a survey of machine-learning solutions and key aspects," *Sensors*, vol. 20, no. 1, p. 109, 2020.

[7] J. Lee, H.-A. Kao, and S. Yang, "Service innovation and smart analytics for industry 4.0 and big data environment," *Procedia Cirp*, vol. 16, pp. 3–8, 2014.

[8] B. Luo, H. Wang, H. Liu, B. Li, and F. Peng, "Early fault detection of machine tools based on deep learning and dynamic identification," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 1, pp. 509–518, 2018.

[9] D. Y. Oh and I. D. Yun, "Residual error based anomaly detection using auto-encoder in smd machine sound," *Sensors*, vol. 18, no. 5, p. 1308, 2018.

[10] R. M. Souza, E. G. Nascimento, U. A. Miranda, W. J. Silva, and H. A. Lepikson, "Deep learning for diagnosis and classification of faults in industrial rotating machinery," *Computers & Industrial Engineering*, vol. 153, p. 107060, 2021.

[11] D. Fernández-Francos, D. Martínez-Rego, O. Fontenla-Romero, and A. Alonso-Betanzos, "Automatic bearing fault diagnosis based on one-class ν-svm," *Computers & Industrial Engineering*, vol. 64, no. 1, pp. 357–365, 2013.

[12] S. Soller, G. Hoelzl, and M. Kranz, "Predicting machine errors based on adaptive sensor data drifts in a real world industrial setup," in *Proceedings of the* 18^{th} *Annual IEEE International Conference on Pervasive Computing and Communications (PerCom2020)*. IEEE, March 23-27 2020, pp. 1–9.

[13] S. Soller, B. Fleischmann, M. Kranz, and G. Hölzl, "Evaluation and adaption of maintenance prediction methods in mixed production line setups based on anomaly detection," in *International Workshop on Pervasive Information Flow (PerFlow'21), at 2021 IEEE International Conference on Pervasive Computing and Communications (Percom 2021), Kassel, Germany*, 2021, pp. 520–525.

[14] S. Soller, M. Kranz, and G. Hoelzl, "Adaptive error prediction for production lines with unknown dependencies," in *Proceedings of the 5th International Conference on Real-time Intelligent Systems (RTIS'20), Biarritz, France, Best Paper Award*, ser. WIMS 2020. New York, NY, USA: Association for Computing Machinery, 2020, pp. 227–234.

[15] G. Hoelzl, S. Soller, and M. Kranz, "Detecting seasonal dependencies in production lines for forecast optimization," *Big Data Research*, vol. 30, p. 100335, 2022.

[16] S. Soller, G. Hoelzl, T. Greiler, and M. Kranz, "Analysis of common prediction models for a fuzzy connected source target production based on time dependent significance," in *Proceedings of the 2022 International Conference on Embedded Wireless Systems and Networks*, ser. EWSN '22. USA: Junction Publishing, 2022, pp. 226–231.

[17] E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrotra, "Dimensionality reduction for fast similarity search in large time series databases," *Knowledge and information Systems*, vol. 3, no. 3, pp. 263–286, 2001.

[18] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *2008 Eighth IEEE International Conference on Data Mining*. IEEE, 2008, pp. 413–422.

[19] B. Schölkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in *Advances in neural information processing systems*, 2000, pp. 582–588.

[20] N. Japkowicz, C. Myers, and M. Gluck, "A novelty detection approach to classification," in *Proceedings of the 14th international joint conference on Artificial intelligence-Volume 1*, 1995, pp. 518–523.

[21] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[22] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient algorithms for mining outliers from large data sets," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 427–438.

[23] P. J. Rousseeuw and K. V. Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, 1999.

[24] M. Goldstein and A. Dengel, "Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm," *KI-2012: Poster and Demo Track*, pp. 59–63, 2012.

[25] A. Lavin and S. Ahmad, "Evaluating real-time anomaly detection algorithms–the numenta anomaly benchmark," in *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2015, pp. 38–44.