Proceedings of the 8th European Conference of the Prognostics and Health Management Society 2024 - ISBN – 978-1-936263-40-0

Process Quality Monitoring Through a LSTM Network Derived from a Rule-Based Approach

Andreas Bernroitner[1] and Roland Eckerstorfer[2]

[1,2] *Plasser und Theurer, Linz, 4020, Austria*
andreas.bernroitner@plassertheurer.com
roland.eckerstorfer@plassertheurer.com

Abstract

The railway infrastructure condition is a crucial factor for the safe and efficient operation of trains. Regular maintenance is inevitable as the track geometry degrades over time due to traffic and environmental effects. To restore the ideal position and provide sufficient durability of ballasted track so called tamping machines are used. These machines lift the track, correct the longitudinal level and the alignment of the track panel and tamp the ballast. During the tamping process the tamping tines penetrate the ballast bed, fill voids and compact the ballast underneath the sleepers by a squeezing movement with superimposed vibration. A detailed description of the tamping cycle can be found on section 2. Monitoring and evaluating this tamping process is essential for maintaining process quality. This can be achieved through a variety of sensors, such as incremental encoders, angle encoders, temperature, pressure, and acceleration sensors, coupled with a measurement unit (DAQ and edge device) to collect, locally store and transmit the data to a cloud. This paper explores the development of a rule-based algorithm for assessing the quality of the tamping process execution in reference to its nominal chronological sequence. The focus is on identifying tamping occurrences and classifying them into acceptable (OK) or non-acceptable (NOK) categories. This involves selecting relevant measurement parameters and processing them, considering the inherent imprecision in real-world processes. Empirical thresholds are established to differentiate between good and bad outcomes. The classification approach has to be sufficiently generic in order to cover a high variety of customized tamping machine types. As each machine is individually designed, the process of generalization is challenging and complex. The paper demonstrates the accuracy and universal applicability of the developed rule set across different tamping machines. The model's effectiveness is validated using the Hold-Out-Test-Set method. Furthermore, the rule-set-achieved outcomes are compared with results gained from an LSTM network. Both the rule-based approach and the neural network demonstrate precision, but the latter requires significantly more effort.

1. Track Maintenance

For a safe and efficient operation of trains a proper track infrastructure is indispensable. Especially on high-speed rail links the quality of the track and its surroundings is crucial. Therefore not only the construction but also the maintenance of the track in order to prevent degradation due to traffic and environmental effects are important. This involves ensuring a clean and dry embedding, sufficient proper ballast underneath the rails, impeccable condition of the sleepers involved and restoring the vertical as well as the horizontal position of the rails. A very detailed description of several track maintenance methods can be found in (Hansmann, 2021).

In Figure 1 an acceptable condition of a track is depicted. Here a sufficient amount of appropriately sized, clean ballast is in place. The positioning of the rails in vertical and horizontal direction is within applicable limits. The durability of the track geometry is ensured through appropriate compaction of the ballast.

Figure 1. Acceptable condition

For the latter tasks so-called tamping machines are used. The process which results in appropriately compacted ballast is referred to as tamping. In Figure 2 an example for such a tamping machine can be seen.

Figure 2. Tamping machine

A crucial part of such working machines is the tamping unit which is visualized in Figure 3. The lower grey colored components are called the tamping tines. They constitute the only components which are in direct contact with the ballast.

Figure 3. Tamping unit

Ultimately, the focus of this paper is the automatized identification of tamping cycles. Subsequently also classifying tamping cycles into acceptable and non-acceptable cycles, hereinafter denoted as OK and NOK respectively, will be done.

In Figure 4 a track with an unacceptable positional deviation can be seen. The ballast condition regarding size, homogeneity and cleanness does not fulfil the minimum criteria either.

Figure 4. Unacceptable condition of the track

In Figure 5 it is obvious that the ballast is in an unacceptable condition. Neither the ballast size nor the cleanness meet the desired conditions. (Soleimanmeigouni I, Ahmadi A, Kumar U., 2018) provide a summary, discussion and classification of existing track geometry measures and track geometry degradation models. Machine learning approaches for diagnosis and prognosis of rail defects are reviewed by (Chenariyan Nakhaee, Hiemstra, Stoelinga, & van Noort, 2019).

Figure 5. Unacceptable condition of the ballast

2. Tamping Process

The main goal of tamping is to correct track faults in longitudinal level and alignment in order to guarantee the operating reliability and ride comfort of the trains. The explanations given in this section are based on (Offenbacher, Koczwara, Landgraf, & Marschnig, 2023) and (Fellinger, 2017). Furthermore, ballast faults like voids beneath the sleepers should be corrected so that the load onto the sleepers is equally distributed and deployed to the underfloor. This also increases track quality before irreversible damages can occur.

A complete tamping cycle can be decomposed into the following sub-processes:

1. Positioning
2. Lifting and Lining
3. Penetrating
4. Filling
5. Compacting
6. Lifting

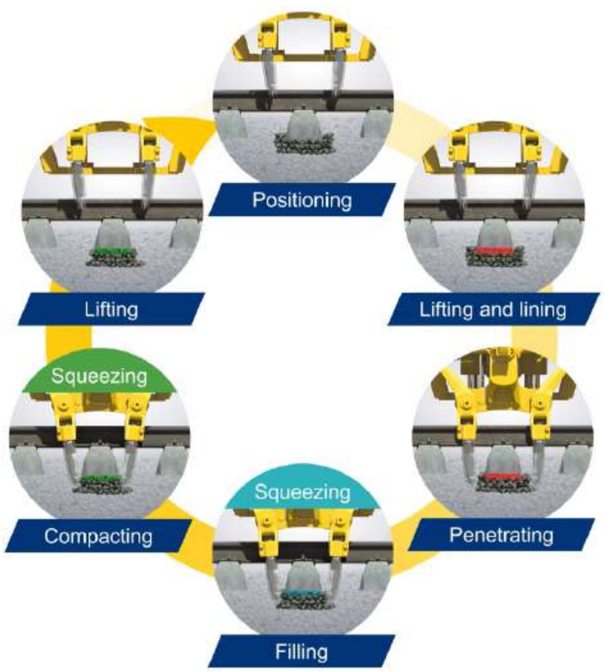

Figure 6. Tamping stages

In the Positioning phase it has to be ensured that the tamping unit is positioned exactly above a sleeper and there is no relative velocity between the unit and the track. Subsequently, in the Lifting and Lining phase the rails are correctly positioned by a separate working unit. Here the rails are lifted and brought into the desired longitudinal and lateral position. Then Penetrating is done and the whole tamping unit is lowered until the tamping tines sink into the surface and the lower position is reached. This is followed by the squeezing movement of the tines which basically comprises two sub processes, Filling (the void caused by the previous Lifting and Lining with ballast) and Compacting (the ballast under the sleeper). Finally, rails are released and the tamping unit is retracted again. This process is known as Lifting. During all of the stages the vibration has to be active. Thus the tamping tines are oscillating with 35 Hz for a smoother penetration and squeezing movement inside the ballast bed (Fischer, 1983).

3. Data Generation/Measurement System

A variety of sensors, such as incremental encoder, angle encoders, temperature, pressure, and acceleration sensors are connected with the control system. An Industrial Internet of Things (IIoT) edge device is fully integrated with the machine control system via the machine network. The device collects and records the data which is transferred to an online platform by means of a mobile broadband connection.

4. Tamping Abstraction

Unfortunately, there are not one-to-one relationships between the recorded measurement signals and the sub-processes as described in Section 2. Additionally, there are further conditions to be fulfilled to assess the quality of the tamping-process, e.g., squeezing (consisting of filling and compacting) shall only be performed when the tamping unit already rests in the down position and not during penetration. On the other hand, it is not relevant to distinguish filling from compacting, but only the process of squeezing and related key parameters as squeezing times are of interest. The sub-process “Positioning” can only be identified by means of the vehicle’s speed, in detail, whether the machine is at standstill or not, but it cannot be checked if it is positioned properly. There are separate assisting tools which deal with proper positioning. For example, there is a camera and image recognition system that makes suggestions to the operator for adjusting the tamping units properly, especially in turnouts. The operator only needs to confirm the suggestions (Plasser und Theurer, 2017). Concluding, the sub-processes as depicted in figure 6 need be represented by sequences based on and created by real signal data. Therefore, the signals are transformed into segments of Boolean representations by means of applying mathematical operations and threshold values, if required. The correct sequence of serial and parallel segments determines the quality or correctness of the tamping process. The proper sequence of segments for an acceptable tamping process is depicted in figure 7.

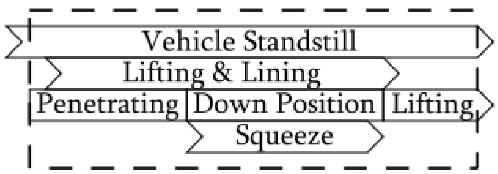

Figure 7. Tamping process based on measurement data

However, real data show deviations from theory, like the overlaps of segments or short unidentified periods between segments (i.e., pauses) which should be in series. Furthermore, differing start or end times of segments which should be synchronous may occur. The inaccuracies are caused by different sampling rates of the individual signals or temporal shifts induced by mathematical operations. Thus, there are parts in the sequences which do not follow the strict theoretical rules but can be considered as valid to a certain extent. The imprecision necessitates the definition of further rules to qualify a tamping process.

5. Long-Short-Term-Memory

The Long Short-Term Memory (LSTM) network was invented by Sepp Hochreiter and Jürgen Schmidhuber in 1997 (Hochreiter & Schmidhuber, 1997). Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) architecture used in the field of deep learning. LSTMs are designed to avoid the long-term dependency problem typical of standard RNNs, enabling them to remember information for long periods. This makes LSTMs particularly useful for tasks involving sequential data, such as time series analysis, natural language processing (NLP), speech recognition, and more. The key to LSTM's ability to retain long-term memory is its cell state, along with its various gates that control the flow of information. An LSTM unit typically comprises the following components:

- Forget Gate f_t
- Input Gate i_t
- Cell State c_t
- Output Gate o_t

The Forget Gate decides what information should be thrown away or kept. It looks at the current input and the previous hidden state and outputs numbers between 0 and 1 for each number in the cell state (C_{t-1}). A value close to 1 means to keep the information, while close to 0 means to forget it. The Input Gate decides what new information will be stored in the cell state. It involves two parts: one Sigmoid layer that decides which values to update, and a Tanh layer that creates a vector of new candidate values that could be added to the state. The cell state is the key innovation of LSTMs. It runs straight down the entire chain, with only minor linear interactions. It's very easy for information to just flow along it unchanged. The cell state is modified by the forget gate and the input gate. The Output Gate determines the next hidden state, which contains information on previous inputs. The hidden state can be used to make predictions. The output gate looks at the current input, the previous hidden state, and the current cell state, and decides what the output should be. These components work together to allow the LSTM to decide when to allow data to enter, when to forget data because it's no longer useful, and when to let it impact the output at the current timestep. This selective memory capability helps LSTMs to perform exceptionally well on tasks where the context or the sequence of data points is important.

A Bidirectional Long Short-Term Memory (Bi-LSTM) network is an extension of the traditional Long Short-Term Memory (LSTM) network. It enhances the original LSTM by providing two layers that process the input sequence in both forward and backward directions. By processing sequences in both directions, Bi-LSTMs can capture context from both the past and the future relative to a specific point in the sequence. The key idea behind a Bi-LSTM is that at any point in time, the network has access to information from both the beginning and the end of the sequence, making it especially powerful for tasks where context from both directions is crucial for understanding or predicting the elements of the sequence. Mathematically, a Bi-LSTM combines the outputs from two separate LSTM layers — one processing the input sequence from start to end and the other processing it from end to start. The outputs of these two LSTMs can be merged in various ways (e.g., concatenation, summation, or averaging) to form a single output that provides a comprehensive context-aware representation of each point in the sequence. Bi-LSTMs are widely used in various sequence modeling tasks, such as natural language processing for named entity recognition, sentiment analysis, and machine translation, as well as in bioinformatics and speech recognition, where understanding the context from both directions can significantly enhance model performance.

In Figure 8 a typical (vanilla-)LSTM is depicted. In the graph the σ stands for the Sigmoid activation and the $\tanh$ for Hyperbolic Tangent activation function. g_t represents the input activation and the x an element wise multiplication.

(De Simone et al., 2023) describe the application of a LSTM model for the failure prediction of rolling stock equipment, in detail of the traction converter cooling system, but also give a rough overview on other LSTM-based prediction algorithms in the railway industry.

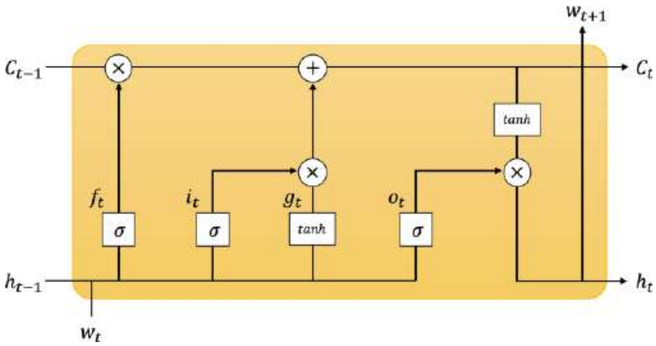

Figure 8. LSTM workflow
(Park & Kim, 2020)

a	b	c	d	e	f	g	h	i
1	1	1	1	0	0	0	0	1
1	1	1	1	0	0	0	0	1
1	1	1	1	1	1	0	0	0
1	1	1	1	1	1	0	0	0
1	1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	0
1	0	1	1	1	1	1	0	0
1	0	1	1	1	1	1	0	0
1	0	1	1	1	1	1	0	0
1	0	1	1	1	1	0	0	0
1	0	1	1	1	1	0	0	0
1	0	1	1	1	1	0	0	0
1	0	1	1	1	0	0	0	1
1	0	1	1	1	0	0	0	1
1	0	1	1	1	0	0	0	1

Figure 9. Tamping identification

6. Rule Based Detection

6.1. OK Tamping cycle detection

In order to identify and evaluate tamping cycles based on the time series of the measurement channels, boolean signals are created and assembled, as depicted in Figure 9. A sampling rate of 10 Hz was chosen in order to ensure an appropriate resolution of the signals. Each row of the visualization in Figure 9 is a time increment. Column **a** stands for "engine running", which means that the engine of the tamping machine has to be switched on and a minimum rotational speed has to be exceeded. The second column **b** represents the tamping cycle initialization which is done by the operator by means of a foot-operated pedal. In the **c** column it is listed whether the superimposed vibration of the tamping tines is activated or not. **d** indicates the proper lifting and lining of the rails. In **e** one can see if the tamping unit's relative velocity falls below a very low threshold value with respect to the rails. This means that "the tamping unit stands still" or it is in a very slow movement at least. The penetrating phase is described in column **f** via checking the downward movement of the tamping units, in detail, it is true if it moves and false if not. Column **g** shows if the tines are in the desired lower position. This is again realized by applying a threshold value to the tamping unit's positional encoder. In column **h** the squeezing movement is depicted. It is true if the tamping tines are moved towards each other to fill and compact the ballast under the sleeper and false else. In the last column **i** the retraction, the lifting of the unit, is depicted. For the consideration of measurement and transmission errors small deviations are tolerated. This means that also segments which are disconnected by only one or two time increments are regarded as one full coherent segment.

The following criteria are established for the tamping cycle identification and classification:

- duration of each individual segment
- simultaneity of segments
- duration of sections with overlapping segments which should not be simultaneous
- serial sequence of segments or detachment of consecutive signals
- duration between consecutive segments

The definition of permissible durations, serial sequences, concurrences etc. requires both profound domain knowledge about the tamping cycle and empirical insights based on real data. For example, the ideal minimum squeezing time, i.e., the duration from start of the filling phase until the end of compaction phase, is defined as about 1.2 seconds by a manufacturer of tamping machines. However, there can be national regulations which specify a deviating squeezing duration. Another example can be the temporal succession of the lifting and penetration phases. Ideally, these two sequences are strictly in series. However, the downward movement of the tamping unit can already start when the lifting of the rail is still in progress provided that a void has formed as soon as the tamping tines enter the ballast. Concluding, a certain duration of parallelism is permissible in this case. Furthermore, it is also acceptable that there is a short pause between the segments. Thus, the definition of such thresholds and tolerances requires experience and sensitivity from the engineers and data analysts. Usually the threshold values are determined empirically or are defined be national regulations depending on where the machine is operated.

In order to get an intuitive feeling about the identification process several consecutive tamping cycles are depicted in Figure 10, where blue sections represent boolean true and orange, boolean false. The columns are identical to those in Figure 9.

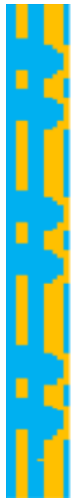

Figure 10. Tamping identification of multiple cycles

6.2. NOK Tamping cycle detection

In order to obtain the desired process quality all sub-processes have to have the appropriate duration and also the sequence of the consecutive sub-processes has to satisfy the correct order, which means that the subsequent signal has to follow the previous one within a certain amount of time. Therefore the following error scenarios can occur:

- vibration off
- incomplete vibration
- relative velocity
- no stand still
- incomplete stand still
- no leveling
- incomplete leveling
- no penetration
- no down position
- incomplete down position
- no squeezing movement
- penetration before lifting the rails
- squeezing before down position
- lifting the tamping unit before squeezing

The time-series signals of an example of detected NOK tamping cycles are depicted in Figure 11 - a better quality of the plot can be found in the Appendix 9, too - where the upper graph shows the lowering position of a tamping unit. Negative values indicate positions of the tamping tines above the rail, zero is approximately the level of the rail's surface and positions greater than 120 mm can be considered as the tines entering the ballast. The lower graph illustrates the machine's velocity in m/h. Based on the developed cycle identification an impermissible overlap of the two sections "vehicle stands still" and "tamping tines are in the ballast" could be detected. The overlap is highlighted in red colour in Figure 11. These overlaps indicate that the tines are already located in the ballast even though the machine is still moving can cause significant wear on or even severe damage of the tamping unit.

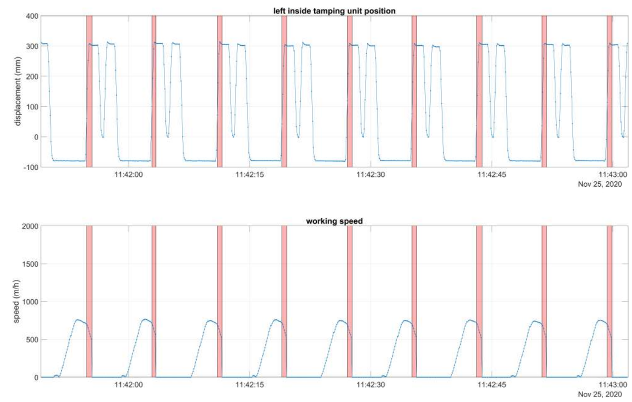

Figure 11. Identified NOK tamping cycle: The tamping unit is already in the ballast even though the machine is still moving (see also Appendix)

7. LSTM Detection

7.1. Architecture

After several trials regarding the structure of the network, the architecture depicted in 9 was chosen.

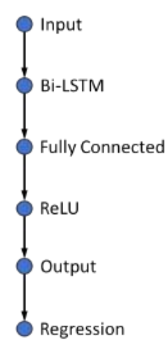

Figure 12. LSTM architecture

The network consists of:

- Input Layer
- Bi-LSTM Layer
- Fully Connected Layer
- ReLU Activation
- Output Layer
- Regression Output

The input layer is the bottom-most layer, where the input, the previously generated boolean signals, is fed to the network.

Subsequently data is passed to the Bi-LSTM layer which consists of 25 hidden units. Bidirectional LSTMs can be useful when the context of the input is needed from both the past and the future of a specific time step. This turned out to be the case in the cycle identification. Following the Bi-LSTM layer, there is a fully connected layer which takes the sequential output from the Bi-LSTM and transforms it into a fixed-size vector. This layer has 10 units, and it is likely responsible for integrating the features learned by the Bi-LSTM layer. After that a non-linearity in form of a ReLU activation function is applied. Therewith the model is allowed to account for non-linear relationships between the features. The next layer is another fully connected output layer with a single unit. This is because the network is designed to output a single continuous regression value. The final layer is a regression output layer with the mean squared error as loss function.

7.2. Training

The analysis workflow was implemented in Matlab and it turned out that training for only 10 epochs with 225 iterations each is sufficient. For the training the timeseries were split into windows of 10 seconds each and a step size of 5 seconds was chosen. Therefore an overlap of 50% occurred intentionally. The training was done with a learning rate of 0.001 on a single GPU. and the training in total only took a couple of minutes. The metric used in training was RMSE (root mean square error).

7.3. Results

In Figure 13 the cycle detection can be seen. This graph can also be found in the appendix. In this visualization the gray rectangles represent the tamping cycles and also their duration. A nearly perfect fit can be found here. There is no visible deviation between the LSTM and the rule based results.

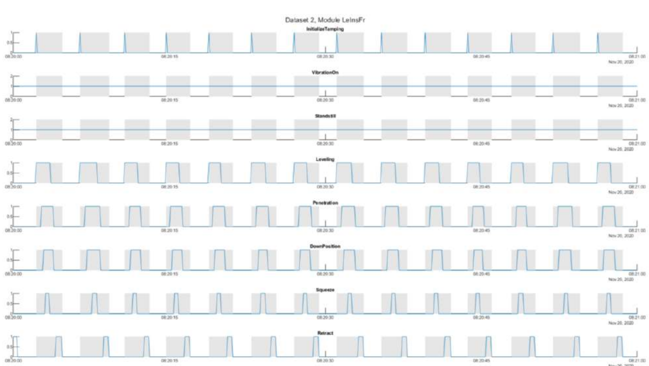

Figure 13. LSTM tamping cycle detection (see also Appendix)

Using the hold-out test set method, an accuracy of 0.98 was achieved.

8. Conclusions

The comparison of the two tested methods for tamping cycle identification, i.e. the rule-based vs. the LSTM approach, it can be concluded that:

1. The accuracy of the rule-based method is approx. 100%, whereas that of the LSTM model is approx. 98% taking only OK detections into consideration. Obviously, the rule-based approach, which basically consists of a set of subsequent if-queries, delivers better results due to the fact that the rules exactly represent the definition of a correct tamping cycle. But the exact representation requires profound domain knowledge of and experience on the tamping procedure and the data acquisition process. When lacking this knowledge and experience the neural network, which defines its own rules by adjusting its learnable parameters, the weights and biases, by evaluating the time series over and over, turns out to be a suitable alternative to still get very accurate results.

2. The implementation effort for the LSTM model is much higher as well as the required hardware and processing resources for training and evaluating the network.

3. The pre-processing of the data and the generation of the boolean sub-processes is the same for both methods.

4. The identification of the NOK tamping cycles is more difficult for the LSTM approach due to the lack of sufficient amount of NOK cycles in real world training data because operating errors rarely occur. A possible solution would be to artificially generate error cases in order to allow the model learn incomplete sequences.

9. Further steps

The LSTM approach as described is capable of identifying OK-cycles. However the NOK-cycles are of higher interest with regards of wear and resulting maintenance. However these cases do not occur sufficiently frequent in real world data. Therefore artificial samples could be generated and be fed to the training set. Another approach could be weighing the very rarely occurring failure cycles higher than the frequently occurring OK cycles in order to balance the training set. Furthermore it should be checked if the found algorithm is generic enough to also fit to other machines and surroundings. Thus it shall be enrolled to different machines operating in different regions of the world in order to compare results and performance subsequently. On the other side also other algorithms shall be implemented and compared. Therefore the time series should again be split into small segments e.g. 0.1s and each of these segments should be classified by different machine learning algorithms according to the features within the respective segment.

REFERENCES

Chenariyan Nakhaee, M., Hiemstra, D., Stoelinga, M., & van Noort, M. (2019). The recent applications of machine learning in rail track maintenance: A survey. In S. Collart-Dutilleul, T. Lecomte, & A. Romanovsky (Eds.), *Reliability, safety, and security of railway systems. modelling, analysis, verification, and certification* (pp. 91–105). Cham: Springer International Publishing.

De Simone, L., Caputo, E., Cinque, M., Galli, A., Moscato, V., Russo, S., ... Giannini, G. (2023). Lstm-based failure prediction for railway rolling stock equipment. *Expert Systems with Applications*, *222*, 119767.

Fellinger, M. (2017). Validierung der instandsetzungsmengen der standardelemente gleis der öbb. *FSV aktuell Schiene*.

Fischer, J. (1983). *Einfluss von frequenz und amplitude auf die stabilisierung von oberbauschotter* (PhD thesis). Technische Universität Graz.

Hansmann, S., Nemetz. (2021). Keeping track of track geometry.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*.

Offenbacher, S., Koczwara, C., Landgraf, M., & Marschnig, S. (2023). A methodology linking tamping processes and railway track behaviour. *Applied Sciences*.

Park, S., & Kim, Y. (2020). A method for sharing cell state for lstm-based language model. *Computer and Information Science*, 81–94.

Plasser und Theurer. (2017). The track maintenance machine 2020. *Plasser und Theurer today*, *46*(132), 24–26.

Soleimanmeigouni I, Ahmadi A, Kumar U. (2018). Track geometry degradation and maintenance modelling: A review. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit.*, *232(1)*, 73-102.

BIOGRAPHIES

Andreas Bernroitner obtained his Master of Science (M.Sc) degree in Mechanical Engineering from the University of Applied Science in 2014, following the completion of his Bachelor of Science (B.Sc) at the same institution. He is presently pursuing a second M.Sc degree in Artificial Intelligence at Johannes Kepler University, Linz. With professional experience as a failure analysis engineer, Bernroitner has been serving as a data scientist within the *Data Science and Analytics* team at *Plasser und Theurer*, Linz, since 2022.

Roland Eckerstorfer graduated in Medical Device Technology on the University of Applied Sciences Upper Austria in Linz in 2009. After several years as measurement engineer in research and development of special purpose machines performing measurements and subsequently the data evaluation and analysis in power plants worldwide and on test rigs, he now is a member of the *Data Science and Analytics* team at *Plasser und Theurer* in Linz since 2021.

APPENDIX

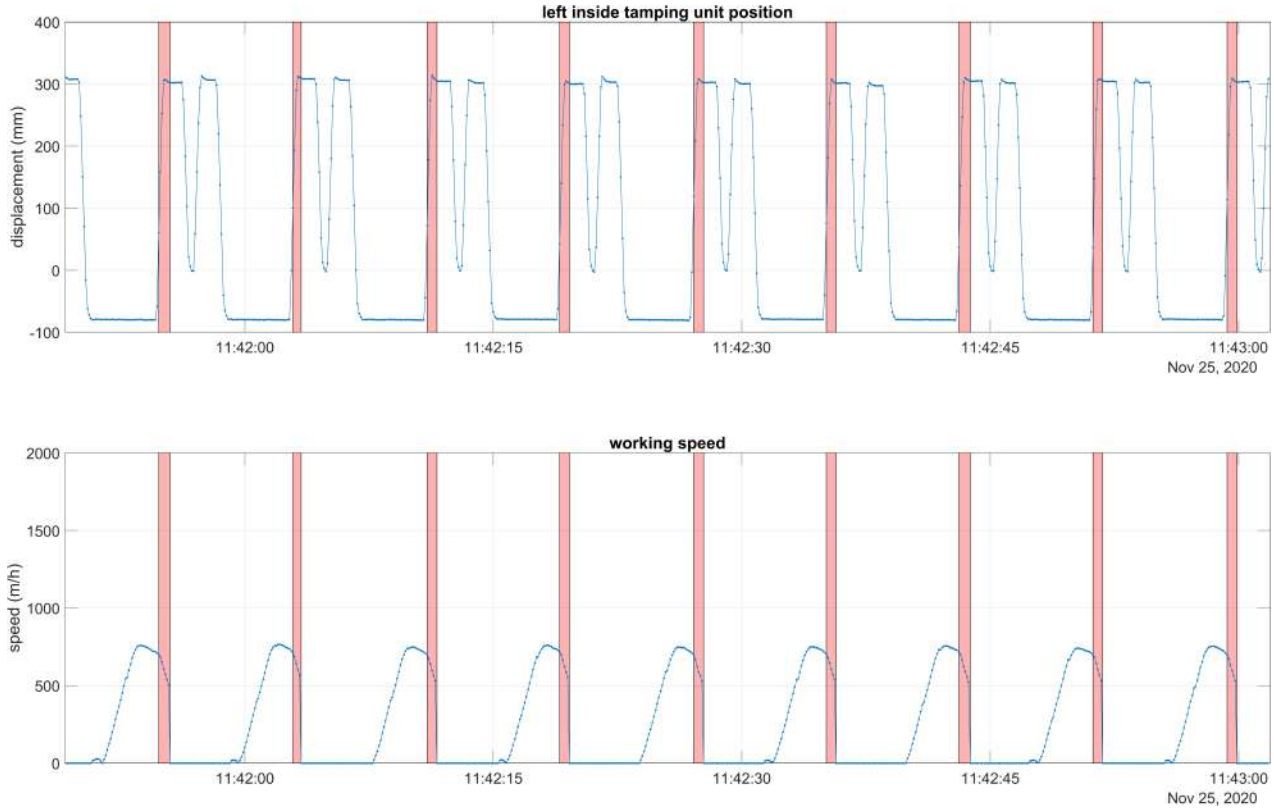

Figure 11. Identified NOK tamping cycle: The tamping unit is already in the ballast even though the machine is still moving

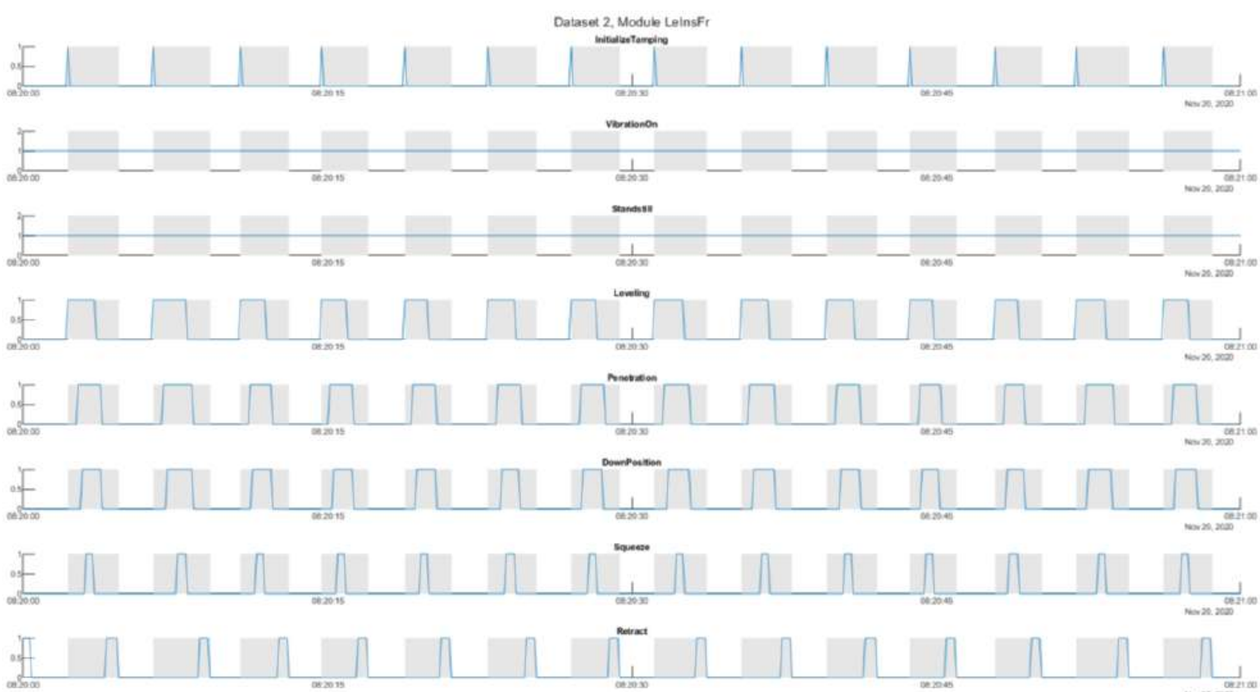

Figure 13. LSTM tamping cycle detection