



Article

Indentation Reverse Algorithm of Mechanical Response for Elastoplastic Coatings Based on LSTM Deep Learning

Xu Long ¹, Xiaoyue Ding ², Jiao Li ², Ruipeng Dong ², Yutai Su ^{2,*} and Chao Chang ^{3,*}

¹ Research & Development Institute, Northwestern Polytechnical University in Shenzhen, Shenzhen 518063, China

² School of Mechanics, Civil Engineering and Architecture, Northwestern Polytechnical University, Xi'an 710072, China

³ School of Applied Science, Taiyuan University of Science and Technology, Taiyuan 030024, China

* Correspondence: suyutai@nwpu.edu.cn (Y.S.); cc@tyust.edu.cn (C.C.)

Abstract: The load-penetration depth (P - h) curves of different metallic coating materials can be determined by nanoindentation experiments, and it is a challenge to obtain stress-strain response and elastoplastic properties directly using P - h curves. These problems can be solved by means of finite element (FE) simulation along with reverse analyses and methods, which, however, typically occupy a lengthy time, in addition to the low generality of FE methodologies for different metallic materials. To eliminate the challenges that exist in conventional FE simulations, a long short-term memory (LSTM) neural network is proposed in this study and implemented to deep learn the time series of P - h curves, which is capable of mapping P - h curves to the corresponding stress-strain responses for elastoplastic materials. Prior to the operation of the neural network, 1000 sets of indentation data of metallic coating materials were generated using the FE method as the training and validating sets. Each dataset contains a set of P - h curves as well as the corresponding stress-strain curves, which are used as input data for the network and as training targets. The proposed LSTM neural networks, with various numbers of hidden layers and hidden units, are evaluated to determine the optimal hyperparameters by comparing their loss curves. Based on the analysis of the prediction results of the network, it is concluded that the relationship between the P - h curves of metallic coating materials and their stress-strain responses is well predicted, and this relationship basically coincides with the power-law equation. Furthermore, the deep learning method based on LSTM is advantageous to interpret the elastoplastic behaviors of coating materials from indentation measurement, making the predictions of stress-strain responses much more efficient than FE analysis. The established LSTM neural network exhibits the prediction accuracy up to 97%, which is proved to reliably satisfy the engineering requirements in practice.

Keywords: elastoplastic coating; indentation; reverse analysis; FE simulation; LSTM neural network



Citation: Long, X.; Ding, X.; Li, J.; Dong, R.; Su, Y.; Chang, C. Indentation Reverse Algorithm of Mechanical Response for Elastoplastic Coatings Based on LSTM Deep Learning. *Materials* **2023**, *16*, 2617. <https://doi.org/10.3390/ma16072617>

Academic Editor: Fernando Gomes de Souza Junior

Received: 26 February 2023

Revised: 19 March 2023

Accepted: 22 March 2023

Published: 25 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Coatings are frequently employed to protect materials and structures in harsh environments and to reduce or stop damage caused by outside forces. Particularly, metallic coatings have a wide variety of applications due to their excellent properties. For instance, in the aerospace industry, titanium-based coatings are frequently used to protect aerospace engines, because of their excellent corrosion resistance and stable strength [1–4]. To achieve corrosion resistance of steel components in automotive industry, aluminum, zinc, and zinc-boron compounds are the most popular metals utilized as coatings [5].

In order to assure the coating's reliability during applications, it is essential to understand and quantify its mechanical qualities, especially the in situ properties after the coating process with high temperature, plastic formation, and strain rate. Apparently, the coating is essentially adhered to the substrate material, and the presence of the substrate material

directly affects the findings of macroscopic testing of coating materials if the substrate effect cannot be reasonably excluded. In this regard, experimental nanoindentation methods have been presented for analyzing the microscale characterization features of materials [6–9]. In instrumented nanoindentations, the indenter penetrates the material's surface at a controlled speed or load, and the applied load P and the penetration depth h of the indenter are simultaneously recorded. By examining the P – h curves under various working situations, a deep understanding can be made for the material's mechanical properties. By following the pioneering Oliver–Pharr model, the hardness and elastic modulus can be obtained based on the P – h curve [10–13]. Nevertheless, the material's stress–strain response cannot be derived directly from the P – h curve. In spite of the fact that the mapping from the P – h curve to the stress–strain curve can be resolved by inverse analysis with finite element (FE) software, this is not only necessary to consider the relationship between several physical quantities during the analysis process, but also it is time-consuming to perform a parametric study of FE simulations. This means that the method simply based on exhausting FE simulations is ineffective in capturing the essence of such an inverse problem for elastoplastic materials.

In light of the aforementioned challenges in the analysis process, machine learning methods are more advantageous than FE simulation at handling nonlinear relationships of data. Machine learning techniques are utilized in industrial manufacturing due to their outstanding capabilities for data analysis [14,15]. The machine learning approach has gained popularity in material inspection analysis because of its low development cost, short development cycle, and excellent predictive performance when dealing with large amounts of data [16]. Deep learning is one of the subfields of machine learning, which has been the major research focus of computer applications in recent years. Compared with conventional machine learning, deep learning can automatically execute the majority of data feature extraction during network training and can work with significantly bigger datasets. On the other hand, machine learning concentrates on applying statistical approaches to data, but deep learning primarily imitates biological neural networks to extract data features step by step. In recent years, convolutional neural networks (CNN), recurrent neural networks (RNN), and generative adversarial networks (GAN) have been the most popular deep learning models [17–21]. These three types of networks have their own application scenarios based on different algorithms. Among them, CNN is frequently used to process and analyze medical images due to its powerful image analysis capability [22,23]. The special feature of GAN is that there are two networks inside it, and during the training process, the two networks fight against each other to continuously approach the optimal solution of the problem. Therefore, GAN is used to generate random data for applications such as image enhancement and data denoising [24–26]. LSTM is derived from RNN, compensating for the fact that RNN cells cannot effectively learn the relevant information of the input data when the input time period is lengthy [27–31]. LSTM is frequently used in text recognition, audio processing, and video processing due to its high effectiveness with time series data [32–34]. For indentation problems, Long et al. [35] adopted the P – h curves of metal materials to train the CNN network; some constitutive parameters are regarded as the training objective. Their datasets are generated by means of FE modeling, and the stress–strain relationship of the metallic material is described by a power-law equation [36,37]. However, the datasets lose certain characteristics during the convolution in the CNN network. Essentially, there is a close relationship between the P – h curve and the constitutive equation, and all data are continuous in the time history. Compared with CNN and GAN networks, a deep learning method based on LSTM could be more advantageous to optimize the reverse process of the mechanical behavior of materials in indentation studies.

In this study, an LSTM neural network is established to perform the inverse calculation of P – h curves to the stress–strain responses of metallic coating materials. Initially, 1000 datasets of matched P – h curves and stress–strain curves are constructed by performing extensive FE simulations. After extracting the coordinates of the P – h curves with equal spacing divisions, the maximum–minimum normalization technique is applied in order

to generate the network's input data. The established network then processes the input data and outputs the corresponding stress–strain response, which completes the crucial training phase for the LSTM neural network. With the trained LSTM network, some independent cases are further examined in the validating phase. This technology is promising to effectively address the challenges of conventional nanoindentation inversion work and markedly enhance computing efficiency. To guarantee the correctness of the proposed procedure, the predicted results from the LSTM network and the FE simulations are compared by in-depth discussions with various statistical evaluation indicators.

2. Indentation Theory and Database Preparation

2.1. Theoretical Basis for Instrumented Indentation

To determine the deformation properties of metallic coating materials, in nanoindentation tests, the indenter is usually controlled to penetrate into the material at a constant load/displacement speed or at a constant indentation strain rate. If no holding stage is considered, the process is separated into two stages based on the direction of the indenter's movement: the loading and the unloading stages. By applying geometric self-similar indenters, the loading and deformation behavior of the material specimen usually satisfies Kick's law [38,39] as given by

$$P = Ch^m, \quad (1)$$

where P is the load applied to the material by the indenter during loading, C represents the curvature during loading, and h is the depth of indentation into the surface of the material. For the exponent, m is equal to 2 for an ideal indentation but is found to be usually less than 2 if uncertainties and imperfections of indenters and test samples are taken into account.

From the measured P – h curves, by continuously monitoring the applied load during the loading stage, material hardness can be determined and identified as a critical parameter to characterize the material properties. Hardness, defined as the average contact pressure on the contact area A_c due to the applied load on the indenter, demonstrates the material's capacity to resist local deformation, specifically plastic deformation, indentation, and scratching. The standard definition of hardness H is

$$H = \frac{P}{A_c}. \quad (2)$$

The contact area A_c can be calculated with the indentation depth for the geometrically self-similar indenter and is approximated as $24.5 h^2$ for a three-sided pyramid Berkovich indenter. In fact, the value of hardness was found to be closely related to the yield strength by the constraint factor (i.e., 3) as proposed by Tabor [40]. In addition to the plastic properties, the Young's modulus, the most important elastic property, can also be determined from the P – h curves by adopting the concept of reduced modulus. Apparently, the elastoplastic properties are implied in the measured P – h curves. Even though intensive efforts [38,41–44] have been made to establish reverse algorithms in recent decades, more reliable and efficient approaches are still unavailable.

In order to identify the stress–strain variation relationship of metal-coated materials during indentation experiments, the elastoplastic properties of the material can be extracted using a reversal method based on the measured material indentation dataset, whereas for metallic or alloy materials, their plastic behavior can be approximated using a power-law constitutive model [37,45,46] in the following form of

$$\sigma = \begin{cases} E\varepsilon & \varepsilon \leq \varepsilon_y \\ R\varepsilon_y^n & \varepsilon \geq \varepsilon_y' \end{cases} \quad (3)$$

where E is Young's modulus, R is the strength factor, n is the strain hardening index, and ε_y is the yield strain corresponding to the yield strength.

It is essential to emphasize that the purpose of this study is to propose an efficient but reliably accurate prediction method of constitutive properties for elastoplastic materials

by instrumented indentations. As a powerful and effective method for data analysis, deep learning allows the dataset consisting of indentation response of materials to function as input data, whilst the output data consists of the material's elastoplastic properties. The prediction model based on deep learning theory not only has excellent prediction capacity, but also can save a lot of time to improve the efficiency of engineering practice, which is of great help to realize the analysis of in situ mechanical properties of metallic coating materials. Therefore, the theories and formulae mentioned above are regarded as the theoretical basis for generating datasets for deep learning.

2.2. FE Simulation of Indentation

In order to generate sufficient reliable data, FE simulations are performed with a three-sided pyramid Berkovich. Note that for a Berkovich indenter, the hardness values of three-dimensional and axisymmetric indentations can be measured over the range from micro to nano-indentations, despite the indentation size effect. In this study, the Berkovich indenter is modeled as an axisymmetric part with a half angle of 70.3° to maintain the equivalent contact area with the three-dimensional indenter. The substrate material is a cylinder with a height of $100\ \mu\text{m}$ and a radius of $100\ \mu\text{m}$, which is discretized with 49,349 axisymmetric elements. With a maximum indentation depth of $2000\ \text{nm}$, the substrate's mesh size of $50\ \text{nm}$ is sufficiently fine to capture the high strain gradient underneath the indenter. Figure 1 shows the experimental model of nano-indentation constructed during the FE simulation. It should be noted that the applied indentation depth is much smaller than the height of the substrate material. The entire indentation process is examined by fixing the radial displacement of the reference point assigned on the indenter. Post-processing work allows for accurate output of the P - h curve and stress-strain curve, laying the groundwork for reliable further numerical verification, where variations of Young's modulus, yield strength and hardening exponent are taken into account. While employing the FE simulation approach for gathering training data, the noise in all data is uniformly reduced to improve model prediction robustness.

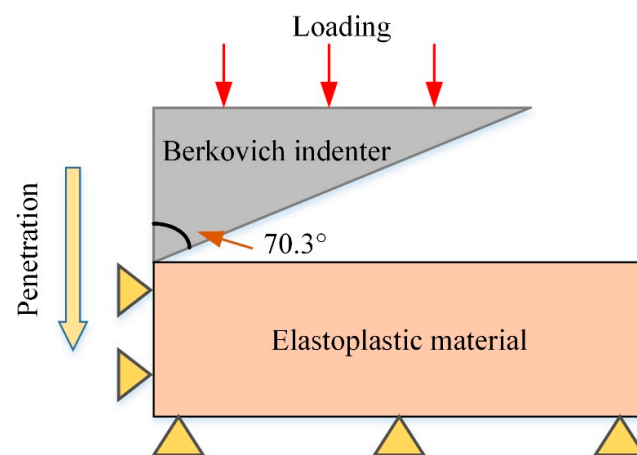


Figure 1. Schematic diagram of FE model for simulating the indentation by a Berkovich indenter.

3. Methodology

3.1. LSTM Neural Network

With the rapid development of machine learning, deep learning as one of its primary branches has been widely applied to scientific and practical engineering problems [47], and the most appropriate deep learning algorithms can be utilized to handle practical engineering problems as effectively as feasible. In the present work, it is crucial to analyze the nanoindentation experimental data generated by the FE simulation program in order to identify an appropriate algorithm. The metallic coating materials indentation dataset has two attributes. On the one hand, it is sequential in the sense that each point on the P - h curve created by the material during the nanoindentation experiment is closely connected,

which conforms to the nonlinear relationship between pressure and material deformation as stated by Kick's law. The second factor is dimension-transformability. Each P - h curve created by the material during indentation trials is mapped by the inversion algorithm onto the associated stress-strain curve.

Based on the features of the metallic coating indentation dataset and the nature of the problem to be solved, LSTM neural networks are quite effective deep learning methods. In 1997, Jürgen Schmidhuber's complete derivation of the LSTM algorithm, which extracts attributes of a dataset over an extended time interval, made its debut [48]. The LSTM neural network uses memory cells and gating units to control the storage and deletion of data. The gating units used are the input gate, which sets the size of the amount of data fed into the network each time. The forgetting gate controls the information that is ignored by the memory cells. The output gate determines the final output sequence feature value. The neural network's operation at time t is illustrated in Figure 2. Analysis of the LSTM neural network's data flow operation and its mathematical model can be expressed as follows [49–53]:

$$f_t = S(W_f \cdot [h_{t-1}, x_t] + b_t), \quad (4)$$

$$i_t = S(W_i \cdot [h_{t-1}, x] + b_i), \quad (5)$$

$$\vec{C}_t = \tanh(W_c \cdot [h_{t-1}, x] + b_c), \quad (6)$$

$$C_t = f_t \times C_{t-1} + i_t \times \vec{C}_t, \quad (7)$$

$$o_t = S(W_o \cdot [h_{t-1}, x_t] + b_o), \quad (8)$$

$$h_t = o_t \times \tanh(C_t), \quad (9)$$

where i_t , f_t , and o_t represent the input gate forgetting gate and the output gate's result, respectively. C_t represents the cell's state at time t , which can be utilized to form the connection between memory cells via \vec{C}_t . h_t represents the state of the hidden layer at moment t . W_x and b_x ($x = i, f, o, c$) are the weight and bias values of the matrices in the LSTM neural network for input gates, forgetting gates, output gates, and cell state updates. S (Sigmoid) and \tanh indicate the activation functions utilized by the appropriate gating units.

The indentation data processed in this study contain metallic materials with a wide range of Young's modulus, as reflected in the series over time. For different materials, the time series are different. Regarding this kind of time series, the LSTM method based on RNN can be a good solution to the problem of time series variation. The data in different time stages are understood in the same network, which can well reflect the stress-strain response pattern in each time stage of the indentation process. The key point of this model is the use of recursive network approach to understand the temporal nature of the indentation process through multi-layer calculations.

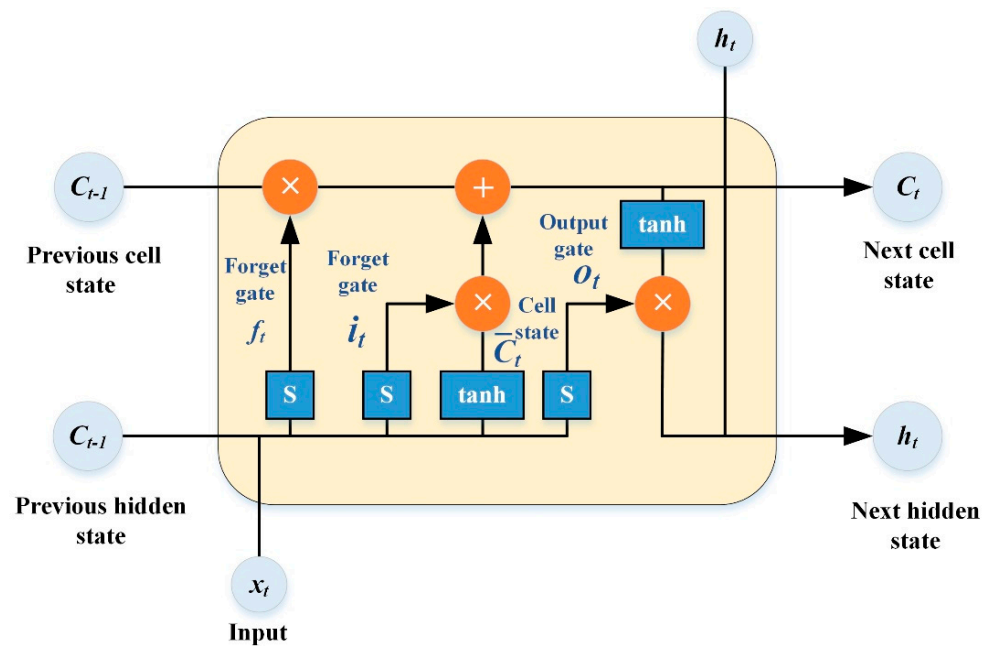


Figure 2. LSTM working schematic at time t .

3.2. Normalization for Data Preparation

After producing sufficient data on metallic coating indentation by the FE model, 1000 sets of data were chosen for network training, of which 90% are training data and 10% are validation data. For data preparation, the horizontal coordinates of the P - h curve of the metal coating material during the loading stage are divided into forty points spaced by fifty nanometers, and the vertical coordinates of these forty points serve as the input for each deep learning training. In order to increase the generalization of the training network to the dataset and the stable robustness of the prediction effect, the network parameter dropout is also set during the construction of the LSTM neural network. This aims to counteract the effect of overfitting in the deep learning process and improve the network's ability to process the data. The workflow of the LSTM neural network constructed in this paper is shown in Figure 3.

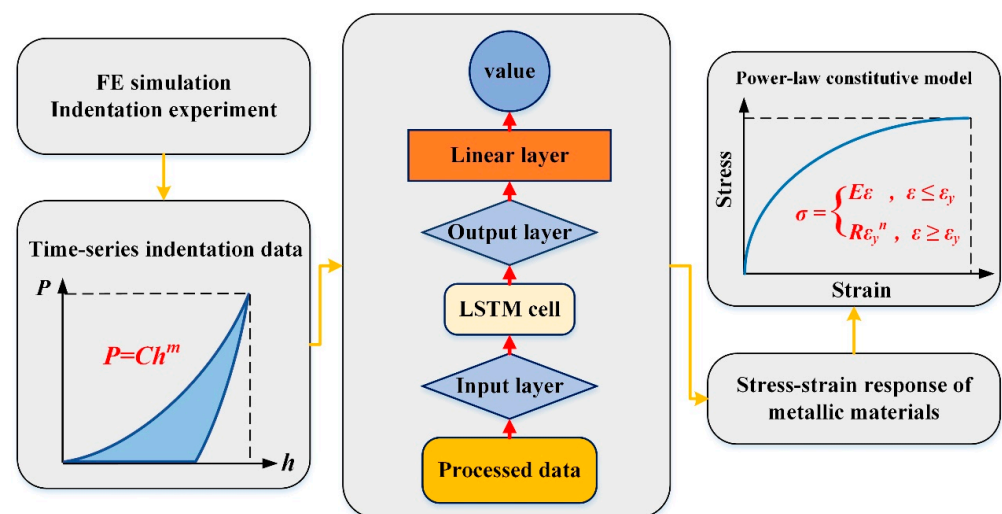


Figure 3. Work process of LSTM neural network.

Normalization of the dataset is an essential component as well. This is due to the fact that the input data and network output data have different scales and units during the training process, as well as a considerable variation in order of magnitude, which

may result in smaller feature quantities being disregarded during training. However, data normalization can not only minimize the influence of scale between input and output responses, but also make the data metrics comparable by standardizing their magnitudes. The dataset normalizing method utilized in this paper is normalization, which has the benefits of effectively enhancing the convergence speed and accuracy of the network and reducing the negative impacts caused by odd sample data. Following are the formulae for the max-min normalization approach:

$$X = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \quad (10)$$

where x is the current input value, X is the input value after reduction to the interval $[0,1]$. x_{\min} and x_{\max} are the minimum and maximum values. In the final output stage, the inverse normalization method can be used to obtain the corresponding prediction results.

3.3. Evaluation Indicators

Through a great number of repetitive calculations, the training stage of the LSTM neural network consists of seeking the optimal solution to the present system of nonlinear equations in the real number range. To effectively handle this optimization problem, the adaptive moment estimation (ADAM) was selected to obtain the optimal real number solution. The ADAM optimizer enables faster gradient reduction than conventional optimization methods, resulting in more accurate updating of the weight and deviation values obtained during computation [54].

In this research, the mean square error (MSE) is used to calculate the loss function in the network, which indicates the error between the predicted and actual values of the network throughout each cycle, and the root mean square error (RMSE) is utilized to complement the error computation. In addition, for a more straightforward evaluation of the performance of the LSTM neural network, the coefficient of determination R^2 is utilized to describe the correlation between the network-predicted values and the target values. The aforementioned evaluation metrics are defined by

$$MSE = \frac{1}{n} \sum_{i=1}^n (x_{\text{pred}} - x_{\text{true}})^2, \quad (11)$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_{\text{pred}} - x_{\text{true}})^2}, \quad (12)$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (x_{\text{pred}} - x_{\text{true}})^2}{\sum_{i=1}^n (x_{\text{true}} - \bar{x})^2} \quad (13)$$

where n represents the total number of samples, x_{pred} represents the value predicted by the LSTM neural network, x_{true} represents the value obtained through validation of the power-law equation, and \bar{x} represents the corresponding mean value.

4. Results and Discussion

4.1. Hyperparameter Setting of LSTM Neural Network

Prior to the network training, the first step is to set the network's hyperparameters. The adequacy of the hyperparameter settings has a direct impact on the predictive power of the final network. Consequently, the intention of the deep learning methodology is to identify the optimal numerical solution to the nonlinear system of equations under various hyperparameter settings. The ideal combination of hyperparameters for the current problem is determined by comparing the network prediction results under each set of hyperparameter settings. The most important hyperparameter is the learning rate, which

indicates the decay efficiency of the network's gradient in each cycle, which is typically set between 10^{-2} and 10^{-6} . When the learning rate is increased, the gradient reduces more quickly and the network's learning efficiency is accelerated, but the best solution will be overlooked during network calculation. Smaller learning rates are appropriate for more precise solutions, but excessively small learning rates result in prolonged training times and poor gradient reduction. Based on the relationship between the P - h curve during the nanoindentation test and the constitutive equation, the final learning rate for this study is determined as 10^{-4} . The number of training cycles of the network has the same features as the learning rate, and 1000 training cycles are used to make the learning curve of the LSTM neural network smooth and stable at a low order of magnitude.

Various types of deep learning networks have unique hyperparameters based on their frameworks. For LSTM neural network, the specific hyperparameters include input size, hidden size, and number of layers, which are defined as explained by the documentation of PyTorch. The number of hidden size reflects the amount of network features in each hidden layer, whereas the number of layers specifies the number of hidden layers utilized for each operation. For the hidden size and layer number, the control variables method was used to set multiple combinations of hyperparameters, and the optimal combination was determined by comparing the loss curves generated from each group. The number of hidden layers is set from 1 to 3, and the hidden size is set to 20, 40, 60, and 80 for each hidden layer, respectively. Upon completion of the test, the loss curves for various combinations are depicted in Figures 4 and 5. The solid lines in Figures 4 and 5 show the loss curves for the training stage, while the dashed lines indicate the loss curves for the validation stage.

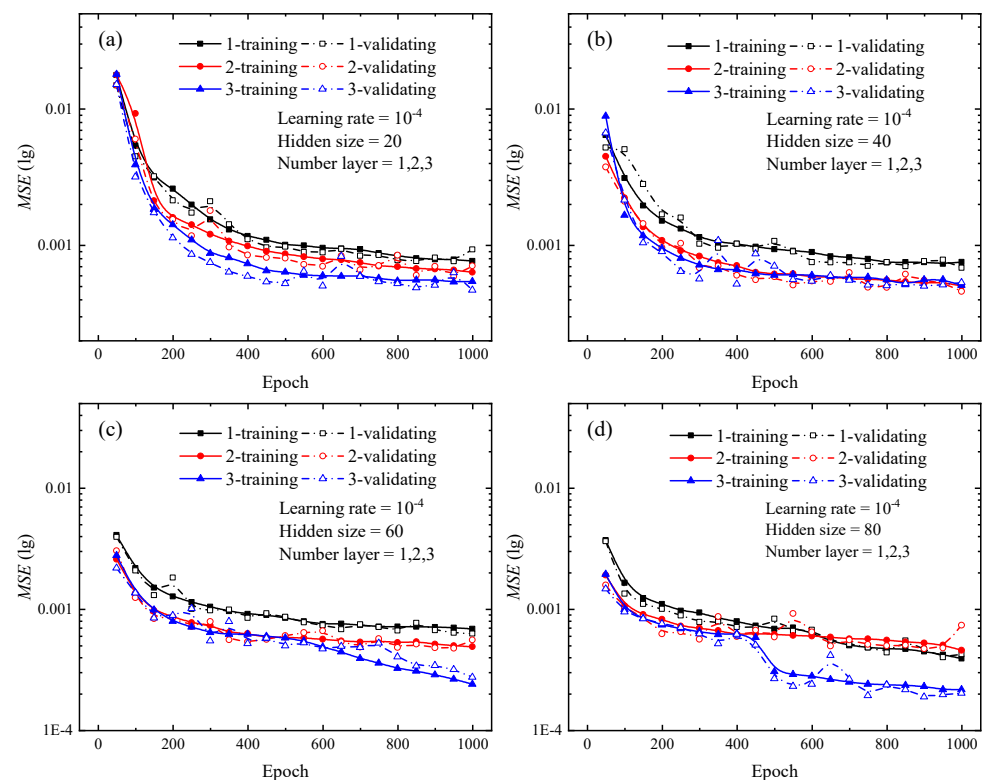


Figure 4. Loss curves of different numbers of hidden layers during the training and validating phrases for indentation predictions. (a) Hidden size of 20; (b) Hidden size of 40; (c) Hidden size of 60; (d) Hidden size of 80. In the legend, the cases of 1, 2, and 3 represent the number of hidden layers in the LSTM neural network.

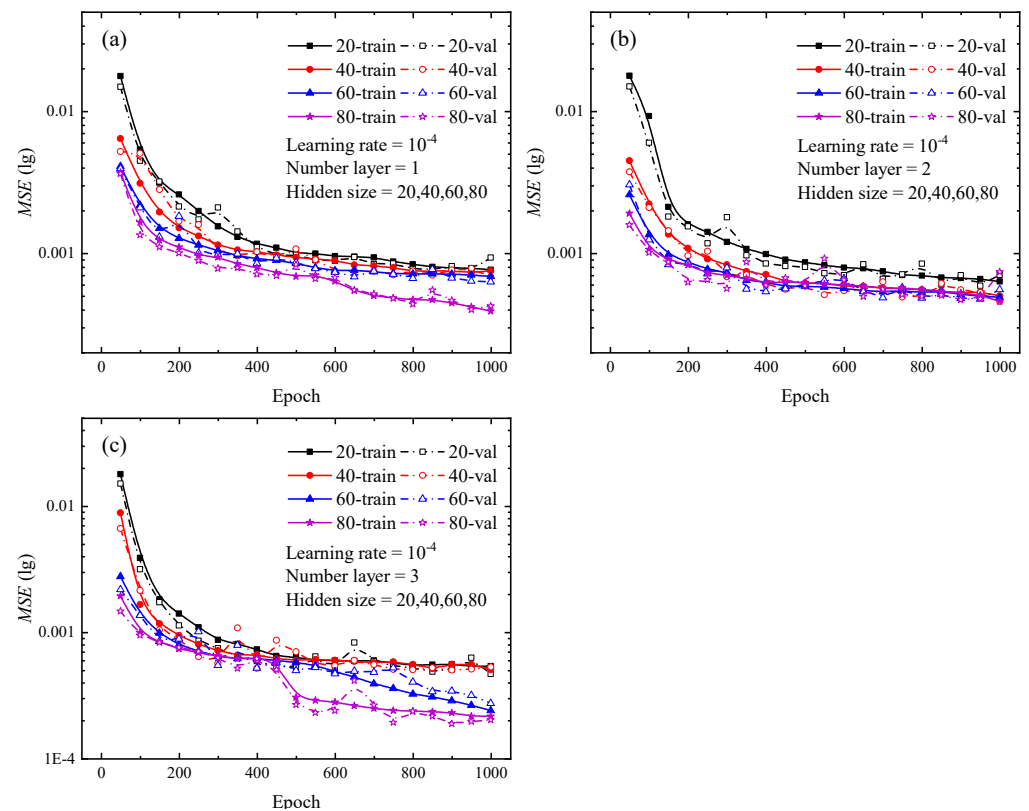


Figure 5. Loss curves of different hidden sizes during the training and validating phrases for indentation predictions. (a) The number of hidden layers is 1; (b) The number of hidden layers is 2; (c) The number of hidden layers is 3. In the legend, the cases of 20, 40, 60 and 80 represent the number of hidden sizes the LSTM neural network.

Once the loss function's value is absolutely minimal, it indicates that the network's predicted value exactly approximates the actual value. Focusing on the loss curves for the combinations of hyperparameters in Figures 4 and 5, the change trend of the loss curve with different combinations of hyperparameters can be initially determined. The larger the hidden size and the number of layers, the steeper the loss curve's decline and the lower the loss value. For the findings depicted in Figures 4 and 5, the LSTM neural network with three hidden layers and 80 hidden units in each hidden layer provides the most effective training performance. During the testing of the optimal hyperparameter combinations for the 12 sets of LSTM neural networks shown above, each network took about 7000 s to complete a full calculation, while the difference value between the longest and shortest run times was only 800 s.

In order to determine the optimal combination of hyperparameters, additional tests are conducted to verify the validity of this trend. Figure 6 depicts the outcomes of the extra tests. Additional studies demonstrate that increasing the number of hidden layers or hidden units has a little influence on the loss curve, and that the final loss remains stable and there is no significant decrease compared to the loss curves in Figures 4 and 5. Adding extra hidden layers or hidden units during network training will dramatically increase the computation and lengthen the training duration. When the number of hidden layers is larger than three or the number of hidden units is greater than or equal to 120, network training requires much more time. This result is approximately 30% longer than the test findings presented in Figures 4 and 5. The combination of hyperparameters for each network and the corresponding run times can be found in Table A1 in Appendix A. Considering the training efficiency and network prediction accuracy, hidden size and number of layers are set to 100 and 3, respectively.

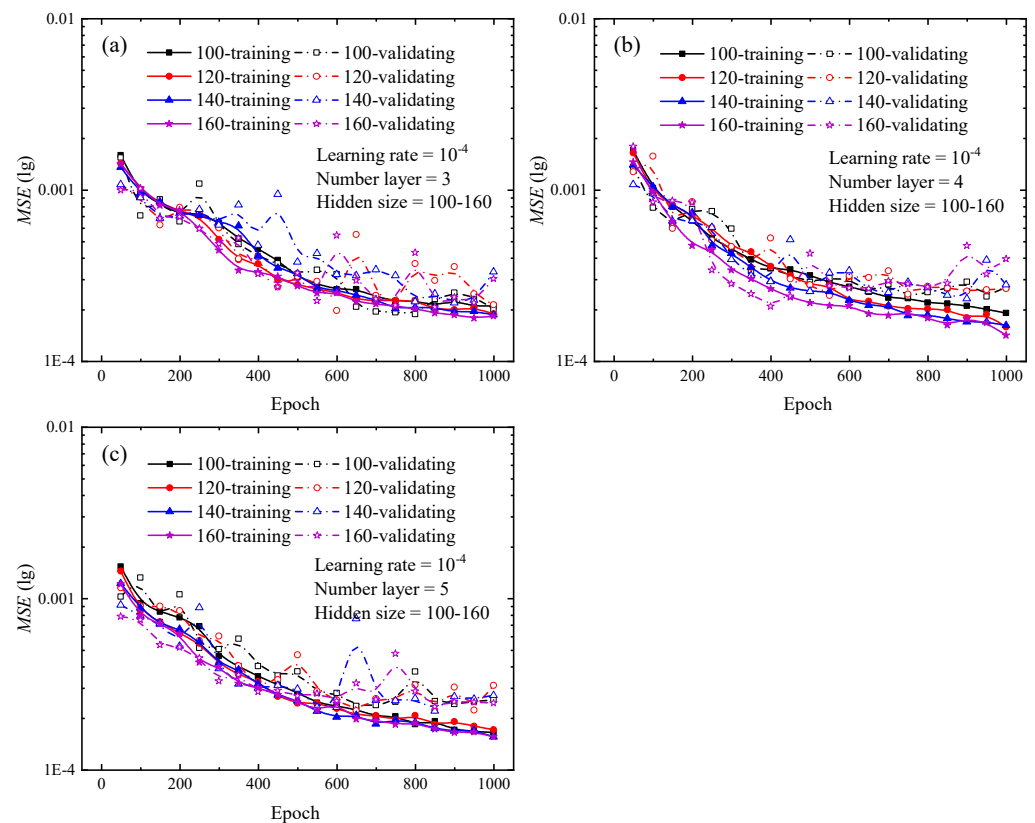


Figure 6. Loss curves of different hidden sizes during the training and validating phrases for indentation predictions. (a) The number of hidden layers is 3; (b) The number of hidden layers is 4; (c) The number of hidden layers is 5. In the legend, the cases of 100, 120, 140 and 160 represent the number of hidden sizes the LSTM neural network.

4.2. LSTM Neural Network Performance Evaluation

To ensure the data-processing resilience of our LSTM neural network, the data used in the training and validation sets in this paper are separated. In addition, the data used for prediction are newly created by the FE simulation when network training is complete. In this study, the coefficient of determination R^2 and the root mean square error (RMSE) are used to evaluate the LSTM neural network. The closer the value of R^2 is to 1, the smaller the difference between the prediction result of the network and the actual value, and the closer the value of RMSE is to zero, the more accurate the network's prediction is. Forty sets of test data are used to put into the network for the validation of the prediction effect. Figure 7 shows the prediction results and the mean absolute percentage error (MAPE) distribution. The formula for the MAPE is defined by

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{x_{pred} - x_{true}}{x_{true}} \right| \times 100\%, \quad (14)$$

Each set of test data contains 40 vertical coordinate points on the P - h curve, corresponding to the pressure values applied by the indenter during the indentation test. After the test data are calculated by the LSTM neural network, the network outputs a stress-strain curve that describes the mechanical behavior of the metal coating material.

The results in Figure 8 show that the LSTM neural network established in this paper can accurately predict the stress-strain response of metallic materials. More importantly, the variation pattern of the predicted values of the network output is consistent with the trend of the elastoplastic behavior of the metallic material described by the power-law equation. In order to observe the performance of the LSTM neural network in practice

more intuitively, a portion of the sample data is randomly selected as listed in Table 1 for further demonstration of the accuracy of stress–strain response of metallic materials.

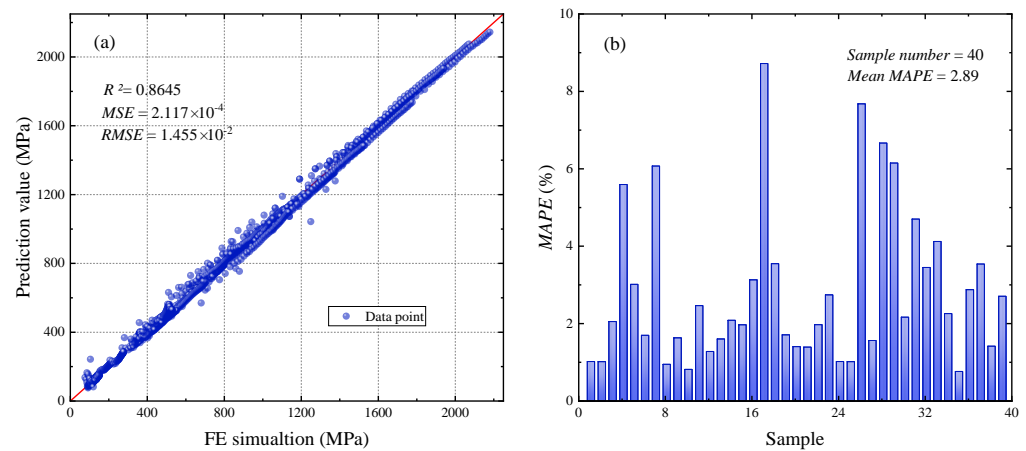


Figure 7. The prediction performance of the LSTM neural network is demonstrated. (a) indicates the prediction effect of the LSTM neural network on 40 sets of sample data, and the predicted stress values are concentrated on the diagonal line of figure. (a,b) shows the MAPE distribution of each sample.

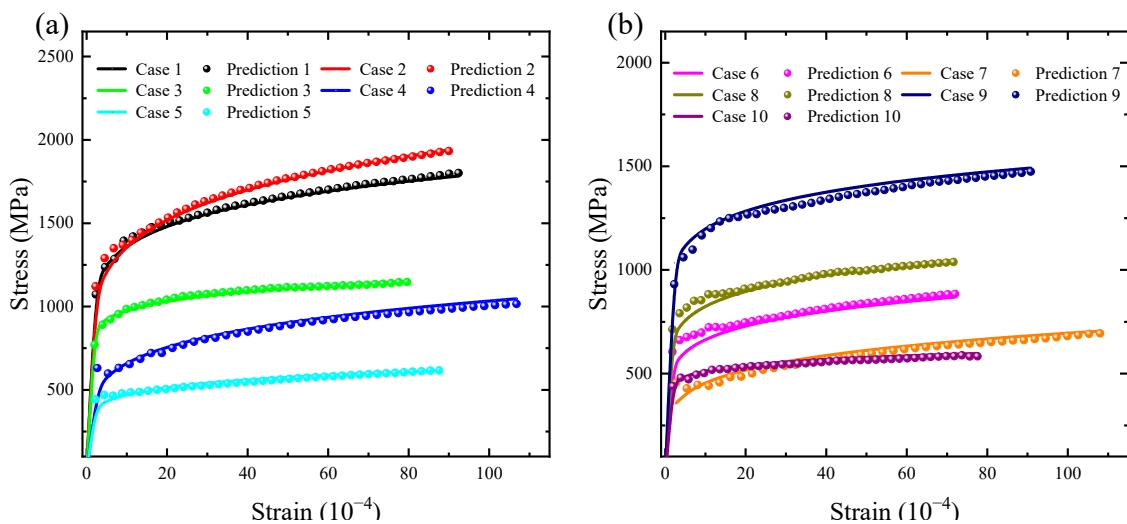


Figure 8. Comparison of predicted stress–strain responses by FE model and the proposed and LSTM neural network (a,b).

Table 1. The constitutive model parameters of the 10 specimens used in the model predictions.

Parameter	E (GPa)	R (GPa)	ϵ_y	n
Case 1	492	2.38	2.31×10^{-3}	1.22×10^{-1}
Case 2	473	2.87	2.25×10^{-3}	1.63×10^{-1}
Case 3	430	1.41	1.99×10^{-3}	7.93×10^{-2}
Case 4	191	1.62	2.67×10^{-3}	1.95×10^{-1}
Case 5	178	0.83	2.19×10^{-3}	1.24×10^{-1}
Case 6	293	1.24	1.80×10^{-3}	1.36×10^{-1}
Case 7	132	1.06	2.70×10^{-3}	1.84×10^{-1}
Case 8	373	1.44	1.79×10^{-3}	1.21×10^{-1}
Case 9	290	0.79	2.27×10^{-3}	3.06×10^{-2}
Case 10	224	1.16	1.94×10^{-3}	1.57×10^{-2}

As listed in Table 1, the Materials 1 to 4 typically represent the datasets prepared in the FE simulation and LSTM network predictions. The solid line indicates the constitutive

relationship of metallic material used for the FE simulation, and the dashed line indicates the stress–strain response predicted by the proposed LSTM neural network from the corresponding P – h response.

5. Conclusions

In this paper, an LSTM-based deep learning method for completing the inversion of P – h curves to material stress–strain response is provided. The prediction results of the network achieved satisfactory results, with a value of 0.8645 for the coefficient of determination R^2 . Moreover, based on the $MAPE$ distribution of the test samples, the prediction accuracy of the LSTM neural network is as high as 97.11. These satisfactory evaluation indicators demonstrate that the network prediction values are generally in agreement with the FE simulation values. The present study explored a wide range of materials with Young's modulus ranging from 200 GPa to 500 GPa. The deep learning model presented in this paper is applicable to indentation data for the vast majority of metallic materials. This study provides more evidence that the power-law instanton may accurately characterize the mechanical behavior of the vast majority of metallic materials. In other words, using a deep learning approach, the relationship between the material's intrinsic structure and the given material indentation data can be effectively reversed, which can then be employed directly in FE computations. The LSTM neural network inverse performance of the intrinsic structure relationship is applicable to a broader range of parameters, and the training dataset includes the majority of typical metallic coating materials, including magnesium, tungsten, and their compounds. The raw data also includes elastic and plastic data, which can comprehensively characterize the mechanical behavior of metallic materials. In comparison to traditional FE modeling and nanoindentation experiments, the LSTM neural network minimizes the consumption of experimental materials on the one hand and greatly improves computational efficiency on the other.

Due to space limitations, the deep learning method proposed in this study is only used for back-calculating the mechanical behavior of materials in the field of nanoindentation research, and only a single power-law equation is used in the generation of the dataset. It is worth noting that deep learning, as a data analysis method, is applicable for arbitrary intrinsic structure models. With the availability of the corresponding dataset, the matching numerical relationships can be derived by network operations. Therefore, it is a desirable way to use deep learning methods to predict the service life of coating structures and even the development of cracks in future research.

Author Contributions: Conceptualization, X.L., X.D. and J.L.; methodology, X.L., X.D., R.D. and Y.S.; software, X.D. and Y.S.; validation, X.D.; formal analysis, J.L.; investigation, X.D. and Y.S.; resources, Y.S.; data curation, Y.S.; Writing—original draft preparation, X.L. and X.D.; writing—review & editing, Y.S. and C.C.; visualization, X.D.; supervision, X.L. and Y.S.; project administration, Y.S.; funding acquisition, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Guangdong Basic and Applied Basic Research Foundation (No. 2023A1515011170), National Natural Science Foundation of China (No. 51508464), the National Key Laboratory Foundation 2022-JCJQ-LB-006 (No. 6142411232212), and the Regional Collaboration Project of Shanxi Province (No. 2022104041101122).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is unavailable due to privacy.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. LSTM neural network different hyperparameter combination and running time.

Hyperparameter	Number Layer	Hidden Size	Training Time (s)
Combination 1	1	20	6.99×10^3
Combination 2	1	40	7.05×10^3
Combination 3	1	60	7.10×10^3
Combination 4	1	80	7.11×10^3
Combination 5	2	20	7.44×10^3
Combination 6	2	40	7.45×10^3
Combination 7	2	60	7.41×10^3
Combination 8	2	80	7.38×10^3
Combination 9	3	20	8.06×10^3
Combination 10	3	40	7.88×10^3
Combination 11	3	60	7.90×10^3
Combination 12	3	80	7.88×10^3
Combination 13	3	100	7.97×10^3
Combination 14	3	120	8.06×10^3
Combination 15	3	140	1.07×10^4
Combination 16	3	160	1.08×10^4
Combination 17	4	100	8.34×10^3
Combination 18	4	120	8.46×10^3
Combination 19	4	140	1.23×10^4
Combination 20	4	160	1.24×10^4
Combination 21	5	100	8.85×10^3
Combination 22	5	120	9.05×10^3
Combination 23	5	140	1.38×10^4
Combination 24	5	160	1.39×10^4

References

- Yang, Z.; Chen, J.; He, G.; Liang, X.; Luo, S. Mechanical Properties and Sand Erosion Damage Mechanism of TiN/Ti Multilayer Coatings after Thermal Cycling Treatment. *J. Mater. Eng. Perform.* **2022**, 1–11. [\[CrossRef\]](#)
- Zeng, Y.; Chen, G.; Wu, C.; Pan, X.; Lin, F.; Xu, L.; Zhao, F.; He, Y.; He, G.; Chen, Q. Thin-Film Platinum Resistance Temperature Detector with a SiCN/Yttria-Stabilized Zirconia Protective Layer by Direct Ink Writing for High-Temperature Applications. *ACS Appl. Mater. Interfaces* **2023**, *15*, 2172–2182. [\[CrossRef\]](#) [\[PubMed\]](#)
- Rahimov, E. Investigation o Wear Mechanisms in Abradable-Blade Contacts. Ph.D. Thesis, University of Sheffield, Sheffield, UK, 2022.
- Belbellaa, Y.; Kribes, N.; Yallese, M.A.; Lekmine, H.; Boutabba, S.; Bezazi, A. Comparative Study to Optimize Surface Roughness of the Titanium Alloy Ti-6Al-4V by Applying Taguchi, RSM and TLBO Methods. *Period. Polytech. Mech. Eng.* **2023**, *67*, 1–11. [\[CrossRef\]](#)
- Dosdat, L.; Petitjean, J.; Vietoris, T.; Clauzeau, O. Corrosion Resistance of Different Metallic Coatings on Press-Hardened Steels for Automotive. *Steel Res. Int.* **2011**, *82*, 726–733. [\[CrossRef\]](#)
- Fischer-Cripps, A.C. A review of analysis methods for sub-micron indentation testing. *Vacuum* **2000**, *58*, 569–585. [\[CrossRef\]](#)
- Hay, J.L. Instrumented Indentation Testing. In *ASM Handbook of Mechanical Testing and Evaluation*; ASM International: Materials Park, OH, USA, 2000.
- Oliver, W.C.; Pharr, G.M. An improved technique for determining hardness and elastic modulus using load and displacement sensing indentation experiments. *J. Mater. Res.* **1992**, *7*, 1564–1583. [\[CrossRef\]](#)
- Oliver, W.C.; Pharr, G.M. Measurement of hardness and elastic modulus by instrumented indentation: Advances in understanding and refinements to methodology. *J. Mater. Res.* **2004**, *19*, 3–20. [\[CrossRef\]](#)
- Fischer-Cripps, A. Critical review of analysis and interpretation of nanoindentation test data. *Surf. Coatings Technol.* **2006**, *200*, 4153–4165. [\[CrossRef\]](#)
- Rauchs, G.; Bardon, J. Identification of elasto-viscoplastic material parameters by indentation testing and combined finite element modelling and numerical optimization. *Finite Elements Anal. Des.* **2011**, *47*, 653–667. [\[CrossRef\]](#)
- Cheng, Y.-T.; Cheng, C.-M. Scaling approach to conical indentation in elastic-plastic solids with work hardening. *J. Appl. Phys.* **1998**, *84*, 1284–1291. [\[CrossRef\]](#)
- Campbell, J.; Thompson, R.; Dean, J.; Clyne, T. Experimental and computational issues for automated extraction of plasticity parameters from spherical indentation. *Mech. Mater.* **2018**, *124*, 118–131. [\[CrossRef\]](#)

14. Sharp, M.; Ak, R.; Hedberg, T., Jr. A survey of the advancing use and development of machine learning in smart manufacturing. *J. Manuf. Syst.* **2018**, *48*, 170–179. [[CrossRef](#)] [[PubMed](#)]
15. Kale, A.P.; Wahul, R.M.; Patange, A.D.; Soman, R.; Ostachowicz, W. Development of Deep Belief Network for Tool Faults Recognition. *Sensors* **2023**, *23*, 1872. [[CrossRef](#)] [[PubMed](#)]
16. Wei, J.; Chu, X.; Sun, X.; Xu, K.; Deng, H.; Chen, J.; Wei, Z.; Lei, M. Machine learning in materials science. *Infomat* **2019**, *1*, 338–358. [[CrossRef](#)]
17. Shrestha, A.; Mahmood, A. Review of Deep Learning Algorithms and Architectures. *IEEE Access* **2019**, *7*, 53040–53065. [[CrossRef](#)]
18. Voulodimos, A.; Doulamis, N.; Doulamis, A.; Protopapadakis, E. Deep Learning for Computer Vision: A Brief Review. *Comput. Intell. Neurosci.* **2018**, *2018*, 7068349. [[CrossRef](#)]
19. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 53. [[CrossRef](#)]
20. Bengio, Y.; Courville, A. Deep Learning of Representations. In *Handbook on Neural Information Processing*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 1–28.
21. Chan, H.-P.; Samala, R.K.; Hadjiiski, L.M.; Zhou, C. Deep Learning in Medical Image Analysis. In *Deep Learning in Medical Image Analysis: Challenges and Applications*; Springer: Cham, Switzerland, 2020; pp. 3–21.
22. Kayalibay, B.; Jensen, G.; van der Smagt, P. CNN-based segmentation of medical imaging data. *arXiv* **2017**, arXiv:1701.03056.
23. Arena, P.; Basile, A.; Bucolo, M.; Fortuna, L. Image processing for medical diagnosis using CNN. *Nucl. Instrum. Methods Phys. Res. Sect. A Accel. Spectrom. Detect. Assoc. Equip.* **2003**, *497*, 174–178. [[CrossRef](#)]
24. Chen, S.; Shi, D.; Sadiq, M.; Cheng, X. Image Denoising with Generative Adversarial Networks and its Application to Cell Image Enhancement. *IEEE Access* **2020**, *8*, 82819–82831. [[CrossRef](#)]
25. Antoniou, A.; Storkey, A.; Edwards, H. Data augmentation generative adversarial networks. *arXiv* **2017**, arXiv:1711.04340.
26. Jin, Y.; Zhang, J.; Li, M.; Tian, Y.; Zhu, H.; Fang, Z. Towards the automatic anime characters creation with generative adversarial networks. *arXiv* **2017**, arXiv:1708.05509.
27. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Comput.* **2019**, *31*, 1235–1270. [[CrossRef](#)] [[PubMed](#)]
28. Staudemeyer, R.C.; Morris, E.R. Understanding LSTM—A tutorial into long short-term memory recurrent neural networks. *arXiv* **2019**, arXiv:1909.09586.
29. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF models for sequence tagging. *arXiv* **2015**, arXiv:1508.01991.
30. Smagulova, K.; James, A.P. A survey on LSTM memristive neural network architectures and applications. *Eur. Phys. J. Spec. Top.* **2019**, *228*, 2313–2324. [[CrossRef](#)]
31. Sherstinsky, A. Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network. *Phys. D Nonlinear Phenom.* **2020**, *404*, 132306. [[CrossRef](#)]
32. Graves, A.; Jaitly, N.; Mohamed, A.-r. Hybrid speech recognition with deep bidirectional LSTM. In Proceedings of the 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, 8–12 December 2013.
33. Zhao, J.; Mao, X.; Chen, L. Speech emotion recognition using deep 1D & 2D CNN LSTM networks. *Biomed. Signal Process. Control.* **2018**, *47*, 312–323. [[CrossRef](#)]
34. Jatakar, K.H.; Mulgund, G.; Patange, A.D.; Deshmukh, B.B.; Rambhad, K.S. Multi-Point Face Milling Tool Condition Monitoring Through Vibration Spectrogram and LSTM-Autoencoder. *Int. J. Perform. Eng.* **2022**, *18*, 570. [[CrossRef](#)]
35. Long, X.; Lu, C.; Shen, Z.; Su, Y. Identification of Mechanical Properties of Thin-Film Elastoplastic Materials by Machine Learning. *Acta Mech. Solida Sin.* **2022**, *36*, 13–21. [[CrossRef](#)]
36. Rice, J.; Rosengren, G. Plane strain deformation near a crack tip in a power-law hardening material. *J. Mech. Phys. Solids* **1968**, *16*, 1–12. [[CrossRef](#)]
37. Long, X.; Jia, Q.; Li, Z.; Wen, S. Reverse analysis of constitutive properties of sintered silver particles from nanoindentations. *Int. J. Solids Struct.* **2020**, *191*, 351–362. [[CrossRef](#)]
38. Dao, M.; Chollacoop, N.; Van Vliet, K.J.; Venkatesh, T.A.; Suresh, S. Computational modeling of the forward and reverse problems in instrumented sharp indentation. *Acta Mater.* **2001**, *49*, 3899–3918. [[CrossRef](#)]
39. Long, X.; Li, J.; Shen, Z.; Su, Y. Dimensionless Analysis to Determine Elastoplastic Properties of Thin Films by Indentation. *Coatings* **2022**, *12*, 1768. [[CrossRef](#)]
40. Tabor, D. *The Hardness of Metals*; Oxford University Press: Oxford, UK, 2000.
41. Cheng, Y.-T.; Cheng, C.-M. Analysis of indentation loading curves obtained using conical indenters. *Philos. Mag. Lett.* **1998**, *77*, 39–47. [[CrossRef](#)]
42. Fu, K.; Chang, L.; Zheng, B.; Tang, Y.; Wang, H. On the determination of representative stress–strain relation of metallic materials using instrumented indentation. *Mater. Des.* **2015**, *65*, 989–994. [[CrossRef](#)]
43. Long, X.; Shen, Z.; Jia, Q.; Li, J.; Dong, R.; Su, Y.; Yang, X.; Zhou, K. Determine the unique constitutive properties of elastoplastic materials from their plastic zone evolution under nanoindentation. *Mech. Mater.* **2022**, *175*, 104485. [[CrossRef](#)]
44. Long, X.; Jia, Q.; Shen, Z.; Liu, M.; Guan, C. Strain rate shift for constitutive behaviour of sintered silver nanoparticles under nanoindentation. *Mech. Mater.* **2021**, *158*, 103881. [[CrossRef](#)]

45. Wu, W.; Qin, F.; An, T.; Chen, P. Experimental and Numerical Investigation of Mechanical Properties of Electroplating Copper Filled in Through Silicon Vias. *IEEE Trans. Compon. Packag. Manuf. Technol.* **2015**, *6*, 23–30. [[CrossRef](#)]
46. Long, X.; Du, C.; Li, Z.; Guo, H.; Yao, Y.; Lu, X.; Hu, X.; Ye, L.; Liu, J. Finite Element Analysis to the Constitutive Behavior of Sintered Silver Nanoparticles Under Nanoindentation. *Int. J. Appl. Mech.* **2018**, *10*, 1850110. [[CrossRef](#)]
47. Meng, D.; Sun, L. Some New Trends of Deep Learning Research. *Chin. J. Electron.* **2019**, *28*, 1087–1091. [[CrossRef](#)]
48. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
49. Li, W.; Wu, H.; Zhu, N.; Jiang, Y.; Tan, J.; Guo, Y. Prediction of dissolved oxygen in a fishery pond based on gated recurrent unit (GRU). *Inf. Process. Agric.* **2020**, *8*, 185–193. [[CrossRef](#)]
50. Nakayama, S.; Arai, S. Dnn-lstm-crf model for automatic audio chord recognition. In Proceedings of the International Conference on Pattern Recognition and Artificial Intelligence, Union, NJ, USA, 15–17 August 2018.
51. Sujadevi, V.G.; Soman, K.P.; Vinayakumar, R.; Sankar, A.U.P. Anomaly Detection in Phonocardiogram Employing Deep Learning. In *Computational Intelligence in Data Mining*; Springer: Singapore, 2019; pp. 525–534. [[CrossRef](#)]
52. Wang, F.; Cen, J.; Yu, Z.; Deng, S.; Zhang, G. Research on a hybrid model for cooling load prediction based on wavelet threshold denoising and deep learning: A study in China. *Energy Rep.* **2022**, *8*, 10950–10962. [[CrossRef](#)]
53. Wang, J.; Li, X.; Li, J.; Sun, Q.; Wang, H. NGCU: A New RNN Model for Time-Series Data Prediction. *Big Data Res.* **2021**, *27*, 100296. [[CrossRef](#)]
54. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.