Computational Materials Science 218 (2023) 111984

Contents lists available at ScienceDirect

Computational Materials Science

journal homepage: www.elsevier.com/locate/commatsci

Full length article

Machine Learning in the development of Si-based anodes using Small-Angle X-ray Scattering for structural property analysis

Philipp Seitz [a], Christian Scherdel [b], Gudrun Reichenauer [b], Jan Schmitt [a,*]

[a] Institute Digital Engineering, University of Applied Sciences Würzburg-Schweinfurt, Ignaz-Schön-Straße 11, 97421 Schweinfurt, Germany
[b] Zentrum für Angewandte, Energieforschung Bayern, Würzburg, Germany

ARTICLE INFO

Dataset link: https://gitlab.vlab.fm.fhws.de/philipp.seitz/machinelearningandsaxs

Keywords:
Machine Learning
Neural Network
Autoencoder
Multi-layer perceptron
SAXS
Material development
Battery material

ABSTRACT

Material development processes are highly iterative and driven by the experience and intuition of the researcher. This can lead to time consuming procedures. Data-driven approaches such as Machine Learning can support decision processes with trained and validated models to predict certain output parameter. In a multifaceted process chain of material synthesis of electrochemical materials and their characterization, Machine Learning has a huge potential to shorten development processes. Based on this, the contribution presents a novel approach to utilize data derived from Small-Angle X-ray Scattering (SAXS) of SiO_2 matrix materials for battery anodes with Neural Networks. Here, we use SAXS as an intermediate, high-throughput method to characterize sol–gel based porous materials. A multi-step-method is presented where a Feed Forward Net is connected to a pretrained autoencoder to reliably map parameters of the material synthesis to the SAXS curve of the resulting material. In addition, a direct comparison shows that the prediction error of Neural Networks can be greatly reduced by training each output variable with a separate independent Neural Network.

1. Introduction

Machine Learning (ML) as a subarea of Artificial Intelligence (AI) is currently distributing in various application fields e.g. material science [1], robotic vision and perception, production control [2] or predictive maintenance. The reasons are, among others, increasing computational power and the availability of tools, which enable also non-experts to use ML-techniques for their application. At the same time, the world is accelerating its electromobility efforts, necessitating continuous improvement efforts in battery technology to achieve range, weight and cost targets.

Integrating Machine Learning techniques into various fields of battery technology describes a future-oriented data-driven approach, which can contribute to faster as well as more targeted development of battery cells as complex electrochemical material systems.

A comment in Nature Materials in October 2020 from Aykol et al. stated that data-driven Machine Learning approaches can contribute to a full exploration of the potential in the application and optimization of battery technologies. At the same time the authors mention that there is a need for experimental data to enable the power of data-driven tools. Here, accessible data with a transparency of experimental setup and conditions as well as a high sample size are the crucial factors [3].

In material development processes (MDP), the experience and intuition of the researcher influence the decision e.g. for the next iteration of synthesis parameter. This can lead to time consuming procedures. Data-driven approaches as Machine Learning can support these complex processes in the way to train models to predict outputs from certain steps in the MDP. In this field the utilization of small data-sets for model training compared to a huge solution space is challenging as well as the multifaceted process chain from material synthesis to the electrochemical characterization of the cell performance.

Based on this motivation the contribution wants to review the actual state of data-driven methods in battery technology briefly after a short introduction into material class synthesis and Small-Angle X-ray Scattering (SAXS) characterization [4]. Then, we present a novel approach to utilize data derived from SAXS of SiO_2 matrix materials for battery anodes with Neural Networks. Here, we use SAXS as an intermediate, high-throughput method to characterize sol–gel based porous materials within the synthesis process. The measured SAXS curves contain information of the particular structure of the material between 0.1 nm and 100 nm e.g. the porosity and pore size of the disordered sol–gel based matrix. The novelty of the presented approach is the utilization of SAXS to provide a fast fingerprint of structural properties of a huge amount of potential SiO_2 matrix materials and use the derived data to train Neural Networks.

The aim of the presented Machine Learning approach with Neural Networks is to predict SAXS curves of resulting materials from specific

* Corresponding author.
E-mail address: jan.schmitt@fhws.de (J. Schmitt).

https://doi.org/10.1016/j.commatsci.2022.111984
Received 12 May 2022; Received in revised form 19 September 2022; Accepted 12 December 2022
Available online 23 December 2022

synthesis parameters to reach pre-defined morphological properties for the first step in anode processing i.e. the provision of an optimized SiO_2 matrix for Si deposition in a later processing state. Doing this, a multi-step method is presented where at first the minimum number of hidden neurons of an autoencoder is found to reliably map the SAXS curve. Then a feed forward net is trained to predict the hidden neuron values by synthesis parameters as input variables. These values represent material property data as well as process data from the synthesis. This combination of separate Neural Networks is efficient in terms of computational power on the one side and makes it possible to forecast the SAXS curve by data from previous steps in the MDP on the other hand. With this, the traditional trial and error, experience-based principle can be replaced with a structured approach and support the development of innovative anode materials as the first of several complex sub-steps to the final battery. Furthermore the Neural Nets open up possibilities to detailed investigations in its trained output parameter space, e.g. can this be used to find specific input parameters which predict optimized outputs to find materials with ideal structural properties. In addition, a direct comparison of predicting higher dimensional output variables with a trained single Neural Net and a trained Net System which predicts each output variable by separately trained single Neural Nets will be made.

Due to this, the contribution is structured as follows: After introducing the sol–gel material class, its synthesis process as well as the SAXS characterization method (see 2, we give a brief literature review in Section 3 of Machine Learning approaches in battery technology and material development. Section 4 introduces the multi-step approach for the application of Neural Networks to predict SAXS curves from substance and process data is presented in detail. Section 5 shows results of this approach, which are critically discussed in Section 6. The paper closes with a conclusion and further research issues.

2. Material class, synthesis and SAXS characterization

2.1. Material class and synthesis

Sol–gel derived nanoporous silica materials (silica aerogels) are investigated since the work of Kistler in the 1930s [5]. High porosities of 50% up to > 95% with dominant mesoporosity (pore size 2–50 nm) are outstanding structural properties of this material class making them candidates for applications e.g. in thermal insulation, filtration, as adsorbers or electrodes [6].

The porous silicas for the current investigation of matrix materials with SAXS and ML were targeting an unusual combination in pore size and porosity, i.e. aiming for small mesopores and high porosities simultaneously. The silicas were synthesized by a 2-step process following the approach described in [7] and [8] - see also Fig. 1. The raw materials used are Tetraethoxysilane (TEOS) as silica source, ethanol as solvent, high-purity water for hydrolysis, hydrochloric acid and ammonia solution to adjust the pH. To provide a set of porous silica samples for the ML-approach and a relevant variation in structural properties, a Design of Experiments (DoE) with 7 factors (synthesis parameters) was set up. To cover a large range of parameter variations and reduce the overall amount of samples, an onion model with 50 variations was chosen instead of a full factorial design.

The combination with SAXS analytics and Machine Learning should give more insights to explore the structures of novel battery anode matrix materials, allowing a huge synthesis and process feature space to be screened effectively e.g. to propose recipe data in terms of an optimal pore size and pore volume.

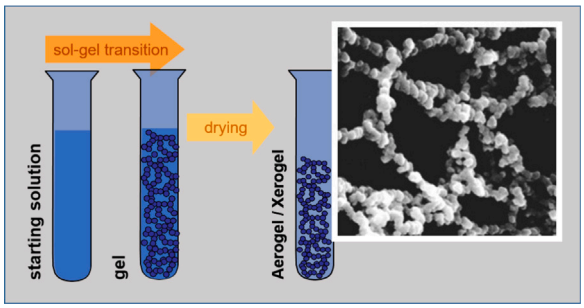

Fig. 1. Scheme of the sol–gel process. After removal of the pore liquid (drying) a 3-dimensional solid network with pores and particles in the nanometer range is obtained as shown in the TEM-image.

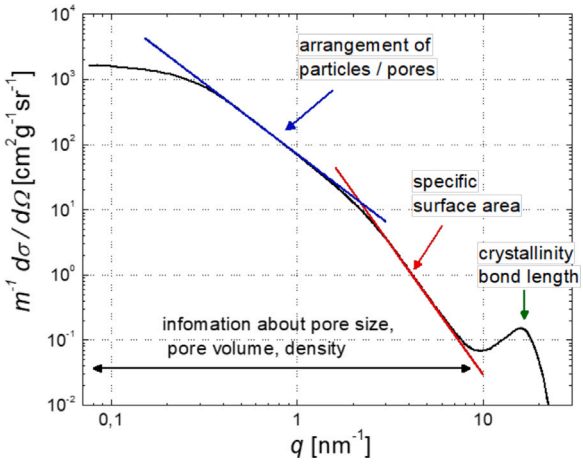

Fig. 2. Scattering curve of a typical sol–gel derived silica gel. The scattering pattern represents a fingerprint of the structural properties containing information about specific surface area, pore size, pore volume (density) and structural arrangement of atoms, pores and particles.

2.2. SAXS characterization

SAXS is a non-destructive and fast characterization method for structural analysis in the range of ca. 0.1 nm to 100 nm. The characterization was performed with a SAXSpoint 1.0 instrument from Anton Paar using $Cu\ K\alpha$ radiation (wavelength $1.54\,\AA$) at two sample detector distances of 109 and 562 mm. Analysis was performed on the dry gels after degassing to avoid artefacts from adsorbed species according to the recommendations given by Scherdel et al. [9]. The scattering intensity was normalized to the mass-specific scattering cross-section $m^{-1}d\sigma/d\Omega$ in units of $cm^2g^{-1}sr^{-1}$, using a glassy carbon reference with a well-known scattering cross-section as standard. Thus, the scattering curves (example in Fig. 2) can be compared without restrictions with each other and contain information about specific surface area, pore size, pore volume (density) and structural arrangement of atoms, pores and particles as well. In other words, the scattering curves provide a structural fingerprint of the porous silica materials under investigation.

Compared to gas adsorption analysis (e.g. for BET surface area and pore size), SAXS allows significantly shorter measurement times (minutes) [10], the analysis of closed porosity as well (inaccessible for the analysis gas) and does not change the sample properties (e.g. deformation, absorption) during characterization. Therefore, the combination of SAXS analytics and Machine Learning is a promising tool for a fast and reliable material development. This should give more insights to explore the structures of novel battery anode matrix materials, allowing a huge synthesis and process feature space to be screened effectively

e.g. to propose recipe data in terms of optimal structural properties (specific surface area, pore size, pore volume and combinations thereof).

3. Brief literature review

The rising number of process parameters in complex process chains and the requirement to develop new materials faster needs alternative approaches to speed up development processes and increase e.g. battery performance [11]. In this context many different ML algorithms have been applied in various fields of material science to

- find new materials or promising material combinations,
- classify materials or properties by recognizing patterns or
- predict structural or performance properties from data subsets.

The overall objective is the piecewise replacement of "experience-" and "intuition-based" decision-making with data-driven methods [12]. This has huge potential to shorten MDP. Oftentimes, analytical relationships cannot be applied due to the large number and interrelations of influencing factors, e.g. during the materials' synthesis and processing and its resulting physical properties. Hence, the challenge is to fit or interpret the results of (intermediate) characterization methods in terms of their structure-process-performance relationship. Here, ML can make a significant contribution [13]. Further, the development of nanoporous materials, as sol–gel materials, by digitization techniques, is strengthened by the statement of Schmidt et al. as Machine Learning as "one of the most exciting tools that have entered the material science toolbox" [14].

We propose the application of ML in battery technology by the utilization of SAXS characterization as a fast intermediate method to evaluate structural parameters during the synthesis process. Hence, the literature review briefly addresses both aspects, ML and SAXS in battery technology in the following two paragraphs.

3.1. Machine learning in battery technology

Research activities in the broad field of battery technology rise, especially through the emerging effort of the global electromobility. Similarly, AI technologies are called as enablers of innovation [15]. The utilization of ML, as a sub-domain of AI, for battery research is addressed in research but not yet widespread [3]. In Table 1 the related research is summarized according to the life-cycle phases and the general research issue and model quality indicator.

Kauwe et al. [16] present how Machine Learning tools can be exploited to predict the properties of battery materials. A data-set of cathode materials and various statistical models are used to predict the specific discharge capacity at 25 cycles. Nakayma et al. [17] introduce two efficient high-throughput computational approaches to explore materials for the ionic conductor of all-solid state batteries. They combine regression based ML and Bayesian optimization to evaluate numerous calculated crystalline structures. Also Sendek et al. do research in the field of data-driven discovery of solid Li-ion conducting materials [19]. A general overview on the application of ML in material discovery and design is given in [20] with an application example of battery monitoring. In [18] ML (ANN, RF) is used to predict the properties of the crystalline system which have a strong correlation to the physical properties. Takagishi et al. [21] make use of ANN and, using three-dimensional virtual structures to design battery electrodes.

The research of ML supported battery production is less represented. Jiang et al. [22] address the understanding of the influence of the electrochemical performance of battery particles' evolving (de)attachment with the conductive matrix. Here, the authors use ML for a deeper statistical analysis. The research group of Herrmann et al. [23] use ML to find interdependencies within the cell production steps. They find the relevant production features with DT and RF algorithms in relation to the resulting cell performance.

In the battery usage phase, Lucu et al. [24] develop a data-driven ageing model for Li-ion batteries with the Gaussian Process framework. Here, the ability of the Gaussian Process model is to learn from new data observations and thus to provide more accurate and confident predictions. Also Richardson et al. [25] use Gaussian Progression Regression (GPR) as a data-driven diagnostic technique. They estimate in-situ capacity by voltage measurements over short periods of galvanostatic operation. In [26] data from impedance spectroscopy as an intermediate method are used to build a GPR model to forecast the states of health, states of charge and battery temperatures. The GPR model takes the entire impedance spectrum as input, without further feature engineering, and automatically determines which spectral features predict degradation e.g. the capacity and remaining useful life. In [27] a Deep Feedforward Neural Networks (DNN) is used for battery state of charge (SOC) estimation, while the training data is generated in the lab by applying drive cycle loads at various ambient temperatures to a Li-ion battery so that the battery is exposed to variable dynamics. An adapted ANN algorithm is used by Hannan et al. [28] to predict SOC. The results show that the proposed method is accurate and robust, as it can accurately examine SOC under different operating conditions. The state of health (SOH) estimation with Support Vector Machines (SVM) as a nonlinear frequency response analysis is shown in [29]. The model has a strong correlation to Lithium-ion battery degradation at mid-frequency range from 1 Hz to 100 Hz.

An example of the application of ML in the recycling phase is given by Senthilselvi et al. [30]. CNN are used to classify images for metal recycling. The model has been trained by pure metal images, while it is tested by captures images.

The brief literature review shows the broad application of ML techniques in the field of battery technology. Especially the SOC and SOH prediction are a major field of research. The usage of data derived from intermediate methods as impedance spectroscopy or SAXS, as shown in this contribution, is less represented. However, the approach offers the possibility to actively manipulate the development process, as no data from already manufactured cells are used. Our contribution focus the first step of battery development as matrix materials determine the later performance of the battery due to its structural properties. Hence, ML is applied to one of the very early in the process chain of the MDP. Here, the SAXS method gives the opportunity for a fast and complete characterization of the relevant structures.

3.2. SAXS analytics for battery technology

As seen in the previous chapter ML is applied for battery technology in various fields. We will propose an approach, which utilizes data from SAXS curves to train a model by an ANN system to predict the SAXS curves from input parameters.

SAXS and ML has been used in several fields of natural science successfully. Chen et al. [31] used extreme gradient boosting (XGBoost) to predict key structural parameters from SAXS of biological macromolecules in the case of double-stranded ribonucleic acid (dsRNA) duplexes. Just like Franke et al. [32] trained a simple k-nearest neighbour method to classify biological macromolecules as compact, extended and flat and to extract information about their maximal particle diameter. [33] investigated which ML-methods where the most accurate ones to extract structural information of SAXS measurements from eleven predefined structural models for common nanostructures resulting in the prior used XGBoost by [31].

In the context of battery research SAXS has also been used to provide a deeper insight into the atomic and pore structures. Saurel et al. [34] focus on three groups of carbonaceous materials: nonporous, microporous and carbide-derived carbons. The SAXS results lead to an improved analytical model description compared to traditional ones. In [35] SAXS is used in-situ to monitor the structural changes of polymer nanocomposites upon heating. In [36] similar work is done with

Table 1
Clustered publications for ML in battery technology.

Source	Development	Production	Usage	Recycling	Material discovery	Processes	SOC/SOH estimation	Age modelling	Model quality	Abbreviations
[16]	X				KRR SVR RFR				$R^2 = 0,51$	KRR = Kernel Ridge Regression SVR = Supported Vector Regression RFR = Random Forrest Regression R^2 = Coefficient of determination
[17]	X				GPR				$R^2 > 0,8$	GPR = Gaussian Progression Regression
[18]	X				RF ANN				$Acc = 80\%$	RF = Random Forest ANN = Artificial Neural Network Acc = Accuracy
[19]	X				LR				$Acc = 50\%$	LR = Logistic Regression
[20]	X				MLFFS				$R^2 = 0,96$	MLFFS = Multi-Layer Filtering Feature Selection
[21]	X				ANN				$R^2 = 0,99$	./.
[22]		X				CNN			./.	CNN = Convolutional Neural Networks
[23]		X				DT RF			$R^2 = 0,75$	DT = Decision Tree
[24]			X					GPR	$RSME = 1,09$	RSME = Root-Mean-Squared Error
[25]			X				GPR		$RMSPE = 0,49$	RSPE = Root mean squared % error
[26]			X					GPR	$R^2 = 0,92$	./.
[27]			X				DNN		$MAE = 1,1\%$	DNN = Deep Neural Network MAE = Mean absolute error
[28]			X				RNARX		$RSME = 0,59$	RNARX = Recurrent nonlinear autoregression
[29]			X				SVM		$R^2 = 1$	SVM = Supporte Vector Machines
[30]				X	CNN				$Acc > 80\%$	./.

Na-ion battery materials. Berhaut et al. [37] study silicon/graphite-based composite anodes during the lithiation process. With SAXS they obtain the nanostructural variations of the silicon phase (SAXS) for a complete picture of the lithium repartition of the prelithiated silicon anode.

The combination of SAXS and ML is, to the best knowledge of the authors, a novel approach to optimize matrix material synthesis. Our multi-step method to design ANN for an optimal prediction of SAXS data and hence suggest optimized synthesis parameter is presented in detail in the next sections.

4. Multi-step method

In the following we present the general concept of the multi-step method in the use of ML and a straight forward approach to its construction. It is afterwards applied to the use case of predicting SAXS curves from given synthesis/process parameters.

4.1. General method

Looking for the best predictions, complex solutions arise by making adjustments to the training algorithms and specializations of the network structure. Kamarthi et al. [38] adapt the backpropagation algorithm by extrapolating weights to reduce training time. In [39] a low-rank factorization to decrease the number of network parameters and also save training time is developed. Yeganeh and Shadma [40] removes useless weights with a generic algorithm and Hunter et al. [41] investigates different network topologies, which also require different training algorithms (see [42]) due to their different basic structures, since the effective Marquardt–Levenberg algorithm can only be used for MLPs [41].

In the course of the presented application, we developed a straight forward implementable method to form a complex as well as fast trainable predictive construct using standard ML-methods. Due to the unknown degree of non-linearity of the parameter spaces we concentrate on using ANNs since they are not restricted to a specific degree of linearity [43,44]. It may be useful to subdivide a big ANN into multiple small ANNs respectively ML-methods depending on intermediate results and their best prediction method appearing in the application. This partitioning of ANNs leads to increasing intrinsic complexity with simultaneously significant reduction of training time. In this way, many different ML methods or ANN topologies can be examined in terms of their quality in a short time and an optimal internal structure can thus be found.

In the case of ANN, the multi-step method also opens up possibilities to derive structure to their inherent black box property. By dividing the respective process steps into individual independent networks, clear tasks are defined and in the event of a problem it is possible to intervene specifically in individual regions without influencing the other networks of the overall structure.

According to [45] it leads to better results if a separate net is trained for each output variable to be predicted instead of using a single net that maps all output variables simultaneously. In this way, for each available intermediate step, not only individual nets but entire independent net-systems are created, which have the same input and jointly generate the input for the adjacent net-system or ultimately the total output of the overall system, which can be a net too. Fig. 3 shows the scheme of the basic idea. The input variables flow into each of the adjacent nets, which perform their own processing and collectively form the input for the next net-system in the chain. Each net can produce one or more output variables.

4.2. General design

A systematic approach is required to design a functional overall net structure. It may be that necessary training values at network junctions can only be determined during setup, as they are generated as input or output by adjacent trained networks. Therefore, it is necessary to

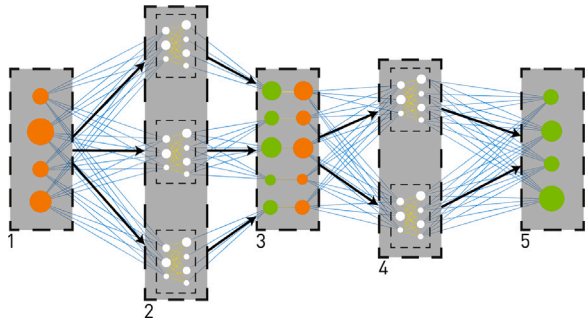

Fig. 3. General scheme of the multi-step method. Each intermediate step for which usable data is available in the process chain can be represented by several nets (e.g. from **1** to **3** via ML-System **2**). As with a single FFN, the pure input variables (**1**) flow into the system and ultimately lead to the sought final output (**5**) via apparent intermediate results (**3**).

specify the type, distribution and task of the respective nodes of the network in advance. In the following, the individual steps to reliably achieve a structured net are presented in general (see Fig. 4):

1. **Define Procedure**
   Define all potential nodes, their internal ML method, connections to neighbouring nodes, and conditional training order.
2. **Train and Optimize**
   Train and validate the nodes according to the appropriate method in the previously defined order and optionally optimize their topology.
3. **Merge**
   Link inputs and outputs of all nodes in their intended order.
4. **Predict**
   Once all the individual nodes have been trained, optimized and fully interconnected, the prediction of sought output variables is now possible.

4.3. Defining procedure in SAXS use-case

In our use-case of finding correlations between synthesis/process parameters and the resulting SAXS curve, the division into two intermediate Neural Network steps was meaningful. On the one hand we choose an autoencoder, which can represent the searched SAXS curve based on a few main features. On the other hand, a network system of standard Feed Forward Nets (FFN) should map the available input parameters as output to the hidden neurons of the mentioned autoencoder.

A fundamental statement about Neural Networks is the universal approximation approach. It states that any continuous function can be approximated arbitrarily close by an FFN with exactly one hidden layer with sufficient number of hidden neurons [43,44]. The autoencoder is designed to do just that. Here, during training, the data set of measured functions is projected onto a certain number of hidden neurons and then returned to the original representation with the best possible match. The more hidden neurons are involved, the smaller the projection error. Here, the goal is to get by with as few hidden neurons as possible. To address the question of "sufficient" amount, several autoencoders are trained for different numbers of hidden neurons and their mean square projection error (MSE) is determined by

$$MSE = \frac{1}{N} \cdot \sum_{i=1}^{N} (f_i - y_i)^2 \tag{1}$$

with N as the number of data-points. f_i as the predicted value from the ML model and y_i as the true data point i.

The sufficient number is reached when the MSE difference to the next larger hidden layer size falls below a threshold value, converges strongly towards zero or deteriorates again.

Once a sufficient number of hidden neurons is found, an autoencoder can be trained whose hidden neurons are used as output setpoints for the preceding FFN.

Regarding item **Define Procedure** the following design order results (see also Fig. 5) :

- The overall network consists of two networks connected in series. One network of FFNs and one autoencoder.
- The number of FFNs is only determined after the required number of hidden neurons has been established for the autoencoder. Thus, it is essential to first optimize and train the autoencoder. Then, for each known hidden neuron value, a separate FFN can be trained and its optimal topology determined.
- Each FFN equally receives the input variables available for the process and is connected to exactly one hidden neuron each as a one-dimensional output.

Since the output dimension of a FFN is significantly reduced from several hundred curve points to very few hidden neuron values by the post-connection of the autoencoder each training can be performed in an exceedingly short time and many different network topologies can be compared. The FFN networks found in this way, in conjunction with the previously trained autoencoder, form the entire prediction chain, which maps a resulting SAXS curve from incoming synthesis/process parameters.

5. Results

In this section, we evaluate the multi-step method according to the systematic approach defined in Section 4.2 and the individual steps taken to create the overall net-system and discuss the results. Nets and autoencoder where trained in MATLAB R2021a. Furthermore, a comparison of results in the use of a single network and a net-system is made.

5.1. Defining procedure

As shown in Fig. 5, the aim is to form a total system of FFN and autoencoder. In this case, the training could not be performed straight forward, since intermediate results were missing during the transition from FFN to the final output, as the hidden neuron values of the autoencoder to be trained are determined from its chosen structure. Thus, before training of the FFN starts, the autoencoder and the required size of the hidden layer had to be determined, which thus defined the target sizes and target values for the output of the FFNs in training. Once the number of hidden neurons was determined, a fixed autoencoder could be trained that would be used to specify the output parameters of the preceding FFNs and also later for final prediction.

Now that the output is also known, it is possible to invest in finding the best possible topology of the preceding FFN system and address issues of partitioning and structure.

5.2. Train nodes

5.2.1. Autoencoder

To determine the minimum number of hidden neurons to be used in the autoencoder, the mean value of the MSE was calculated from 10 trained autoencoders and the absolute improvement at different hidden layer sizes was compared. Here, a sigmoid encoding function and a linear decoding function lead to the best results. There is a continuous decrease of the MSE with increasing size of the hidden layer. Comparing the MSEs by subtracting the following value $i+1$ from the actual one i, we notice an abrupt improvement deceleration at the number of 6 neurons with a low standard deviation as seen in Fig. 6.

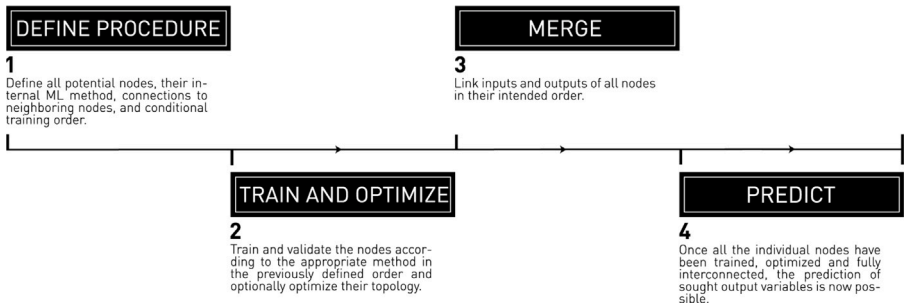

Fig. 4. Systematic approach to design a functional overall structure of the multi-step method.

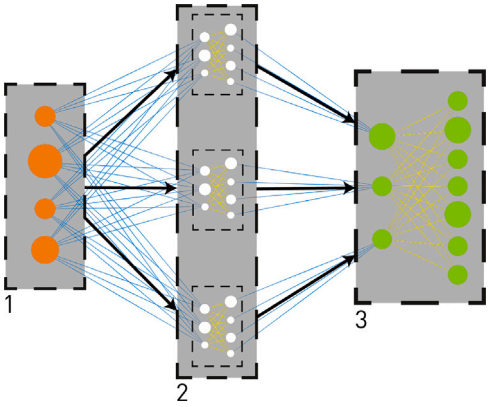

Fig. 5. Scheme of the multi-step method in the use-case. A system of several FFNs (**2**) each maps the input variables (**1**) to one ore more of the hidden neurons of the autoencoder, which in turn forms and outputs the resulting SAXS curve from the incoming values (**3**).

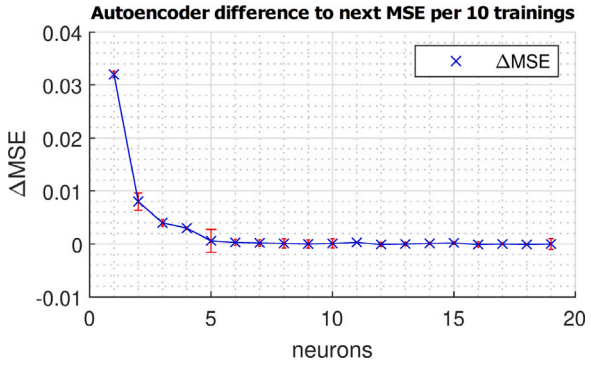

Fig. 6. MSE differences from actual to next hidden layer size of the autoencoder. For more than 6 hidden neurons there are no bigger improvements further on.

It shows increasing the intrinsic information with > 6 hidden neurons has no more significant benefit and the SAXS curve can be described sufficiently in this way.

The examination of the mapping areas of the predicted SAXS curves for the individual neurons does not lead to any useful insight. As shown in Fig. 7 features interpreted by the autoencoder are not obvious, closed subsections as they would be subdivided according to a human logical assessment. Rather, the mapping results collectively from the interaction of several to all neurons. Occasionally, some points show up that are not represented by one single neuron, but no useful interpretation can be derived.

5.2.2. Feed forward net

For the final linkage of synthesis and process parameters to the output of the SAXS curve, a connection between the mentioned input parameters and the determined 6 hidden neurons of the autoencoder was further required. Seven process parameters were chosen as input values, which were variably configured in the synthesis of the materials:

1. target density ρ_{target} (assuming that all silane in the liquid volume is converted to SiO_2)
2. molar ratio of water to TEOS x_{HT}
3. (calculated) pH
4. ageing time in multiples of the gelation time
5. pH of the pore liquid during ageing
6. temperature T
7. gas flow during controlled ambient pressure drying

Since the processing and measurement of a single synthesis material is rather time consuming, the framework of a Design of Experiments (DoE) was used for generating the necessary training data, which ensures the highest possible variability with the lowest possible number of data sets in the input parameters of the experiments. Through the evaluation of the DoE with onion modification, 50 variabilities of input synthesis parameters were given, based on which the respective materials were synthesized, their SAXS curves measured and ultimately the required FFNs could be trained.

Note: In the further sections, the topology of an FFN is described using the notation $[il\ hl_1\ hl_2 \dots hl_n\ ol]$.
Where il *is the number of neurons in the input layer,* hl_i *is the number of neurons in the* i*th hidden layer and* ol *is the number of neurons in the output layer of the FFN.*

To find the right net topology the usefulness of the division into single work steps by our multi-step method becomes apparent. The training of one single net with the same structure as it results in the division into two separate nets took a similar amount of time as the training of more than 1000 different divided nets (e.g. one FFN of [7 10 10 10 6 949] compared to two separate nets of [7 10 10 10 6] and [6 949]). To search for the overall structure with the smallest approximation error (SSE) we trained all possible structure layouts from one to three hidden layers and all of their combinations between 1 and 10 neurons. The SSE is defined as

$$SSE = \sum_{i=1}^{N} (f_i - y_i)^2 \tag{2}$$

Multiple training per specific network topology is necessary since each training has different initial conditions and sampling which generate different results.

In the presented use-case, the best results with the present data sets were obtained for a network topology of [7 8 6 10 6] and the use of the *tanh* activation function. Fig. 8 shows the SSE of each hidden neuron. Since the hidden neuron values can only lie in the value range between 0 and 1, the absolute SSE simultaneously reflect relative errors. In the present case, the average prediction error of all Hidden Neurons was 0.101 ± 0.028 with the smallest mean prediction error in the second neuron at 0.066 ± 0.059 and the largest in the first neuron at 0.143 ± 0.108.

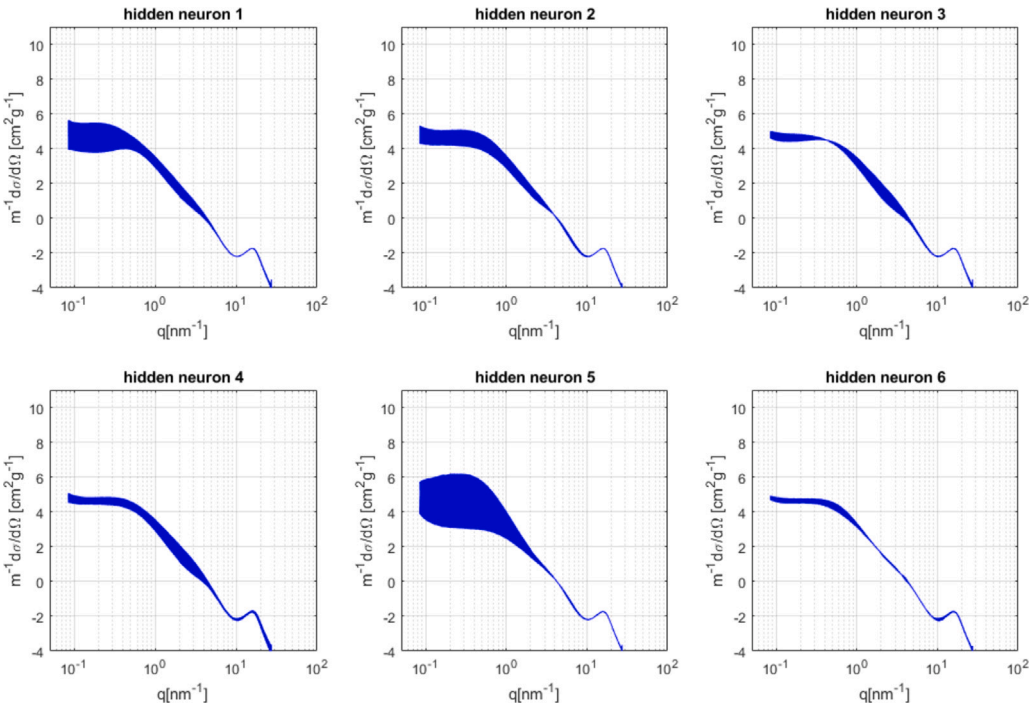

Fig. 7. Mapping areas of individual hidden neurons. No logical task partitioning of individual curve segments to individual neurons can be observed. Only the collective signal output generates a meaningful SAXS curve.

Table 2
Net structures of the optimal found networks to the respective Hidden neuron. No structural relationships can be identified that provide indications for a construction rule for optimal network topologies.

Topology FFN	Hidden neuron Autoencoder
[7 7 10 10 1]	#1
[7 7 10 9 1]	#2
[7 5 4 6 1]	#3
[7 9 5 3 1]	#4
[7 10 8 5 1]	#5
[7 7 3 10 1]	#6

5.2.3. Single FFN vs. FFN-system

As mentioned in the beginning of this paragraph, splitting the single FFN into a network system of FFNs should lead to better results [45]. For this purpose, a separate network was trained for each hidden neuron of the autoencoder and its respective optimal topology was determined according to the established procedure of finding the best single net with the results seen in Table 2. No structural relationships can be identified that provide indications for a construction rule for optimal network topologies. Thus, it looks like successive testing of all possible structures is inevitable to find the best nets. Fig. 8 shows the comparison of the prediction errors between the single net results and the net system results. The average prediction error of all hidden neurons was 0.004 ± 0.011 with the smallest mean prediction error in the third neuron at 0.002 ± 0.007 and the largest in the sixth neuron at 0.017 ± 0.082. Compared to the single FFN, this reduced the average prediction error by 96% with the smallest improvement at 80.8% and the largest at 97.7%.

5.3. Merge

Now, the subsystems had been successfully trained and their respective topologies optimized, the overall structure could be generated by linking respective input and output neurons. Each parameter set leads

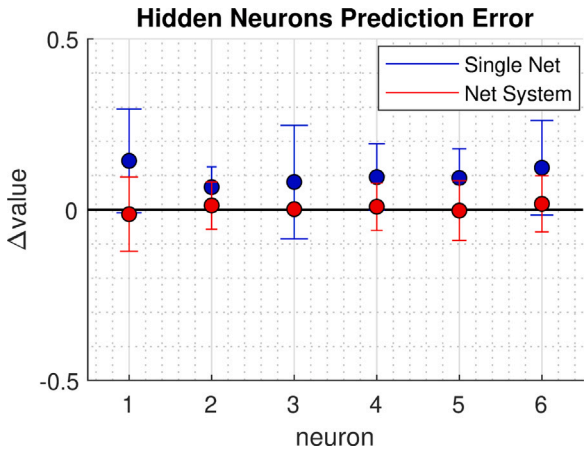

Fig. 8. Comparison of the mean prediction errors with associated standard deviation in the prediction of the hidden neuron values between Single Net and Net System. The average prediction error of the Single Net was 0.101 ± 0.028 with the smallest mean prediction error in the second neuron at 0.066 ± 0.059 and the largest in the first neuron at 0.143 ± 0.108. On average, with the Net System the prediction error was reduced by 96%. The smallest improvement was 80.8% and the largest 97.7%.

in the prediction through the respective FFN to the attached hidden neuron of the autoencoder, which then interpreted an associated SAXS curve from all six hidden neuron values. Fig. 9 shows an example of the comparison between the measured SAXS curve and its predicted counterpart using their defined parameter set.

6. Discussion

When the prediction errors were examined more in detail, it was noticed that in some cases the errors of the net-system had the same structure as the errors of the single net. If a hidden neuron value was over-/underestimated in the optimal Single Net, it was also over-/underestimated in the Network System, only at a smaller distance from

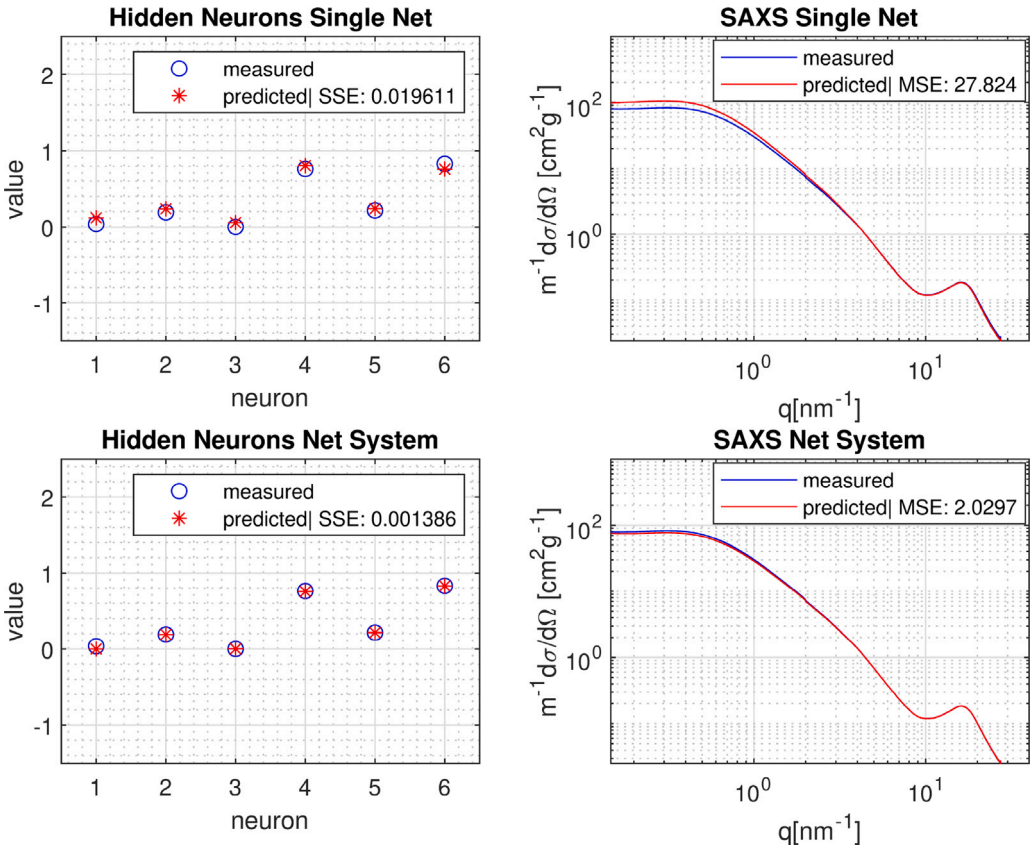

Fig. 9. Comparison of measured SAXS curve and predicted counterpart with one parameter set of the material synthesis. Top: Prediction result with Single Net. Bottom: Prediction result with Net System.

the set-point value. This suggests that for the Single Net there also exists a topology that gives the same results as the Net System found here. For this, larger numbers of hidden layers and hidden neurons would have to be further investigated for the FFN, which however again very quickly results in much more computing time. In the conclusion, the use of the FFN system is certainly in the advantage, since for the same computational effort, better results can be produced. Moreover it is not clear whether the search for the best network topologies should focus more on wider or deeper networks.

In this way, complex interrelationships, which are no longer analytically comprehensible due to the processing and measurement of the results, can be reasonably subdivided into small partial predictions, which, when combined in the current application, yield a resulting SAXS curve to be predicted on the basis of the synthesis/process parameters using just a few data sets.

Separating higher dimensional outputs into individual systems, with each having one output, gave better results under optimal topologies with improvements up to 97.7% compared to the standard prediction with one single FFN.

7. Conclusion and further research

With the presented multi-step method many different network structures can be investigated in a short time and optimized net topologies for the given task can be found. The comparison of the successively better results when varying the network topologies indicates a large dependence on the quality of the predictive ability of the best network found.

In the development of anode materials, the novel combination of SAXS, as a fast characterization method and ML proves to be a symbiosis with high potential. Above all, it can lead to a more targeted development process from the very beginning and, among others, enables the application of mathematical optimization methods by artificially adapting the synthesis process (here: matrix materials for battery electrodes). This provides an opportunity to replace the traditional trial and error, experience-based, principle with a structured approach and to support the development of innovative anode materials. In the first of several complex sub-steps to the final battery, the knowledge obtained brings confidence. All additional steps up to the final product must be pursued in the future.

In conclusion, the investigations have shown that the multi-step method can be successfully used in material development to adapt the complex synthesis process with standard machine learning methods. Thus, the possibility is given to provide good estimates based on pure synthesis parameters with small prediction errors.

In addition of using standard neural networks, the question arises whether other ML methods or specialized networks such as Bridged Multilayer Perceptron [41] or Fully Connected Cascade [41] can provide better results.

With the current results, no correlation between the optimal network topology and the structure of the data could be determined in initial observations. It may be possible to derive a procedure that replaces the successive testing of all possible combinations of hidden layer sizes.

There needs to be further investment in exploring the predictive capabilities within and beyond the training data. The goal of machine adaptation of development steps is usually to use them to optimize specific parameters. In the current case, a trained network could dictate which parameters would need to be set for a desired SAXS curve. The idea is to use a trained overall network as a substitute for the time-consuming synthesis process, and thus, in combination with a numerical optimization procedure, to find the ideal input parameters that provide the desired output. Depending on the requirements, the

optima sought do not necessarily have to lie in the range of possible interpolation, but may also require the ability to extrapolate.

There is also the question of how much data is sufficient to successfully train a regression. Since we worked with DoE, it is not clear whether the results would also have been possible without DoE, as this equidistantly covers the search space and thus possibly also provides the maximum information content with the smallest possible experimental size.

CRediT authorship contribution statement

Philipp Seitz: Conceptualization, Methodology, Software, Validation, Formal analysis, Data curation, Writing, Writing – review & editing, Visualization. **Christian Scherdel:** Methodology, Validation, Resources, Data curation, Writing – review & editing. **Gudrun Reichenauer:** Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Jan Schmitt:** Conceptualization, Investigation, Writing, Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The raw/processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study and remain confidential so far. The code to apply the multi-step method for various data is available under https://gitlab.vlab.fm.fhws.de/philipp.seitz/machinelearningandsaxs.

Acknowledgements

The authors gratefully acknowledge the *Bayerisches Staatsministerium für Wirtschaft, Landesentwicklung und Energie* for funding the project *LeMO2n - Lernende Multi-Skalen-Optimierung für SiO2-basierende Anodenmaterialien*, Grant no. 0703/68362/298/21/16/22/17/23/18/24.

References

[1] C. Scherdel, E. Miller, G. Reichenauer, J. Schmitt, Advances in the development of sol-gel materials combining small-angle X-Ray scattering (SAXS) and machine learning (ML), Processes 9 (4) (2021) 672.
[2] B. Engelmann, S. Schmitt, E. Miller, V. Bräutigam, J. Schmitt, Advances in machine learning detecting changeover processes in cyber physical production systems, J. Manuf. Mater. Process. 4 (4) (2020) 108.
[3] M. Aykol, P. Herring, A. Anapolsky, Machine learning for continuous innovation in battery technologies, Nat. Rev. Mater. 5 (10) (2020) 725–727.
[4] I.C. Secretary, Particle Size Analyses – Small Angle X-ray Scattering (SAXS), Standard ISO 17867:2020, International Organization for Standardization, Geneva, CH, 2020, URL https://www.iso.org/standard/69213.html.
[5] S. Kistler, Coherent expanded aerogels, Rubber Chem. Technol. 5 (4) (1932) 600–603.
[6] M.A. Aegerter, N. Leventis, M. Koebel, Advances in sol-gel derived materials and technologies, in: Aerogels Handbook, Springer, New York, NY, USA, 2011.
[7] G.W. Scherer, D.M. Smith, X. Qiu, J.M. Anderson, Compression of aerogels, J. Non-Cryst. Solids 186 (1995) 316–320.
[8] C. Scherdel, G. Reichenauer, Highly porous silica xerogels without surface modification, J. Supercrit. Fluids 106 (2015) 160–166.
[9] C. Scherdel, G. Reichenauer, The impact of residual adsorbate on the characterization of microporous carbons with small angle scattering, Carbon 50 (8) (2012) 3074–3082.
[10] C. Schlumberger, C. Scherdel, M. Kriesten, P. Leicht, A. Keilbach, H. Ehmann, P. Kotnik, G. Reichenauer, M. Thommes, Reliable surface area determination of powders and meso/macroporous materials: Small-angle X-ray scattering and gas physisorption, Microporous Mesop. Mater. 329 (2022) 111554.
[11] A. Karakoç, Ö. Keleş, A predictive failure framework for brittle porous materials via machine learning and geometric matching methods, J. Mater. Sci. 55 (11) (2020) 4734–4747.
[12] R. Ramprasad, R. Batra, G. Pilania, A. Mannodi-Kanakkithodi, C. Kim, Machine learning in materials informatics: recent applications and prospects, Npj Comput. Mater. 3 (1) (2017) 1–13.
[13] D.M. Dimiduk, E.A. Holm, S.R. Niezgoda, Perspectives on the impact of machine learning, deep learning, and artificial intelligence on materials, processes, and structures engineering, Integr. Mater. Manuf. Innov. 7 (3) (2018) 157–172.
[14] J. Schmidt, M.R. Marques, S. Botti, M.A. Marques, Recent advances and applications of machine learning in solid-state materials science, Npj Comput. Mater. 5 (1) (2019) 1–36.
[15] I.M. Cockburn, R. Henderson, S. Stern, 4. The Impact of Artificial Intelligence on Innovation: An Exploratory Analysis, University of Chicago Press, 2019.
[16] S.K. Kauwe, T.D. Rhone, T.D. Sparks, Data-driven studies of li-ion-battery materials, Crystals 9 (1) (2019) 54.
[17] M. Nakayama, K. Kanamori, K. Nakano, R. Jalem, I. Takeuchi, H. Yamasaki, Data-driven materials exploration for li-ion conductive ceramics by exhaustive and informatics-aided computations, Chem. Rec. 19 (4) (2019) 771–778.
[18] M.A. Shandiz, R. Gauvin, Application of machine learning methods for the prediction of crystal system of cathode materials in lithium-ion batteries, Comput. Mater. Sci. 117 (2016) 270–278.
[19] A.D. Sendek, E.D. Cubuk, E.R. Antoniuk, G. Cheon, Y. Cui, E.J. Reed, Machine learning-assisted discovery of solid Li-ion conducting materials, Chem. Mater. 31 (2) (2018) 342–352.
[20] Y. Liu, T. Zhao, W. Ju, S. Shi, Materials discovery and design using machine learning, J. Materiomics 3 (3) (2017) 159–177.
[21] Y. Takagishi, T. Yamanaka, T. Yamaue, Machine learning approaches for designing mesoscale structure of li-ion battery electrodes, Batteries 5 (3) (2019) 54.
[22] Z. Jiang, J. Li, Y. Yang, L. Mu, C. Wei, X. Yu, P. Pianetta, K. Zhao, P. Cloetens, F. Lin, et al., Machine-learning-revealed statistics of the particle-carbon/binder detachment in lithium-ion battery cathodes, Nature Commun. 11 (1) (2020) 1–9.
[23] A. Turetskyy, S. Thiede, M. Thomitzek, N. von Drachenfels, T. Pape, C. Herrmann, Toward data-driven applications in lithium-ion battery cell manufacturing, Energy Technol. 8 (2) (2020) 1900136.
[24] M. Lucu, E. Martinez-Laserna, I. Gandiaga, K. Liu, H. Camblong, W. Widanage, J. Marco, Data-driven nonparametric Li-ion battery ageing model aiming at learning from real operation data-Part B: Cycling operation, J. Energy Storage 30 (2020) 101410.
[25] R.R. Richardson, C.R. Birkl, M.A. Osborne, D.A. Howey, Gaussian process regression for in situ capacity estimation of lithium-ion batteries, IEEE Trans. Ind. Inform. 15 (1) (2018) 127–138.
[26] Y. Zhang, Q. Tang, Y. Zhang, J. Wang, U. Stimming, A.A. Lee, Identifying degradation patterns of lithium ion batteries from impedance spectroscopy using machine learning, Nature Commun. 11 (1) (2020) 1–6.
[27] E. Chemali, P.J. Kollmeyer, M. Preindl, A. Emadi, State-of-charge estimation of Li-ion batteries using deep neural networks: A machine learning approach, J. Power Sources 400 (2018) 242–255.
[28] M.A. Hannan, M.H. Lipu, A. Hussain, P.J. Ker, T.I. Mahlia, M. Mansor, A. Ayob, M.H. Saad, Z. Dong, Toward enhanced state of charge estimation of lithium-ion batteries using optimized machine learning techniques, Sci. Rep. 10 (1) (2020) 1–15.
[29] N. Harting, R. Schenkendorf, N. Wolff, U. Krewer, State-of-health identification of lithium-ion batteries based on nonlinear frequency response analysis: First steps with machine learning, Appl. Sci. 8 (5) (2018) 821.
[30] A. Senthilselvi, V. Sellam, S.A. Alahmari, S. Rajeyyagari, Accuracy enhancement in mobile phone recycling process using machine learning technique and MEPH process, Environ. Technol. Innov. 20 (2020) 101137.
[31] Y.-L. Chen, L. Pollack, Machine learning deciphers structural features of RNA duplexes measured with solution X-ray scattering, IUCrJ 7 (5) (2020) 870–880.
[32] D. Franke, C.M. Jeffries, D.I. Svergun, Machine learning methods for X-ray scattering data analysis from biomacromolecular solutions, Biophys. J. 114 (11) (2018) 2485–2492.
[33] P. Tomaszewski, S. Yu, M. Borg, J. Rönnols, Machine learning-assisted analysis of small angle X-ray scattering, in: 2021 Swedish Workshop on Data Science, SweDS, IEEE, 2021, pp. 1–6.
[34] D. Saurel, J. Segalini, M. Jauregui, A. Pendashteh, B. Daffos, P. Simon, M. Casas-Cabanas, A SAXS outlook on disordered carbonaceous materials for electrochemical energy storage, Energy Storage Mater. 21 (2019) 162–173.
[35] G. Sandí, H. Joachin, R. Kizilel, S. Seifert, K.A. Carrado, In situ SAXS studies of the structural changes of polymer nanocomposites used in battery applications, Chem. Mater. 15 (4) (2003) 838–843.
[36] M. Povia, J. Sottmann, G. Portale, K.D. Knudsen, S. Margadonna, S. Sartori, Operando SAXS/WAXS on the AP/C as the anode for Na-Ion batteries, J. Phys. Chem. C 122 (11) (2018) 5917–5923.
[37] C.L. Berhaut, D.Z. Dominguez, D. Tomasi, C. Vincens, C. Haon, Y. Reynier, W. Porcher, N. Boudet, N. Blanc, G.A. Chahine, et al., Prelithiation of silicon/graphite composite anodes: Benefits and mechanisms for long-lasting Li-Ion batteries, Energy Storage Mater. 29 (2020) 190–197.
[38] S.V. Kamarthi, S. Pittner, Accelerating neural network training using weight extrapolations, Neural Netw. 12 (9) (1999) 1285–1299.

[39] T.N. Sainath, B. Kingsbury, V. Sindhwani, E. Arisoy, B. Ramabhadran, Low-rank matrix factorization for deep neural network training with high-dimensional output targets, in: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE, 2013, pp. 6655–6659.

[40] A. Yeganeh, A. Shadman, Using evolutionary artificial neural networks in monitoring binary and polytomous logistic profiles, J. Manuf. Syst. 61 (2021) 546–561.

[41] D. Hunter, H. Yu, M.S. Pukish III, J. Kolbusz, B.M. Wilamowski, Selection of proper neural network sizes and architectures—A comparative study, IEEE Trans. Ind. Inform. 8 (2) (2012) 228–240.

[42] B.M. Wilamowski, N.J. Cotton, O. Kaynak, G. Dundar, Computing gradient vector and Jacobian matrix in arbitrarily connected neural networks, IEEE Trans. Ind. Electron. 55 (10) (2008) 3784–3790.

[43] A. Kratsios, The universal approximation property, Ann. Math. Artif. Intell. 89 (5) (2021) 435–469.

[44] K. Hornik, Approximation capabilities of multilayer feedforward networks, Neural Netw. 4 (2) (1991) 251–257.

[45] D.F. Cook, C.T. Ragsdale, R. Major, Combining a neural network with a genetic algorithm for process parameter optimization, Eng. Appl. Artif. Intell. 13 (4) (2000) 391–396.