

# Detecting Anomalous Multivariate Time-Series via Hybrid Machine Learning

Anika Terbuch<sup>1</sup>, Paul O’Leary<sup>2</sup>, Negin Khalili-Motlagh-Kasmaei<sup>3</sup>, Peter Auer<sup>4</sup>, Alexander Zöhner<sup>5</sup>,  
and Vincent Winter<sup>6</sup>

**Abstract**—This article investigates the use of hybrid machine learning (HML) for the detection of anomalous multivariate time-series (MVTS). Focusing on a specific industrial use-case from geotechnical engineering, where hundreds of MVTS need to be analyzed and classified, has permitted extensive testing of the proposed methods with real measurement data. The novel hybrid anomaly detector combines two means for detection, creating redundancy and reducing the risk of missing defective elements in a safety relevant application. The two parts are: 1) anomaly detection based on approximately 50 physics-motivated key performance indicators (KPIs) and 2) an unsupervised variational autoencoder (VAE) with long short-term memory layers. The KPI captures expert knowledge on the properties of the data that infer the quality of produced elements; these are used as a type of auto-labeling. The goal of the extension using machine learning (ML) is to detect anomalies that the experts may not have foreseen. In contrast to anomaly detection in streaming data, where the goal is to locate an anomaly, each MVTS is complete in itself at the time of evaluation and is categorized as anomalous or nonanomalous. The article compares the performance of different VAE architectures [e.g., long short-term memory (LSTM-VAE) and bidirectional LSTM (BiLSTM-VAE)]. The results of using a genetic algorithm to optimize the hyperparameters of the different architectures are also presented. It is shown that modeling the industrial process as an assemblage of subprocesses yields a better discriminating power and permits the identification of interdependencies between the subprocesses. Interestingly, different autoencoder architectures may be optimal for different subprocesses; here two different architectures are combined to achieve superior performance. Extensive results are presented based on a very large set of real-time measurement data.

**Index Terms**—Artificial intelligence in measurement and instrumentation, hybrid learning, key performance indicator (KPI), long short-term memory (LSTM)-variational autoencoder (VAE), outlier detection, time-series.

Manuscript received 17 September 2022; revised 5 December 2022; accepted 20 December 2022. Date of publication 12 January 2023; date of current version 20 January 2023. This work was supported by the COMET Program within the K2 Center “Integrated Computational Material, Process and Product Engineering (IC-MPPE)” (this program is supported by the Austrian Federal Ministries for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK) and for Labor and Economy (BMAW), represented by the Austrian Research Promotion Agency (FFG), and the Federal States of Styria, Upper Austria, and Tyrol) under Project 886385. The Associate Editor coordinating the review process was Dr. Valentina Bianchi. (Corresponding author: Anika Terbuch.)

Anika Terbuch, Paul O’Leary, and Negin Khalili-Motlagh-Kasmaei are with the Chair of Automation, Department of Product Engineering, Montanuniversität Leoben, 8700 Leoben, Austria (e-mail: anika.terbuch@unileoben.ac.at).

Peter Auer is with the Chair of Information Technology, Department Mathematics and Information Technology, Montanuniversität Leoben, 8700 Leoben, Austria.

Alexander Zöhner and Vincent Winter are with Keller Grundbau Ges.m.b.H., 1110 Vienna, Austria.

Digital Object Identifier 10.1109/TIM.2023.3236354

## I. INTRODUCTION

THIS article investigates the use of hybrid machine learning (HML) [1] to infer the quality of industrially produced elements, for which final quality control is not possible. In the specific use-case analyzed and presented here, underground foundation columns are being produced; however, since they are subsurface their final quality cannot be inspected directly. Consequently, the quality must be inferred through the analysis of measurement data acquired in real-time from the instrumented rig. This requires the analysis of large sets of multivariate real-time measurement data (MVRTD), emanating from instrumented industrial equipment. This is a pertinent topic since industrial IoT [2] is making ever-increasing volumes of measurement data available for analysis. Furthermore, HML, for example, physics-informed neural networks [3], is an emerging research topic and is considered the best approach to obtaining reliable results in conjunction with the analysis of the measurement data from physical systems [4], [5], [6]. Although this article is focused on a specific use-case, many of the results are relevant in other applications; in particular, the examination of different autoencoder architectures for the unsupervised analysis of multivariate time-series (MVTS) data and the combination with a genetic algorithm to optimize the hyperparameters.

The importance of machine learning (ML) in the field of instrumentation and measurement [4], [7] is increasing with the growing volumes of measurement data recorded in industrial processes. However, as shown in [5], the two communities, i.e., the machine learning and the instrumentation and measurement community, use different terminologies. As a result, we feel it is important to have interdisciplinary publications that bridge both the topics.

As described in [7], ML architectures should be chosen with care when applied in safety-relevant applications, as is the case here. For this reason, an HML approach was chosen, which augments an outlier detection technique based on multiple physics-based metrics, with unsupervised ML. The HML techniques were successfully applied in other fields related to instrumentation and measurement, e.g., in [8] and [9]. Many ML approaches presented in literature use publicly available datasets to perform ablation studies [10] and performance benchmarking of different architectures. A survey on different methods for time-series forecasting [11] on publicly available time-series datasets suggested that hybrid approaches are superior to pure statistical or ML approaches. The same study

concluded that more complex models do not necessarily lead to higher accuracy. This is important, since high accuracy, in the field of instrumentation and measurement, is a highly desirable feature [5], [6].

Especially in anomaly detection, benchmarks need to be treated with care, since most suffer from one or all the following issues [12]: the problem is too trivial and can be solved in a few lines of code; the anomaly density is too high; if supervised learning is used, the ground truth is mislabeled and often there are no data available after the anomaly has occurred, and Wu and Keogh [12] called this as the *run to failure bias*. The study done on the comparison of the classical pure ML methods, pure (deep) ML methods, and the combination of both, HML, suggests that there is no pure ML method that can outperform a hybrid system [13].

### A. New Contributions

The initial results of this work were published in [14]; however, since then, significant modifications have been made and improved results obtained. This article presents these extensions and new results.

- 1) An HML framework that captures a priori knowledge in the form of key performance indicators (KPI), which is used to implement autolabeling, permitting a truly unsupervised learning. Furthermore, the combination of two means for anomaly detection, KPI and ML, is introduced; this leads to a redundant anomaly detection system and a higher likelihood for detecting anomalies.
- 2) The conceptual framework for handling MVTS measurement data in an object-oriented way is presented; covering all the tasks from data ingestion to outlier detection. The data are handled together with the pertinent meta-data, events, and segments. The concept that industrial processes can be described as consisting of subprocesses is discussed. KPIs are defined for the subprocesses separately and this leads to a hierarchical anomaly detection in the subprocesses. Aggregating the results yields a more granular classification of anomalies and reveals possible interactions between the subprocesses.
- 3) A clear definition and algorithmic implementation for the *outlierness* is provided, i.e., a metric for the extent to which a time-series is considered to be an outlier by the KPI-based anomaly detector.
- 4) The statistical analysis of the performance of autoencoders with LSTM and BiLSTM layers for the subprocesses is presented.
- 5) The implementation of the autoencoder training has been modified, to significantly reduce the amount of padding required within the mini-batches during training.

### B. Organization of the Article

This article is organized as follows: in Section I, an introduction to the topic of HML for anomaly detection in multivariate time-series is given; in Section II, insights into the use-case are presented; Section III discusses the structured data handling for large amounts of industrial real-time data. The first part of the hybrid framework based on KPI is

shown in Section IV, whereas the anomaly detection based on ML can be found in Section V, and the results of the comparison of architectures are presented in Section VI. The article is concluded with insights gained and further work in Section VII.

## II. USE-CASE AND NATURE OF THE DATA

The vibro-ground improvement techniques are used for almost 100 years to support buildings and infrastructure works; however, there is still no standardized method for quality control for ground improvement techniques [15]. The increasing economic pressure in the construction industry and a higher risk awareness accelerated the change process of traditional, mostly manually performed techniques, toward automated solutions for quality control [16]. The produced foundation columns are subsurface, and therefore, the quality control needs to be based on indirect methods and reliable labeling for supervised ML is not a feasible solution. For the last 30 years monitoring KPIs, which were derived from the recorded data from instrumented production rigs, built the basis for geotechnical and quality evaluations of the process [17]. These data are also utilized in installation reports which to this day are the core of quality control. For each foundation column, an installation report is created, which needs to be manually evaluated by geotechnical experts [15]. However, evaluating hundreds, if not thousands, of installation reports with limited human resources increases the risk that the controls are not performed with the due-diligence required. For this reason, our framework is used to support the geotechnical expert at this task and refining the KPI definitions to cover as many anomalies as possible. The KPI refinement is driven by new process insights, gained by the knowledge discovery process, performed after ML anomaly detection, to increase the interpretability of the anomaly detection framework, which is essential in critical infrastructure and safety-relevant applications [18]. The opportunity costs of false negative are much higher than of false positive detection. Therefore, the two anomaly detection results are combined with a logical or—if an MVTS is flagged by one of the means for anomaly detection as anomalous it is considered as anomalous and undergoes further investigations by geotechnical experts.

In more detail, the use-case presented here is the vibro ground improvement process. The goal is to improve the bearing capacity of the ground so that more stable foundations can be produced for a building. However, this leads to a safety-relevant system, since failure to detect an anomalous foundation column can lead to a local instability of a building. The process consists of four phases: *run-in*, process preparation is performed; *penetration*, whereby a rig penetrates the ground with a large vibrator until a predefined bearing capacity is reached; *compaction*, is the repetitive process of withdrawing the vibrator, introducing gravel into the ground and compacting with the vibrator; and *run-out* finalizing the process.

A typical building site may have between  $250 \leq m \leq 1500$  such foundation columns. Fusing the location data with machine data enables the georeferenced viewing of the data associated with the foundation columns. This supports

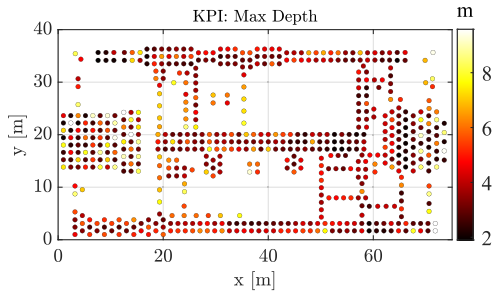


Fig. 1. Georeferenced data from an exemplary building site. Each point corresponds to the location of an underground foundation column. The color of the point represents the value of some specific KPIs; in the case shown here, it is the depth required to achieve a specific load-bearing capacity. There is a real-time MVTS associated with each of these foundations, and for this site, there are  $m = 637$  points.



Fig. 2. Here a series of three instrumented rigs can be seen on a building site. This gives an impression of the scale of operations and the relevance of location registered data.

the geotechnical experts in establishing an overview of the geotechnical properties and enables an automatic detection of changing the subsurface properties. An example of such a view is shown in Fig. 1; this site has  $m = 637$  foundation columns. The georeferencing permits the identification of systematic changes in the subsurface ground properties over a site. This is important since reliable detection of anomalies requires the separation of systematic changes from local anomalous behavior.

A series of instrumented rigs work in parallel on a single building site; see Fig. 2, producing the required foundation columns. The instrumented rigs have  $n = 9$  sensor channels used in this process. The sensor data acquired in real-time are sampled at  $f_s = 1$  Hz. This yields an MVTS for each foundation column; an example of such data can be seen Fig. 3. Due to changes in the subsurface properties, the time required to produce the foundation may vary significantly; as a result, the time-series have strongly varying lengths. For the site data shown in Fig. 1, there is a median number of time-steps per foundation of  $t_{\text{med}} = 345$  and an interquartile range (IQR) of  $t_{\text{IQR}} = 169$ . A  $t_{\text{med}}$  of 345 time-steps with 1-Hz sampling frequency corresponds to an average lead time of 345 s. Consequently, the computational methods must be capable of analyzing, the execution of the production process with strongly varying lead time, which leads to recordings of the process with widely varying lengths. To summarize, there are approximately 1000 MVRTD associated with a single building site, each of these time-series having  $n = 9$  sensor

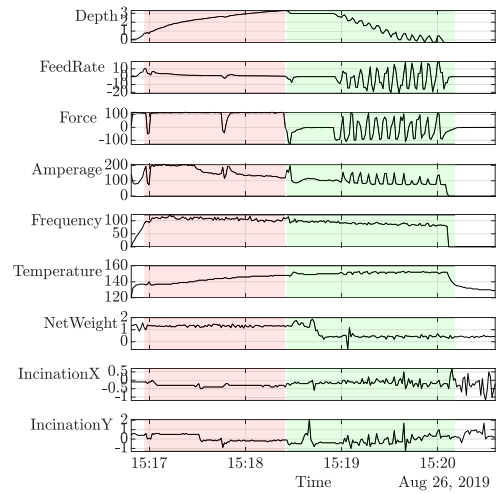


Fig. 3. Example of the real-time data acquired from the rig while producing a single foundation column. One such MVTS is acquired per foundation column. The four phases of the process can be seen: run-in (white), penetration (red), compaction (green), and run-out (white). Only the penetration and compaction phases are relevant for the quality of the foundation column.

channels and several hundred time-steps recorded per sensor. Furthermore, there are multiple building sites to be monitored.

### III. STRUCTURED HANDLING OF DATA

A significant portion of this project is associated with managing such volumes of multivariate measurement data, together with the corresponding metadata, in a systematic manner. There are tens of thousands of MVRTD that need to be handled. Consequently, a systematic data ingestion process has been defined. In this process, each of the MVTD is ingested. In the same step, events and segments are determined by applying a rule-based system. By performing data fusion, the data from computer-aided design (CAD) are merged with machine data for each of the foundations. This provides for the georeferencing of the data and enables the comparison of planned versus executed work. An object class for an MVTS has been defined. It provides a container for the MVRTD and augments this with the respective metadata, events, and segments. In this manner, all data required to segment and process the data from a specific foundation column are contained in a single MVTS object. These objects are saved in binary HDF5 [19] format and made available via a standard interface. This facilitates the exchange of data and fast loading into memory.

In addition, an index has been implemented with one column per foundation element and one row per indexing value, e.g., KPI. This permits the identification of MVTS from either its metadata or the values of the associated KPI or by an indexing value emanating from the ML evaluation of the data.

### IV. KEY PERFORMANCE INDICATORS

Given the collection of MVTS, the task now is to individually characterize them by a series of metrics, called KPIs here. The goal of the technical KPI is to capture expert knowledge about the process and physical properties that can be expressed in terms of scalar metrics computed from the real-time machine data. These provide the physics-based side of the hybrid learning process.

The penetration and compaction phases are definitive for the quality of the executed foundation column, whereas the complete process, including run-in and run-out, may be relevant for the efficiency. Consequently, KPIs are defined for and categorized according to: penetration, compaction, and complete element. The categorization is achieved via metadata and not hard coded. In previous work [14], these phases were analyzed together. However, defining KPI in separate categories provides the possibility of a more granular classification of the MVTS according to the subprocesses. For example, the penetration phase may be anomalous, whereas the compaction is not. Currently, there are more than  $p = 50$  different KPIs computed.

#### A. Exemplary Technical KPI

Here, two new exemplary KPIs<sup>1</sup> are presented. They are physics-based metrics and permit the identification of certain types of anomalies. Furthermore, they can be computed using the map-reduce paradigm [20], opening the door to parallel computation, if required.

The incremental work as a function of time  $w(t)$  is computed from the amperage  $a(t)$  multiplied with the operating voltage (electrical work) and the product of force  $f(t)$  and penetration rate  $dd(t)/dt$  (mechanical work). This computation has the same temporal resolution as the MVTS data and corresponds to the map step in map-reduce. Then the total work  $W$ , required to perform penetration, is the integral over  $w(t)$

$$W = \int_{t_s}^{t_e} w(t)dt \quad (1)$$

where by  $t_s$  and  $t_e$  correspond to the start and end times for penetration, respectively, and this aggregation corresponds to the reduce step. The total work required to complete penetration  $W$  is used as a metric to characterize each MVTS.

The second KPI presented  $L$  is the ratio of the traversed length to the depth penetrated. Given  $d(t)$  the depth as a function of time,  $L$  is computed as follows:

$$L = \frac{\int_{t_s}^{t_e} |d(t)|dt}{\int_{t_s}^{t_e} d(t)dt}. \quad (2)$$

Consider the exemplary MVTS shown in Fig. 4: this corresponds to a straightforward penetration of the ground and yields  $L = 1$ . In contrast, consider Fig. 5: as can be seen, there were difficulties penetrating the ground and the vibrator had to be retracted several times and the process restarted. This behavior is considered anomalous; however, it has predictive value, since together with the georeferencing it can be used to determine whether this is a local random anomaly, or if the subsurface properties are changing systematically over a site.

#### B. Heat-Maps and Outlier Detection via KPI

Currently, more than  $p = 50$  KPI are computed, in three categories, for each foundation column. The KPI can be organized

<sup>1</sup>It is not possible to present details on more of the KPIs due to the limited space available in this article.

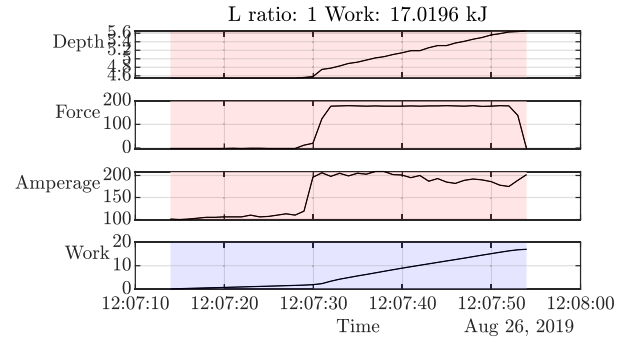


Fig. 4. Depth  $d(t)$ , force  $f(t)$ , and amperage  $a(t)$  are the channels that form the MVTS of a single foundation column. The work channel  $w(t)$ , underlined in blue, is a derived variable, computed from the above three channels and has the same temporal resolution as the real-time data from the machine. The integral over  $w(t)$  yields the total work required to perform penetration. This dataset corresponds to the almost ideal process of creating a column. The  $L$  ratio, indicated in the header, refers to the ratio of traversed distance to depth penetrated.

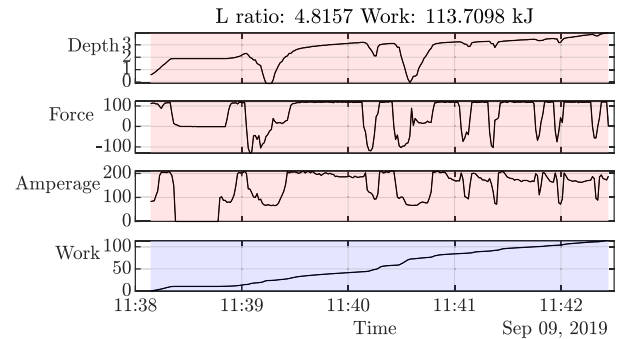


Fig. 5. Data similar to Fig. 4 but for an anomalous foundation column. This is an example where the operator had difficulties penetrating the ground with the vibrator. Both the  $L$  ratio and the total work performed are physics-relevant metrics and are suitable to identify this type of anomaly.

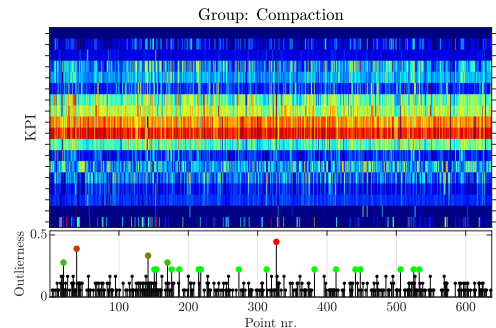


Fig. 6. Heat-map and outlieriness for the KPI relevant to the compaction phase of the process, and the data are for all the foundation columns on the building site. Top: Heat-map, whereby the horizontal dimension refers to the number of the individual points shown in Fig. 1; the KPIs are mapped to the vertical axes. The coloring of the elements on each row corresponds to the relative value of that KPI at each point. Bottom: The outlieriness for each foundation column.

as a matrix  $T(j, k)$ , whereby each row  $j$  corresponds to a KPI and each column  $k$  to a specific foundation point. Normalizing the KPI matrix  $T$ , using min-max normalization, by row provides a uniform scaling for the graphical representation, e.g., in Fig. 6 the KPI heat-map for the compaction phase and in Fig. 7 the heat-map of the KPI relevant for all phases is shown.

In addition, the KPI matrix  $T$  is used to detect MVTS that are statistical outliers; this infers the possibility of an anomalous foundation column. The foundation columns are subsurface, the precludes access to the *ground-truth*, and

**Algorithm 1** Pseudo/Code for the Computation of *Outlierness*


---

```

procedure OUTLIERNESS( $T$ )
  ▷ Given the matrix  $T$  of KPI compute the outlierness for each
  foundation column.
   $B \leftarrow \text{false}(\text{size}(T))$            ▷ Initialize binary matrix
   $[o, p] \leftarrow \text{size}(T)$ 
  for  $j = 1, \dots, o$  do
    ▷ For each row compute upper and lower bounds
     $t \leftarrow T(j, :)$                  ▷ Extract row vector  $t$ 
    ▷  $q_{25}(t)$  and  $q_{75}(t)$  refer to the 25% and 75% percentiles of the
    vector  $t$ , whereas  $IQR$  refers to the interquartile range.
     $b_u \leftarrow q_{75}(t) + 1.5 IQR(t)$ 
    ▷ Compute upper bound  $b_u$ 
     $b_l \leftarrow q_{25}(t) - 1.5 IQR(t)$ 
    ▷ Compute lower bound  $b_l$ 
    for  $k = 1, \dots, p$  do           ▷ For each column
      ▷ Detect individual outliers if  $v$  is outside the bounds
       $v \leftarrow T(j, k)$ 
      if  $\text{not}(b_l \leq v \leq b_u)$  then
         $B(j, k) \leftarrow \text{true}$ 
      end if
    end for
    ▷ Sum up each column to obtain the result vector  $r$ 
  for  $k = 1, \dots, p$  do
     $r(k) \leftarrow \sum_{j=1}^o B(j, k)$ 
  end for
  return  $r$                        ▷ The vector of outlierness values.
end procedure

```

---

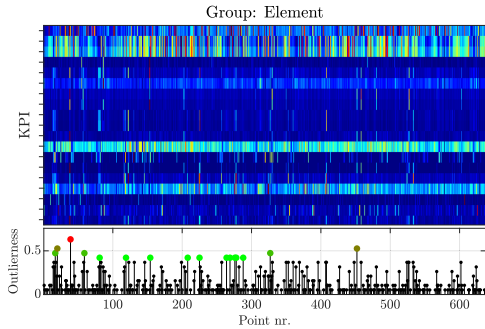


Fig. 7. Heat-map and outlierness for the KPI relevant to the complete element produced; see also Fig. 6.

consequently, an inference must be used at this point. The term *outlierness* is coined here, and it is a relative measure of the degree to which the MVTS can be considered to be an outlier. It is the number of KPI for which the column data are a statistical outlier divided by the total number of KPI being considered. The pseudocode for the computation of the outlierness is shown in Algorithm 1.

## V. ML-BASED ANOMALY DETECTION

As described in [18], autoencoders are a popular choice for anomaly detection in critical infrastructure. The task of identifying patterns that are not present in data from normal operations is called anomaly detection [18]. Anomaly detection can also be described as a binary classification problem with one class containing the anomalous data and one class containing the nonanomalous samples [21]. Modeling the normal behavior in physical systems is often not a feasible solution, if it is unknown or simply too complex. In ML

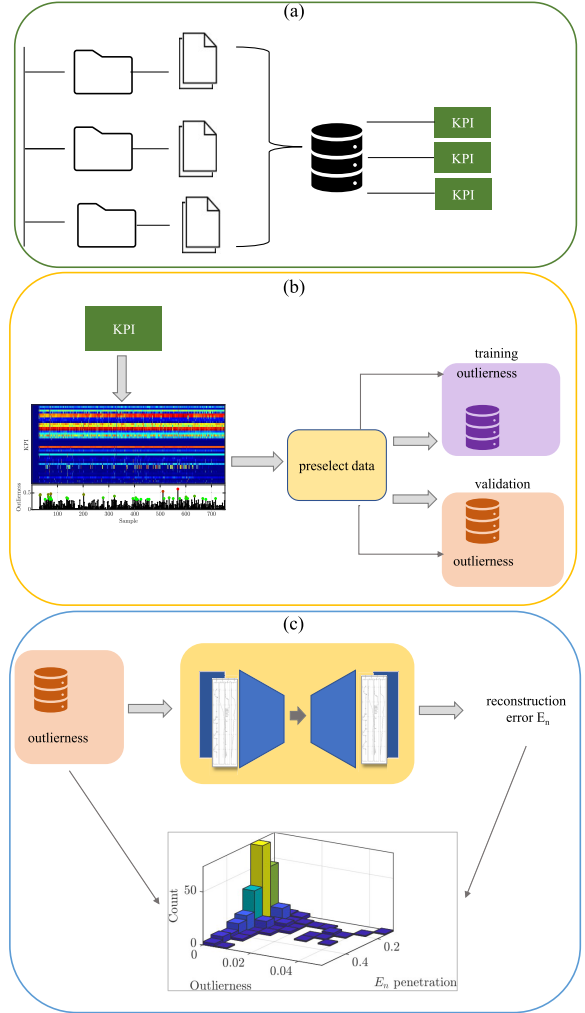


Fig. 8. Interplay of the components of our hybrid framework. (a) Files are recorded at different sites and mapped to a storage location. From these data, the KPIs are calculated as part of the data-ingestion process. (b) From KPI visualizations—e.g., the heat-map is created and the outlierness—rank of the MVTS—is calculated. Based on the outlierness, the files are separated in a training and validation set. (c) Process of getting the autoencoder anomaly score for the data of the validation set using a trained ML model. By passing the validation set through the autoencoder, the reconstruction error can be calculated. In the end, the two anomaly scores per MVTS for the KPI-based anomaly detection and the autoencoder-based anomaly detection are combined in a bivariate histogram as a starting point for the knowledge discovery process.

approaches for anomaly detection, this normal behavior is learned/inferred from provided data samples [21]. A simplified drawing of the architecture used is shown in Fig. 8.

Most of the lately invented architectures for anomaly detection in time-series data are designed for streaming data for the multivariate [22], [23] or univariate case [24] using sliding window approaches. The industrial use-case presented in this article, however, deals with a high number of MVTS data files with recordings of varying length. Each MVTS is complete at the time of evaluation. In this manner, it is more closely related to boundary value problems, than locating a changing pattern in streaming data.

To apply the autoencoder to the samples of varying length, a novel resampling approach was developed (see Section V-D), which reduces the effects of resampling to a minimum.

This approach differs from the previously mentioned, since the goal is not to localize the position of the occurrence of an anomaly, but to detect whether a built column, an MVTS, is anomalous as a whole. In contrast to approaches (e.g., [25]) which apply ML on KPI data, in this framework the KPI data are used as a separate mean for anomaly detection and for the unsupervised training set construction. The learning algorithm, however, is applied on the recorded time-series data. In addition, the available process knowledge should be incorporated into the hybrid anomaly detector to acquire explainable and physically consistent results.

### A. Objective-Based Preprocessing

To meet different objectives, different preprocessing steps need to be performed to extract and enhance the special characteristics of the data. The objective can also be dependent on the subprocess monitored. In this section, the preprocessing steps regarding outlier detection in terms of the quality of the foundation columns are shown.

1) *Feature Selection*: Using the whole set of channels is in most cases not beneficial for the ML performance. A higher number of features used results in a higher number of learnable parameters, increased model complexity, and training time. The goal is to reduce the number of features used while preserving the relevant information and eliminating redundant features [26].

Despite using unsupervised ML, knowledge about the monitored system and the different recorded channels should be used when deriving the subset of channels. In this approach, the channels from which the work performed as a function of time (see Figs. 4 and 5) can be derived were chosen. Depending on the goal that should be archived with ML, a different subset of channels may be required as input to the ML model. Two possible examples of goals that require a different channel selection are anomaly detection regarding the foundation column quality or anomaly detection regarding the production efficiency of the ground improvement process.

2) *Rule-Based Segmentation*: Based on the definition of rules and events using expert knowledge, the MVTS are segmented into subprocesses. This is done because separate ML models are trained for the separate phases. The subprocesses are indicated in Fig. 3 with different colors.

3) *Data Trimming*: The objective of anomaly detection for the case study is to detect anomalies in the sense of quality. The ground improvement process contains process pauses, e.g., logistical difficulties in gravel delivery. These pauses do not have an impact on the effectiveness of the performed process but the process efficiency. To meet our objective, these pauses when no drilling is performed are removed from the MVTS used for ML. After trimming the data by removing the process pauses, in MVTS with anomalous process behavior discontinuities can occur. These discontinuities caused by operating errors and other process anomalies can be detected using appropriate techniques [27], [28]. This is done to use these anomalous cases in a knowledge discovery process. This is based on previous work published in [29].

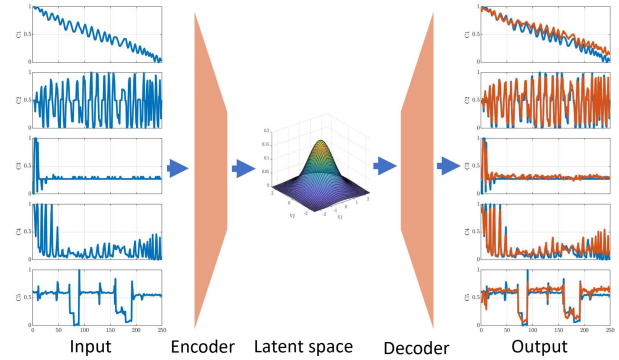


Fig. 9. Schematic representation of an autoencoder with the encoder that encodes the input features into latent variables and the decoder that reconstructs the lower dimensional representation available in the latent space.

### B. Architecture Selection

An autoencoder is an ML structure performing two mappings [30].

- 1) *Encoder*: Mapping  $\Psi$  from the original data to the latent variables in the latent space.
- 2) *Decoder*: Mapping  $\Sigma$  from the latent space to the reconstructed signal.

The goal is to extract latent variables that allow for a minimal amount of distortion in the reconstructed signal [30], [31]. The dimension of the latent space  $n_z$  is for anomaly detection chosen to be  $n_z < b$ ,  $b$  being the dimension of the encoded data [32]. Autoencoders and variational autoencoders (VAEs), which differ in the regularization of their cost function, were successfully used for anomaly detection in time-series in the past [33]. A schematic representation of the ML architecture is shown in Fig. 9.

The assumption behind anomaly detection using VAEs is performing some kind of dimensionality reduction and the assumption that anomalies contain nonrepresentative features which cannot be encoded into the lower dimensional space [34], [35]. As described previously, the three-channel MVTS used for ML is converted into a lower dimensional latent representation. The degree of dimensionality reduction is determined by the dimension of the latent space, the latent dimension  $n_z$ ; in this work,  $n_z = 1$  was chosen to archive a sufficiently high degree of dimensionality reduction.

A recurrent neural network (RNN) was incorporated into both, the encoder and the decoder, to handle the sequential data [36], namely, a long short-term memory (LSTM) [37] layer. Unidirectional LSTM is processing the input data in the forward direction, whereas bidirectional LSTM (BiLSTM) is processing it in a forward and backward direction [38]. Bidirectional models can only be used when the whole data sample that they are applied to is available [36]. This condition is fulfilled in the use-case presented here, since the data analyzed are not streaming data. BiLSTM performs well, especially when the value of time-step  $t$  depends on both, prior and past time-steps [39]. This is the case in boundary value problems. The drilling process described in this article can be seen as a (homogeneous) boundary value problem since the drilling starts at the surface and ends at the surface.

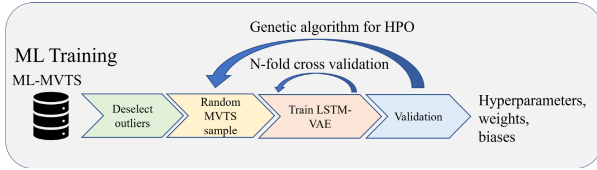


Fig. 10. Process from ML-MVTS to trained ML model incorporating HPO.

### C. KPI-Supervised Training

As described in Section V, most of the data, gathered with physical systems, are unlabeled and reliable labeling is hard to archive [21]. Because of these circumstances, an unsupervised anomaly detection approach was chosen. Unsupervised anomaly detection is done under the assumption that the training set  $\Gamma \subset \Phi$ , whereas  $\Phi$  is the set of all available samples, contains mostly samples from normal operations but could contain a small number of still unknown anomalies [13], [40]. This training set is constructed using the outlieriness calculated with the KPI [13] for this reason, and we call it *KPI-supervised training* as the anomalous samples are excluded and autolabeled using the outcome of the KPI classifier. This excludes the already known anomalies covered by the KPI framework.

The training set is constructed using the outcome of KPI anomaly detection. The proposed framework does not require human labeling, since it uses the results gained with the KPI classifier to construct a training set only consisting of MVTS which are considered to be nonanomalous. The whole process of preprocessing and training is illustrated in Fig. 10.

### D. Length-Preserving Training

The training procedure based on stochastic gradient descent was adjusted to apply on MVTS of different time-series lengths and minimize the effects of resampling to a common length which is required for gradient calculation. During training, the samples are sorted according to their time-series length and only resampled to a common length, downsampled to the shortest length of the samples forming a mini-batch; see Fig. 11. After training the autoencoder in this new way, it can reconstruct samples while preserving their real length. This is desirable since an unusual time-series length, the process length of an industrial process, can be a sign of anomalous process execution. This avoids the necessity to resample the time-series to a common time-series length [41], [42] or to pad the data with zeros to the length of the longest time-series [43], [44], which adds information and has an impact on the learning performance, especially when used with LSTM [45].

The cost function optimized with stochastic gradient descent is a regularized [46] version of the least-squares cost function minimizing the reconstruction error. Being  $Y_k$  the  $k$ th sample of a mini-batch and  $\hat{Y}_k$  the reconstructed MVTS, i.e., the output of the autoencoder the reconstruction error  $E_r$  of a mini-batch of size  $m$  is given as<sup>2</sup>

$$E_r = \sum_{k=1}^m \|Y_k - \hat{Y}_k\|_F^2. \quad (3)$$

<sup>2</sup> $\|A\|_F$  denotes the Frobenius norm of the matrix  $A$ .

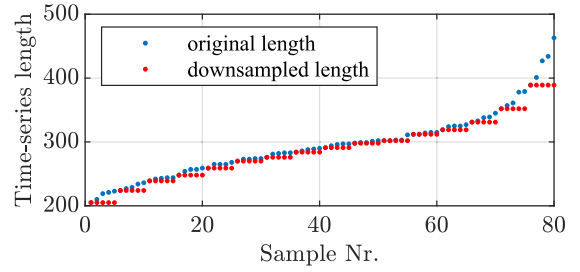


Fig. 11. Samples forming a mini-batch of the training dataset are downsampled to the shortest time-series length of the mini-batch and the training dataset is not resampled to a common time-series length.

In the VAE, the regularization is based on the Kullback–Leibler divergence<sup>3</sup> between two distributions  $\Omega$  and  $\Pi$  given as  $D_{\text{KL}}(\Omega\|\Pi)$  [47], [48]. The loss function of a VAE can be written as [47] and [48]

$$E_r = E_r - \beta D_{\text{KL}}(\Omega\|\Pi) \quad (4)$$

with  $\Omega$  being a prior chosen by the user, often a multivariate Gaussian distribution with zero mean and a diagonal covariance matrix. The second distribution  $\Pi$  is the learned distribution in the latent space. In the formulation introduced by [48]  $\lambda = 1$ , in this case, the objective of the learning algorithm is minimizing the *evidence lower bound* (ELBO) [47]. In our approach, we use a so-called  $\beta$ -VAE [49] where the Lagrangian multiplier  $\beta$  is a hyperparameter that is optimized during hyperparameter optimization (HPO), described in Section V-E, to ensure that the balancing between the two terms of the loss function is optimal.

### E. Hyperparameter Optimization

The hyperparameters have a significant impact on the performance of the network [50]. The optimal performance of an ML architecture on a dataset is obtained by optimizing the hyperparameters. The available hyperparameters depend on the architecture chosen and their optimal value on the problem and the corresponding data. The search space of the HPO grows exponentially with the number of parameters optimized. This combined with the long training times of ML architectures mostly heuristics or meta-heuristics are used [36], [43].

In this framework, a genetic algorithm is used for the HPO, and the following hyperparameters were optimized.

- 1) Trained epochs:  $e \in [50, 150]$ .
- 2) Mini-batch size  $s \in [2, n]$ .
- 3) Number of neurons encoder  $n_{\text{ne}} \in [10, 100]$ .
- 4) Number of neurons decoder  $n_{\text{nd}} \in [10, 100]$ .
- 5) Learning rate  $\alpha \in [0.003, 0.1]$ .
- 6) Regularization factor  $\beta \in [0, 50]$ .

The genetic algorithm adjusted for this framework is described in more detail in [51] and was executed with the following settings: 15 individuals per generation, threefold cross-validation for the fitness estimation, and a maximum of ten generations. Moreover, a combination of two crossover functions was used to obtain a better exploration of the search space.<sup>4</sup>

<sup>3</sup>The Kullback–Leibler divergence is not symmetric and therefore cannot be used as a measure.

<sup>4</sup>Further information on the genetic algorithm and implementation details can be found in [51].

TABLE I  
RESULTS OF THE HPO

Architecture	sub-process	$e$	$n_{ne}$	$n_{nd}$	$\alpha * 10^5$	$s$	$\beta$
VAE-LSTM	penetration	69	40	56	1998	11	5
VAE-LSTM	compaction	132	39	47	3689	15	8
VAE-BiLSTM	penetration	117	28	39	5987	17	9
VAE-BiLSTM	compaction	132	39	47	3689	15	8

### F. Anomaly Detection Using Autoencoders

The last step of anomaly detection using autoencoders is performing the binary classification task. In this step, the model trained on the subset  $\Gamma$  is applied to the whole available dataset  $\Phi$ .

When using autoencoders, the output is the reconstructed signal from which the reconstruction error can be derived [33], [44], [52]. From the reconstruction error, the anomaly score is calculated. The anomaly score used here for the sample  $y$  with a time-series length  $t$  is calculated as follows:

$$E_a = \sum_{c=1}^t \|y^{(c)} - \hat{y}^{(c)}\|_F \quad (5)$$

with  $y^{(c)}$  denoting the measurement taken at time-step  $c$  and  $\hat{y}^{(c)}$  the reconstructed measurement at time-step  $c$ . Because of the changing time-series length, the anomaly score  $E_a$  is normalized by the number of time-steps  $t$ , which leads to a normalized anomaly score  $E_n$

$$E_n = \frac{E_a}{t}. \quad (6)$$

To get the classification, a threshold  $u$  is needed to represent the boundary between the classes. Since the distribution of reconstruction errors is positive semi-definite, a skewness adjustment needs to be performed [53], [54], [55]. The MVTS with  $E_n < u$  get assigned to the class of nonanomalous samples and MVTS with  $E_n \geq u$  are marked as anomalous.

## VI. ARCHITECTURE COMPARISON AND RESULTS

The novel hybrid framework was applied to the case study introduced in Section II. Separate autoencoders were trained for the analyzed subprocesses, compaction, and penetration. The following architectures were compared.

- 1) *VAE-LSTM*: VAE with one LSTM layer in the encoder and the decoder.
- 2) *VAE-BiLSTM*: VAE with one BiLSTM layer in the encoder and decoder.

### A. HPO Results

The HPO was executed four times to cover all the two compared architectures; the two phases and the results are shown in Table I.

### B. Architecture Comparison

In this section, different architectures are evaluated phase-wise with metrics based on the anomaly score which is derived from the reconstruction error of trained autoencoders. Because of the nature of ML problems, random initialization

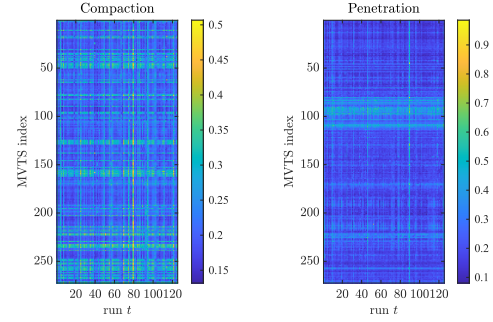


Fig. 12. Visualization of the fluctuation in the anomaly scores  $E_r$  of MVTS of a site when the training procedure is repeated  $r = 125$  times for the subprocesses compaction (left) and penetration (right).

TABLE II  
SUBPROCESSWISE ML ARCHITECTURE COMPARISON

statistical quantity	penetration		compaction	
	VAE-LSTM	VAE-BiLSTM	VAE-LSTM	VAE-BiLSTM
$mean(mean(E_{nn}))$	0.1997	0.2175	0.1986	0.2319
$median(median(E_{nn}))$	0.1883	0.1992	0.1966	0.2247
$mean(mean(E_{na}))$	0.3918	0.4303	0.3161	0.2880
$median(median(E_{na}))$	0.3821	0.4053	0.2353	0.2483
$mean(distance\ means)$	0.1921	0.2128	0.1251	0.0550
$median(distance\ medians)$	0.1932	0.2110	0.0516	0.0275

of learnable parameters, and an overdetermined solution space, their performance varies when executing them multiple times on the same dataset using the same hyperparameters. These observations were previously described in [51].

The visualization of the anomaly score of the trained architectures is shown in Fig. 12. The performance fluctuation of the training on the same data with the same hyperparameters can be observed on the vertical lines. The horizontal lines indicate that the sample was reconstructed with a similar anomaly score in the majority of the  $t$  trained autoencoders.

To get an estimate of the performance of the architectures,  $r = 125$  instances of each ML architecture were trained and then an outlier detection was performed. The goal was to identify the best architecture in terms of reliability, the variance of the outcome should be low, and at the same time the distance between the two groups should be maximized. The obtained results are shown in Table II.<sup>5</sup>

The results of the comparison of the ML architectures as shown in Table II are interpreted as follows.

- 1) The mean of the means of the reconstruction errors of samples classified by the autoencoder architectures as nonanomalous is lower for VAE-LSTMs in both the subprocesses; the corresponding results occur calculating the median of medians.
- 2) The mean of the means of the anomaly scores of samples classified by the autoencoder architectures as anomalous is higher for BiLSTM in the penetration subprocess. This can also be seen in the box-plot shown in Fig. 13. A higher mean of the distance between nonanomalous and anomalous samples is desirable, since it makes the two groups better distinguishable.

Based on the results, the following selection of architectures was made: for the two subprocesses, different architectures

<sup>5</sup>  $E_{nn}$  indicates the normalized anomaly score of samples that are assigned to the class nonanomalous; correspondingly,  $E_{na}$  indicates the normalized anomaly score of samples that are assigned to the class anomalous.



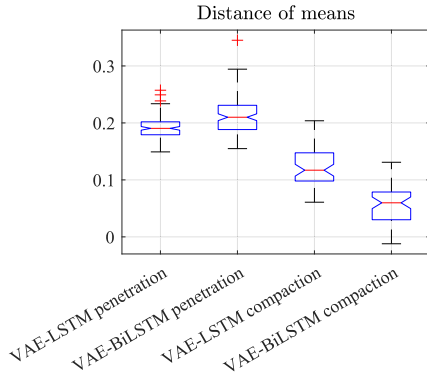


Fig. 13. Box-plot showing the distance of the means of the two classes; nonanomalous and anomalous; for the compared architectures.

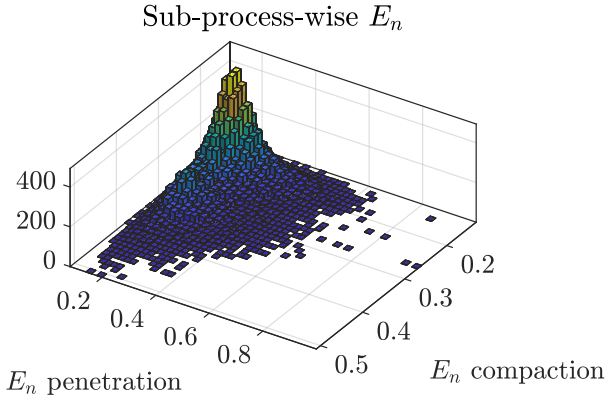


Fig. 14. Bivariate histogram of the anomaly score  $E_n$  for the two subprocesses gained by training a population of  $t = 125$  autoencoders with optimized hyperparameters for each of the subprocesses.

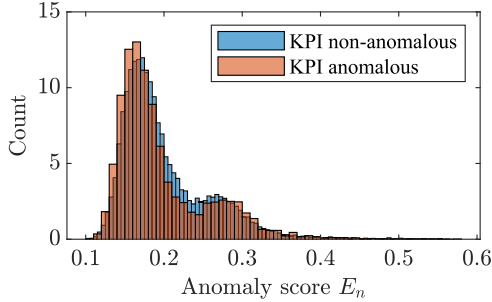


Fig. 15. Probability density function of the reconstructions of a trained VAE-BiLSTM architecture for the compaction phase; the two sets of MVTS were constructed using the KPI classifier.

were chosen; for penetration, autoencoders incorporating BiLSTM layers outperformed autoencoders using LSTM layers; for compaction, the architectures using LSTM layers performed significantly better.

### C. Subprocesswise Anomaly Detection

The bivariate histogram shown in Fig. 14 indicates that a high anomaly score in one of the two phases does not correlate with a high anomaly score in the other phase. A correlation coefficient  $r = -0.0628 \ll -0.3$  indicates a very weak negative linear relationship [56] between the anomaly scores of the two phases. This is consistent with the experts' institution of this case study; if the penetration phase is anomalous, this does not result in an anomalous compaction subprocess.

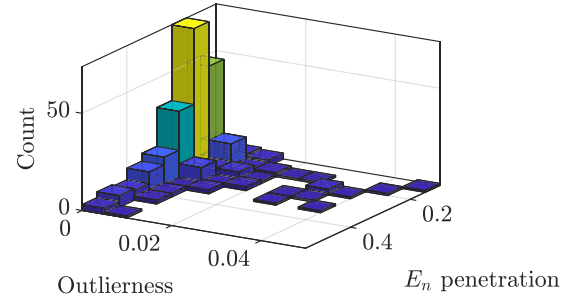


Fig. 16. Bivariate histogram of the anomaly score  $E_n$  for a trained VAE-BiLSTM and the KPI outlieriness for the penetration phase.

### D. Results KPI-Supervised Training

As described in Section V-C, the training sets were constructed using the outlieriness. However, when analyzing the distribution of the anomaly scores of both the groups, MVTS which are considered, based on the KPI analysis, to be anomalous and those considered to be no-anomalous, the distributions shown in Fig. 15 of the anomaly score were obtained. It can be seen that the outcome is two nearly identical probability density functions. Based on these results the KPI-supervised training needs further refinement to produce meaningful, distinctive results. This could be caused by the KPI covering all the sensor channels or by the fact that the KPI classifier and autoencoder anomaly detection cover other anomalies; see Fig. 16.

However, summed up it can be said that a high reconstruction error of an optimized and trained autoencoder does not lead to a high outlieriness and vice versa. This can be seen in Fig. 16, where the best performing architecture for compaction (VAE-BiLSTM) was trained with optimized hyperparameters to obtain the anomaly score and the KPIs covering the penetration phase were used to calculate the outlieriness. As already reported in [14], the two methods identify a common set of MVTS having both low anomaly scores and low outlieriness. Based on these, two means of anomaly detection two classifications can be derived and combined to cover a wider variety of anomalies; since the application shown is safety-relevant, every additionally detected anomaly is strengthening the process safety.

## VII. CONCLUSION

A new framework for the handling of MVTS data recorded in industrial applications was presented. The presented framework standardizes the data handling and makes the HML classifier applicable to a wide variety of use cases. Multiple technical KPIs that capture physical knowledge were developed. Furthermore, additional refinements of (Bi-)LSTM autoencoders were performed to make them better suitable for MVTS data with varying lengths. This permits the construction of ML architectures that preserve the real length of the MVTS after applying ML.

A new approach of unsupervised training set construction using the outcome of the KPI anomaly detection was investigated, which needs further refinement. A comparison of LSTM-VAE and BiLSTM-VAE was performed and led to using different recurrent layers for the present subprocesses of

the investigated case study. It can be concluded that training separate ML models for the subprocesses is beneficial since the anomaly scores of the two phases are not correlated and it also gives insights into which of the subprocesses the recorded data are anomalous.

Further work should be done in the direction of unsupervised training set construction and evaluation metrics for unsupervised ML architectures and training functions which are robust against outliers in the training set. The comparison shows that different architectures of different sizes are optimal for the different subprocesses. It should be noted that it is concluded after this study that due to the extensive knowledge discovery process and the derived preprocessing techniques, only a low number of not excluded anomalies is still present in the compaction phase. One indicator for that is the overall low number of detected anomalies and the overall low anomaly score of the subprocess.

## REFERENCES

- [1] C. Alippi, A. Ferrero, and V. Piuri, "Artificial intelligence for instruments and measurement applications," *IEEE Instrum. Meas. Mag.*, vol. 1, no. 2, pp. 9–17, Jun. 1998.
- [2] A. Morato, S. Vitturi, F. Tramarin, and A. Cenedese, "Assessment of different OPC UA industrial IoT solutions for distributed measurement applications," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (IMTC)*, May 2020, pp. 1–6.
- [3] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.*, vol. 378, pp. 686–707, Mar. 2019.
- [4] M. Khafer and S. Shirmohammadi, "Applied AI in instrumentation and measurement: The deep learning revolution," *IEEE Instrum. Meas. Mag.*, vol. 23, no. 6, pp. 10–17, Sep. 2020.
- [5] S. Shirmohammadi and H. Al Osman, "Machine learning in measurement Part 1: Error contribution and terminology confusion," *IEEE Instrum. Meas. Mag.*, vol. 24, no. 2, pp. 84–92, Apr. 2021.
- [6] H. Al Osman and S. Shirmohammadi, "Machine learning in measurement Part 2: Uncertainty quantification," *IEEE Instrum. Meas. Mag.*, vol. 24, no. 3, pp. 23–27, May 2021.
- [7] A. Rashed and S. Shirmohammadi, "A novel method to estimate measurement error in AI-assisted measurements," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (IMTC)*, May 2022, pp. 1–5.
- [8] O. Rippel, N. Schönfelder, K. Rahimi, J. Kurniadi, A. Herrmann, and D. Merhof, "Panoptic segmentation of animal fibers," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (IMTC)*, May 2022, pp. 1–6.
- [9] P. Assumpcao, C. Oliveira, W. Melo, and L. Carmo, "Sensors fingerprints using machine learning: A case study on dam monitoring systems," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (IMTC)*, May 2022, pp. 1–6.
- [10] P. R. Cohen and A. E. Howe, "How evaluation guides AI research: The message still counts more than the medium," *AI Mag.*, vol. 9, no. 4, p. 35, Dec. 1988.
- [11] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "The M4 competition: 100,000 time series and 61 forecasting methods," *Int. J. Forecasting*, vol. 36, no. 1, pp. 54–74, Jan. 2020.
- [12] R. Wu and E. Keogh, "Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress," *IEEE Trans. Knowl. Data Eng.*, early access, Sep. 14, 2021, doi: 10.1109/TKDE.2021.3112126.
- [13] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, "Do deep neural networks contribute to multivariate time series anomaly detection?" 2022, *arXiv:2204.01637*.
- [14] A. Terbuch, P. O'Leary, and P. Auer, "Hybrid machine learning for anomaly detection in industrial time-series measurement data," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf. (IMTC)*, May 2022, pp. 1–6.
- [15] P. Nagy and D. Adam, "Quality control of deep vibro compaction based on the vibrator movement," in *Proc. 17th Eur. Conf. Soil Mech. Geotechnical Eng.*, Reykjavik, Island, 2019, pp. 1–8.
- [16] R. S. Pugh, "Settlement of floor slabs on stone columns in very soft clays," *Proc. Inst. Civil Eng. Geotechnical Eng.*, vol. 170, no. 1, pp. 16–26, Feb. 2017.
- [17] K. Kirsch and F. Kirsch, *Ground Improvement by Deep Vibratory Methods*. England, U.K.: Taylor & Francis, 2017.
- [18] H. S. Mavikumbure, C. S. Wickramasinghe, D. L. Marino, V. Cobilean, and M. Manic, "Anomaly detection in critical-infrastructure using autoencoders: A survey," in *Proc. 48th Annu. Conf. IEEE Ind. Electron. Soc. (IECON)*, Oct. 2022, pp. 1–7.
- [19] M. Folk, A. Cheng, and K. Yates, "HDF5: A file format and I/O library for high performance computing applications," in *Proc. Supercomputing*, vol. 99, 1999, pp. 5–33.
- [20] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, 2008.
- [21] I. Steinwart, D. Hush, and C. Scovel, "A classification framework for anomaly detection," *J. Mach. Learn. Res.*, vol. 6, pp. 211–232, Jun. 2005.
- [22] L. Zhou, Q. Zeng, and B. Li, "Hybrid anomaly detection via multihead dynamic graph attention networks for multivariate time series," *IEEE Access*, vol. 10, pp. 40967–40978, 2022.
- [23] M. Munir, S. A. Siddiqui, M. A. Chattha, A. Dengel, and S. Ahmed, "FuseAD: Unsupervised anomaly detection in streaming sensors data by fusing statistical and deep learning models," *Sensors*, vol. 19, no. 11, p. 2451, May 2019.
- [24] Z. Ji, J. Gong, and J. Feng, "A novel deep learning approach for anomaly detection of time series data," *Sci. Program.*, vol. 2021, pp. 1–11, Jul. 2021.
- [25] Y. Zhao, X. Zhang, Z. Shang, and Z. Cao, "DA-LSTM-VAE: Dual-stage attention-based LSTM-VAE for KPI anomaly detection," *Entropy*, vol. 24, no. 11, p. 1613, Nov. 2022.
- [26] R.-C. Chen, C. Dewi, S.-W. Huang, and R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," *J. Big Data*, vol. 7, no. 1, pp. 1–26, Dec. 2020.
- [27] D. Ninevski and P. O'Leary, "A convolutional method for the detection of derivative discontinuities," in *Proc. 21st Int. Carpathian Control Conf. (ICCC)*, Oct. 2020, pp. 1–5.
- [28] D. Ninevski and P. O'Leary, "Detection of derivative discontinuities in observational data," in *Proc. Int. Symp. Intell. Data Anal.* Cham, Switzerland: Springer, 2020, pp. 366–378.
- [29] A. Terbuch, A. Zöhrer, V. Winter, P. O' Leary, N. Khalili-Motlagh-Kasmaei, and G. Steiner, "Quality monitoring in vibro ground improvement—A hybrid machine learning approach," *Geomechanics Tunnelling*, vol. 15, no. 5, pp. 658–664, Oct. 2022.
- [30] P. Baldi, "Autoencoders, unsupervised learning, and deep architectures," in *Proc. ICML Workshop Unsupervised Transf. Learn.*, I. Guyon, G. Dror, V. Lemaire, G. Taylor, and D. Silver, Eds. Bellevue, Washington, DC, USA, vol. 27, Jul. 2012, pp. 37–49. [Online]. Available: <https://proceedings.mlr.press/v27/baldi12a.html>
- [31] D. E. Rumelhart and J. L. McClelland, *Learning Internal Representations by Error Propagation* (A Bradford Book). Cambridge, MA, USA: MIT Press, 1987, pp. 318–362.
- [32] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, U.K.: MIT Press, 2016.
- [33] D. Park, Y. Hoshi, and C. C. Kemp, "A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder," *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 1544–1551, Jul. 2018.
- [34] T. Kieu, B. Yang, and C. S. Jensen, "Outlier detection for multidimensional time series using deep neural networks," in *Proc. 19th IEEE Int. Conf. Mobile Data Manage. (MDM)*, Jun. 2018, pp. 125–134.
- [35] T. Amarbayasgalan, V. H. Pham, N. Theera-Umpon, and K. H. Ryu, "Unsupervised anomaly detection approach for time-series in multi-domains using deep reconstruction error," *Symmetry*, vol. 12, no. 8, p. 1251, Jul. 2020.
- [36] J. F. Torres, D. Hadjout, A. Sebba, F. Martínez-Álvarez, and A. Troncoso, "Deep learning for time series forecasting: A survey," *Big Data*, vol. 9, no. 1, pp. 3–21, Feb. 2021.
- [37] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [38] F. U. M. Ullah, A. Ullah, I. U. Haq, S. Rho, and S. W. Baik, "Short-term prediction of residential power energy consumption via CNN and multi-layer bi-directional LSTM networks," *IEEE Access*, vol. 8, pp. 123369–123380, 2020.
- [39] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The performance of LSTM and BiLSTM in forecasting time series," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2019, pp. 3285–3292.

- [40] A. Garg, W. Zhang, J. Samaran, R. Savitha, and C.-S. Foo, "An evaluation of anomaly detection and diagnosis in multivariate time series," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 6, pp. 2508–2517, Jun. 2022.
- [41] Z. C. Lipton, D. C. Kale, C. Elkan, and R. C. Wetzell, "Learning to diagnose with LSTM recurrent neural networks," in *Proc. 4th Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds. San Juan, Puerto Rico, May 2016, pp. 1–18.
- [42] W. Zhang, Z. Lin, and X. Liu, "Short-term offshore wind power forecasting—A hybrid model based on discrete wavelet transform (DWT), seasonal autoregressive integrated moving average (SARIMA), and deep-learning-based long short-term memory (LSTM)," *Renew. Energy*, vol. 185, pp. 611–628, Feb. 2022.
- [43] S. Bouktif, A. Fiaz, A. Ouni, and M. A. Serhani, "Multi-sequence LSTM-RNN deep learning and metaheuristics for electric load forecasting," *Energies*, vol. 13, no. 2, p. 391, Jan. 2020.
- [44] M. Thill, W. Konen, H. Wang, and T. Bäck, "Temporal convolutional autoencoder for unsupervised anomaly detection in time series," *Appl. Soft Comput.*, vol. 112, Nov. 2021, Art. no. 107751.
- [45] M. Dwarampudi and N. V. S. Reddy, "Effects of padding on LSTMs and CNNs," 2019, *arXiv:1903.07288*.
- [46] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization Inverse Problems*. Amsterdam The Netherlands: Springer, 2000.
- [47] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2014. [Online]. Available: <http://arxiv.org/abs/1312.6114>
- [48] D. P. Kingma and M. Welling, *An Introduction to Variational Autoencoders*. New York, NY, USA: Now Publishers, 2019, pp. 1–13.
- [49] I. Higgins et al., "Beta-VAE: Learning basic visual concepts with a constrained variational framework," in *Proc. 5th Int. Conf. Learn. Represent. (ICLR)*, Toulon, France, Apr. 2017. [Online]. Available: <https://openreview.net/forum?id=Sy2fzU9gl>
- [50] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space Odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2016.
- [51] A. Terbuch, "LSTM hyperparameter optimization: Impact of the selection of hyperparameters on machine learning performance when applied to time series in physical systems," M.S. thesis, Dept. Product Eng., Chair Automat., Montanuniversitaet Leoben, Leoben, Austria, 2021.
- [52] M. Liu, Z. Xu, and Q. Xu, "DeepFIB: Self-imputation for time series anomaly detection," 2021, *arXiv:2112.06247*.
- [53] E. Vandervieren and M. Hubert, "An adjusted boxplot for skewed distributions," in *Proceedings in Computational Statistics*. Prague, Czech Republic: Springer-Verlag, 2004, pp. 1933–1940.
- [54] M. Hubert and S. Van Der Veken, "Outlier detection for skewed data," *J. Chemometrics*, vol. 22, nos. 3–4, pp. 235–246, 2008.
- [55] M. Hubert and E. Vandervieren, "An adjusted boxplot for skewed distributions," *Comput. Statist. Data Anal.*, vol. 52, no. 12, pp. 5186–5201, 2008.
- [56] B. Ratner, "The correlation coefficient: Its values range between +1/−1, or do they?" *J. Target. Meas. Anal. Mark.*, vol. 17, no. 2, pp. 139–142, Jun. 2009.



**Anika Terbuch** received the B.Sc. and M.Sc. degrees in industrial logistics from Montanuniversität Leoben, Leoben, Austria, in 2019 and 2021, respectively.

She is currently a University Assistant with the Chair of Automation, Montanuniversität Leoben, Leoben. Her research focuses on embedding physical a priori knowledge into hybrid machine learning architectures.

**Paul O'Leary** received the B.A. degree in mathematics and the B.A.I. degree in engineering from Trinity College Dublin, Dublin, Ireland, both in 1982, the M.Sc. degree in electronic engineering from PII, Eindhoven, The Netherlands, in 1984, and the Ph.D. degree from the University of Pavia, Pavia, Italy, in 1991.

He is currently a Professor of automation with the University of Leoben, Leoben, Austria. His research focuses on mathematical methods to extract information, knowledge, and understanding from data.



**Negin Khalili-Motlagh-Kasmaei** received the master's degree in urban engineering from Azad University, Zanjan, Iran, in 2015. She is currently pursuing the Ph.D. degree with the Chair of Automation, Montanuniversität Leoben, Leoben, Austria.

She is working as a Project Assistant at the Chair of automation, Montanuniversität Leoben. Her research focuses on multivariate time-series data analysis, anomaly detection, rule-based key performance indicators, and developing observational digital twins.



**Peter Auer** received the Ph.D. degree in mathematics from the Vienna University of Technology, Vienna, Austria, in 1992.

From 1988 to 1992, he was a Researcher at the Institute for Statistics and Probability Theory and the Institute for Applied Computer Science, Vienna University of Technology. From 1992 to 2002, he was an Assistant Professor and later an Associate Professor at the Institute for Theoretical Computer Science, Graz University of Technology, Graz, Austria. From 1995 to 1996, he was a Visiting Research

Scholar at the University of California at Santa Cruz, Santa Cruz, CA, USA. Since 2003, he has been a Full Professor of information technology with the Montanuniversität Leoben, Leoben, Austria.

Dr. Auer is a member of the ACM. He is an Action Editor of the *Journal of Machine Learning Research* and an Editorial Board Member of the *Machine Learning Journal*.



**Alexander Zöhrer** received the master's degree in civil engineering and the Ph.D. degree from the Graz University of Technology, Graz, Austria, in 2003 and 2006, respectively.

In 2007, he joined Company Keller Grundbau Ges.m.b.H., Vienna, Austria, as a Design Engineer. Since 2015, he has been the Head of the Engineering Department of the Business Unit South-East Europe and Nordics. In this position he is responsible for any research and development activities as well as the engineering support including design calculations

for all construction sites.

Dr. Zöhrer is a member of the EFFC Sustainability Working Group as well as a member of various global working groups at Keller.



**Vincent Winter** received the M.Sc. degree in civil engineering from the Vienna University of Technology, Vienna, Austria, in 2002.

Since 2000, he has been working in various positions with Company Keller Grundbau Ges.m.b.H., Vienna, starting as a site manager and later on as a project manager for sites of all special foundation works. Currently, he is a Product Line Manager for Vibro Techniques for Keller South East Europe and Nordics, supporting big and difficult projects and being responsible for the development of the Vibro

Techniques at Keller. As an expert for vibro ground improvement techniques, he was involved in many national as well as a number of international projects.