



An end-to-end machine learning approach with explanation for time series with varying lengths

Manuel Schneider¹ · Norbert Greifzu^{1,2} · Lei Wang³ · Christian Walther⁴ · Andreas Wenzel^{1,2} · Pu Li⁵

Received: 24 June 2023 / Accepted: 14 January 2024 / Published online: 19 February 2024
© The Author(s) 2024

Abstract

An accurate prediction of complex product quality parameters from process time series by an end-to-end learning approach remains a significant challenge in machine learning. A special difficulty is the application of industrial batch process data because many batch processes generate variable length time series. In the industrial application of such methods, explainability is often desired. In this study, a 1D convolutional neural network (CNN) algorithm with a masking layer is proposed to solve the problem for time series of variable length. In addition, a novel combination of 1D CNN and class activation mapping (CAM) technique is part of this study to better understand the model results and highlight some regions of interest in the time series. As a comparative state-of-the-art unsupervised machine learning method, the One-Nearest Neighbours (1NN) algorithm combined with dynamic time warping (DTW) was used. Both methods are investigated as end-to-end learning methods with balanced and unbalanced class distributions and with scaled and unscaled input data, respectively. The FastDTW and DTAIDistance algorithms were investigated for the DTW calculation. The data set is made up of sensor signals that was collected during the production of plastic parts. The objective was to predict a quality parameter of plastic parts during production. For this research, the quality parameter will be a difficult or only destructively measurable parameter and both methods will be investigated for their applicability to this prediction task. The application of the proposed approach to an industrial facility for producing plastic products shows a prediction accuracy of 83.7%. It can improve the reference method by approximately 1.4%. In addition to the slight increase in accuracy, the CNN training time was significantly reduced compared to the DTW calculation.

Keywords Time series classification (TSC) varying length · CNN with masking-layer · CAM for TSC · Quality prediction · Injection moulding

✉ Manuel Schneider
m.schneider@hs-sm.de

✉ Norbert Greifzu
norbert.greifzu@iosb-ast.fraunhofer.de

¹ Faculty Electrical Engineering, Embedded Diagnostics Systems, Schmalkalden University of applied sciences, Blechhammer 6, 98574 Schmalkalden, Thuringia, Germany

² Fraunhofer Institute of Optronics, System Technologies and Image Exploitation, IOSB-AST Ilmenau, Fraunhofer IOSB, Am Vogelherd 90, 98693 Ilmenau, Germany

³ Department Intelligent Energy Systems, Fraunhofer Institute for Integrated Systems and Device Technology IISB, Fraunhofer IISB, Schottkystraße 10, 91058 Erlangen, Bavaria, Germany

⁴ Bauhaus-Universität Weimar, Institute for Structural Mechanics, Marienstraße 15, 99423 Weimar, Thuringia, Germany

⁵ Department of Computer Science and Automation, Technische Universität Ilmenau, Ehrenbergstraße 29, 98684 Ilmenau, Thuringia, Germany

1 Introduction

Artificial intelligence (AI) has been widely applied in the industry over the last two decades [1, 2]. Machine learning (ML), including deep learning (DL), as a subfield of AI, is becoming increasingly important for research in image and signal processing, as well as for industrial applications [3, 4].

In [5], the focus was on the detection of defects in plastic parts with a convolutional neural network (CNN) through image analysis in combination with edge computing and Internet of Things (IoT) systems. Some challenges of quality prediction in the context of big data in the field of Industry 4.0 were mentioned in [6].

In this paper, our focus is on time series with a high sampling rate and different lengths, but a small number of records. In this context, the topic of TSC has gained importance as a key component of signal processing from ML.

In [7] a new approach for adaptive multi-scale pooling and the use of temporal encoding was used to improve classification accuracy, especially on short data obtained as partial time series.

Dempster et al. [8] described the use of the HYbrid Dictionary-ROCKET Architecture (HydRa) model and compared it with the RandOm Convolutional KErnel Transform (ROCKET) algorithm for extracting and counting symbolic patterns in time series. It is shown that ROCKET is an outstanding algorithm capable of classifying time series accurately and quickly [9].

In [10], a deep neural network for TSC was used, which was the most exhaustive study of DNNs for TSC. A total of 8730 deep learning models have been trained on 97 time series data sets. Another method is the shepard interpolation neural networks (SINN) model which provides a shallow learning approach with minimal use of training samples. The SINN networks are more interpret-able than other neural networks. It learns metric features of the trained time series. In [11] a leverage novel SINN architecture was bench-marked to other TSC algorithms. However, the choice of ML models and algorithms for time series classification is limited by classification accuracy, computational time and the ability of interpretation.

Batch production processes are extremely popular in the process industry. A distinct property of such processes is that the process data are time series with different lengths by different batch operations. The analysis of non-periodic time series from batch production processes is the aim of this study. Such signals are often measured in discontinuous production processes, e.g. injection moulding of plastic parts. The product quality of a batch process will be influenced by the operating parameters.

The first step in obtaining information about the production process is data acquisition. It consists of process parameters, quality values, or sensor signals. Figure 1 illustrates the steps to build a prediction model using ML. The upper part shows the steps mentioned in [12]. The lower part of Fig. 1 describes the proposed approach in this study. The data are accumulated by sensor signals from inside the mould cavity during the production of plastic parts. A brief description of the process is given in Sect. 3.1. The next step of the upper part in Fig. 1 would be the task of feature extraction. Feature extraction can be much more difficult than building the ML model and in some cases requires more expertise on the production process. The advantage of the proposed end-to-end method consists of the simultaneous computation of the features, prediction of the product quality, and the visualization by the CAM algorithm to validate the model decision. This requires much less effort for feature extraction and helps to get clues where the ML model makes the decision of the classification problem.

Another important aspect is to the computation time taken for training the model. One way is the use of batch computation. According to [13], when training neural networks using batch normalization (BN), the batch size is an important hyper-parameter to speed up the training time. To optimize the training processes, a batch size greater than one and parallel computing will be the best solution. [14] used some regularization techniques with a batch size greater than one. In general, a convolutional neural network (CNN) with a batch size greater than one cannot be trained with time series of different lengths. A solution without using CNN could be realized with 1NN with DTW. This method is one of the most popular and traditional algorithms for time series classification [15–17]. In addition, in [18] the DTW was described for varying time series. Bagnall and Lines [19] has stated that it is very difficult to outperform the 1NN-DTW combination.

In this work, we compare this method with a proposed 1D CNN using a masking layer for TSC for varying lengths. Both methods are investigated without feature extraction steps. As an application case for verifying the proposed ML model, discontinuous time series from a plastics production process are used in this study.

In [20, 21] the focus was on TSC and comprehensible quality prediction in the plastic injection moulding process. Monitoring the quality of plastic injection moulded parts is often difficult and expensive. With rising energy costs, quality prediction using ML algorithms has become a new focus for more and more manufacturers. Finding the right process parameters is a time-consuming task. To reduce this time and to find suitable process parameters e.g. pressure, temperature, and injection speed, ML methods were used in [1, 4].

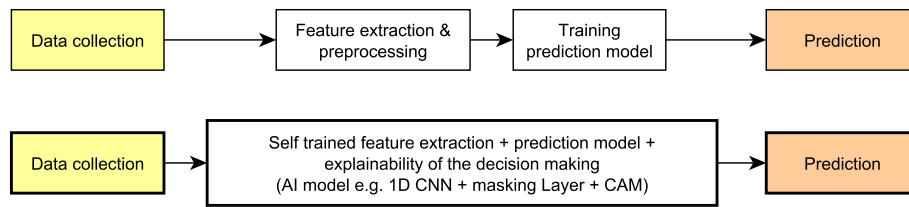


Fig. 1 Steps to design a prediction model using machine learning. The upper path shows the steps from [12]. The path below shows the steps proposed in this study using end-to-end learning with explainability for decision-making

In addition, the study of the interpretability of black-box models from DL is a promising research topic called eXplainable Artificial Intelligence (XAI) according to [22, 23]. In [24], Shapley Additive Explanations (SHAP) are used to interpret neural networks on Process State Points (PSPs) extracted from cavity pressure profiles. The study by [25] describes a method for multivariate data series classification using CNN and CAM.

In this study, we use visualization of the activation for interpretability of the 1D CNN model using CAM according to [26]. The input data for the models are sensor signals from inside the cavity of an injection mould during the production of the plastic part. The major contribution of this study lies in the TSC of non-periodic signals with different lengths. For better comparability, the 1NN with DTW and the 1D CNN with masking layer will be trained on the same data set without feature extraction.

1.1 Basic principles of 1NN with DTW

In this section, we give an overview of the reference method used for TSC. As a reference method without feature extraction, a DTW with 1NN is chosen. According to [19], this method is hard to beat for classification problems. The problem of classification models with unequal sample lengths was discussed in [17], where the Proximity Forest and DTW methods were used to deal with this type of time series.

DTW is a distance-based method for dealing with time series for clustering, classification and similarity search [27]. DTW can also be used for time series with unequal sampling rates or series of different lengths. For classification tasks, DTW combined with k-Nearest Neighbours (kNN) leads to a powerful model with high accuracy. Therefore, we chose such a model as a reference model to compare our results. For details of the DTW calculation see [28] and [29]. In order to gain a better understanding of the functionality of DTW, a brief overview is given as follows.

DTW compares two signals (x , e.g. Eq. 1 and y , e.g. Eq. 2) and results in a distance measurement of one signal to the other series. This matrix is called the cost matrix $C_{(m,n)}$, where $C \in \mathbb{R}^{M \times N}$ according to [30]. The absolute cost matrix between the signals is computed using Eq. 3.

The distance can be used for classification problems, for example with 1NN classifiers. Using this distance value, the accumulated cost matrix $D_{(m,n)}$ is calculated according to Eqs. 4 to 8.

$$x(n) = (2, 1, 1, 8, 8, 8, 6, 1, 1) \tag{1}$$

$$\{n \in \mathbb{N}, 1 \leq n \leq N\} \tag{2}$$

$$y(m) = (2, 2, 0, 8, 7, 4, 3) \tag{3}$$

$$\{m \in \mathbb{N}, 1 \leq m \leq M\} \tag{4}$$

$$C_{(m,n)} = |x(n) - y(m)| \tag{5}$$

$$D_{(1,1)} = C_{(1,1)} \tag{6}$$

$$D_{(m,1)} = C_{(m,1)} + D_{(m-1,1)} \tag{7}$$

$$D_{(1,n)} = C_{(1,n)} + D_{(1,n-1)} \tag{8}$$

$$D_{(m,n)} = C_{(m,n)} + d_{(m,n)} \tag{9}$$

$$d_{(m,n)} = \underset{\text{argmin}}{\begin{cases} D_{(m,n-1)}, \\ D_{(m-1,n-1)}, \\ D_{(m-1,n)} \end{cases}} \tag{10}$$

Figures 2 and 3 show the results of the simple example with x and y .

For the visualization of the difference between the two signals, the optimal warping path p^* with the minimum costs in the accumulated cost matrix can be calculated. Algorithm 1 shows the steps for calculating the optimal warping path in a different manner according to [30].

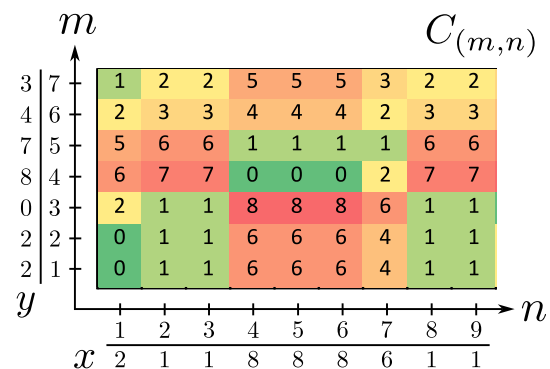


Fig. 2 Result of $C_{(m,n)}$ for element-wise absolute differences between series x and y

Algorithm 1 Optimal Warping Path

```

Input:  $D_{(m,n)}$ 
calculating  $p^*$  in reverse order starting with  $p_l = (M, N), l \in \mathbb{N}, 1 \leq l \leq (M + N)$ 
start with  $p_l = D(m, n)$ 
while  $m = 1 \ \& \ n = 1$  do
    if  $n = 1$  then
         $p_{(l-1)} := D(m - 1, 1),$ 
    end if
    if  $m = 1$  then
         $p_{(l-1)} := D(1, n - 1),$ 
    end if
    if otherwise then
         $p_{(l-1)} := d_{(m,n)}$ 
    end if
end while
 $p_l = (1, 1), l = 1$  finished
    
```

The accumulated cost matrix $D_{(m,n)}$ and the calculation of the optimal warp path are the key elements of the DTW algorithm. However, this method is not well suited for obtaining information about regions of interest in the time series. The reference method used has another disadvantage: prediction of long time series can be very slow for naive DTW computation. An idea to speed up the full DTW calculation is to use parallel computing on CPUs or GPUs [31]. Other ideas to speed up DTW computations as described in [32] are part of further work. In this study, we use parallel CPUs for the calculations.

2 Problem description

This study focuses on the following objectives. First, the classification of time series with varying lengths is considered. This type of time series is often encountered in problems with real data, such as data sets with varying

production parameters. This leads to the task of predicting the quality of injection-moulded plastic parts from sensor signals during production cycles.

To solve this problem, sensor signals from inside an injection mould for the production of plastic parts were collected. The label for the ML model is a quality parameter of the products. However, this quality parameter is difficult to measure, which increases the cost of quality management. To address this problem, we investigate two classification models to predict the quality parameter in order to realize comprehensive quality monitoring during production. As a hard-to-measure critical quality parameter, the force between the part and a force gauge should be predicted with the ML algorithms.

We use a state-of-the-art ML method, 1NN with DTW, to treat this type of time series and a CNN with CAM model for comparison. Our proposed method consists of a 1D CNN layer with a masking layer, a hidden CNN layer with Global Average Pooling (GAP) and a dense layer. Detailed information can be given in Sect. 4.1.

The final objective is to make it possible to explain and understand the classification decision made by the 1D CNN classification model. For human operators, it will be essential to recognize or understand the decision from the model. A CAM algorithm will be used to colour the activation of some regions in the time signal for the classification decision.

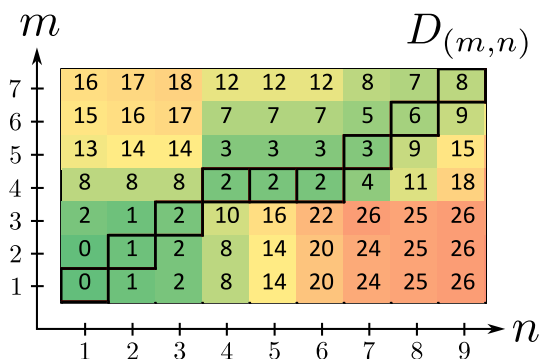


Fig. 3 Resulting accumulated cost matrix $D_{(m,n)}$ from Fig. 2 according to Eq. 7 and visualization of optimal warping path p^* for this example

3 Materials

3.1 Data acquisition and preprocessing

In each dataset, we recorded six sensor signals during the production cycle from inside the cavity of an injection

mould. All sensor signals were acquired at a sampling rate of 1000 kHz. The produced parts were numbered, matched to the dataset and quality checked by quality management (QM). Two thermocouples in the cavity, two in the mould and two piezoelectric cavity pressure sensors were used for data acquisition. Figure 4 shows schematically the mould cavity with the sensor positions at part 1 (p_1) and part 2 (p_2), respectively.

The piezoelectric sensors are amplified by a chopper charge amplifier with an analogue-to-digital converter (ADC) described in [20, 33].

The course of the cavity internal sensors is available as a time series, whereby their length differs from product to product and between the DOE steps. Figure 5 shows two pressure signals from inside the cavity of part one.

The relevant areas of the plastic part, where the critical quality parameter corresponds, are in a hard-to-measure position, and a tactile distance measurement process is too time-consuming and expensive. In addition, indirectly measuring the force based on the dimensions is also difficult to realize. Moreover, the tensile test has an effect on the surface of the tested parts which have to be destroyed after inspection. This leads us to an indirect measurement of the quality parameter. The next section describes the proposed 1D CNN model with masking layer and its use for time series classification with different lengths.

3.2 Production process and design of experiments

To predict the quality from sensor signals during the production of plastic products, we first designed a series of experiments for the production of plastic parts and data collection. This is made by using the Taguchi method and Minitab statistical software. This design of experiment (DOE) method can be used to visualize the effects of process parameters and product quality. In [34], an overview of injection moulding and, in particular, the most important process parameters for the production of plastic parts is presented.

Our Taguchi plan consists of the following parameters: the hot runner temperature T_{HR} , the mould temperature T_M ,

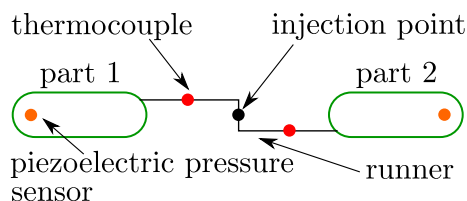


Fig. 4 Mould design with two cavities and the sensor positions of the thermocouples and the piezoelectric pressure sensors. The production machine is an Arburg A 320 S

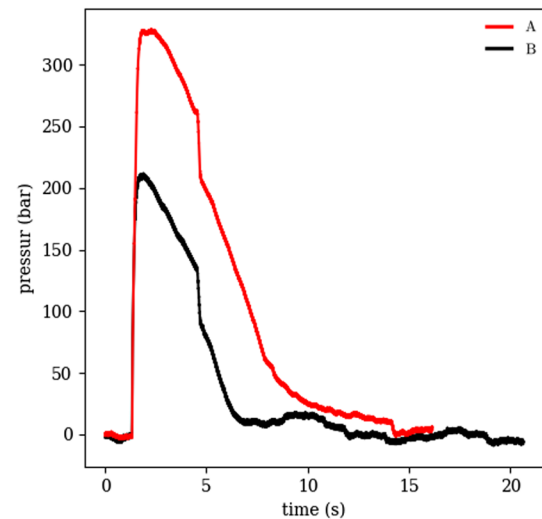


Fig. 5 An example of piezoelectric pressure signals (p_1) from two different DOE parameter sets. The different lengths can be seen. The signal 'A' (red curve) shows a fast injection process with high injection pressure and signal 'B' (black curve) shows a slow process with lower injection pressure

the velocity of the injected melt v_m , the post-pressure P_a , the cooling water flow rate \dot{V}_W , and the cooling time t_c . An extra parameter is built by the combination of $(T_M; t_c)$. All of the above factors were varied to a high, medium and low level as applicable by the machine and tool limits. The combination of two factors is equivalent to a simultaneous change in the same direction. Details of the Taguchi Plan are confidential. Further information on the Taguchi method can be found in [35].

A combination of Taguchi, response surface method, and nondominated sorting genetic algorithm II (NSGA-II) was used to optimize the injection moulding process of fibre-reinforced composites [36]. The advantage of using DOE is that it helps to collect sensor and production data in a small space close to the working point for producing high-quality plastic parts. In fact, it is the easiest way to produce plastic parts with different quality parameters in a controlled way.

In addition to the process and quality parameters mentioned above, some sensor signals from inside the mould cavity were also sampled. Now we need to determine which sensor has the highest effect to predict the product quality. The sensors we used are shown schematically in Fig. 4 in Sect. 3.1. To understand the result of the DOE with the information from the sensor signals, some further steps are necessary. Here we use Principal Component Analysis (PCA) in [37] to identify the influence of the process parameters on the shrinkage behaviour in the production of plastic moulded gears. PCA or ANalysis of VAriance (ANOVA) method is an option to extract information about the sensor signals and their influence by the

DOE parameter set. This method helps to reduce the dimension of the classification problem, from a multivariate to a univariate classification model. Thus it also helps to reduce the training time and the complexity of the model. The signals with higher fluctuations are more influenced by the DOE parameters. For the calculation, the mean of the signals is calculated and used as input to ANOVA. The results of the DOE and ANOVA are shown in Sect. 5.

Our DOE analysis aims to predict the quality of the plastic parts. It is shown that the tensile force between the plastic parts and the gauge used is the critical quality parameter. If the force is less than a critical value, the product will be treated as a “good” (P positive) part. All products with a tensile force greater than or equal to the critical value will be labelled as “not good” (N negative) parts. The amount of good and bad parts varies considerably in the data collected. The ratio of good parts to bad parts is 1:4. Some effects from balanced and unbalanced data distributions are described in [38]. Table 1 summarizes the properties of the data set, i.e. the collected time series p_2 .

It is shown in Table 1 that a property of the recorded data set is the unbalanced label distribution. To estimate the influence of the unbalanced data, different data processing schemes are employed with adjusted data distributions. As a result, the number of bad and good parts in the training data is re-distributed. In Tables 2 and 3 the subdivision of the two variants with the corresponding data distributions for the 1NN and 1D CNN classifiers are shown.

Two graphic processing units (GPUs) of an Nvidia DGX-1 deep learning cluster were used for our computational implementation. Python and Keras [39] with Tensorflow [40] were used for data preparation, classification and visualization.

4 Methods

4.1 Proposed 1D CNN model with masking layer

In this study, a method is developed based on a CNN model with an additional masking layer to deal with variations in

Table 1 Properties of the collected time series p_2

| | |
|----------------------|--|
| Max. time length | 22.140 s |
| Total number (n) | 726 |
| Quality criterion | Force on part 2 |
| Categories | $F_2 < \text{Critical value}$, good parts $F_2 \geq \text{Critical value}$, bad parts |
| Number of good parts | 166 ($\approx 23\%$) |

Table 2 Representation of the unbalanced training data distribution

| | Training data | Test data | Σ |
|----------|---------------|-----------|----------|
| Good | 132 | 34 | 166 |
| Bad | 448 | 113 | 561 |
| Σ | 580 | 147 | 727 |

Table 3 Representation of the balanced training data distribution

| | Training data | Test data | Σ |
|----------|---------------|-----------|----------|
| Good | 132 | 34 | 166 |
| Bad | 132 | 113 | 165 |
| Σ | 264 | 147 | 331 |

the length of time-series. The feature of the proposed 1D CNN is that the whole set of signals is examined and not only parts of it. A major advantage of a masking layer is that the training time can be reduced by using a batch size larger than 1 for series with different lengths. In addition, training with batch normalization can increase the accuracy of the classification model, according to [14].

The proposed model has two convolutional layers with a Rectified Linear Unit (ReLU) layer, as shown in Fig. 6.

The masking method is often used for image analysis for example human pose estimation [41]. Here, we apply this method for time series analysis. Figure 7 shows the functionality of the masking-layer for time series with varying lengths.

The description of the masking layer is part of the Keras deep learning API that we used for our development. Further information on masking can be found in [42].

For the masking layer, a few more steps need to be taken. First, the data sets must be stacked. For DataFrames in Python this step automatically fills all shorter series with nan values to the length of the largest series. The next step is to replace the nan values with the masking value. The masking value has to be a value that doesn't exist in the time series. The function of the masking layer is to mask (skip) all values in the time series that are equal to the masking value at those time steps. This is carried out in the first step as shown in Fig. 6. The next step is to perform the first convolution (orange) layer. At the end of each convolution layer, a ReLU layer is used. After the ReLU layer, the outputs are provided by a blue box. The output (output 2) of the last convolutional layer is the input to the Global Average Pooling (GAP) layer. Finally, the dense layer provides the output units to the two final neurons to obtain

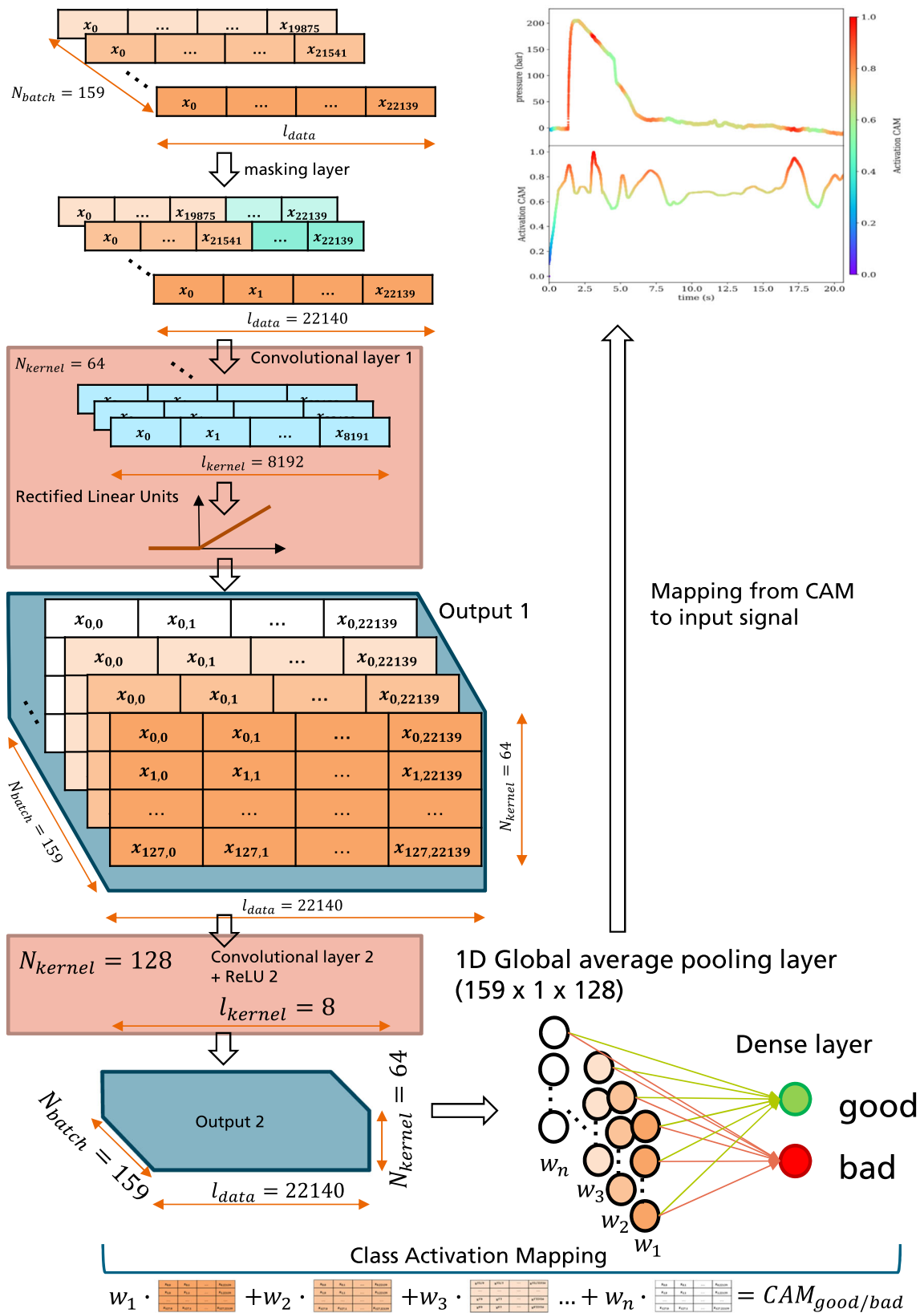


Fig. 6 The proposed CNN model for the classification task with two convolutional layers, ReLU, GAP layer inspired by a Fully Convolutional Neural Network (FCNN) from [16]

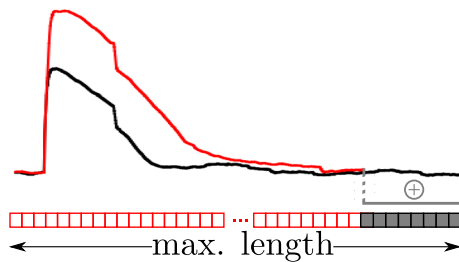


Fig. 7 Functionality of the masking layer. Shorter time series (red signal) are supplemented by large negative values not included in the signal itself (masking value / grey values). The information from the masking layer is passed to the next layers of the CNN model. The masked values are excluded from the back-propagation algorithm

the predicted quality classification. The combination of output 2, the weights w_x and the dense layer forms the CAM activation. The activation result is mapped onto the input time series, producing the coloured signal in regions of high activation for the corresponding class.

To improve the generalization of the model to unknown or new data and to avoid overfitting, the data are randomly mixed into training, test, and validation subsets. The validation data are required for the evaluation of the performance of the model. The random mixing ensures that each subset contains enough data features to reliably guarantee the model results, even if the data volume is relatively small. In addition, to bring the features of the dataset to the same order of magnitude, all input data are re-scaled to the interval $[0, 1]$ using normalization. The corresponding target data are pre-processed for classification by one-hot coding. Furthermore, to reduce the training time, the sensor signals used are selected by an ANOVA method. All plastic parts produced are inspected and labelled by the QM department.

The last step of our study is to explain the classification results. The CAM algorithm described in [26] is used for this task with some adjustments. Equations 9 to 11 [26] are used to calculate the activation back to the GAP layer. This activation can be visualized as a heat map on the input signal. As a result, this method helps to better understand the classification decision and find interesting regions in the time series to match with the injection moulding process.

$$S_c = \sum_k w_k^c \sum_{x,y} f_k(x,y) \quad (9)$$

$$= \sum_{x,y} \sum_k w_k^c f_k(x,y) \quad (10)$$

$$M_c(x,y) = \sum_k w_k^c f_k(x,y) \quad (11)$$

5 Results

5.1 DOE & ANOVA analysis

Based on the developed DOE, 726 records were collected, of which 165 are described as good and 561 as bad parts. The result of the regression analysis by the developed DOE and the process parameters have a R^2 of 91.62%. The regression equation from the DOE is as follows:

$$F_1 = -8.6 - 1.135t_c + 0.4718T_M + -0.189\dot{V} + 0.1091v_m + 0.0221T_{HR} + 0.00026P_a \quad (12)$$

This result shows which process parameter has more influence on the DOE regression model. However it is not possible to determine the influence of environmental conditions, the quality of the plastic, and the sensor properties from this regression model. The information content of the process parameters themselves is not the same as the sensor signals from inside the cavity. The calculation of the variances leads to the result shown in Fig. 8. It is shown that the pressure sensor p_2 has the highest variance.

ANOVA is used for sensors with similar variances, types, and positions. Table 4 shows the grouped sensors and the results of the ANOVA with f-value and its significance. Due to some problems with the sensors T_{M2} , the signals T_{M1} and T_{M2} cannot be used for further investigations.

It is shown from ANOVA that sensor group 1 has a larger f-value and a smaller significance. Therefore, the second cavity pressure sensor p_2 is selected as a representative sensor signal for the classification model. The following results are divided into classification approaches by the reference model and the proposed 1D CNN with masking layer model. In addition, the visualization of the activation for the interpretability of the 1D CNN classification model is compared by using CAM according to [26].

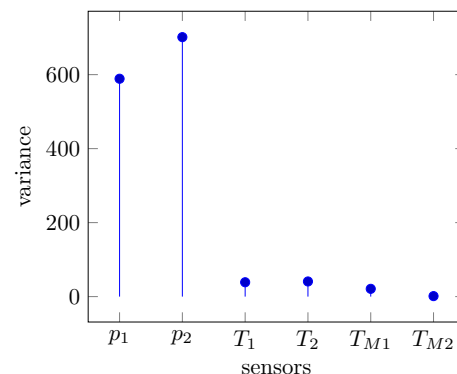


Fig. 8 Plot of the variances for the investigated sensor signals: p_1 , p_2 , T_{M1} , T_{M2} , T_1 and T_2 . The highest variance of the sensor shows p_2

Table 4 Sensor groups from ANOVA

| Group number | Grouped sensors | f Value | Significance |
|--------------|------------------|---------|--------------|
| 1 | P_1, P_2 | 33.01 | 1.12e−8 |
| 2 | T_1, T_2 | 3.77 | 0.05 |
| 3 | T_{M1}, T_{M2} | N/A | N/A |

5.2 Results by 1NN with DTW and 1D CNN with masking layer

Figure 9 shows a DTW result for two pressure signals from the injection moulding process. The DTW calculation was done in parallel with 48 jobs. This reduces the calculation time to about 12 h by using the DTAIDistance and 24 h for the FastDTW algorithm. A run of the DTW algorithm on one job took between 16 s (balanced) and 27 s (unbalanced). For the 1NN prediction, the minimum value of the corresponding input class with the smallest DTW value leads to the decision for the classification.

Table 5 shows the results for the unbalanced FastDTW model without scaling. The data distribution of this classification model is improved to balance the amount of good and bad parts within the training set. The result for the balanced model and unknown test data is displayed in

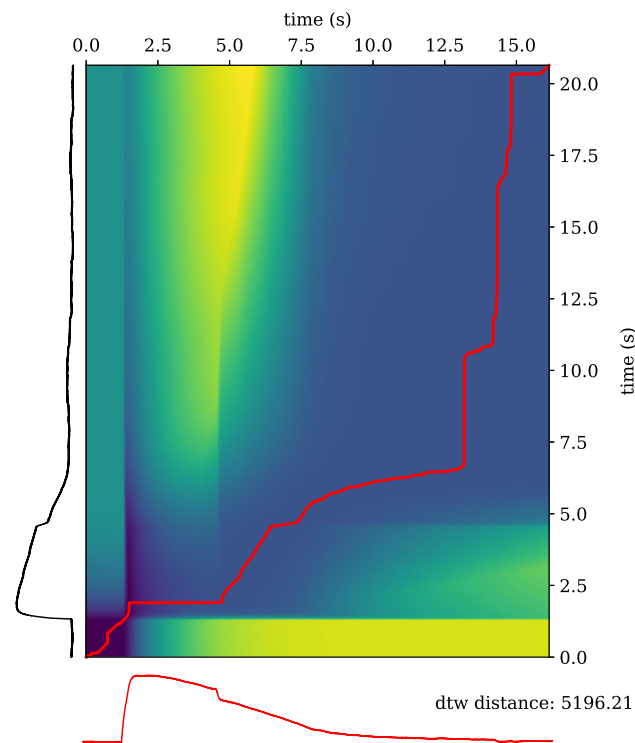


Fig. 9 Result of the DTW for two pressure signals. The calculated DTW distance, in this case, is around 5196. The heat-map shows the accumulated cost matrix. The red line shows the optimal warping path between the two signals

Table 6. For the given training-test split, the results from the FastDTW calculation are slightly better than those from the DTAIDistance. The results from the DTAIDistance DTW and all other results can be found in Sect. 8. The same unknown test dataset is used to compare the results for all models. The corresponding overall classification accuracy for unknown test data of the 1NN with FastDTW classification model is calculated according to Eq. 13 [43]. The variable r_i stands for the correct predictions and S denotes the total number of predicted records.

$$Acc = 100\% \cdot \frac{r_i}{S} \tag{13}$$

$$r_i = TN + TP \tag{14}$$

$$S = TP + FP + FN + TN \tag{15}$$

Table 7 lists the hyperparameters used to train the 1D CNN models. For training the 1D CNN model, the computation time of an epoch requires 3 s for the unbalanced training dataset. The prediction time per dataset needs about 0.0035 s. The same initialization parameters were used to train all models for unbalanced, unbalanced scaled, balanced and balanced scaled. The validation split for Keras was set to 0.3.

The first result represents the data distribution of the 1D CNN classification model with the unbalanced data distribution. Table 8 shows the corresponding confusion matrix. The overall classification accuracy using Eq. 13 with 1D CNN with a masking layer is calculated to 83.7%.

The 1D CNN classification model results with balanced training data are presented as follows. The results of the confusion matrix of the test data are presented in Table 9.

The corresponding training process and overall classification accuracy of the test data of this 1D CNN classification model are shown in Fig. 11.

5.2.1 CAM results & visualization

For generating the CAM result, the best models from the training were loaded. The visualization of CAM activation is shown in Fig. 12 The signal and the implemented CAM of an

Table 5 Confusion matrix result for unknown test data of the 1NN with FastDTW classification (test case: 1)

| | | 1NN with FastDTW | |
|---------|---|--------------------|-------------------|
| | | N (bad) | P (good) |
| Experte | N | 101 (89.4%) | 12 (10.6%) |
| | P | 14 (41.2%) | 20 (58.8%) |

The distribution of the training data is unbalanced and without scaling. According to equation 13, the accuracy is 82.3%

For better readability the values for true positive and true negative are in bold

Table 6 Confusion matrix result of unknown test data for the 1NN FastDTW balanced and without scaling (test case: 2)

| | | 1NN with FastDTW | |
|---------|---|-------------------|-------------------|
| | | N | P |
| Experte | N | 67 (59.3%) | 46 (40.7%) |
| | P | 7 (20.6%) | 27 (79.4%) |

According to equation 13, the accuracy is 63.9%

For better readability the values for true positive and true negative are in bold

Table 7 Used hyperparameters in training the experimental 1D model

| Hyperparameters | Learning rate | Batch size | Epoch | Optimizer |
|-----------------|---------------|------------|-------|-----------|
| | 0.0001 | 32 | 1500 | Adam |

Table 8 Confusion matrix for the unknown test data of the 1D CNN classification model with unbalanced data distribution without scaling (test case: 9)

| | | 1D CNN | |
|---------|---|-------------------|-------------------|
| | | N | P |
| Experte | N | 98 (86.7%) | 15 (13.3%) |
| | P | 9 (26.5%) | 25 (73.5%) |

According to Eq.13, the accuracy is 83.7%

For better readability the values for true positive and true negative are in bold

unbalanced CNN classification model show the result of a good predicted part. The heat-map from the CAM algorithm shows which region of the time series has more influence on the model decision. The activation process is shown in Fig. 12 for a good classified and in Fig. 13 for a bad classified signal. The results from CAM are also mapped to the sensor signal. This result depends on the DOE parameter set and may change for other critical quality parameters. Both results are predicted by model 9 from Sect. 8.

Table 9 Confusion matrix for the unknown test data of the 1D CNN classification model with balanced data distribution without scaling (test case: 10)

| | | 1D CNN | |
|---------|---|------------------|-------------------|
| | | Bad | Good |
| Experte | N | 71(62.8%) | 42 (37.2%) |
| | P | 8 (7.1%) | 26 (23.0%) |

According to Eq. 13, the accuracy is 66.0%

For better readability the values for true positive and true negative are in bold

6 Discussion

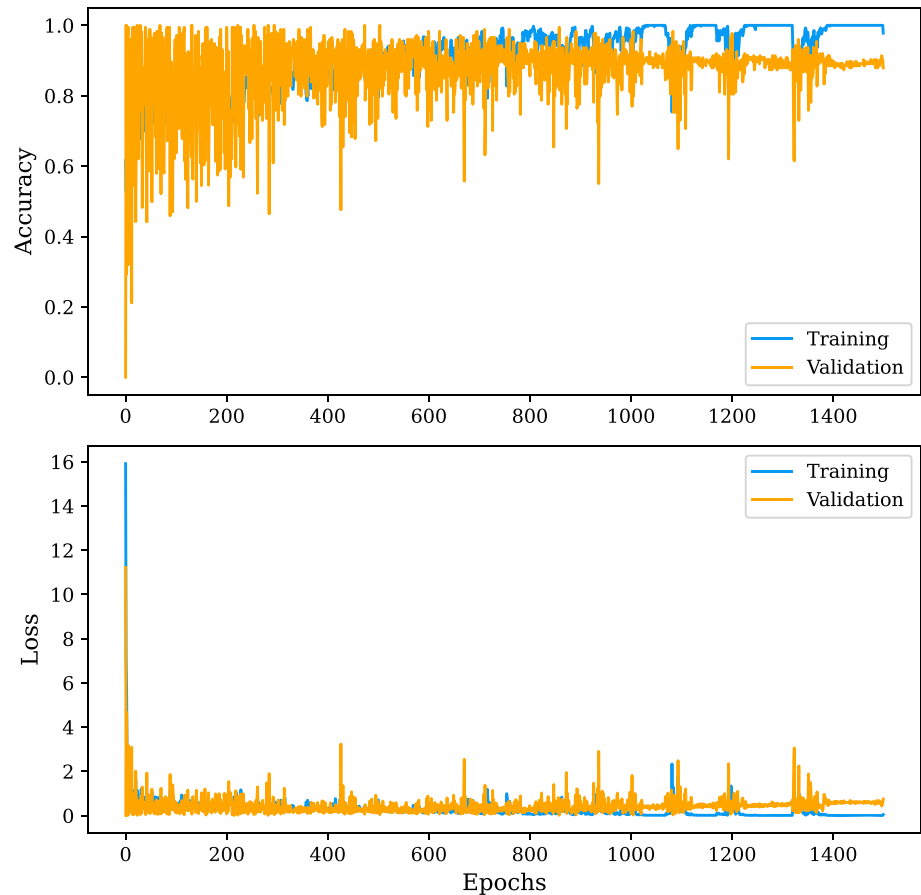
Figure 8 shows the variance of the sensor signals. A high variance in this context means a greater influence on the DOE parameter set. For this type of classification task, the ANOVA method is suitable to reduce the training time and complexity of the classification models. Compared to the DOE, it can be seen that the ANOVA method solves the problem from a different direction. The result of DOE represents the influence of the part quality by the variation of the process parameters. The quality parameter can also be influenced by the machine, plastic quality level, environmental conditions, and many other factors. All these parameters influence the sensor signals. The position of the sensor, the sensor technology, or the ADC also affects the significance of the ANOVA result. However, it is shown from our results that the melt temperature has less influence on the ANOVA result. For this type of sensor signal, small changes in the DOE parameter set will reduce the variance of the mean. Another aspect could be the size of the plastic part and the sensor positions, but both parameters cannot be changed. The problem with the T_{M2} signal could be a hardware problem, but analysis of a single of this type is impossible.

The results of the DTW algorithms used are very satisfactory. Both DTW algorithms were tested under the same conditions. The slightly better accuracy of the FastDTW algorithm compared to the DTAIDistance algorithm is unexpected, but this is not the main aspect of our study. The training time of the DTAIDistance algorithm is half that of the FastDTW algorithm, but it is much slower than that of the proposed 1D CNN with a masking layer.

From Fig. 10, it can be seen that the validation loss of the 1D CNN classification models falls rapidly at the beginning of the training and increases during that. This is an indication of overfitting of the model to the over-represented labels in the unbalanced training. The high fluctuation at the beginning could be a result of the small training data and the validation split ratio of 0.8 in combination with the chosen batch size. With an unbalanced dataset, the probability of picking more bad parts is much higher than getting good parts for the training batch. This could be the reason for the high variations at the beginning of the training process.

Therefore, this could be reduced by using a balanced dataset for training or increasing the batch size. The balanced training is shown in Fig. 11, resulting in a better training process during the first training epochs. However, in the middle of training, the distance between the training loss and the validation loss increases, and the model starts to overfit and the validation accuracy cannot be improved. A possible reason could be the implementation of a dropout

Fig. 10 Training process of 1D CNN classification model with original data distribution. The blue lines represent the training data and the orange is the validation data set



layer [44] and early stopping, but we believe that the main problem would be the small number of recorded datasets.

The results in Fig. 12 from the CAM algorithm show high activations in the pressure signals during the production phase, which agrees well with the experts' opinion. A part of the compression phase, where the internal pressure increases rapidly, is recognized as an important region for predicting good parts. The highest CAM activation for bad parts is shown in the middle and the end of the pressure curve where the cavity pressure drops rapidly or has a step downward. These areas correspond to the technical meaning of the injection moulding process and show strong changes where the internal pressure drops or rises rapidly. Such region of the injection moulding process is called the after-pressure and holding phase which has a greater impact on the product quality than other phases. Some other CAM results are displayed in Sect. 8 which shows how different the CAM results can be for different types of trained models. Some models show a high rise at the beginning of the pressure signal, others present the step at the end of the signal. The difference in the accuracy offers an indication of which model is better and how the model should be trained for a small number of collected data.

On the other hand, there are results from CAM that are not easy to understand even for experienced moulders. This

could be caused by a non-optimal experimental design or overfitting of the classifier. The proposed CNN with CAM can highlight regions of interest in the pressure curves that contribute to its prediction. The reference model INN with DTW can also produce graphical output directly from the distance calculation of DTW. However, its warp path is difficult to interpret and only contains information about the phase and periodicity of the inputs. As a result, the reference model is not able to realize this kind of information as the CAM do.

7 Conclusions

The contributions of this study can be summarized as: a) the development of a fast classification model for TSC with up to 83.7 % accuracy for a hard-to-measure prediction task, b) the application of CNN with a masking layer for time series with different lengths, c) the combination with the CAM helps to lead to a better understanding of the classification results, d) better results by CNN with CAM than a INN with DTW for this type of data, and e) using both methods to predict hard-to-measure quality parameters on real-world data.

Fig. 11 Training process of the 1D CNN classification model with balanced training data distribution. The blue lines represent the training dataset and the orange is the validation dataset

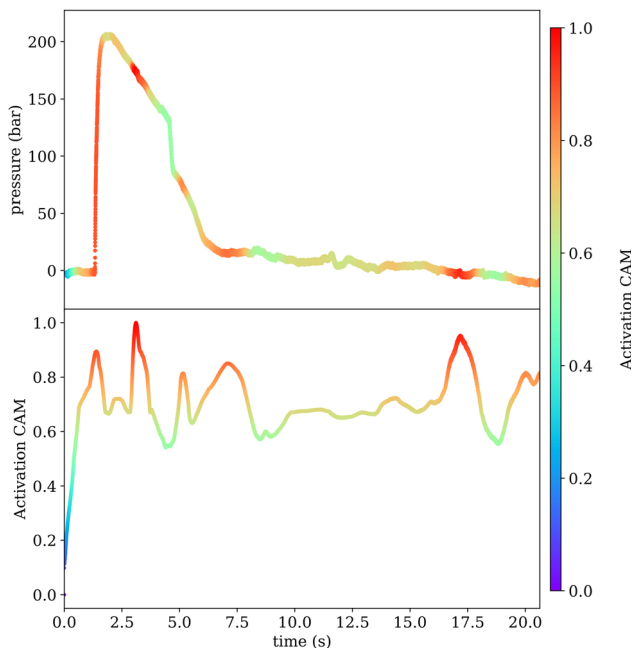
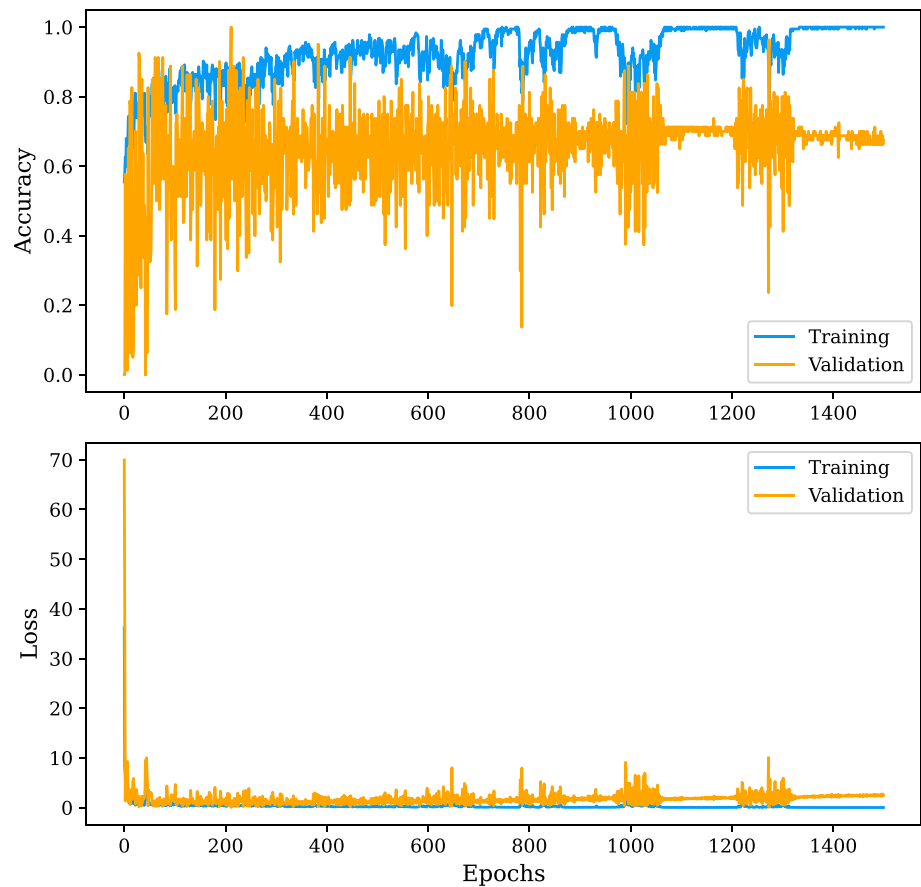


Fig. 12 CAM visualization by the 1D CNN classification unbalanced trained model from Table 8. The upper time series shows a pressure signal with the coloured CAM activation (lower curve). The signal shows a good classified cavity pressure curve resulting from test scenario 9

Further details on the main findings of this study are as follows. With the proposed 1D CNN and a masking layer, a fast classification for TSC of discontinuous signals with different lengths can be realized. The state-of-the-art 1NN with DTW is a suitable method to compare the classification results, but it is computationally expensive for a large number of time series. A simple way to decrease the computation time for the classification models is to reduce the multivariate classification problem to a univariate one. This was done by using the ANOVA method to determine the signal with the largest variance. In comparison to the 1NN with DTW, this makes the prediction task much faster. The overall classification accuracy of this model reaches 82.3 %. The confusion matrix and overall classification accuracy of the 1D CNN models with unbalanced data distribution is 83.7 % and with balanced distribution 66.0 %. The result of the unbalanced data distribution is $\approx 1.4\%$, which is better than the best 1NN with DTW model. The results for the two DTW algorithms (FastDTW and DTAIDistance) are also shown in the tables in Sect. 8.

The classification results with the implemented masking layer are promising especially for the hard-to-predict quality parameter. The computation time of the reference model was between 12 and 24 h, which is too slow for prediction tasks for this type of data. The model calculation

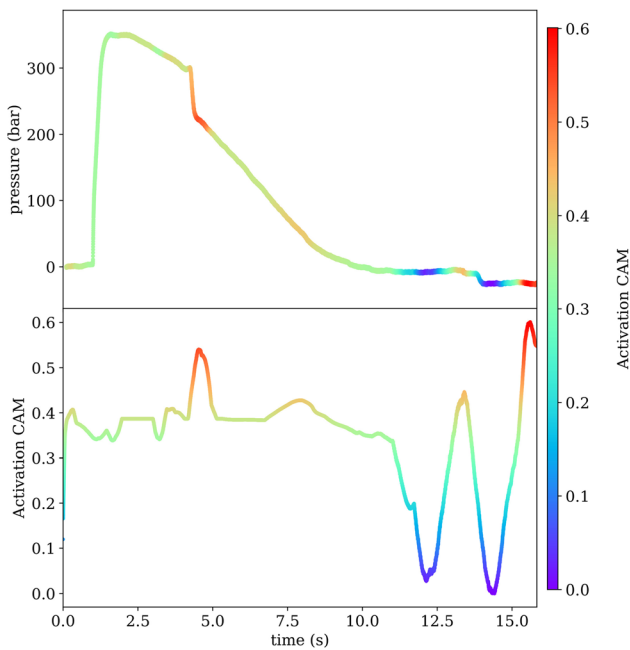


Fig. 13 CAM visualization result for a bad classified pressure signal, also resulting from test scenario 9 from Table 8

with this train and test subsets for the INN with DTW in 12 h/24 h could be realized only by parallel computing with 48 tasks. The FastDTW [45] is slower than the DTAIDistance [46] and leads to the same result as in the literature. A 3.4 % better accuracy by FastDTW compared to the DTAIDistance algorithm is negligible. The difference between the computation times was already analysed in [47] and could therefore be verified.

Compared to the CNN model with a masking layer, the average training time for the unbalanced dataset is 33 min, and for the balanced dataset, it is further reduced to 16 min. In addition, the 1D CNN with a masking layer delivers significantly faster predictions and has improved accuracy using the same dataset. This highlights the importance of training the CNN on diverse series with a batch size greater than one, which is part of our recommendation.

In subsequent applications, the use of up to 200 epochs proves to be sufficient for effective training. This large number of epochs provides a more robust basis for comparing different models. Notably, this approach not only reduces training time but also maintains the superior performance of the developed CNN for this specific data type.

While the accuracy and loss metrics show a favourable trajectory for the training data, a plateau is observed for the test data set. This observation is indicative of the complexity of the learning task, with the training model showing a tendency to overfit due to the limited number of records.

The properties of all models used in this paper are summarized in Sect. 8. The higher activation during the production phases in the pressure signals corresponds well with

the meaning of the injection experts. On the other hand, some CAM results are difficult to understand, which could be due to the random training initialization. At these points, other sensors like the temperature may have a slightly greater influence on the results, but this information is not part of this work. The classification methods produce good results without the need for a time-consuming feature extraction algorithm. The major advantage of the 1D CNN model with a masking layer is the fast training time and the ability to obtain some regions of interest from the CAM algorithm. This made the proposed model more efficient for application to TSC with varying length series, especially in discontinuous production processes.

Regarding future work, the input data of the proposed 1D CNN models are the data sets of only one cavity pressure. To avoid overfitting or to reduce the scatter of classification accuracy and loss value, it might be better to use more signals or simulation data. In addition, a combination with the second pressure sensor or with the melt temperature sensors will be interesting for improving the results. This requires a multivariate method to extract some information from the sensor signals. PCA in [37] was used to reduce the influence of process parameters on the shrinkage behaviour in the manufacture of plastic moulded gears. In [36], a combination of the Taguchi method, response surface methodology, and nondominated sorting genetic algorithm II (NSGA-II) was used to optimize the injection moulding process of fibre-reinforced composites. For further work, PCA will be the next method to investigate the reduction in sensor signals in the context of multivariate classification tasks.

Another approach could be transfer learning. Some interesting aspects have been mentioned in [48]. Furthermore, it is conceivable to combine regression and classification approaches using so-called multi-output models to address the problem of unbalanced datasets on the one hand and to ensure compatibility with other explanatory models on the other hand. In this way, model-agnostic explanatory models could complement CAM explanations in the future. An implementation of our approach as a real-time quality prediction method for plastic injection moulding in Apache Spark may be possible. As stated in [49], to meet the demands of digital transformation in Industry 4.0, a near real-time quality prediction method for plastic injection moulding could be effectively realized using Apache Spark.

A final point is the possibility of exploring Batch Normalization (BN) methods and Residual neural NETWORK (ResNET) models for this type of data with masking layers and varying lengths. This can increase classification accuracy and may be more efficient for small and unbalanced datasets.

Moreover, research on LSTM and GRU models or the classification of other datasets can be topics of future work.

Appendix

See Tables 10, 11, 12 and Figs. 14, 15, 16, 17, 18 and 19.

Table 10 Summary of the characteristics of all used models in my master thesis

| Test case | | Reference method INN with FastDTW | | | | | | | | |
|-------------------------------|-------|-----------------------------------|---------|----------|---------|------------|---------|----------|---------|---------|
| | | 1 | | 2 | | 3 | | 4 | | |
| Training | | Unbalanced | | Balanced | | Unbalanced | | Balanced | | |
| Input scaled | | – | | – | | ✓ | | ✓ | | |
| Confusion matrix ^a | TN | FP | 101 | 12 | 67 | 46 | 104 | 9 | 74 | 39 |
| | (≈ %) | (≈ %) | (89.4%) | (10.6%) | (59.3%) | (40.7%) | (92.0%) | (8.0%) | (65.5%) | (34.5%) |
| | FN | TP | 14 | 20 | 7 | 27 | 18 | 16 | 11 | 23 |
| | (≈ %) | (≈ %) | (41.2%) | (58.8%) | (20.6%) | (79.4%) | (52.9%) | (47.1%) | (32.4%) | (67.6%) |
| Accuracy | | 82.3% | | 63.9% | | 81.6% | | 66.0% | | |

^aN for negative and P for a positive label, T true and F false prediction

Table 11 Summary of the characteristics of all used models in my master thesis

| Test case | | Reference method INN with DTAIDistance DTW | | | | | | | | |
|-------------------------------|-------|--|---------|----------|---------|------------|---------|----------|---------|---------|
| | | 5 | | 6 | | 7 | | 8 | | |
| Training | | Unbalanced | | Balanced | | Unbalanced | | Balanced | | |
| Input scaled | | – | | – | | ✓ | | ✓ | | |
| confusion matrix ^a | TN | FP | 97 | 16 | 74 | 39 | 101 | 12 | 72 | 41 |
| | (≈ %) | (≈ %) | (85.8%) | (14.2%) | (65.5%) | (34.5%) | (89.4%) | (10.6%) | (63.7%) | (36.3%) |
| | FN | TP | 15 | 19 | 7 | 27 | 17 | 17 | 7 | 27 |
| | (≈ %) | (≈ %) | (44.1%) | (55.9%) | (20.6%) | (79.4%) | (50.0%) | (50.0%) | (20.6%) | (79.4%) |
| accuracy | | 78.9% | | 68.7 % | | 80.3 % | | 67.3 % | | |

^aN for negative and P for a positive label, T true and F false prediction

Table 12 Summary of the characteristics of all used models in my master thesis

| Test case | | Proposed method 1D CNN with masking layer | | | | | | | | |
|-------------------------------|-----------------|---|---------|----------|---------|------------|---------|----------|---------|---------|
| | | 9 | | 10 | | 11 | | 12 | | |
| Training | | Unbalanced | | Balanced | | Unbalanced | | Balanced | | |
| Input scaled | | – | | – | | ✓ | | ✓ | | |
| Confusion matrix ^a | TN | FP | 98 | 15 | 71 | 42 | 76 | 37 | 21 | 92 |
| | (≈ %) | (≈ %) | (86.7%) | (13.3%) | (62.8%) | (37.2%) | (67.3%) | (32.7%) | (18.6%) | (81.4%) |
| | FN | TP | 9 | 25 | 8 | 26 | 5 | 29 | 1 | 33 |
| | (≈ %) | (≈ %) | (26.5%) | (73.5%) | (7.1%) | (23.0%) | (14.7%) | (85.3%) | (0.9%) | (29.2%) |
| | Figure CAM good | | Fig. 12 | | Fig. 14 | | Fig. 16 | | Fig. 18 | |
| | Figure CAM bad | | Fig. 13 | | Fig. 15 | | Fig. 17 | | Fig. 19 | |
| | Accuracy | | 83.7% | | 66.0% | | 71.4% | | 36.7% | |

^a N for negative and P for a positive label, T true and F false prediction

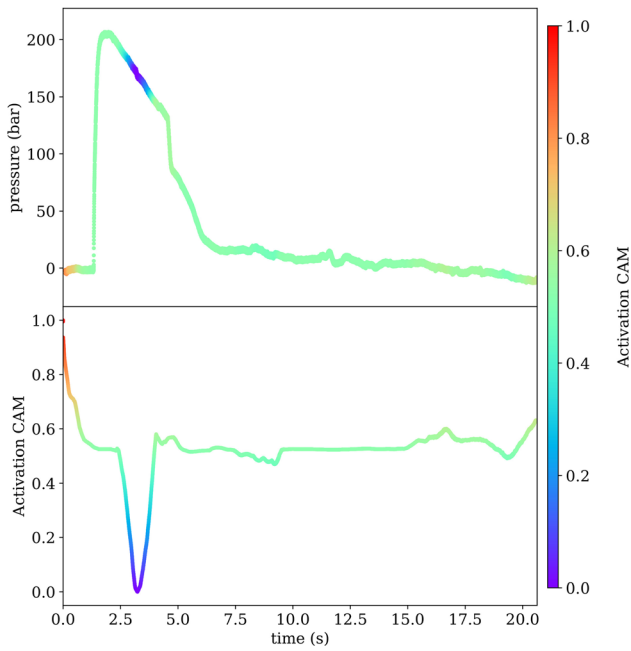


Fig. 14 CAM visualization of 1D CNN classification, a balanced trained model without scaling from test scenario 10 for a good part

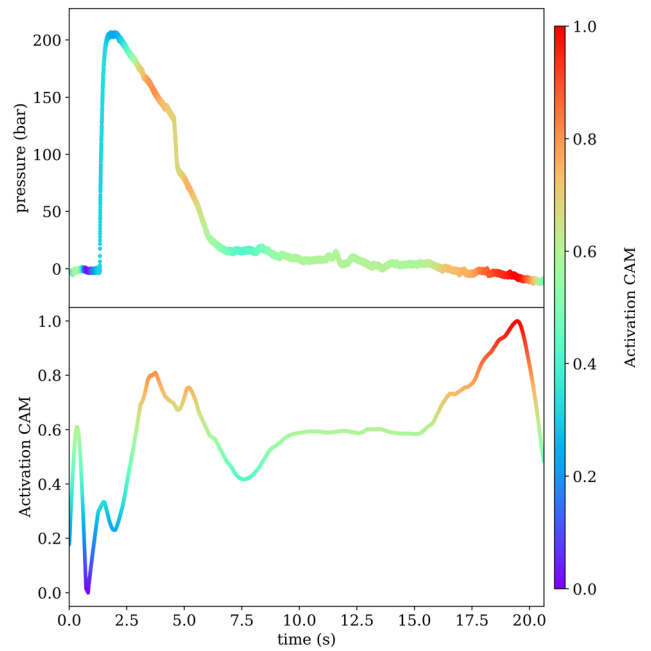


Fig. 16 CAM visualization of 1D CNN classification, an unbalanced trained model with scaling from test scenario 11 for a good part

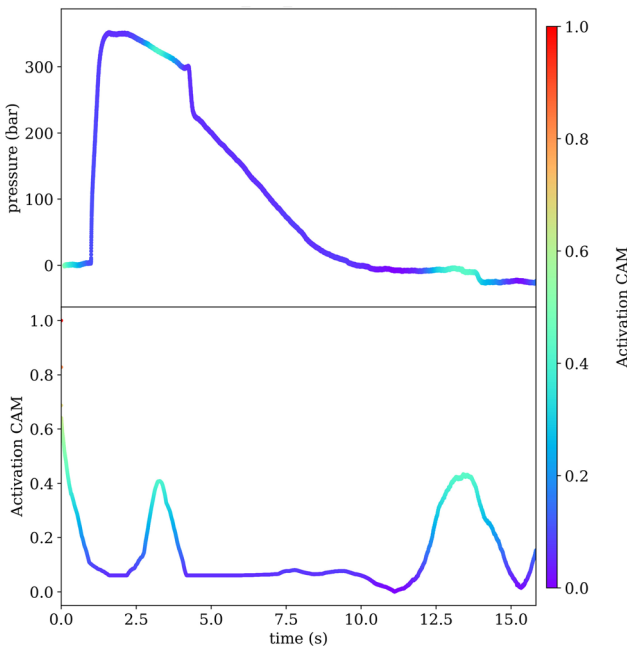


Fig. 15 CAM visualization of 1D CNN classification, a balanced trained model without scaling from test scenario 10 for a bad part

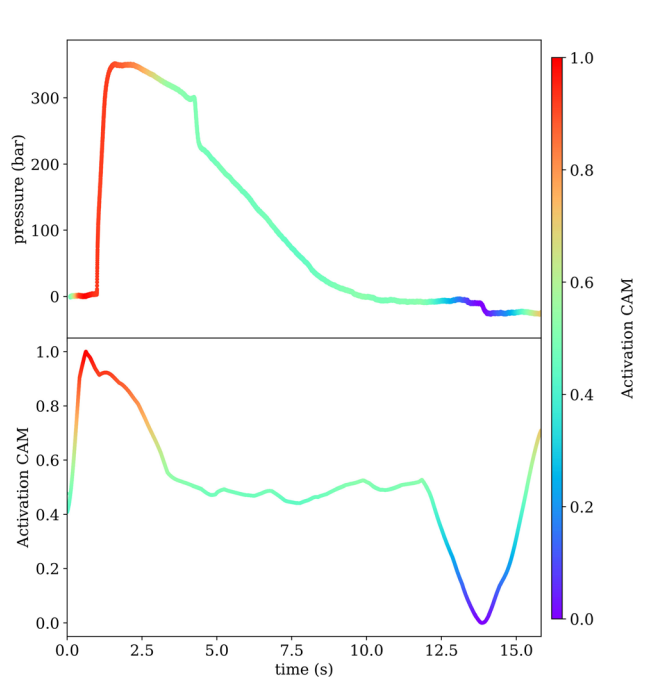


Fig. 17 CAM visualization of 1D CNN classification, an unbalanced trained model with scaling from test scenario 11 for a bad part

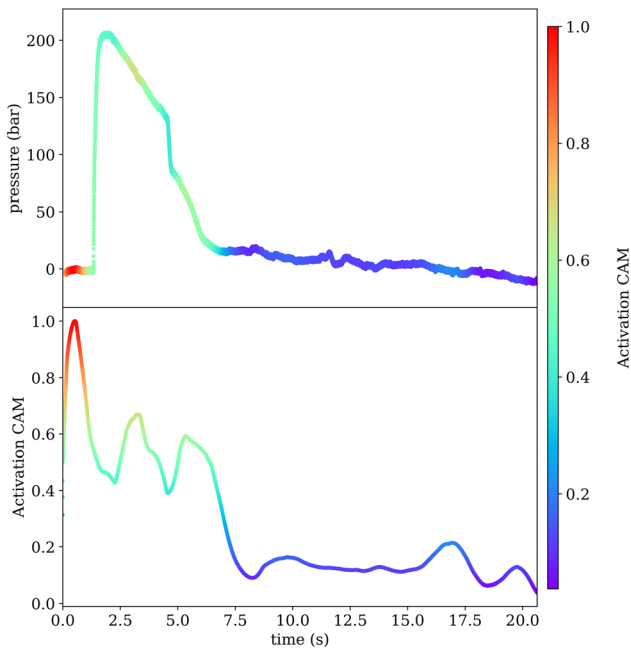


Fig. 18 CAM visualization of 1D CNN classification, a balanced trained model with scaling from test scenario 12 for a good part

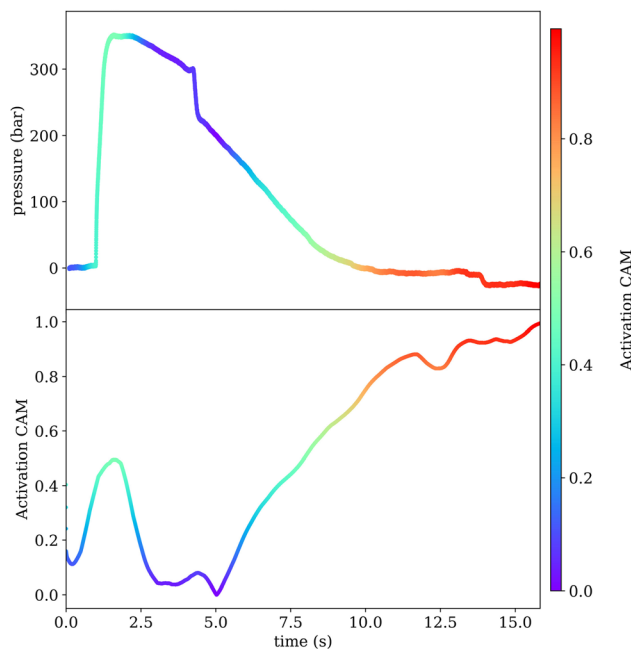


Fig. 19 CAM visualization of 1D CNN classification, a balanced trained model with scaling from test scenario 12 for a bad part

Author contributions PL, AW, and CW contributed to the study's conception and design. Material preparation and data collection were performed by NG and MS. LW, NG, and MS were responsible for the data analysis, developed models, and interpretation of the results. The first draft of the manuscript was written by MS and modified by PL,

and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL. This work has been supported by the German federal government's special fund at BMBF [Grant Nos.: 16DKWN078A, 16DKWN078B]. The authors thank the reviewers for their instructive comments and suggestions.

Data availability The datasets generated and/or analysed in the current study are not available to the public due to the license of the industrial partner, but can be viewed by the corresponding authors upon a justified request. The data sets are not part of the funded projects.

Code availability To enable and encourage further research in this area, we have published some code snippets on the following website: https://github.com/MScSchneider/CNN_masking_layer_CAM_for_timeseries.

Declarations

Declarations The authors declare that they have no conflict of interest.

Ethical approval This article does not involve human subjects for data collection. There is no need for ethical approval.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Sadeghi BHM (2000) A BP-neural network predictor model for plastic injection molding process. *J Mater Process Technol* 103(3):411–416. [https://doi.org/10.1016/s0924-0136\(00\)00498-2](https://doi.org/10.1016/s0924-0136(00)00498-2)
2. Smith ML, Smith LN, Hansen MF (2021) The quiet revolution in machine vision - a state-of-the-art survey paper, including historical review, perspectives, and future directions. *Comput Ind* 130:103472. <https://doi.org/10.1016/j.compind.2021.103472>
3. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521(7553):436–444. <https://doi.org/10.1038/nature14539>
4. Im D, Lee S, Lee H, Yoon B, So F, Jeong J (2021) A data-centric approach to design and analysis of a surface-inspection system based on deep learning in the plastic injection molding industry. *Processes* 9(11):1895. <https://doi.org/10.3390/pr9111895>
5. Ha H, Jeong J (2021) CNN-based defect inspection for injection molding using edge computing and industrial IoT systems. *Appl Sci* 11(14):6378. <https://doi.org/10.3390/app11146378>
6. Rousopoulou V, Nizamis A, Vafeiadis T, Ioannidis D, Tzovaras D (2020) Predictive maintenance for injection molding machines

- enabled by cognitive analytics for industry 4.0. *Front Artif Intell.* <https://doi.org/10.3389/frai.2020.578152>
7. Sawada A, Miyagawa T, Ebihara A, Yachida S, Hosoi T (2022) Convolutional neural networks for time-dependent classification of variable-length time series. In: 2022 International joint conference on neural networks (IJCNN), pp. 1–8. <https://doi.org/10.1109/IJCNN55064.2022.9892605>
 8. Dempster A, Schmidt DF, Webb GI (2023) Hydra: competing convolutional kernels for fast and accurate time series classification. *Data Min Knowl Disc* 37(5):1779–1805. <https://doi.org/10.1007/s10618-023-00939-3>
 9. Bier A, Jastrzebska A, Olszewski P (2022) Variable-length multivariate time series classification using ROCKET: a case study of incident detection. *IEEE Access* 10:95701–95715. <https://doi.org/10.1109/access.2022.3203523>
 10. Fawaz HI, Forestier G, Weber J, Idoumghar L, Muller P-A (2019) Deep learning for time series classification: a review. *Data Min Knowl Disc* 33(4):917–963. <https://doi.org/10.1007/s10618-019-00619-1>
 11. Smith KE, Williams P (2018) Time series classification with shallow learning shepard interpolation neural networks. In: Mansouri, A., El Moataz, A., Nouboud, F., Mammass, D. (Eds.) *International Conference on Image and Signal Processing*, pp. 329–338. https://doi.org/10.1007/978-3-319-94211-7_36
 12. Kvaktun D, Hoffmann A, Schiffers R (2022) Analysis of feature extraction algorithms for quality prediction using machine learning in injection molding. *Procedia CIRP* 112:590–595. <https://doi.org/10.1016/j.procir.2022.09.059>
 13. Yong H, Huang J, Meng D, Hua X, Zhang L (2020) Momentum batch normalization for deep learning with small batch size. *Springer*. https://doi.org/10.1007/978-3-030-58610-2_14
 14. Thakkar V, Tewary S, Chakraborty C (2018) Batch normalization in convolutional neural networks — a comparative study with CIFAR-10 data. *IEEE*. <https://doi.org/10.1109/eaic.2018.8470438>
 15. Lines J, Bagnall A (2014) Time series classification with ensembles of elastic distance measures. *Data Min Knowl Disc* 29(3):565–592. <https://doi.org/10.1007/s10618-014-0361-2>
 16. Wang Z, Yan W, Oates T (2017) Time series classification from scratch with deep neural networks: a strong baseline. *IEEE*. <https://doi.org/10.1109/ijcnn.2017.7966039>
 17. Tan CW, Petitjean F, Keogh E, Webb GI (2019) Time series classification for varying length series. *arXiv*. <https://doi.org/10.48550/ARXIV.1910.04341>
 18. Finkeldey F, Volke J, Zarges J-C, Heim H-P, Wiederkehr P (2020) Learning quality characteristics for plastic injection molding processes using a combination of simulated and measured data. *J Manuf Process* 60:134–143. <https://doi.org/10.1016/j.jmapro.2020.10.028>
 19. Bagnall A, Lines J (2014) An experimental evaluation of nearest neighbour time series classification. Technical report, School of Computing Sciences, University of East Anglia (June)
 20. Seul T, Wenzel A, Schneider M, Röstel P, Jahn R, Schlutter R (2015) It's the inner Values that Count. *Kunststoffe International*
 21. Schneider M, Jahn A, Greifzu N, Fränzel N (2016) Entwicklung Eines Unipolaren Differentiellen Ladungsverstärkers Für die Anwendung in Eingebetteten Diagnosesystemen zur Druckmessung in Spritzgussmaschinen. In: 18. GMA/ITG-Fachtagung Sensoren und Messsysteme, pp. 782–789. <https://doi.org/10.5162/sensoren2016/P9.2>
 22. Anneken M, Veerappa M (2022) eXplainable Artificial Intelligence (XAI)-Toolbox. <https://www.iosb.fraunhofer.de/de/projekte-produkte/XAIToolbox.html>
 23. Veerappa M, Anneken M, Burkart N, Huber MF (2022) Validation of XAI explanations for multivariate time series classification in the maritime domain. *J Comput Sci* 58:101539. <https://doi.org/10.1016/j.jocs.2021.101539>
 24. Gim J, Rhee B (2021) Novel analysis methodology of cavity pressure profiles in injection-molding processes using interpretation of machine learning model. *Polymers* 13(19):3297. <https://doi.org/10.3390/polym13193297>
 25. Boniol P, Meftah M, Remy E, Palpanas T (2022) dcam: dimension-wise class activation map for explaining multivariate data series classification. *Proceedings of the 2022 International Conference on Management of Data (SIGMOD '22)*, June 12–17, 2022, Philadelphia, PA, USA. <https://doi.org/10.1145/3514221.3526183> arXiv:2207.12165 [cs.LG]
 26. Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A (2016) Learning deep features for discriminative localization. *IEEE*. <https://doi.org/10.1109/cvpr.2016.319>
 27. Shokoohi-Yekta M, Hu B, Jin H, Wang J, Keogh E (2016) Generalizing DTW to the multi-dimensional case requires an adaptive approach. *Data Min Knowl Disc* 31(1):1–31. <https://doi.org/10.1007/s10618-016-0455-0>
 28. Tavenard R, Faouzi J, Vandewiele G, Divo F, Androz G, Holtz C, Payne M, Yurchak R, Rußwurm M, Kolar K, Woods E (2020) Tslern, a machine learning toolkit for time series data. *J Mach Learn Res* 21(118):1–6
 29. Meert W, Hendrickx K, Van Craendonck T, Robberechts P, Blockeel H, Davis J (2020). *DTAIDistance* Zenodo. <https://doi.org/10.5281/ZENODO.7158824>
 30. Müller, M (2007) Dynamic time warping, 69–84. https://doi.org/10.1007/978-3-540-74048-3_4
 31. Schmidt B, Hundt C (2020) *cuDTW++: ultra-fast dynamic time warping on CUDA-enabled GPUs*. Springer. https://doi.org/10.1007/978-3-030-57675-2_37
 32. Mueen A, Keogh E (2016) Extracting optimal performance from dynamic time warping. *ACM*. <https://doi.org/10.1145/2939672.2945383>
 33. Schneider M, Jahn A, Greifzu N, Fränzel N (2017) Development of a chopper charge amplifier for measuring the cavity pressure inside injection moulding tools and signal optimisation with a Kalman filter. *J Sens Sens Syst* 6(1):199–210
 34. Farooque R, Asjad M, Rizvi SJA (2021) A current state of art applied to injection moulding manufacturing process – a review. *Mater Today: Proceed* 43(1):441–446. <https://doi.org/10.1016/j.matpr.2020.11.967>
 35. Roy RK (2010) *A primer on the Taguchi method*. Society of Manufacturing Engineers. ISBN: 9780872638648
 36. Li K, Yan S, Zhong Y, Pan W, Zhao G (2019) Multi-objective optimization of the fiber-reinforced composite injection molding process using taguchi method, RSM, and NSGA-II. *Simul Model Pract Theory* 91:69–82. <https://doi.org/10.1016/j.simpat.2018.09.003>
 37. Mehat NM, Kamaruddin S, Othman AR (2014) Optimized injection molding of unfilled and glass filled PA6 gears. *Int J Manuf Eng* 2014:1–8. <https://doi.org/10.1155/2014/719462>
 38. Trommer M, Wenzel A (2016) *Regelungs- und Informationstechnik. Automatische Identifikation und Selektion von untypischen Datenat - Automatisierungstechnik Methoden und Anwendungen der Steuerungs-*, 64(1):19–28. <https://doi.org/10.1515/auto-2015-0003>
 39. Chollet F et al. (2015) *Keras* <https://keras.io>
 40. Martín A et al. (2015) *TensorFlow: large-scale machine learning on heterogeneous systems*. Software available from <https://www.tensorflow.org/>
 41. He K, Gkioxari G, Dollár P, Girshick R (2017) *Mask r-cnn*. Facebook AI research [arXiv:1703.06870](https://arxiv.org/abs/1703.06870) [cs.CV]

42. Zhu S, Chollet F (2023) Understanding masking & padding. online https://keras.io/guides/understanding_masking_and_padding/
43. Trommer M (2017) Beitrag zur Anwendung von Support-Vektor-Maschinen zur robusten nichtlinearen Klassifikation komplexer biologischer Daten. PhD thesis, Technische Universität Ilmenau, Ilmenau (May). https://www.db-thueringen.de/receive/dbt_mods_00032299
44. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 15(56):1929–1958
45. Salvador S, Chan PK (2004) FastDTW: toward accurate dynamic time warping in linear time and space. Dept. of Computer Sciences Florida Institute of Technology
46. Wannesm Khendrickx Yurtman A, Robberechts P, Vohl D, Ma E, Verbruggen G, Rossi M, Shaikh M, Yasirroni M, Todd Zielinski W, Van Craenendonck T, Wu S (2022) wannesm/dtaidistance: v2.3.5. Zenodo . <https://doi.org/10.5281/ZENODO.5901139>
47. Wu R, Keogh EJ (2022) FastDTW is approximate and generally slower than the algorithm it approximates. *IEEE Trans Knowl Data Eng* 34(8):3779–3785. <https://doi.org/10.1109/tkde.2020.3033752>
48. Tercan H, Guajardo A, Heinisch J, Thiele T, Hopmann C, Meisen T (2018) Transfer-learning: bridging the gap between real and simulation data for machine learning in injection molding. *Procedia CIRP* 72:185–190 <https://doi.org/10.1016/j.procir.2018.03.087>
49. Uguroglu E (2021) Near-real time quality prediction in a plastic injection molding process using apache spark. *IEEE*. <https://doi.org/10.1109/iscsic54682.2021.00059>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.