A GENERATIVE NEURAL NETWORK MODEL FOR THE QUALITY PREDICTION OF WORK IN PROGRESS PRODUCTS

A PREPRINT

Guodong Wang
Institute of Computer Engineering
Technische Universität Wien
guodong.wang@tuwien.ac.at

Anna Ledwoch
Department of Engineering
University of Cambridge
aml79@cam.ac.uk

Ramin Hasani
Institute of Computer Engineering
Technische Universität Wien
ramin.hasani@tuwien.ac.at

Radu Grosu
Institute of Computer Engineering
Technische Universität Wien
radu.grosu@tuwien.ac.at

Alexandra Brintrup[*]
Department of Engineering
University of Cambridge
ab702@cam.ac.uk

ABSTRACT

One of the key challenges in manufacturing processes is improving the accuracy of quality monitoring and prediction. This paper proposes a generative neural network model for automatically predicting work-in-progress product quality. Our approach combines an unsupervised feature-extraction step with a supervised learning method. An autoencoding neural network is trained using raw manufacturing process data to extract rich information from production line recordings. Then, the extracted features are reformed as time-series and are fed into a multi-layer perceptron for predicting product quality. Finally, the outputs are decoded into a forecast quality measure. We evaluate the performance of the generative model on a case study from a powder metallurgy process. Our experimental results suggest that our method can precisely capture the defective products.

Keywords Autoencoder · Generative Models · Quality Prediction · Time-delayed Neural Networks · Powder Metallurgy

1 Introduction

Quality control is the process of monitoring, predicting and improving the quality of products within manufacturing processes [1]. The common practice in industry is to adopt statistical process control (SPC) toolkits for tackling quality control issues [2, 3], which includes sampling from batches produced during a process, and examining the associated control chart containing sample dimensions, as well as the the upper and lower bounds of the sample dimensions forming the specification around the dimension [3, 4, 5]. Any product samples either greater than the upper limit or lower than the bottom limit suggest increased likelihood of products that violate quality control specifications. The quality control specifications of work-in-progress (WIP) products on subsequent processes are gradually tightened as they are passed through one process stage to another [6].

During past decades, many valuable works have been developed for SPC, such as statistical mechanics [7] and data-driven methods [8, 9]. Statistical mechanics-based approaches make use of statistical features such as mean, variance, maximum, minimum, kurtosis, skewness and root-mean-square (RMS), as the direct or indirect indicators of the products' quality [10], to gain a better understanding of the status of the WIP products. However, due to the increased complexity of production systems, statistical mechanics are not sufficiently geared to handle increased data and uncertainty from modern factories [11]. More specifically, unnatural patterns have been traditionally identified through statistical tests. However, these are only able to detect pattern presence, not which one occurs. Moreover, these techniques suffer from errors such as false alarms and missed disturbances.

[*] Corresponding Author, Address: 17 Charles Babbage Rd, Cambridge CB3 0FS, United Kingdom.

With the recent development of more advanced data collection and monitoring such as the Internet of Things based systems [12], [13, 14] and improved computational power manufacturers are increasingly investigating machine learning based methods for predictive quality control. For example, the Chemical industry has a significant amount of uncertainty within the manufacturing processes, due to subtle differences between raw materials and varying operating conditions. To address the quality variance problem in the chemical industry, Wang et al. [15, 16] proposed data-driven based multi-model approaches to capture the time-series relationships among historical production quality and the quality of final products. Similar research activity was conducted by Aumi et al. [17] in which the authors proposed a model predictive control framework for modeling the temporal connection. Whereas, Yu [18] used a finite mixture model based Gaussian process regression approach to conducting online quality prediction for nonlinear chemical processes, which demonstrates that data-driven methods like partial least squares based techniques are also beneficial for manufacturing process monitoring and diagnosis [19, 20]. Industrial process monitoring is another indirect way of estimating WIP product quality. For example, Hasani et al. [21] proposed a machine learning suite for bearing component diagnostics and prognostics. Estimating the performance shifting of production systems could provide additional useful information for better predicting the quality characteristics of WIP products. The popularity of machine learning is due to better accuracy without the need for manually detecting and interpreting patterns [22]. However, the main setbacks from machine learning based approaches have thusfar included their ability to handle noise and imbalance in data.

In our study, we aim to predict quality violations at time $t + 1$ based on historical sample data up to time t. Quality control engineers randomly select N samples out of M WIP products, in which $N \ll M$, and measure respective quality characteristics. The quality characteristic in the paper is a given dimension of a metal part produced by powder metallurgy for the automotive industry. The process under consideration is the Compaction stage, where mixed powder particles are poured into a mold and compacted with high force so that they maintain their shape before the subsequent Sintering process. The measurement data is given in millimeters. Since manual inspection of samples is time and resource consuming, the amount of samples that can be analyzed is low. Low samples sizes mean that we can not extract useful information by directly inputting the data into a feature-extraction ANN. With a limited amount of samples, any probability assumption about the product quality is not appropriate. In addition, out of the limited sample set, even fewer samples violate quality limits, creating an imbalanced dataset. Therefore, the research gap we address here is how to predict the continuous change in product quality with a small, imbalanced dataset, without any underlying assumption of the probability distribution.

We do so by:

- Showing that a ANN structured in an auto-encoding topology can provide a rich latent space representation of powder metallurgy process data automatically. By auto-extracting features, the algorithm predicts the quality of the products better than the existing approaches, even with a small data sample. The auto-encoding network enables us to reduce the dimensionality of the data in a latent space representation. Synthetic Minority Over-sampling Technique (SMOTE) method is used to mitigate data imbalance.

- Illustrating that our method can capture the latent probability distribution of the experimental data set. ANN, by its nature, has the ability of learning the probability distribution of the provided data set. With fine-tuned hyper-parameters and well-designed training process, we demonstrate that our method can do so (Figure 3). Our neural network not only can replicate the experimental data points, but also can generate new points, which fit the same distribution pattern. It is worth to note that our experimental dataset, by its nature, contains noise, hence even though the predicted points are not exactly the same as the actual ones (i.e., few predicted samples violate the control limits), from a distribution point of view, we can claim that the proposed method is capable of capturing the probability distribution of the provided dataset. This claim is guaranteed by the training algorithm of ANN, i.e., addition of a regularization item to the cost function of the ANN, e.g., formula (6).

  Besides providing a reasonable degree of accuracy in predicting next step quality character, with an average prediction accuracy close to 80%, the probability distribution of the predicted data covers a wider range than the original data. With a wider-ranged probability distribution, our method can overcome the noisy parts of the original data set randomly generated at products sampling phase.

As a result, our method can effectively capture features of defective products and is at least 10% better at predicting unqualified sample products by analyzing historical data.

The paper is organized as follows: In section 2, we present the state-of-the-art predictive quality control methods. In section 3, we introduce the data preprocessing steps, and explain the auto-encoder method developed. In Section 4, we test our method on a case example from the powder metallurgy industry and compare performance with the existing approaches. Section 5 concludes our work.

2 Literature Review

Predictive quality control is an important issue for customer satisfaction in the manufacturing sector [23] and non-conforming products create a waste of time and production resources [24].

The roots of SPC as a quality control mechanism go back to the 1920s, when W.A. Shewhart first proposed the use of control charts [25], to detect out-of-control signals indicating the existence of special causes that affect process stability by providing a visual indication of the behavior of critical quality variables. Common patterns in control charts facilitate the detection of a set of possible causes - for example, shift patterns may indicate changes in material, machine, or operator, while trend patterns may indicate tool wear. A depth survey of control chart recognition literature can be found in [22].

In the last decade, researchers have argued that machine learning based approaches, in particular, ANNs could produce more accurate results and thus numerous studies have been developed [26], for instance, Monostori and Lee [27, 28] mention that machine learning should be adopted as one of the critical tools to upgrade the intelligence level of factory machines. Wuest et al. [14] introduced a more practical application scenario and showed that a supervised learning algorithm performed efficiently on product sate data. Recent achievement in computer vision techniques are also extracting for manufacturing e.g., Blasco et al. [29] successfully deployed an image processing algorithm for efficiently controlling the products quality. In what follows we first briefly go through the classical machine learning based approaches and argue their advantages and disadvantages, then focus on ANNs based methods.

Melhem et al., [30] introduced an approach for wafer quality prediction by adopting the historical data of health indicator. Melhem et al. suggested to use principal component analysis (PCA) to extract significant indicators first, and then applied the indicator as labels to the training process in Bayesian networks, regression models, neural networks, SVMs, K-Near Neighborhood (KNN) and other clustering methods. The proposed approach demonstrated a reasonable performance however it was noted that the indicator extraction method would not generalize to multiple applications. Chou et al., [31] successfully combined genetic algorithm with SVMs, to predict the wafer quality. Arif et al. [32], suggested a mixture PCA method with an Iterative Dichotomiser algorithm (ID3), for multi-stage quality prediction, while Kao et al., [33] adopted classification and rule mining approach for solving a similar problem. Diao et al. [34], introduced another analytical tool called an improved dominant factor (DFs), which is based on an improved principle component analysis (iPCA), for dynamic quality control. The shortcoming of the proposed method is similar to that of PCA-based techniques. Essentially, PCA is a linear map [35] which is not suitable for handling the uncertainty of our dataset.

Model-based approaches are also popular [36]. Rajagopal and Castillo [37] proposed a Bayesian method that is able to set a tolerance limit on one or more responses to provide a given probability of conformance, which was used to determine the optimal setting of the control factors. A novel structure for Bayesian network was proposed to monitor and predict products quality of wafers during the manufacturing process [38]. Although model-based techniques may provide good performance depending on the application, they are designed to solve specific problems and require prior knowledge on the system under test. Thus the analysis and modeling processes are not entirely automated. There are two aspects we would like to emphasize in our approach regarding the automation property of ANN-based solutions. First, compared with model-based techniques, ANN-based methods are essentially data-driven, in which the domain knowledge is gained through ANN training processes and stored in the form of neural weights. Secondly, the advanced training algorithms enable neural networks to be self-updating in the online learning fashion and automatically detect the dynamical changes of systems. In order to capture the dynamical changes, the self-updating conducts the following works including re-searching proper activation functions and updating the topology and neural weights for the ANN.

Neural network-based methods for quality prediction have been especially popular, due to their ability in modeling non-linear dependencies. For example, neural network-based methods are used for dynamic quality prediction in plastic injection processes [39, 40, 41, 42]. The differences among these works are the topology of neural structures, which is dependent on sampling data. Chen et.al [39] proposed a back-propagation neural network based self organized mapping method, sun et.al [40] used a three-layer feedforward neural network, Iniesta et.al [41] suggested to recurrent neural networks, and Taghizadeh et.al [42] proposed to use a four-layer feed-forward neural network. Under various experimental environment and the specific application scenarios, these ANNs-based have been approved through their superior performance of production quality prediction. Semiconductor industry is another interesting filed that ANN based solutions can play a key role.For example, Hasani et.al [43, 44] applied different recurrent neural networks for speeding up the production quality validation of wafers. The simulation results demonstrated that ANN-based production quality control method can reduce significant amount of examination time. The general procedure of developing an ANN for quality prediction involves 1) synthesizing data at the initial step, 2) extracting useful information from raw data and presenting extracted information as new input-features of a neural network, and 3) choosing a specific neural network structure based on practical application.

The way users apply ANNs for solving quality control issues can be classified into two categories: The ANN is simply applied for implementing quality control without input feature processing. For example, Pugh [45] might be the first researcher to use Artificial Neural Networks (ANNs) for detecting mean shifts of control charts [42]. A self-organization map by using the back-propagation learning technique for neural network was tested to predict product quality within a plastic injection process [39]. The main setback in these works is that only the utilized shallow feed forward neural network cannot be adapted for extracting all the necessary features in similar problems. Although, users may input the neural network with handcrafted features while such methods lack generalization and automation.

The other method used with ANNs are feature-based techniques [46], which extract input-features automatically. For example, with two-step feature learning, Bai et al., [47] used a regression neural network to assess the product quality. Two-step feature learning methods such as the one proposed in [47], eases the pre-processing stages by automatically extracting features from raw sensory data. These features are then learned by a second neural network to provide a prediction. Hasani et al., [44] designed a compositional neural-network for monitoring and predicting the product quality of complex analog circuits. Other methods, with significant improvement of modeling performance, included wavelet-ANN [48, 49], statistical features-ANN [50, 10] and deep neural network [51, 1].

A notable advantage of neural network-based methods is their automated learning process. However, it is worth noting that learning might be difficult in when not having enough data and imbalanced data. For any of these cases, it has been noted that neural networks were harder to train and it was harder to achieve satisfactory predictive outcomes [52]. In our case study, we face both issues: the way the quality control process is currently implemented prevents from collecting more than few samples from each batch of products in every sampling operation. Secondly, control limit violation situations are quite rare.

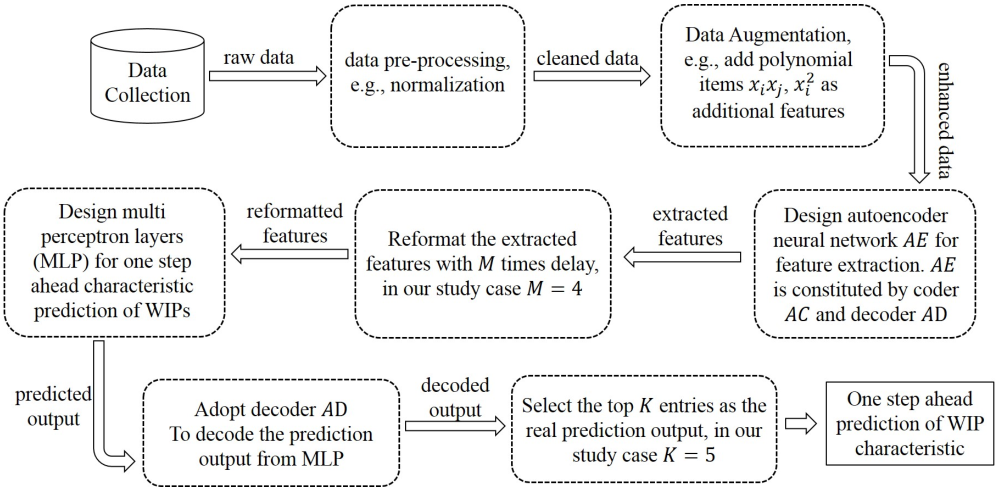

Data Collection
raw data
data pre-processing, e.g., normalization
cleaned data
Data Augmentation, e.g., add polynomial items $x_i x_j$, x_i^2 as additional features
enhanced data
Design autoencoder neural network AE for feature extraction. AE is constituted by coder AC and decoder AD
extracted features
Reformat the extracted features with M times delay, in our study case $M = 4$
reformatted features
Design multi perceptron layers (MLP) for one step ahead characteristic prediction of WIPs
predicted output
Adopt decoder AD To decode the prediction output from MLP
decoded output
Select the top K entries as the real prediction output, in our study case $K = 5$
One step ahead prediction of WIP characteristic

Figure 1: Model construction in quality characteristic prediction of WIP products.

To tackle this issue, we propose a generative neural network model for predicting the quality evolving trend of WIPs. A graphical abstract of how to apply the proposed method is depicted in Figure 1. In specific, we first adopt the data balancing technique called SMOTE [53], then an autoencoding neural network architecture prepares informative features from rebalanced data. We then apply a feature based technique, namely the autoencoder neural network with a time-delayed multi-layer perception. The last step is decoding the coded prediction. In summary, the contribution of our work includes using the SMOTE methodology for data imbalance, and conducting the prediction on auto-coded data instead of using normalized inputs, as auto-encoding can handle uncertainty and noise from original data set well and is thus more promising for handling small sample sizes. Next, we postulate our methodology.

3 The Generative Neural Network Quality Prediction Method

The main objective of our model is to predict the WIP product quality at time t, based on the measurement of the product status at time $t - 1$, ..., 1. We only consider one specific quality characteristic since in our industrial domain of application if any one of the attributes of the product is out of the control-chart boundaries, the product is considered disqualified. The quality characteristic used refers to a certain dimension such as the diameter or length of a WIP product.

3.1 Pre-processing methodologies

There are three pre-processing stages before data was fed into a learning system including 1) interweaved input data extension, 2) data-class balancing, and 3) normalization.

3.1.1 Interweaved input data extension

At each stage of product monitoring process, a small sub-sample of the products are randomly selected to undergo quality assessment. The ratio of sample to population size in our problem domain is 5:2000. With such a limited number of products the neural network cannot decode a reasonable global behavior of the products since it struggles with over-fitting issues. To tackle this problem, we interweaved multiplicative input samples, T x 6 matrix, and enriched the data set with more hidden information, as follows. Suppose matrix X is the original data without additional labels:

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} & x_{1,5} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} & x_{2,5} \\ \dots & & & & \\ x_{t,1} & x_{t,2} & x_{t,3} & x_{t,4} & x_{t,5} \end{bmatrix} \tag{1}$$

where t is the time at which we sample within the specific product quality monitoring period, then the left-hand-side pairs shown in the below vector:

$$[x_{1,7}, ..., x_{1,20}] = [\ x_{1,1} \cdot x_{1,1} \quad x_{1,1} \cdot x_{1,2} \quad \dots \quad x_{1,1} \cdot x_{1,5}\ ] \tag{2}$$

stands for new augmented data to be incorporated to train the neural network. Equation 2, illustrates how we extended the first row of the status matrix X. The operations follow for all rows in a similar fashion respectively. In this way we extend the product status matrix to a T x 21 with the last column corresponding to the labels of the sample.

3.1.2 Data-class balancing

When sampling products to perform quality control, it is usually the case that the number of positive and negative samples is not uniformly distributed in the dataset. Here, a negative sample indicates a product within control limits, and the positive sample indicates a product that violates control limits. In our four study cases, the imbalance ratio of negative/positive samples is $66.82, 35.46, 5.83$ and 32.19, respectively. This imbalance weakens the generalizability of a data-driven approach. To tackle this, we adopted SMOTE [53], to re-balance the collected dataset.

The SMOTE technique is an over-sampling method in which the minority class is over-sampled by generating new (synthetic) samples instead of simply over-sampling with replacement, which is done by introducing synthetic samples along the line segments joining any / all of the k minority class nearest neighbors [53]. Here, k is the number of synthetic samples the user wants to generate for re-balancing the data set. The value of k is usually defined by the ratio of majority/minority classes. In order to clearly show the idea of SMOTE, a simple case is presented in the pseudo (Algorithm 1).

```
Data: Samples from Minority Class
Result: Generate some samples which belong to the minority class $X_{generated}$
initialization: $k$ - identify nearest neighbors based on data imbalance ratio;
while $i \leq k$ do
    Consider a sample $(x_1, x_2, x_3, x_4, x_5)$ and let $(n_1, n_2, n_3, n_4, n_5)$ be one of its nearest neighbors ;
    $X_{generated} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} + \begin{bmatrix} rand(0,1)_1 \\ rand(0,1)_2 \\ rand(0,1)_3 \\ rand(0,1)_4 \\ rand(0,1)_5 \end{bmatrix} \cdot \begin{bmatrix} x_1 - n_1 \\ x_2 - n_2 \\ x_3 - n_3 \\ x_4 - n_4 \\ x_5 - n_5 \end{bmatrix}$
end
```

Algorithm 1: A simple example of generating synthetic samples by SMOTE.

According to the theory of SMOTE, we re-balanced our experimental data sets by following steps: First, each row of the matrix of products status is plotted as a point in a 5-dimension space. Then, SMOTE calculates the distance between every point and its neighbors. Based on the imbalance degree of the dataset, one can decide how many points SMOTE should generate artificially to re-balance the data. For instance, if there are three products within control limits and one product that violates it, SMOTE augments the dataset to find two more positive samples around the region to re-balance the dataset.

Data imbalance issue is the common barrier in data analysis [54]. Bootstrapping as one of the possible solutions has been applied widely in many scenarios. The core idea of bootstrap is resampling examples from the provided data set. Hence, in the augmented dataset, there are many replicated samples, which show that this type of solutions is not effective [53]. On the other hand, SMOTE generates new samples by considering the similarity between the generated data point and the original sample. From above-presented example, it is clear that SMOTE generates new samples in the range of the distance between a specific sample and its neighbors.

Note that such samples are utilized during the training to increase the generalisation of the method but have no contribution to the testing environment.

3.1.3 Normalization

The final step for data preprocessing is resetting outliers and normalizing all feature values into the range [0, 1] by a normalization postulated in Equation (3).

$$Y = \frac{(Y_{max} - Y_{min}) \cdot (X - X_{min})}{(X_{max} - X_{min}) + Y_{min}} \tag{3}$$

here, Y_{max} and Y_{min} are user defined upper bound (defined as 1) and lower bound (defined as 0), respectively. X_{max} and X_{min} are the maximum and minimum values of collected data. Regarding the aforementioned operation to reset outliers, our principle is to reset then slightly lower or greater than X_{min} or X_{max}, respectively.

The pre-processed data are then used to train neural network models, which are described as follows.

3.2 Stacked unsupervised and supervised learning systems

We initially designed an autoencoding neural network for automatically extracting meaningful information from the augmented X.

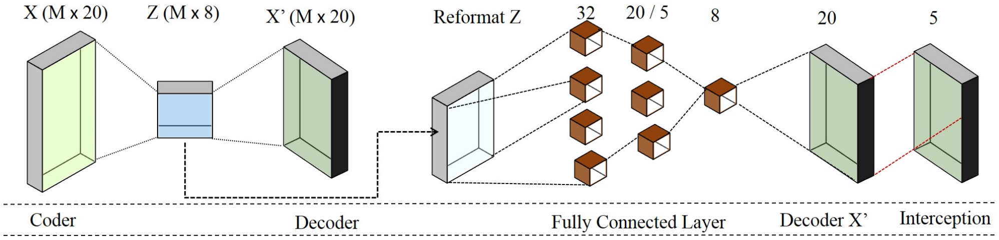


Figure 2: The abstract structure of our proposed generative neural network method for product quality prediction.

According to the technical pipeline (Figure 1) of our method, the collected WIP product status (X) is fed into a naive autoencoder. The trained autoencoder then learns how to code the inputs. With new inputs fed into autoencoder, the coded information need to be prepared to be input to the fully connected layer for the quality prediction. Based on trial and error, we found that WIP product status is strongly connected to the past 4 states. This is the reason for Z reformation after coding operation. After fully connected layers are adopted, the coded prediction is output. ANN-based methods are essentially data-driven techniques, i.e., the topology of neural networks are strongly relay on the variables associated with running environment. Hence, according to the data characters of experiments S1, S2, S3, and S4, we design two different neural network structures for the fully connected layer, i.e., the size of hidden layer for experiments S1, S2 and S4 is 20, respectively. The corresponding hidden layer size of experiment S4 is 5. The trained Decoder X' is then applied again for decoding the prediction results. The "Interception" simply copies the outputs from Decoder X' and selects the top 5 entries as the values of predicted values of the feature (i.e., the quality characteristic).

The autoencoder maps the augmented X to a latent space representation. Then, a simple multi-layer perceptron (MLP) is applied to the latent coded data to predict the quality of a product in a next step transition. To decrypt the coded prediction, we feed them into a decoding layer to obtain a rescaled output, which falls into the range of [0, 1]. We then make use of Equation 3 and the predefined control limits to get the predicted values back to real feature (i.e., quality characteristic) values. The actual samples from the augmented X are picked up at the last decoding layer, as the final predicted values of quality characteristic (Figure 2).

In the following, we describe the structure of each neural network component. The autoencoder [55] provides an automatic way to learn a low dimensional latent representation for input features by reproducing inputs to at their output.

Accordingly, one can define $x \in \mathbb{R}^{D_x}$, which is a D_x dimension input to the autoencoder; The encoder initially maps x to a lower dimension vector $z \in \mathbb{R}^{D}$, and generates an estimation of x, $\hat{x} \in \mathbb{R}^{D}$:

$$z = f(Wx + b_1);\;\; \hat{x} = f(W^T z + b_2), \tag{4}$$

where f is a sigmoid function, $W \in \mathbb{R}^{D_x \times D}$ is the weight matrix, and vectors $b_1 \in \mathbb{R}^{\mathbb{D}}$, $b_2 \in \mathbb{R}^{D_x}$ represent the bias values.

Many advanced ANN training algorithms have been proposed recently, e.g., Adam [56], Adagrad [57]. These training algorithms are mainly designed for speeding up the training process of complex data sets with highly non-convex cost functions, which require significant amount of computational resources in terms of both power and memory space. While most of our experimental data points fall into a small predefined range in the means of a SPC chart. Hence, our cost function is smoothly evolved and the values of the data points are intensive, which can be handled by a basic training algorithm. Therefore, from the practical application point of view, we adopted the scaled conjugate gradient (SCG) learning algorithm [58] for training the autoencoder. The cost function for the network is represented as C_1 (Equation 5) and is optimized by SCG subsequently:

$$C_1 = \underbrace{\frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{I}(x_{in} - \hat{x}_{in})^2}_{\text{mean squared error}} + \underbrace{\lambda R_{L2}}_{L_2\text{ regularization}} \tag{5}$$

The first item in C_1 represents the mean squared error (MSE). The second item is a L_2 regularization that is used for tackling overfitting issue, where λ is the regularization coefficient. L_2 regularization is computed as follows:

$$R_{L_2} = \frac{1}{2}\sum_{j}^{t}\sum_{i}^{k}(w_{ji})^2 \tag{6}$$

Where, n and k are the number of samples and the number of features in the training data, respectively.

The pre-trained autoencoder then provides rich features for given input data. The latent space is used to train the MLP. The cost function for the MLP is chosen to be the same as C_1, with a slight difference in the number of neurons and network layers. We name this objective function as C_2. We stack the autoencoder together with the MLP as a complete model. A fine-tuning training stage is then deployed on the overall network topology where the objective function is determined as the summation of the two as $C_{stacked} = C_1 + C_2$.

4 Experimental setting and results

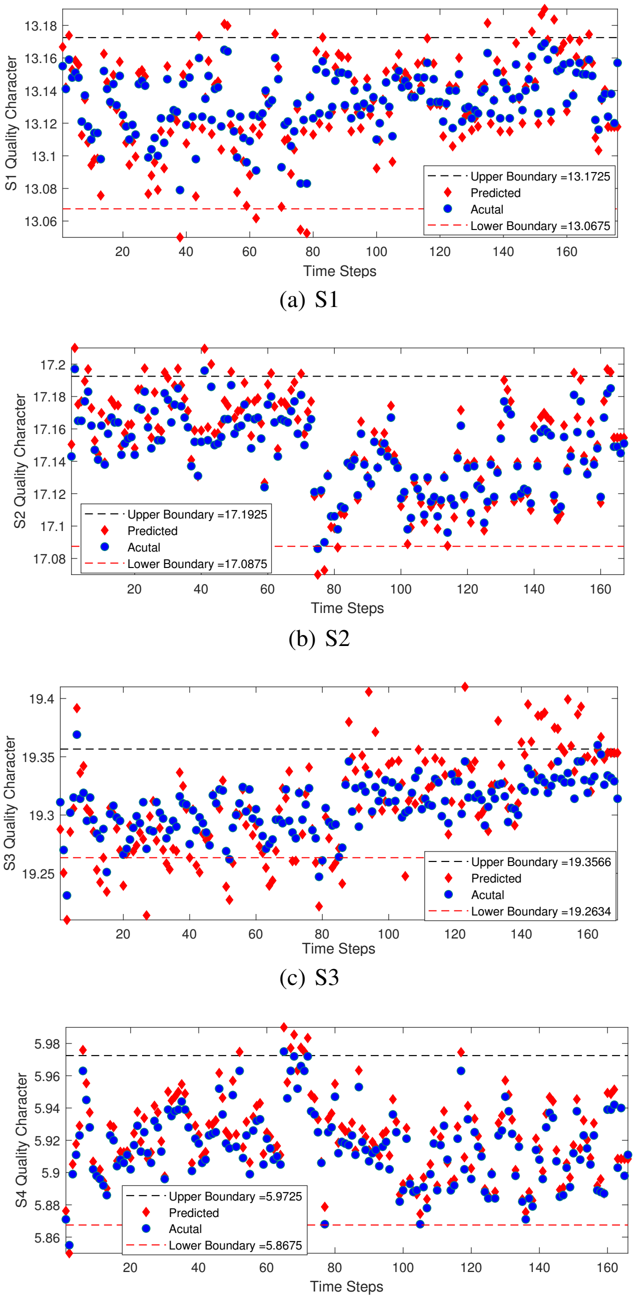
In this section, we evaluate the performance of our proposed method for quality prediction on a dataset collected from a powder metallurgy process. We initially describe the specification of the data before applying our method and assessing its performance. Finally, we compare our results with existing methods applied for solving similar problems. The process under query involves four steps, including powder blending (mixing), compaction, sintering and post-production processes [59]. Our dataset is from the compaction stage of a given product, where five samples are randomly selected out of a batch of 2000 products at each time step. A given product quality characteristic is measured for these five samples. Figure 4 presents the prediction generated by our method. At each time step, we calculate the statistical summary of our prediction and the corresponding original data point, i.e., minimum, first quartile, median, third quartile, and maximum. In order to validate the prediction accuracy of our method, the overlap between prediction and the actual data points summary is adopted. According to the plotted statistical boxes, a reasonable degree of accuracy over time is observed. "A reasonable degree" means that our prediction is not only able to cover the sampled but also some unseen data points, which is the reason why the boxes of predicted points (red) are always slightly bigger than the actual ones.

The specification of the experimental dataset is summarized in Table 1. In our experiments, we select four different groups of dataset for evaluating the performance of our method, i.e., experiment S1, S2, S3, and S4. The corresponding sub data sets come from four different production orders, i.e., O1, O2, 03 and O4, respectively.

4.1 Experimental Results

We implemented our method in MATLAB by using the Neural Network Toolbox. The training process is performed on a Microsoft Azure NC-Series virtual machine powered by one NVIDIA Tesla K80 GPU.

In order to train and test all four different orders, we designed the neural network topologies as depicted in Table 2. We performed the experiments with initially training the autoencoder to obtain a feature extractor. Before the training



(a) S1

(b) S2

(c) S3

(d) S4

Figure 3: Episodes of the algorithm's predictions in action. Red diamond is the predicted value, and blue circles are the actual value of the attribute of a product. The samples outside of the bounds are considered bad samples.

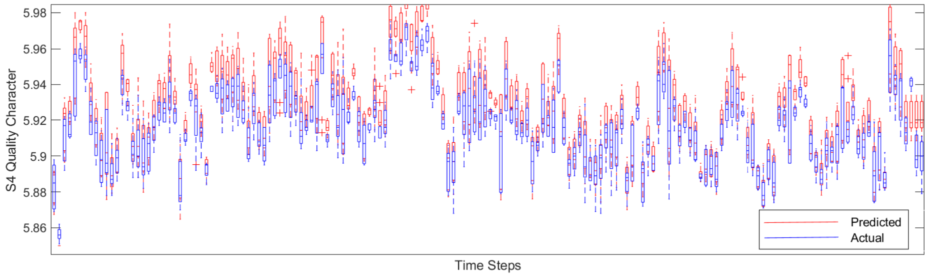


Figure 4: Box plot of the sample product quality together with prediction from our method for S4.

Table 1: Experiment data set of 4 different orders from FB1060 part. The data set was collected within 365 days.

Experiment	**# of Samples**	**Sample Size**	**Order #**
S1	1899	1×5	O1
S2	1896	1×5	O2
S3	1911	1×5	O3
S4	1892	1×5	O4

processes, we split the re-balanced dataset into 70% negative samples and 30% positive samples. 70% of the data in each class is used for training the autoencoder, 15% for the validation and 15% for the testing.

For the MLP, we reshaped the latent space of the autoencoder. We determined the intermediate products' status at time t, which has a strong correlation with the four time-lagged states of $t-1$, $t-2$, $t-3$, $t-4$. Then, the re-shaped inputs of the MLP are organized in the form of Z, as:

$$Z = \begin{bmatrix} z_{1,1} & z_{1,2} & z_{1,3} & z_{1,4} & z_{1,5} \\ z_{1,2} & z_{1,3} & z_{1,4} & z_{1,5} & z_{1,6} \\ \dots & & & & \end{bmatrix} \tag{7}$$

Where, $z_{i,j}$ is the coded output from the autoencoder shown in Figure 2. For instance, the autoencoder structure for experiment S1 is 20-8-20, i.e. the size of $z_{i,j}$ is 8. After reformatting the coded Z, the size of inputs of fully connected layers is 4×8.

Time-lagged inputs, together with the MLP form a time-delayed neural network architecture which incorporates the temporal dependencies in the data. The recurrent neural network RNN structure is deployed to overcome the vanishing and exploding gradient problems in the standard RNNs [60].

Our experimental results are shown in Figures 3(a) - 3(d), for four experiments. For all experimental settings, our method reasonably predicted the quality of the products. We observed that our algorithm predicts the samples that are outside or close to the specification boundaries, with higher degrees of sensitivity. The specification boundaries in all testbeds are delicately narrow. This is because high quality products are required and therefore, a sensitive prediction is demanding. Experiment S3 has higher number of miss-predictions due to higher level of noise when compared with other experiments. A different neural network topology is utilized for this case (32-5-10) to overcome this issue. Selecting the best ANN structure for a given dataset would be a largely a trial-error process, and would be out of the scope of the paper as there is no one-size-fits-all rule. There are some tools can help users automatically search the best neural network structure, for instance, Bayesian optimizer based toolkits, or Google published product called AutoML platform, or the deep learning community developed toolkit called hyperopt. Our experimental evaluations suggest that

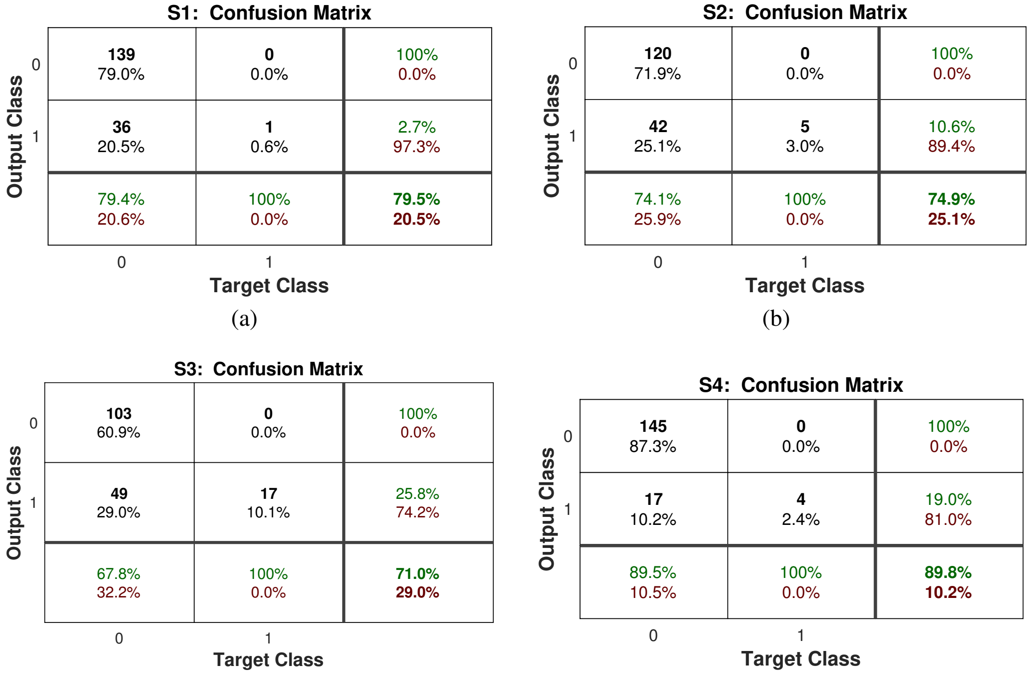
Table 2: Neural network structure for different orders. AE: autoencoder, FC: Fully Connected Layer. The activation function for both networks is a logistic sigmoid function.

Experiment	**Structure of AE**	**Structure of FC**
S1	20-8-20	32-20-8
S2	20-8-20	32-20-8
S3	20-8-20	32-5-20
S4	20-8-20	32-20-8

Table 3: Configuration of the compared methods in the experiments S_1, S_2, S_3 and S_4. γ is the mean values of the experimental dataset.

Compared Methods	Configuration
NN based	network structure: 20-32-32-19-5, activation function: SIGMOID
K-NN based	$K = 2$
SVM based	one class SVM, $\gamma = \frac{1}{2\sigma^2}$, $C = 2$

our method generalizes well even on samples that have never been seen during the training. We quantitatively measured the predictive classification performance of our method and illustrated it by the confusion plots shown in Figures 5(a), 5(b), 5(c) and 5(d). In each sub-figure, the prediction accuracy for positive and negative classes was computed. The absolute accuracy for these tests were calculated as follows: 79.5% for S1, 74.9% for S2, 71.0% for S3 and 89.8% for S4. More importantly, we observe a good performance for the method on capturing most of WIP products which violate control limits. In our case study, with four different orders, our method predicted aproximately 100% of samples that violated control limits.



(a)

(b)

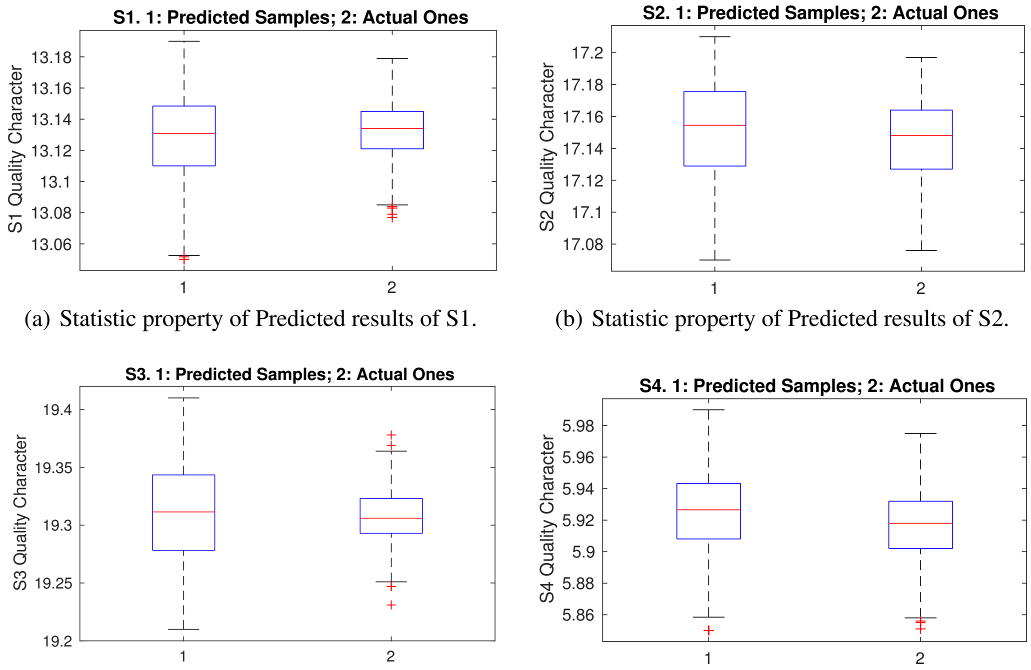
(c)

(d)

Figure 5: Confusion matrices for 4 testbeds, which includes positive class, negative class, prediction precision, recall rate and F-Measure.

Figure 4 represents a box plot for the experiment S4. We can empirically evaluate a reasonable degree of the prediction accuracy. To quantify the overall prediction performance for each testbed, we average the candle for the actual and the predicted points and demonstrate them in Figure 4.

Figure 4.1 visualizes the performance of our method in all the experiments. The prediction boxes show that our predictions essentially capture the majority of the desired outputs with few outliers. These prediction outliers may come from our policy of resetting the outliers from the original data set and the data generated by the data balance technique SMOTE. Additionally, the predicted boxes are larger than the boxes of the desired outputs due to the randomness of the selected samples.



(a) Statistic property of Predicted results of S1.

(b) Statistic property of Predicted results of S2.

(c) Statistic property of Predicted results of S3.

(d) Statistic property of Predicted results of S4.

Figure 6: Statistic analysis of prediction. The box plot shows the median, 25% quartile, 75% quartile and the range.

4.1.1 Comparison with existing methods

In this section, we compare our results with the existing methods for quality prediction. As mentioned previously, ANN-based approaches for quality control, especially for quality prediction, attracted a lot of research and application attentions [39]. K-NN [61] and SVM [31]. Therefore, in the comparison, we selected the NN, K-NN, and SVM based approaches as benchmark methods and compared their performance on the dataset with our method. The corresponding configuration of the compared methods is presented in Table 3. Table 4 represents a comparison between our results with existing solutions. As discussed, we manually added labels for every 5 samples which sampled at the same time, i.e., 0 for the ones all fall into the control range, otherwise, labeled as "1". Therefore, the prediction accuracy of our method is calculated based on the correct classification of different samples. Under such evaluation, the results demonstrate the superiority of the classification performance of our method in terms of identifying true positives. It worth noting that NN based method is better than our method in terms of "accuracy". However, in order to synthetically evaluate the experimental results, "PC rate (the accuracy of capturing defective products)" is another important factor need to be considered. Therefore, to balance the trade-off between accuracy and PC rate, we sacrifice the partial accuracy to make our model have a stronger capability of capturing defective WIPs. In addition our method is fully data-driven and automated. Table 4, shows that neural network-based methods can realize high degrees of accuracy. Feature extraction can presumably be a big issue for industrial data analysis. Hence, the modeling effort for the K-NN and SVM-based methods is critically limited. The auto-encoded approach can predict higher number of true positives, which is very important in quality control as this measure would indicate that it is more likely that detected control limit violation might not be a true control limit violation (false alarms), but there is a lower probability that the method will miss an occurrence of quality issues when they are present (missed disturbances), which is crucial for customer satisfaction. Additionally, by equipping our method with an automated feature extraction toolkit, we generalize better by predicting correctly 98% of products violating control limits.

5 Conclusion

In this paper, we introduce a generative neural network method for predicting WIP product quality. As currently reported machine learning methods in this domain suffer from manual feature extraction, and noise, our method is based on an unsupervised learning setting and a time-delayed feed-forward neural network. By conducting the prediction of product status on coded inputs, our generative neural network method can tackle such issues with the limited

Table 4: Quantitative and qualitative comparison of the performance amongst existing approaches and our method on powder metallurgy dataset of a quality character of intermediate products. NN, Simple Feed-Forward Neural Network. k-NN, k-near neighbors algorithm. SVM, Support Vector Machine. PC Rate, true negative rate, i.e., positive class rate for capturing bad products. ++: Highly satisfies. +: Satisfies. -: The attribute is not covered -+: The attribute is fairly covered.

Algorithm	Accuracy	PC Rate	Generalization	Automated	unsupervised
GNNQP	**78.8.0%**	**98.0%**	**++**	**+**	**++**
NN based	89.0%	55.6.0%	+	+	+
k-NN based	54.0%	89.3.0%	-	-	-+
SVM based	65.0%	31.0%	-	-	-+

amount of measurement data. According to the experimental results, reasonable predictive classification accuracy is achieved, and our method precisely captures the majority of the WIP products that violate control limits during the manufacturing process. Furthermore, autoencoder as an unsupervised feature extractor introduces an automated way for generating features from original data set without adding prior knowledge or presumption to a data set. We concluded the performance of our methodology over four different products using the same quality characteristic.

Additionally, we tried to balance the prediction performance between the general prediction accuracy and the accuracy of capturing the defective products. It is worth noting that the acceptance rate of misclassified products would be industry dependent. For our case, the misclassification rate is better than the currently deployed manual inspection method because of the trade off between labour time saved and unit value of misclassified items. However we are not claiming that our method is the final answer to industrial quality control - it is, rather, a study showing what is possible with the use of a state of the art automated ANN better methods that improve misclassification would be an important stream of research to be pursued. It is also important to note that the main optimisation target of the classification process should be the correct classification of “defected products”. This target means that actual defected products are captured at the expense of having more risk of misclassification in this category (precision). It would be far more damaging to classify a defective product as a good quality product than vice versa. When a product is classified as defected, it may not be automatically discarded but an operator may be alerted – hence business processes may be adjusted in the use of machine learning algorithms, until more confidence is built.

As part of our future work, we are planning to improve our method to process multiple product quality characteristics simultaneously, instead of only handling one specific characteristic at one time. Moreover, we plan to extend our method as a technical framework which could be applied to various industrial applications. Then efforts from the potential users should be devoted to adjusting the hyper-parameters of our method, e.g., depth of autoencoder and MLP, width of every neural layer, activation function. Hyper-parameters, could be adjusted either through manual trials or by adopting auto-machine learning techniques to automatically choose neural network structure that fits the best to a given dataset.

Acknowledgment

This work was partially supported by the Doctoral college - Cyber-Physical Production Systems project, funded by Technische Universität Wien, Vienna, Austria. The Co-author Guodong Wang finished part of the work during his academic visiting at University of Cambridge, UK, as a visiting PhD student with Dr. Alexandra Brintrup. Guodong Wang and Dr. Alexandra Brintrup conducted the experiments. Additionally, Dr. Anna Ledwoch, Ramin M. Hasani, and Prof. Dr. Radu Grosu were involved in the experimental results analysis.

References

[1] Tian Wang, Yang Chen, Meina Qiao, and Hichem Snoussi. A fast and robust convolutional neural network-based defect detection model in product quality control. *The International Journal of Advanced Manufacturing Technology*, 94(9-12):3465–3471, 2018.

[2] Douglas C Montgomery. *Introduction to statistical quality control*. John Wiley & Sons (New York), 2009.

[3] Thomas P Ryan. *Statistical methods for quality improvement*. John Wiley & Sons, 2011.

[4] Sotiris Bersimis, Stelios Psarakis, and John Panaretos. Multivariate statistical process control charts: an overview. *Quality and Reliability engineering international*, 23(5):517–543, 2007.

[5] Chi-Jie Lu, Yuehjen E Shao, and Po-Hsun Li. Mixture control chart patterns recognition using independent component analysis and support vector machine. *Neurocomputing*, 74(11):1908–1914, 2011.

[6] James D Williams, William H Woodall, and Jeffrey B Birch. Statistical monitoring of nonlinear product and process quality profiles. *Quality and Reliability Engineering International*, 23(8):925–941, 2007.

[7] Sofiane Achiche, Luc Baron, Marek Balazinski, and Mokhtar Benaoudia. Online prediction of pulp brightness using fuzzy logic models. *Engineering Applications of Artificial Intelligence*, 20(1):25–36, 2007.

[8] Lei Yang and Jay Lee. Bayesian belief network-based approach for diagnostics and prognostics of semiconductor manufacturing systems. *Robotics and Computer-Integrated Manufacturing*, 28(1):66–74, 2012.

[9] Yu-Chuan Su, Min-Hsiung Hung, Fan-Tien Cheng, and Yeh-Tung Chen. A processing quality prognostics scheme for plasma sputtering in tft-lcd manufacturing. *IEEE Transactions on semiconductor manufacturing*, 19(2):183–194, 2006.

[10] Guodong Wang and Radu Grosu. Milling-tool wear-condition prediction with statistic analysis and echo-state networks. In *Proceedings of S2M'16, the International Conference on Sustaniable Smart Manufacturing*, 2016.

[11] Jay Lee, Hung-An Kao, and Shanhu Yang. Service innovation and smart analytics for industry 4.0 and big data environment. *Procedia Cirp*, 16:3–8, 2014.

[12] Klas Meyer, Simon Kern, Nicolai Zientek, Gisela Guthausen, and Michael Maiwald. Process control with compact nmr. *TrAC Trends in Analytical Chemistry*, 83:39–52, 2016.

[13] Ni-Bin Chang, Kaixu Bai, and Chi-Farn Chen. Integrating multisensor satellite data merging and image reconstruction in support of machine learning for better water quality management. *Journal of environmental management*, 201:227–240, 2017.

[14] Thorsten Wuest, Christopher Irgens, and Klaus-Dieter Thoben. An approach to monitoring quality in manufacturing using supervised machine learning on product state data. *Journal of Intelligent Manufacturing*, 25(5):1167–1180, 2014.

[15] David Wang, Jun Liu, and Rajagopalan Srinivasan. Data-driven soft sensor approach for quality prediction in a refining process. *IEEE Transactions on Industrial Informatics*, 6(1):11–17, 2010.

[16] David Wang. Robust data-driven modeling approach for real-time final product quality prediction in batch process operation. *IEEE Transactions on Industrial Informatics*, 7(2):371–377, 2011.

[17] Siam Aumi, Brandon Corbett, Tracy Clarke-Pringle, and Prashant Mhaskar. Data-driven model predictive quality control of batch processes. *AIChE Journal*, 59(8):2852–2861, 2013.

[18] Jie Yu. Online quality prediction of nonlinear and non-gaussian chemical processes with shifting dynamics using finite mixture model based gaussian process regression approach. *Chemical Engineering Science*, 82:22–30, 2012.

[19] S Joe Qin. Survey on data-driven industrial process monitoring and diagnosis. *Annual reviews in control*, 36(2):220–234, 2012.

[20] Shen Yin, Guang Wang, and Huijun Gao. Data-driven process monitoring based on modified orthogonal projections to latent structures. *IEEE Transactions on Control Systems Technology*, 24(4):1480–1487, 2016.

[21] Ramin M Hasani, Guodong Wang, and Radu Grosu. An automated auto-encoder correlation-based health-monitoring and prognostic method for machine bearings. *arXiv preprint arXiv:1703.06272*, 2017.

[22] Wafik Hachicha and Ahmed Ghorbel. A survey of control-chart pattern-recognition literature (1991–2010) based on a new conceptual classification scheme. *Computers & Industrial Engineering*, 63(1):204–222, 2012.

[23] Marcello Colledani, Tullio Tolio, Anath Fischer, Benoit Iung, Gisela Lanza, Robert Schmitt, and József Váncza. Design and management of manufacturing systems for production quality. *CIRP Annals-Manufacturing Technology*, 63(2):773–796, 2014.

[24] Mélanie Noyel, Philippe Thomas, Patrick Charpentier, Andre Thomas, and Thomas Brault. Implantation of an on-line quality process monitoring. In *Industrial Engineering and Systems Management (IESM), Proceedings of 2013 International Conference on*, pages 1–6. IEEE, 2013.

[25] Walter Andrew Shewhart. *Economic control of quality of manufactured product*. ASQ Quality Press, 1931.

[26] Amitava Mitra. *Fundamentals of quality control and improvement*. John Wiley & Sons, 2016.

[27] László Monostori. Cyber-physical production systems: roots, expectations and r&d challenges. *Procedia Cirp*, 17:9–13, 2014.

[28] Jay Lee, Behrad Bagheri, and Hung-An Kao. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing Letters*, 3:18–23, 2015.

[29] José Blasco, Sandra Munera, Nuria Aleixos, Sergio Cubero, and Enrique Molto. Machine vision-based measurement systems for fruit and vegetable quality control in postharvest. In *Measurement, Modeling and Automation in Advanced Food Processing*, pages 71–91. Springer, 2017.

[30] Mariam Melhem, Bouchra Ananou, Mohand Djeziri, Mustapha Ouladsine, and Jacque Pinaton. Prediction of the wafer quality with respect to the production equipments data. *IFAC-PapersOnLine*, 48(21):78–84, 2015.

[31] Pao-Hua Chou, Menq-Jiun Wu, and Kuang-Ku Chen. Integrating support vector machine and genetic algorithm to implement dynamic wafer quality prediction system. *Expert Systems with Applications*, 37(6):4413–4424, 2010.

[32] Fahmi Arif, Nanna Suryana, and Burairah Hussin. A data mining approach for developing quality prediction model in multi-stage manufacturing. *International Journal of Computer Applications*, 69(22), 2013.

[33] Hung-An Kao, Yan-Shou Hsieh, Cheng-Hui Chen, and Jay Lee. Quality prediction modeling for multistage manufacturing based on classification and association rule mining. In *MATEC Web of Conferences*, volume 123, page 00029. EDP Sciences, 2017.

[34] Guangzhou Diao, Liping Zhao, and Yiyong Yao. A dynamic quality control approach by improving dominant factors based on improved principal component analysis. *International Journal of Production Research*, 53(14):4287–4303, 2015.

[35] Chris Chatfield. *Introduction to multivariate analysis*. Routledge, 2018.

[36] Libo Sun, Shunsheng Guo, Yibing Li, and Maw Maw Htay. Quality prediction model based on mechanical product gene. *Advanced Science, Engineering and Medicine*, 5(10):1090–1096, 2013.

[37] Ramkumar Rajagopal and Enrique Del Castillo. A bayesian method for robust tolerance control and parameter design. *IIE Transactions*, 38(8):685–697, 2006.

[38] Guodong Wang, Ramin M Hasani, Yungang Zhu, and Radu Grosu. A novel bayesian network-based fault prognostic method for semiconductor manufacturing process. In *Industrial Technology (ICIT), 2017 IEEE International Conference on*, pages 1450–1454. IEEE, 2017.

[39] Wen-Chin Chen, Pei-Hao Tai, Min-Wen Wang, Wei-Jaw Deng, and Chen-Tai Chen. A neural network-based approach for dynamic quality prediction in a plastic injection molding process. *Expert systems with Applications*, 35(3):843–849, 2008.

[40] Xiao Fang Sun, Peng Fei Zhu, Ying Jun Lu, and Hai Tian Pan. Prediction of quality index of injection-molded parts by using artificial neural networks. In *Advanced Materials Research*, volume 291, pages 432–439. Trans Tech Publ, 2011.

[41] A Alvarado-Iniesta, DJ Valles-Rosales, JL García-Alcaraz, and A Maldonado-Macias. A recurrent neural network for warpage prediction in injection molding. *Journal of applied research and technology*, 10(6):912–919, 2012.

[42] S Taghizadeh, A Özdemir, and O Uluer. Warpage prediction in plastic injection molded part using artificial neural network. *Iranian Journal of Science and Technology. Transactions of Mechanical Engineering*, 37(M2):149, 2013.

[43] Ramin M Hasani, Dieter Haerle, and Radu Grosu. Efficient modeling of complex analog integrated circuits using neural networks. In *Ph. D. Research in Microelectronics and Electronics (PRIME), 2016 12th Conference on*, pages 1–4. IEEE, 2016.

[44] Ramin M Hasani, Dieter Haerle, Christian F Baumgartner, Alessio R Lomuscio, and Radu Grosu. Compositional neural-network modeling of complex analog circuits. In *Neural Networks (IJCNN), 2017 International Joint Conference on*, pages 2235–2242. IEEE, 2017.

[45] G Allen Pugh. Synthetic neural networks for process control. *Computers & industrial engineering*, 17(1-4):24–26, 1989.

[46] Jian-bo Yu and Li-feng Xi. A neural network ensemble-based model for on-line monitoring and diagnosis of out-of-control signals in multivariate manufacturing processes. *Expert systems with applications*, 36(1):909–921, 2009.

[47] Yun Bai, Zhenzhong Sun, and Jun Deng. Manufacturing quality prediction based on two-step feature learning approach. In *Sensing, Diagnostics, Prognostics, and Control (SDPC), 2017 International Conference on*, pages 260–263. IEEE, 2017.

[48] M Shafafi Zenoozian and Sakamon Devahastin. Application of wavelet transform coupled with artificial neural network for predicting physicochemical properties of osmotically dehydrated pumpkin. *Journal of Food Engineering*, 90(2):219–227, 2009.

[49] Wenchuan Wang, Miao Zhang, and Xinggao Liu. Improved fruit fly optimization algorithm optimized wavelet neural network for statistical data modeling for industrial polypropylene melt index prediction. *Journal of Chemometrics*, 29(9):506–513, 2015.

[50] AB Johnston, LP Maguire, and TM McGinnity. Downstream performance prediction for a manufacturing system using neural networks and six-sigma improvement techniques. *Robotics and computer-integrated manufacturing*, 25(3):513–521, 2009.

[51] Peng Wang, Robert X Gao, and Ruqiang Yan. A deep learning-based approach to material removal rate prediction in polishing. *CIRP Annals*, 66(1):429–432, 2017.

[52] Andrew M Dai and Quoc V Le. Semi-supervised sequence learning. In *Advances in Neural Information Processing Systems*, pages 3079–3087, 2015.

[53] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[54] Guillaume Lemaître, Fernando Nogueira, and Christos K Aridas. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *The Journal of Machine Learning Research*, 18(1):559–563, 2017.

[55] Y-lan Boureau, Yann L Cun, et al. Sparse feature learning for deep belief networks. pages 1185–1192, 2008.

[56] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[57] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[58] Martin Fodslette Møller. A scaled conjugate gradient algorithm for fast supervised learning. *Neural networks*, 6(4):525–533, 1993.

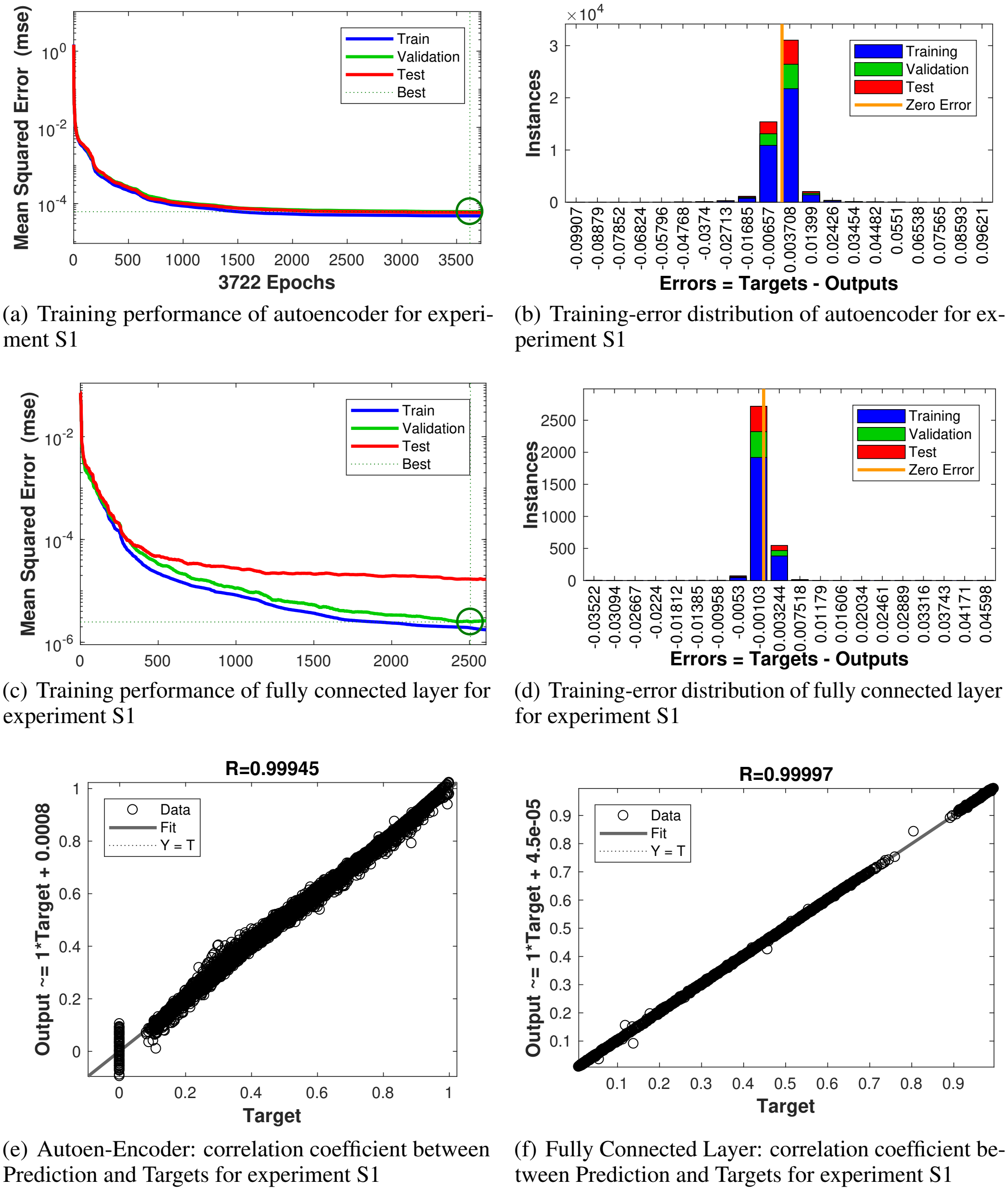
[59] Amir Arifin, Abu Bakar Sulong, Norhamidi Muhamad, Junaidi Syarif, and Mohd Ikram Ramli. Material processing of hydroxyapatite and titanium alloy (ha/ti) composite as implant materials using powder metallurgy: a review. *Materials & Design*, 55:165–175, 2014.

[60] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

[61] Ming Luo, Ying Zheng, and Shujie Liu. Data-based fault-tolerant control of the semiconductor manufacturing process based on k-nearest neighbor nonparametric regression. In *Intelligent Control and Automation (WCICA), 2012 10th World Congress on*, pages 3008–3012. IEEE, 2012.

A Training Status of Generative Neural Network

The same neural network structures are applied on four sub data sets, which includes the structure for autoencoder and the structure for multi perceptron (perceptron layer). The experimental results demonstrate the generic of our method. With same structure of generative neural network, we look into the training process and show how the neural networks get trained (Figures 7(a) - 10(f)). The first rows of Figure 7(a) - 10(f) are the training status of autoencoder. The second rows of of Figure 7(a) - 10(f) are the training status of fully connected layer. In the following figures, R is the correlation coefficient between our prediction and the targets.



(a) Training performance of autoencoder for experiment S1

(b) Training-error distribution of autoencoder for experiment S1

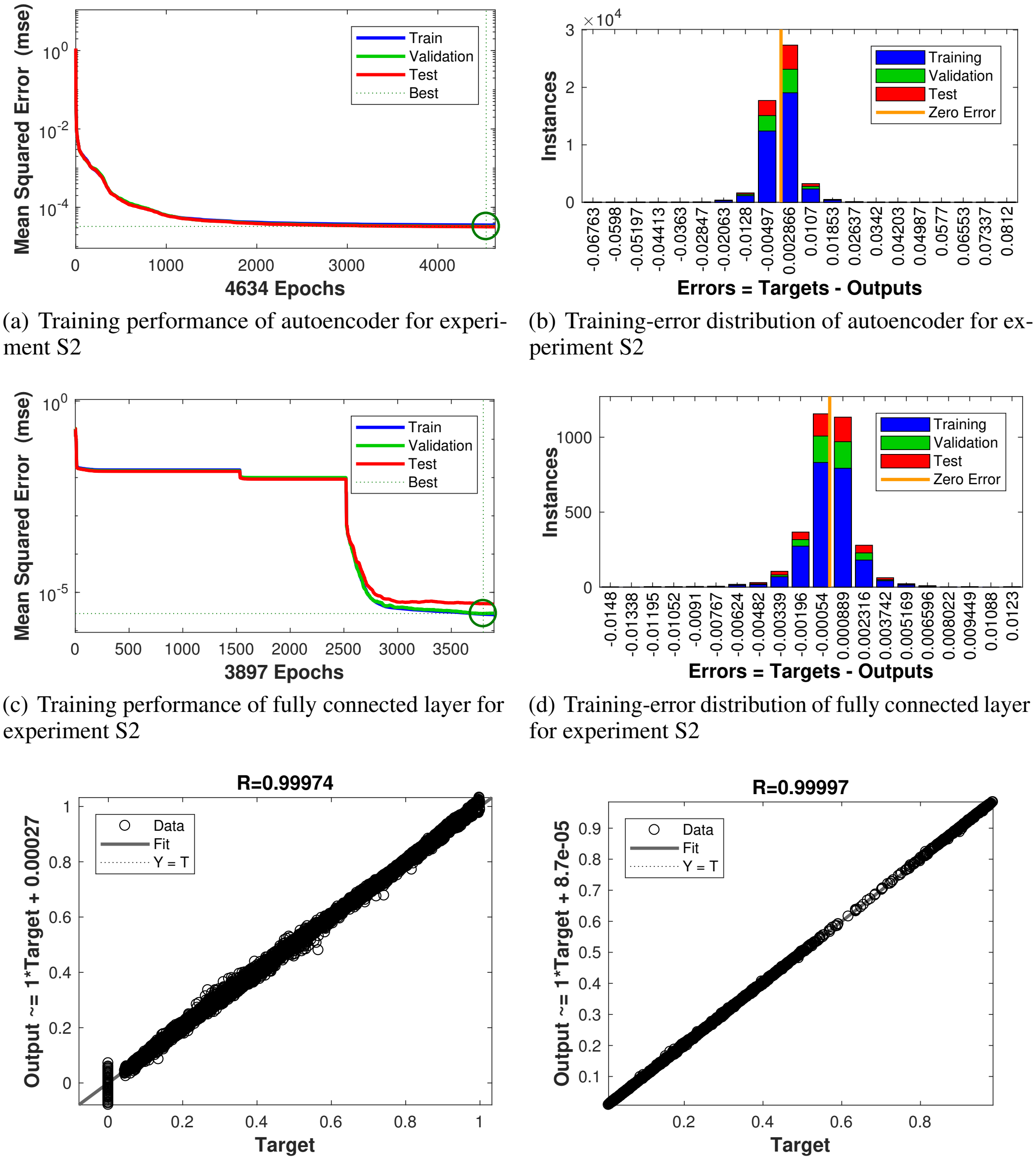
(c) Training performance of fully connected layer for experiment S1

(d) Training-error distribution of fully connected layer for experiment S1

(e) Autoen-Encoder: correlation coefficient between Prediction and Targets for experiment S1

(f) Fully Connected Layer: correlation coefficient between Prediction and Targets for experiment S1

Figure 7: Internal training status of GNNQP for S1. After 3500 and 2500 steps iterations, autoencoder and Perceptron layer smoothly reach stable points. The histogram figures show that the training, validation and testing errors (Mean-Squared-Error) of autoencoder and perceptron layer are mostly close to 0.0036 and 0.001, respectively. Last two regression analysis which is performed on coded targets and coded prediction. It shows the general training performance of our method on this data set.



(a) Training performance of autoencoder for experiment S2

(b) Training-error distribution of autoencoder for experiment S2

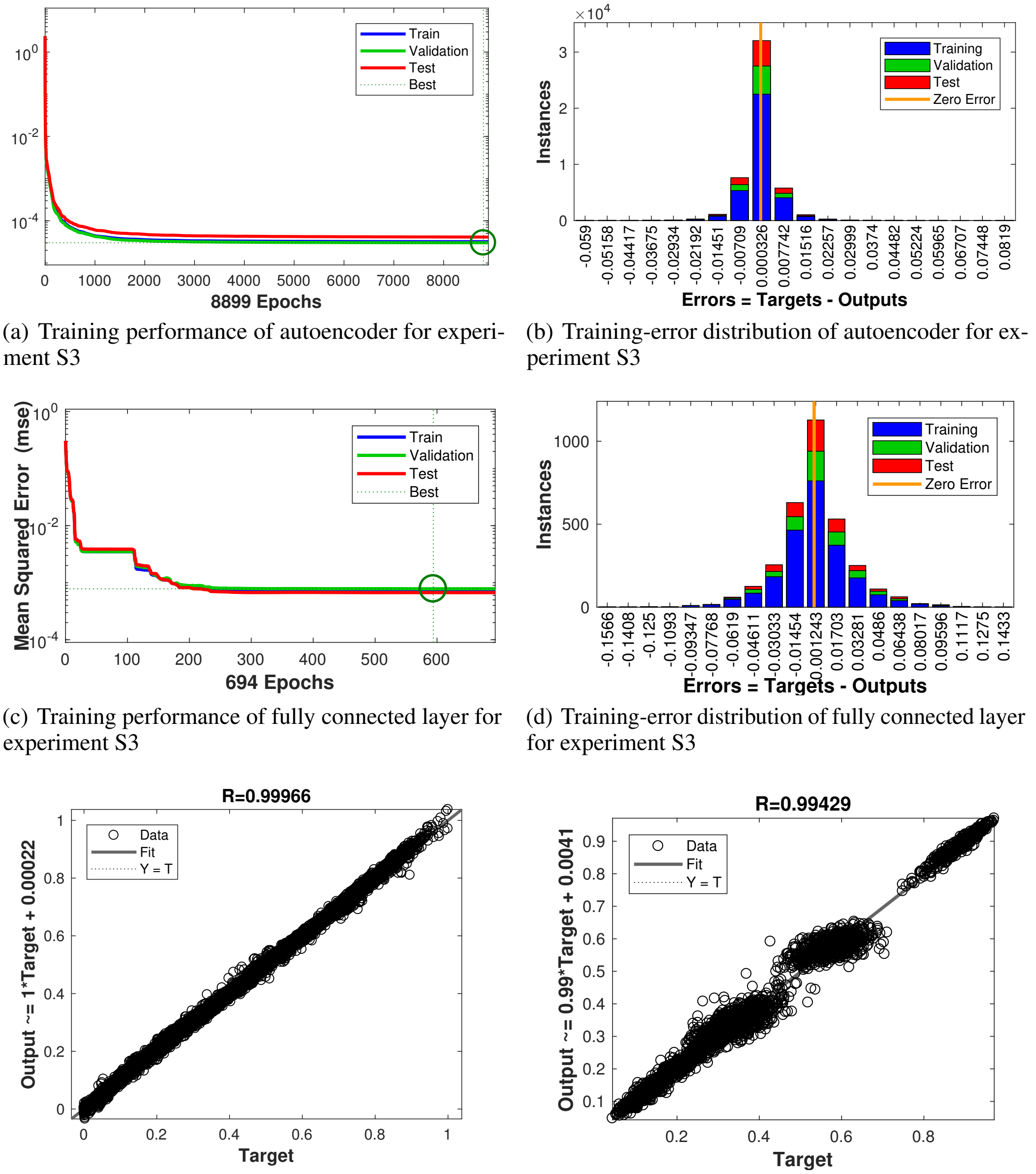
(c) Training performance of fully connected layer for experiment S2

(d) Training-error distribution of fully connected layer for experiment S2

(e) Autoen-Encoder: correlation coefficient between Prediction and Targets for experiment S2

(f) Fully Connected Layer: correlation coefficient between Prediction and Targets for experiment S2

Figure 8: Internal training status of GNNQP for S2. The training processes are stable after 4000 time iteration. It worth to note that the training performance of perceptron layer has stair-style training error reduction which is related to the initial values of neural network. The maximum training time also has effect on this issue. Theoretically, with more training iteration, the curve of training performance of perceptron layer should be similar as the autoencoder training status, which is reduced smoothly.



(a) Training performance of autoencoder for experiment S3

(b) Training-error distribution of autoencoder for experiment S3

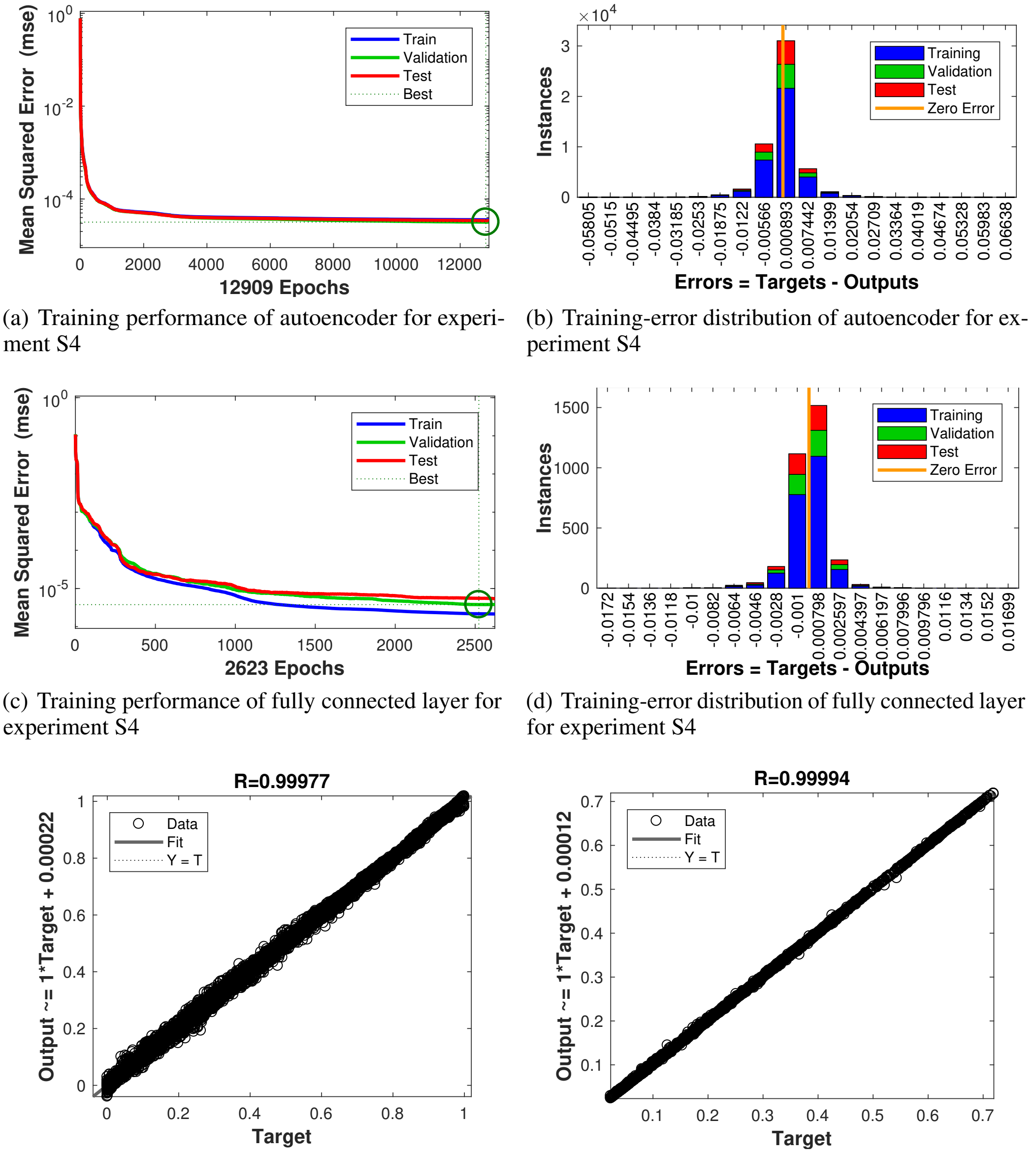
(c) Training performance of fully connected layer for experiment S3

(d) Training-error distribution of fully connected layer for experiment S3

(e) Autoen-Encoder: correlation coefficient between Prediction and Targets for experiment S3

(f) Fully Connected Layer: correlation coefficient between Prediction and Targets for experiment S3

Figure 9: Internal training status of GNNQP for S3. This sub data set is fully imbalanced. From the prediction results of coded outputs of perceptron layer, there are gaps between our prediction and targets. This is also reflected on the error histogram, where has amount of non-zero errors from perceptron layer.



(a) Training performance of autoencoder for experiment S4

(b) Training-error distribution of autoencoder for experiment S4

(c) Training performance of fully connected layer for experiment S4

(d) Training-error distribution of fully connected layer for experiment S4

(e) Autoen-Encoder: correlation coefficient between Prediction and Targets for experiment S4

(f) Fully Connected Layer: correlation coefficient between Prediction and Targets for experiment S4

Figure 10: Internal training status of GNNQP for S4. The training status shows the superiority of our method. GNNQP gradually reaches the stable points with 12000 and 2500 iteration for autoencoder and perceptron layer, respectively. Most of the errors close to 0.0003 and 0.0012. Regression analysis shows the high modeling performance of our method.