JMEPEG
https://doi.org/10.1007/s11665-024-10167-5

1059-9495/$19.00

ORIGINAL RESEARCH ARTICLE

Machine Learning- and Finite Element-Based Temperature- and Rate-Dependent Plasticity Model: Application to the Tensile Behavior

Bo Zhang, Yang Yang, Hao Wu, Yida Zhang, Quanyi Wang, Hong Zhang, Yongjie Liu, and Qingyuan Wang

Submitted: 30 April 2024 / Revised: 28 August 2024 / Accepted: 15 September 2024

This paper aims to investigate the in-depth relationship between plastic hardening models based on J2 plasticity theory with temperature and strain rate using machine learning methods. A data set related to temperature and strain rate is constructed by randomly generating equivalent plastic strain increments. To replace the conventional process of solving equivalent plastic strain increments using Newton's iterative method under isotropic hardening and associated flow rules, artificial neural network (ANN) and support vector regression (SVR) models are developed to predict equivalent plastic strain increments efficiently. The hyperparameters of these models are systematically optimized to identify the most suitable configurations. After determining the optimal hyperparameters, the model is trained. The plastic hardening models based on ANN and SVR are then implemented within the Abaqus User MATerial (UMAT) subroutine, and their performance is validated. For this purpose, a single element and tensile test simulation were applied to evaluate the accuracy of ANN and SVR. The results indicate that the ANN model's performance initially improves with increasing temperature but subsequently deteriorates. In contrast, the strain rate appears to have minimal impact on the ANN model's performance. On the other hand, the SVR model's performance remains unaffected by variations in both temperature and strain rate. Furthermore, the SVR model demonstrates higher accuracy, requires fewer hyperparameters and is more suitable for this application. Nonetheless, the ANN model's results remain within acceptable limits.

Keywords artificial neural network, finite element analysis, machine learning, plasticity model, support vector regression

1. Introduction

Plastic hardening theory is a fundamental framework in material mechanics, used to describe the processes of plastic deformation in metals and other materials (Ref 1, 2). This theory is essential for predicting the mechanical behavior of materials under various conditions, such as changes in temperature and strain rate, which are common in real-world applications. Recent advancements in computational modeling and experimental techniques have enhanced our understanding of plastic hardening, leading to the development of more accurate constitutive models capable of predicting material behavior across a wide range of conditions (Ref 3). For example, the integration of thermomechanical coupling into constitutive models has significantly improved predictions of plastic hardening in materials exposed to high temperatures and dynamic loading conditions. This approach accounts for the interplay between thermal and mechanical factors, offering a more comprehensive understanding of material behavior (Ref 4-6). Additionally, the influence of strain rate on plastic hardening has been extensively studied. High strain rate deformation often results in different hardening responses compared to low strain rates, due to factors like adiabatic heating and the limited time available for dislocation motion (Ref 7, 8). These models provide more precise predictions of material performance, thereby contributing to the optimization of material design and manufacturing processes.

Machine learning aims to enable computer systems to perform specific tasks without explicit programming by learning patterns and rules from data (Ref 9, 10). This technology has become integral to contemporary science and technology, finding applications across various fields (Ref 11). In recent years, the continuous advancements in machine learning have introduced new approaches to the study of metal plastic hardening. Dong Phill Jang et al. (Ref 12) proposed a constitutive model based on artificial neural network (ANN) to predict the elastoplastic behavior of J2 plasticity. Hongchun Shang et al. (Ref 13) used an artificial neural network (ANN) model to characterize the dynamic hardening behavior of 5182-O aluminum alloy plate under the coupling effect of strain rate

Bo Zhang, Yang Yang, Hao Wu, Yida Zhang, Quanyi Wang, Hong Zhang, and **Yongjie Liu**, Failure Mechanics and Engineering Disaster Prevention Key Laboratory of Sichuan Province, College of Architecture and Environment, Sichuan University, Chengdu 610065, China; and Key Laboratory of Deep Underground Science and Engineering, Ministry of Education, Sichuan University, Chengdu 610065, China; and **Qingyuan Wang**, Failure Mechanics and Engineering Disaster Prevention Key Laboratory of Sichuan Province, College of Architecture and Environment, Sichuan University, Chengdu 610065, China; Key Laboratory of Deep Underground Science and Engineering, Ministry of Education, Sichuan University, Chengdu 610065, China; and School of Architecture and Civil Engineering, Chengdu University, Chengdu 610106, China. Contact e-mail: zzhanghong@scu.edu.cn.

Published online: 08 October 2024

and temperature. Nikolaos N. Vlassis et al. (Ref 14) constructed a new deep learning framework, which is dedicated to the training of interpretative elastoplastic models, achieving thermodynamic consistency and excellent learning ability. Annan Zhang et al. (Ref 15) successfully expressed the stress–strain response of levy von Mises material with isotropic hardening through the neural network model and effectively simulated in the element and structure simulation. Burcu Tasdemir et al. (Ref 16) captured the complex behavior of materials under random non-monotonic uniaxial loading through the neural network model trained by machine learning and achieved accurate prediction, which was successfully applied to the elastic–plastic response of commercial pure titanium. Xin Li et al. (Ref 17) proposed a strengthened constitutive model based on machine learning method, combining data-driven concept and basic plasticity theory. In the study of mechanical response of Ti-6Al-4 V titanium alloy, the accurate prediction highly correlated with the experimental results is achieved, which reveals the potential of this method in computational mechanics.

However, to the best of our knowledge, no research has thoroughly explored the relationship between plastic hardening models based on J2 plasticity theory with respect to temperature and strain rate using machine learning methods. In response to this gap, this paper begins by detailing the process of gathering experimental data. It then introduces the J2 plasticity hardening theory, covering key elements such as the yield function, associative plastic flow rule and its incremental formulation. Following this, the paper provides an overview of machine learning, including the acquisition of datasets and the fundamental principles of two machine learning models: artificial neural networks (ANN) and support vector regression (SVR). The finite element analysis results of the machine learning-based plastic hardening model are then examined, with a focus on the effects of temperature, strain rate and strain on both ANN and SVR models. Their respective strengths and weaknesses are analyzed, culminating in a summary that looks forward to future developments and potential next steps in this area of research.

The main innovations of this study are as follows:

1) This study is based on the J2 plasticity hardening theory and uses machine learning models to predict the incremental equivalent plastic strain. Predicting using the incremental method is more suitable for three-dimensional finite element simulations.
2) This study applies both ANN and SVR machine learning models and discusses the advantages and disadvantages of each. Additionally, it investigates the impact of strain, temperature and strain rate on both models.
3) This study proposes a novel method for generating datasets. By processing the raw experimental data, meet the training needs of machine learning models based on J2 plastic hardening theory.

2. Constitutive Model and Finite Element Implementation

2.1 Plastic Constitutive Theory

2.1.1 Yield Criterion. The yield criterion is used to judge whether the material is in the yield state. The von Mises yield function is defined as follows:

$$f = \sigma_e - \sigma_y \tag{Eq 1}$$

where $\sigma_e = \sqrt{\frac{1}{2}\left[(\sigma_1-\sigma_2)^2+(\sigma_1-\sigma_3)^2+(\sigma_2-\sigma_3)^2\right]}$ is von Mises equivalent stress (Ref 18) and σ_y is the yield stress. So, the yield criterion is:

$$\begin{cases} f<0: \text{ Elastic deformation} \\ f=0: \text{ Plastic deformation} \end{cases} \tag{Eq 2}$$

In theory, there is no state where $f > 0$. If $f > 0$, the yield function needs to be modified to ensure that $f = 0$.

Geometrically, in the three-dimensional principal stress space, the yield criterion is expressed as a cylinder, which is called the yield surface. If the current stress state is inside the yield surface, corresponding to $f < 0$, no yield occurs; if the current stress state is on the yield surface, corresponding to $f = 0$, the material is just at the point of yielding. If the current stress state is outside the yield surface, corresponding to $f > 0$, it is proved that the current stress state will produce plastic strain. It is necessary to modify the yield surface so that the stress state is just on the yield surface (Ref 19).

2.1.2 Hardening Theory. Hardening theory is the characterization of the subsequent yield surface of a material after yielding. It can be divided into three categories: isotropic hardening theory, follow-up hardening theory and mixed hardening theory (Ref 20). In the process of uniaxial tension, the hardening of the material is linear isotropic hardening.

The isotropic hardening theory considers that the loaded yield surface expands uniformly and isotropically, and the yield stress is related to the equivalent plastic strain, and the equivalent stress is only related to the stress state:

$$f(\boldsymbol{\sigma}, p) = \sigma_e(\boldsymbol{\sigma}) - \sigma_y(p) \tag{Eq 3}$$

Under different temperatures and strain rates, the constitutive model and material property will change. So, the yield stress is not only a function of the equivalent stress, but also a function of the equivalent stress, temperature and strain rate:

$$f(\boldsymbol{\sigma}, p, T, S) = \sigma_e(\boldsymbol{\sigma}) - \sigma_y(p, T, S) \tag{Eq 4}$$

2.1.3 Associative Plastic Flow Rule. For associative flow, the direction of the plastic strain rate tensor is considered to be perpendicular to the tangent direction at the load point on the yield surface (Ref 21) which could be expressed by yield function:

$$d\boldsymbol{\varepsilon}^p = d\rho \frac{\partial f}{\partial \boldsymbol{\sigma}} \tag{Eq 5}$$

where $d\boldsymbol{\varepsilon}^p$ is the plastic strain increment and $\boldsymbol{\sigma}$ is the stress tensor. This shows that the direction of the plastic strain increment is determined by $\frac{\partial f}{\partial \boldsymbol{\sigma}}$, and the magnitude is determined by the plastic multiplier $d\rho$.

The plastic flow direction of the first component is as follows:

$$\frac{\partial f}{\partial \sigma_1} = \frac{\frac{1}{2}[(\sigma_1 - \sigma_2) + (\sigma_1 - \sigma_3)]}{\sqrt{\frac{1}{2}\left[(\sigma_1 - \sigma_2)^2 + (\sigma_1 - \sigma_3)^2 + (\sigma_2 - \sigma_3)^2\right]}} = \frac{3}{2}\frac{\sigma_1'}{\sigma_e} \quad \text{(Eq 6)}$$

Other components have similar results, written in tensor form as follows:

$$d\boldsymbol{\varepsilon}^p = d\rho \frac{3}{2}\frac{\sigma'}{\sigma_e} \quad \text{(Eq 7)}$$

where $\sigma' = \boldsymbol{\sigma} - \frac{1}{3}(\sigma_1 + \sigma_2 + \sigma_3)$ is the deviatoric stress tensor and von Mises stress σ_e can be calculated by σ':

$$\sigma_e = \sqrt{\frac{3}{2}\sigma' : \sigma'} \quad \text{(Eq 8)}$$

Next, consider the equivalent plastic strain increment dp:

$$dp = \sqrt{\frac{2}{9}\left[\left(d\varepsilon_1^p - d\varepsilon_2^p\right)^2 + \left(d\varepsilon_1^p - d\varepsilon_3^p\right)^2 + \left(d\varepsilon_2^p - d\varepsilon_3^p\right)^2\right]} \quad \text{(Eq 9)}$$

Substitute Equation (5) into Equation (7), and find: $dp = d\rho$. Equation (5) can be written as follows:

$$d\varepsilon^p = dp \frac{3}{2}\frac{\sigma'}{\sigma_e} \quad \text{(Eq 10)}$$

2.2 Finite Element Implementation

2.2.1 Incremental Constitutive Equation. If the current load step is n, the strain increment at a certain point in the $n + 1$ increment step can be obtained through the equilibrium equation in increment form. At this time, all known quantities are as follows: strain tensor $\boldsymbol{\varepsilon}_n$, stress tensors $\boldsymbol{\sigma}_n$, plastic strain tensor $\boldsymbol{\varepsilon}p$ n and equivalent plastic strain p_n of step n and strain increment $d\boldsymbol{\varepsilon}_{n+1}$ of step $n + 1$.

According to the generalized Hooke's law:

$$\boldsymbol{\sigma}_{n+1} = 2G\boldsymbol{\varepsilon}_{n+1}^e + \lambda \mathrm{tr}\left(\boldsymbol{\varepsilon}_{n+1}^e\right)\mathbf{1} \quad \text{(Eq 11)}$$

where $G = \frac{E}{2(1+v)}$, $\lambda = \frac{vE}{(1+v)(1-2v)}$ and E is Young's model, and v is Poisson's ratio. Elastic strain can be decomposed into:

$$\boldsymbol{\varepsilon}_{n+1}^e = \boldsymbol{\varepsilon}_n^e + d\boldsymbol{\varepsilon}_{n+1}^e = \boldsymbol{\varepsilon}_n^e + d\boldsymbol{\varepsilon}_{n+1} - d\boldsymbol{\varepsilon}_{n+1}^p \quad \text{(Eq 12)}$$

So:

$$\boldsymbol{\sigma}_{n+1} = 2G\left(\boldsymbol{\varepsilon}_n^e + d\boldsymbol{\varepsilon}_{n+1} - d\boldsymbol{\varepsilon}_{n+1}^p\right) + \lambda \mathrm{tr}\left(\boldsymbol{\varepsilon}_n^e + d\boldsymbol{\varepsilon}_{n+1} - d\boldsymbol{\varepsilon}_{n+1}^p\right)\mathbf{1} \quad \text{(Eq 13)}$$

Because of the principle of plastic incompressibility, $\mathrm{tr}\left(d\boldsymbol{\varepsilon}_{n+1}^p\right) = 0$:

$$\boldsymbol{\sigma}_{n+1} = 2G\left(\boldsymbol{\varepsilon}_n^e + d\boldsymbol{\varepsilon}_{n+1}\right) + \lambda \mathrm{tr}\left(\boldsymbol{\varepsilon}_n^e + d\boldsymbol{\varepsilon}_{n+1}\right)\mathbf{1} - 2Gd\boldsymbol{\varepsilon}_{n+1}^p \quad \text{(Eq 14)}$$

$2G\left(\boldsymbol{\varepsilon}_n^e + d\boldsymbol{\varepsilon}_{n+1}\right) + \lambda \mathrm{tr}\left(\boldsymbol{\varepsilon}_n^e + d\boldsymbol{\varepsilon}_{n+1}\right)\mathbf{1}$ is the trial stress, and write it as $\boldsymbol{\sigma}$tr $n + 1$:

$$\sigma_{n+1} = \sigma_{n+1}^{tr} - 2Gd\varepsilon_{n+1}^p = \sigma_{n+1}^{tr} - 2Gdp_{n+1}\frac{3}{2}\frac{\sigma_{n+1}'}{\sigma_{e,n+1}} \quad \text{(Eq 15)}$$

Trace both sides of the equation at the same time:

$$tr(\sigma_{n+1}) = tr\left(\sigma_{n+1}^{tr}\right) \quad \text{(Eq 16)}$$

$$\sigma_{n+1}' + \frac{1}{3}tr(\sigma_{n+1})1 = \sigma_{n+1}^{tr} - 3Gdp_{n+1}\frac{\sigma_{n+1}'}{\sigma_{e,n+1}} \quad \text{(Eq 17)}$$

$$\left(1 + 3G\frac{dp_{n+1}}{\sigma_{e,n+1}}\right)\sigma_{n+1}' = \sigma_{n+1}^{tr} - \frac{1}{3}tr\left(\sigma_{n+1}^{tr}\right)1 = \sigma_{n+1}^{tr'} \quad \text{(Eq 18)}$$

$$\left(1 + 3G\frac{dp_{n+1}}{\sigma_{e,n+1}}\right)^2 \sigma_{n+1}' : \sigma_{n+1}' = \sigma_{n+1}^{tr'} : \sigma_{n+1}^{tr'} \quad \text{(Eq 19)}$$

$$\sigma_{e,n+1} + 3Gdp_{n+1} = \sigma_{e,n+1}^{tr} \quad \text{(Eq 20)}$$

Eventually, our yield condition is:

$$\begin{aligned} f &= \sigma_{e,n+1} - \sigma_{y,n+1} \\ &= \sigma_{e,n+1}^{tr} - 3Gdp_{n+1} - \sigma_{y,n+1}(p_n + dp_{n+1}, T, S) = 0 \end{aligned} \quad \text{(Eq 21)}$$

This is a nonlinear equation for dp_{n+1}, which can be solved by Newton iteration method. The plastic strain increment $d\boldsymbol{\varepsilon}p$ $n + 1$ can be obtained after this:

$$d\varepsilon_{n+1}^p = dp_{n+1}\frac{3}{2}\frac{\sigma_{n+1}'}{\sigma_{e,n+1}} = dp_{n+1}\frac{3}{2}\frac{\sigma_{n+1}^{tr'}}{\sigma_{e,n+1}^{tr}} \quad \text{(Eq 22)}$$

3. Implementation Process

Read the strain increment of the current incremental step $d\boldsymbol{\varepsilon}$ $n + 1$ through the incremental form of the balance equation.

Calculate the trial stress and substitute it into the yield function to determine whether yielding has occurred:

$$\sigma_{n+1}^{tr} = \sigma_n + 2Gd\varepsilon_{n+1} + \lambda tr(d\varepsilon_{n+1})1 \quad \text{(Eq 23)}$$

$$f = \sigma_{e,n+1}^{tr} - \sigma_{y,n} = \sqrt{\frac{3}{2}\left(\sigma_{n+1}^{tr'} : \sigma_{n+1}^{tr'}\right)} - \sigma_{y,n} \quad \text{(Eq 24)}$$

If $f \leq 0$, update the stress according to elastic conditions and $dp_{n+1} = 0$, else calculate the equivalent plastic strain increment dp_{n+1} and plastic strain increment $d\boldsymbol{\varepsilon}p$ $n + 1$ through Equation (21) and Equation (22).

Then, calculate elastic strain increment $d\boldsymbol{\varepsilon}e$ $n + 1$ and stress increment $d\boldsymbol{\sigma}_{n+1}$:

$$d\boldsymbol{\varepsilon}_{n+1}^e = d\boldsymbol{\varepsilon}_{n+1} - d\boldsymbol{\varepsilon}_{n+1}^p \quad \text{(Eq 25)}$$

$$d\boldsymbol{\sigma}_{n+1} = 2Gd\boldsymbol{\varepsilon}_{n+1}^e + \lambda \mathrm{tr}\left(d\boldsymbol{\varepsilon}_{n+1}^e\right)\mathbf{1} \quad \text{(Eq 26)}$$

Update stress $\boldsymbol{\sigma}_{n+1}$ and equivalent plastic strain p_{n+1}:

$$\sigma_{n+1} = \sigma_n + d\sigma_{n+1} \quad \text{(Eq 27)}$$

$$p_{n+1} = p_n + dp_{n+1} \quad \text{(Eq 28)}$$

4. Source and Selection of Datasets

4.1 Acquisition of Datasets

According to Equation (21), to solve the nonlinear equation about dp_{n+1} when $\sigma tr\ e,n+1$, p_n , T, S and G are known is the task. At the same time, the shear modulus G can be expressed as a function of T and S. In this way, the feature dimension of the input data is reduced by one dimension, which can not only simplify the model, but also enhance the generalization ability of the model.

Set the value of p_i from 0 to 0.1, the value of temperature from 50 to 650°C and the value of strain rate from 10^{-5} to 10^{-1}. The elastic model E takes the yield stress divided by 0.002, and Poisson's ratio is 0.33. The training set, validation set and test set are selected as shown in Table 1.

Under a specific temperature and strain rate, the input and output vectors of the condition are generated as follows:

Generate dp_1 from 0 to 0.0002 randomly, and normalize it:

$$\mathbf{O}_1 = [dp_1] \tag{Eq 29}$$

where $\mathbf{O}_1$ is the first output vector. According to Equation (21), first input vector $\mathbf{I}_1$:

$$\mathbf{I}_1 = \left[0, \sigma_y(dp_1, T, S) + 3Gdp_1, T, S\right] \tag{Eq 30}$$

where σ_y is a hardening function that is related to temperature and strain rate. Similarly:

If $p_{i\text{-}1} + dp_i \geq 0.1$, end and normalize input vector and output vector. Normalization allows the preprocessed data to be limited to a certain range, thus eliminating the undesirable effects caused by singular sample data. Normalization allows the preprocessed data to be limited to a certain range, thus eliminating the undesirable effects caused by singular sample data. Normalization speeds up the speed of gradient descent to find the optimal solution and also speeds up the convergence of the training network (Ref 22, 23). In this study, Min-Mix normalization is used. For variable x, the normalization formula is:

$$x_i' = \frac{x_i - \max(x_i)}{\max(x_i) - \min(x_i)} \tag{Eq 32}$$

For the strain rate, since there is a large order of magnitude difference between the different strain rates, we first take a negative logarithm with a base of 10 and normalize it afterward. Figure 1 shows the flowchart of the process.

4.2 Selection of Yield Function

Due to the current lack of sufficiently many (30 or more) experimental data for the same material at different temperatures and strain rates, a preexisting hardening function had to be chosen to generate the data.

Johnson–Cook model was able to better describe the complex stress–strain behavior of metallic materials at high strain rates and high temperatures and only requires a small amount of experimental data for parameter fitting. Therefore, it is widely used in simulation and analysis in the fields of impact dynamics, metal forming and material processing, providing valuable reference and support for engineering design (Ref 24). Despite the utility and accuracy of the Johnson–Cook model under certain conditions, its limitations need to be taken into account when applying it, and appropriately revised and validated in the context of specific situations (Ref 25). In our study, a modified Johnson–Cook model is chosen as the hardening function as proposed by Nitin Kotkunde et al. (Ref 26). The expression is as follows:

$$\begin{cases} \mathbf{O}_2 = [dp_2] \\ \mathbf{I}_2 = \left[p_0 + dp_1, \sigma_y(dp_1 + dp_2, T, S) + 3Gdp_2, T, S\right] = \left[p_1, \sigma_y(dp_1 + dp_2, T, S) + 3Gdp_2, T, S\right] \\ \vdots \\ \mathbf{O}_i = [dp_i] \\ \mathbf{I}_i = \left[p_{i-2} + dp_{i-1}, \sigma_y(p_{i-1} + dp_i, T, S) + 3Gdp_i, T, S\right] = \left[p_{i-1}, \sigma_y(p_{i-1} + dp_i, T, S) + 3Gdp_i, T, S\right] \end{cases} \tag{Eq 31}$$

Table 1 Selections of training set, validation set and test set

	Strain rate, s^{-1}				
Temperature, °C	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
50	Training set	Training set	Training set	Test set	Training set
150	Training set	Test set	Validation set	Training set	Training set
250	Test set	Training set	Training set	Training set	Validation set
350	Training set	Training set	Training set	Validation set	Training set
450	Validation set	Training set	Training set	Training set	Training set
550	Training set	Validation set	Training set	Training set	Test set
650	Training set	Training set	Test set	Training set	Training set

$$\sigma = \left(A_1 + B_1\varepsilon + B_2\varepsilon^2\right)\left(1 + C_1 \ln \dot{\varepsilon}^*\right) \exp\left[\left(\lambda_1 + \lambda_2 \ln \dot{\varepsilon}^*\right)\left(T - T_r\right)\right] \quad \text{(Eq 33)}$$

where A_1, B_1, B_2, C_1, λ_1 and λ_2 are the material constants whose value is shown in

Table 2. And σ is von Mises stress, ε is the equivalent plastic strain, $\dot{\varepsilon}^*$ is the ratio of the strain rate to the reference strain rate, T is the temperature, and T_r is the reference temperature. The value of reference strain rate is 10^{-5}, and the value of reference temperature is 50°C.

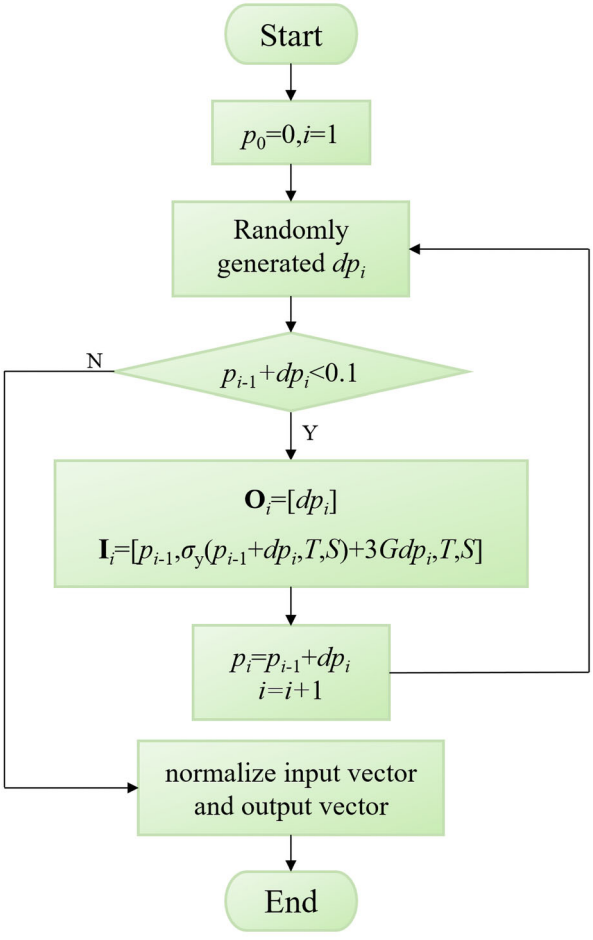

Fig. 1 Flowchart for generating a dataset

5. Machine Learning

Although dp_{n+1} can be solved by Newton iteration method, this method has great limitations. First of all, to solve dp_{n+1} by Equation (Ref 23), it is necessary to know the specific functional analytical formula of yield stress σ_y. Secondly, if the initial value is not properly selected, the result may not converge. So, machine learning is used to try to solve dp_{n+1}.

5.1 Artificial Neural Network

Artificial neural network (ANN) is inspired by biological neural network and used to calculate nonlinear problems (Ref 27). ANN consists of an input layer, an output layer and one or more hidden layers. Each hidden layer accepts the data of the previous layer, applies weights and biases to calculate after nonlinear transformation through the activation function and transmits the calculation results to the next hidden layer until reaching the output layer (Ref 28-30). Figure 2 shows an ANN structure with three hidden layers using ReLU function as the activation function.

If there is no activation function or a linear activation function is introduced, no matter how complex the constructed neural network is, the final output is a linear combination of inputs. The existence of activation functions enables neural networks to approximate nonlinear functions, which can enable neural networks to be applied to more nonlinear models (Ref 31). The Sigmoid and Tanh were early proposed activation functions. After that, ReLU and ReLU-like activation functions were proposed and showed better results (Ref 32). In this study, one of the ReLU-like activation functions, leaky ReLU, is used as the activation function. The expressions of the above four activation functions are as follows:

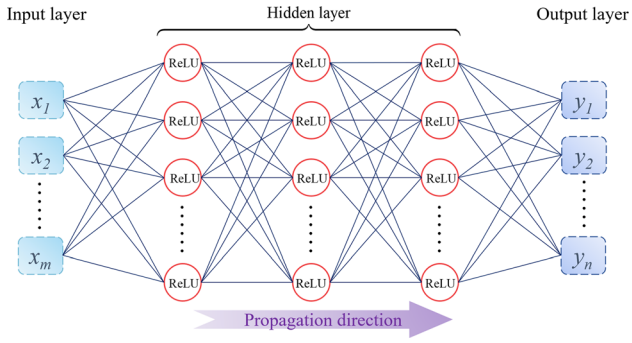

Fig. 2 Structure of artificial neural network

Table 2 Material constants for modified Johnson–Cook model

Parameter	A_1, MPa	B_1, MPa	B_2, MPa	C_1	λ_1	λ_2
Value	952.4	2977.4	−12218.3	0.0027	−0.0012	3.75×10^{-6}

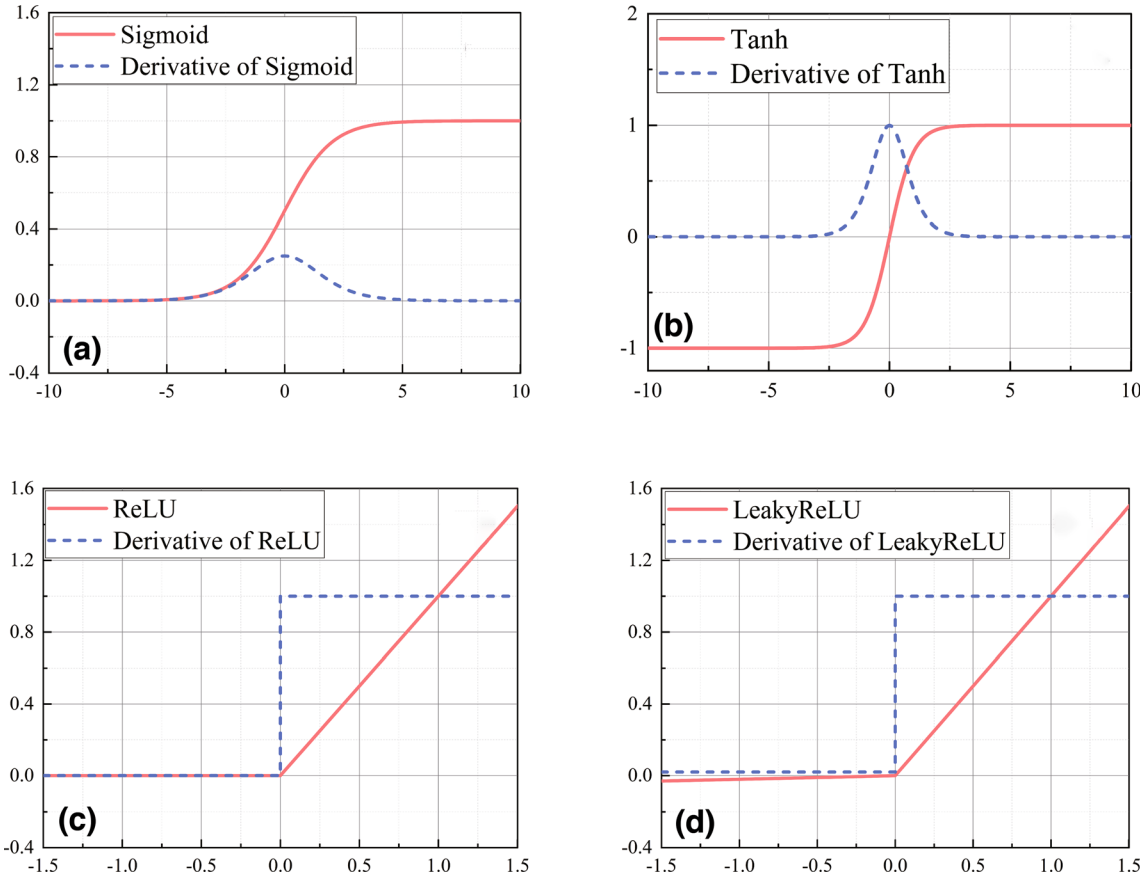

Fig. 3 (a) Sigmoid activation function, (b) Tanh activation function, (c) ReLU activation function and (d) Leaky ReLU activation function

$$g(x) = \frac{1}{1+e^{-x}}, \text{ Sigmoid}$$
$$g(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}, \text{ Tanh}$$
$$g(x) = \begin{cases} x, x \geq 0 \\ 0, x < 0 \end{cases}, \text{ ReLU} \quad \text{(Eq 34)}$$
$$g(x) = \begin{cases} x, x \geq 0 \\ \alpha x, x < 0 \end{cases}, \text{ Leaky ReLU}$$

where α is a number greater than zero, and usually, the value of a is about 0.01. Figure 3 shows the images of the above four activation functions and their derivative images. Through the image, it is not difficult to find that compared to Sigmoid and Tanh, Leaky ReLU does not suffer from serious gradient disappearance. There will be no dead neurons like ReLU.

The mean square error loss function (MSE) is expressed as:

$$loss = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \quad \text{(Eq 35)}$$

where $\hat{y}_i$ is the forecast value, y_i is the factual true value, and *loss* is the loss function. The process of ANN training is to continuously optimize the weights and biases through the gradient descent algorithm to make the loss function as small as possible. In the store library of Python, many gradient descent algorithms have been packaged to train the ANN model conveniently and quickly, such as SGD, momentum and Adam. Among them, Adam is a very popular algorithm in the field of deep learning. The empirical results show that the Adam algorithm performs well in practice and has a great advantage over other algorithms (Ref 33). So, the Adam optimizer will be chosen for the training of the ANN model.

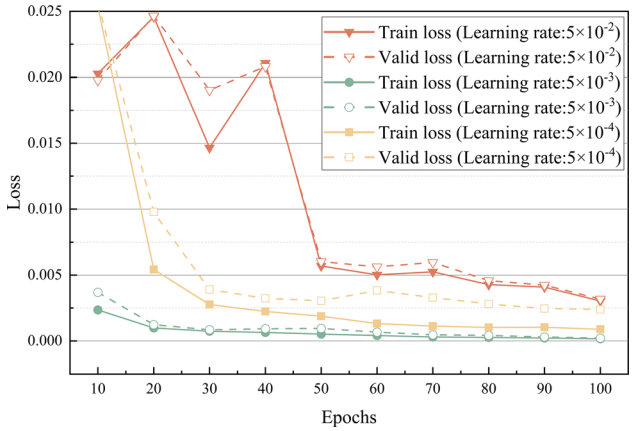

Fig. 4 The value of loss function at different learning rates.

Select the batch size to be 256, three hidden layers and 20 neurons in each layer. In this state, determine the value of learning rate. The value of loss function of the training set and validation set when the learning rate is 5×10^{-2}, 5×10^{-3} and 5×10^{-4}, respectively, is shown in Fig. 4. When the learning rate is 5×10^{-2}, the loss function cannot converge to a smaller value, and oscillation occurs. This means that a smaller learning

rate is needed. When the learning rate is 5×10^{-3}, the loss function can converge to a smaller value. When the learning rate is 5×10^{-4}, the convergence speed of the loss function is slow, and the loss of the validation set is significantly greater than the loss of the training set, resulting in overfitting. To sum up the above, the learning rate is determined to be 5×10^{-3}.

Simultaneously, we explored the impact of neural network structure on ANN performance. We adjusted the number of hidden layers and the number of neurons in each layer. The learning rate was set to 5×10^{-3}. The train loss and valid loss are shown in Fig. 5. It is evident that with this learning rate, the best training results are achieved with 3 hidden layers, each containing 20 neurons.

The finally determined hyperparameters are shown in Table 3.

5.2 Support Vector Regression (SVR)

Support vector regression is developed on the basis of support vector machines, which have strong nonlinear fitting and generalization capabilities (Ref 34, 35). SVR needs to find a hyperplane $f(\mathbf{x}) = \mathbf{W}^T\mathbf{x} + b$ so that all points are between the tube formed by the hyperplanes $f(\mathbf{x}) - \varepsilon$ and $f(\mathbf{x}) + \varepsilon$and , at the same time, ensure that the tube is as wide as possible (Ref 36-38), that is:

$$\begin{cases} \min \dfrac{1}{2}\|\mathbf{W}\|^2 \\ s.t. \left| y_i - \left(\mathbf{W}^T\mathbf{x}_i + b\right)\right| \le \varepsilon, \forall i \end{cases} \quad \text{(Eq 36)}$$

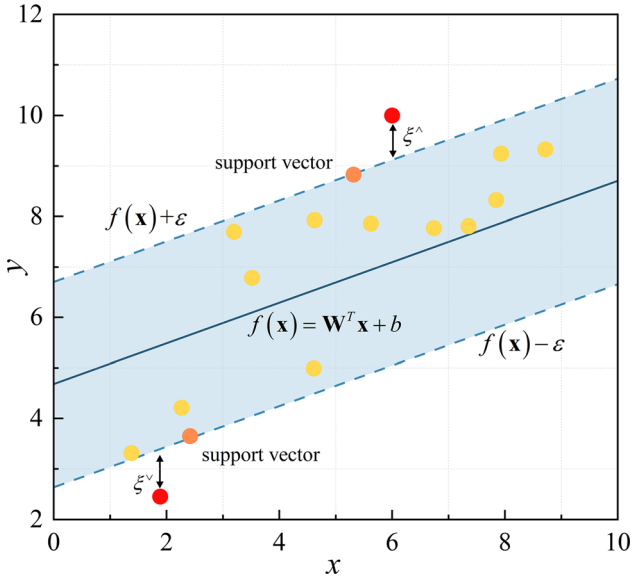

Fig. 5 The value of loss function at different network structures. L is the number of hidden layers, and N is the number of neurons per layers.

On the one hand, maximizing the interval can ensure that as many data points as possible are inside the interval band, on the other hand, it can prevent the model from over fitting.

It is not difficult to find that the determination of tube is independent of the internal points, but only related to the points on the boundary, which determine the upper and lower hyperplanes, so these points are called support vectors.

In real tasks, it is often difficult to directly determine the appropriate ε to ensure that most of the data can be in the interval band. However, SVR wants all the training data to be in the interval band, so it adds the upper and lower bound relaxation variable $\xi^\wedge$ and $\xi^\vee$, which allows some samples to be not in the interval band. If a sample is in the interval band, then $\xi^\wedge = \xi^\vee = 0$; if a sample is above the interval band, then $\xi^\wedge \neq 0, \xi^\vee = 0$; and if a sample is below the interval band, then $\xi^\wedge = 0, \xi^\vee \neq 0$ (Ref 37-41). Figure 6 shows the visualization model of SVR.

So now the optimization problem becomes:

$$\begin{cases} \min \dfrac{1}{2}\|\mathbf{W}\|^2 + C\sum_{i=1}^{n}\left(\xi_i^\wedge + \xi_i^\vee\right) \\ s.t. -\varepsilon - \xi_i^\vee \le y_i - \left(\mathbf{W}^T\mathbf{x}_i + b\right) \le \varepsilon + \xi_i^\wedge, \forall i \\ \xi_i^\vee \ge 0, \xi_i^\wedge \ge 0, \forall i \\ \xi_i^\vee \times \xi_i^\wedge = 0, \forall i \end{cases} \quad \text{(Eq 37)}$$

where C is the penalty factor and ε is the tube size, which are two important hyperparameters. C indicates the importance of the loss caused by outliers (Ref 42), and ε reflects the tolerance of deviation. And n is the number of samples.

If the regression effect of our data in the existing dimension is not good, so kernel functions are introduced to map the data to the high-dimensional space. Gaussian kernel function is a commonly used and effective kernel function. The specific method of using kernel function and the determination of SVR optimal solution are relatively complex, so you can refer to previous related articles (Ref 43, 44). Finally, the regression equation is:

$$\begin{aligned} f(\mathbf{x}) &= \sum_{i=1}^{m} \boldsymbol{\alpha}_i \kappa(\mathbf{x}_i - \mathbf{x}) + b \\ \kappa(\mathbf{x}_i - \mathbf{x}) &= \exp\left(-\gamma\|\mathbf{x}_i - \mathbf{x}\|^2\right) \end{aligned} \quad \text{(Eq 38)}$$

where m is the number of support vectors and $\mathbf{x}_i$ is the support vector. And γ is also a hyperparameter.

First fix the hyperparameter C to 4, which expresses the degree of penalization of the outliers. The effect from C can be balanced by adjusting ε. The validation set loss and training set loss under different γ and ε are shown in Fig. 7.

It can be found that when ε is large, the model is underfitted; as ε gets progressively smaller, the model is fitted better and better. Surprisingly, even when ε is very small, the model is not overfitted for most of γ. For γ, as it gets progressively larger, the fit of the model first gets better at $\gamma = 0.7$ where it works best

Table 3 Hyperparameters of ANN model

	Algorithm	Learning rate	Activation function	Epochs	Number of hidden layers	Number of neurons per layer
Selection	Adam	5×10^{-3}	Leaky ReLU	100	3	20

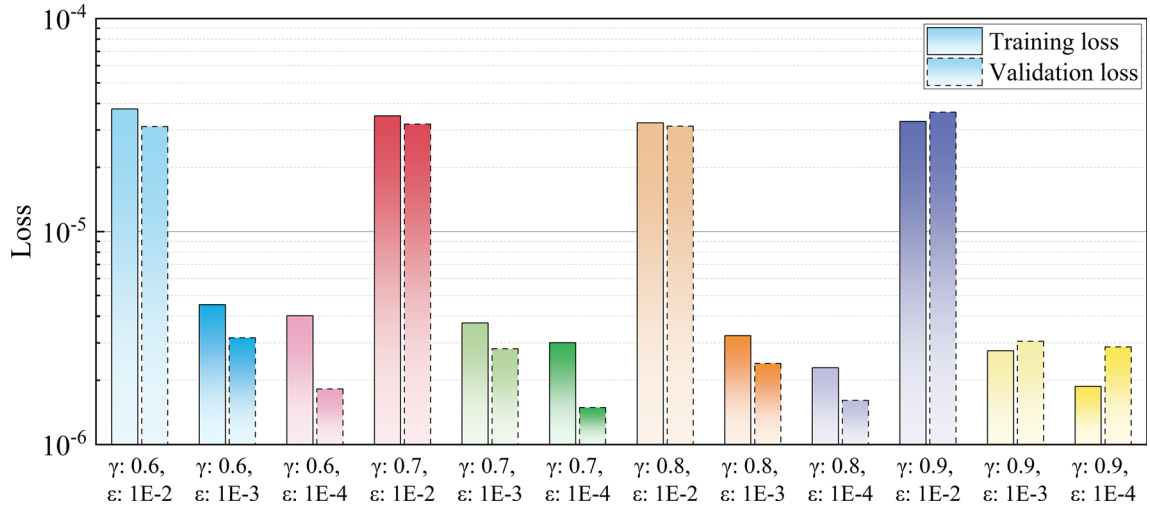

Fig. 6 The visualization model of SVR.

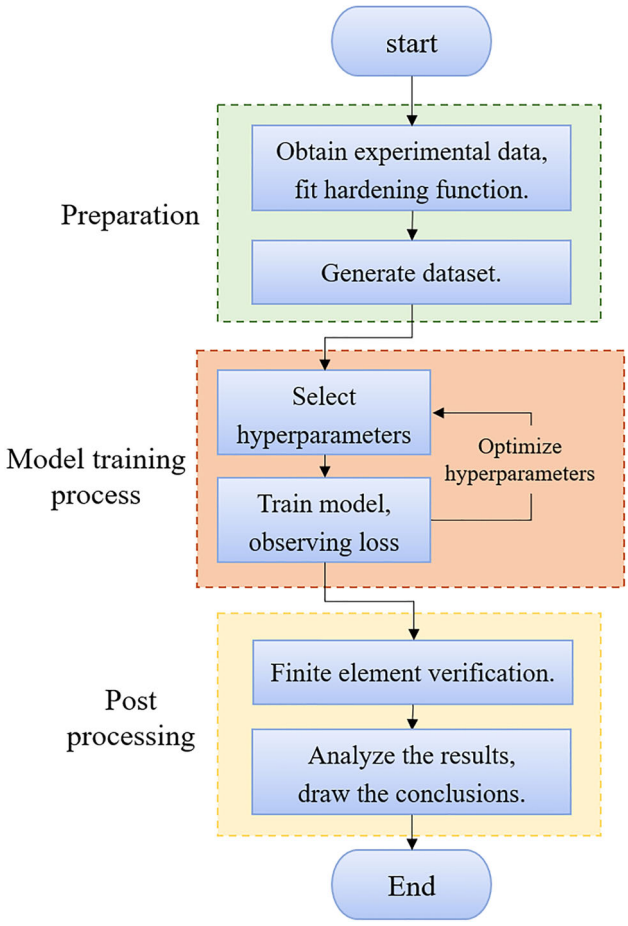

Fig. 7 The validation set loss and training set loss under different γ and ε.

Table 4 Hyperparameters of SVR model

	C	ε	γ
Selection	4	10^{-4}	0.7

and then gets worse, and even when $\gamma = 0.9$, the model has a tendency to overfit.

The optimal values of C, ε and γ of SVR model are shown in Table 4.

Figure 8 illustrates the overall technical workflow of the study, outlined as follows: Initially, experimental data are collected under various temperatures and strain rates, followed by the fitting of the hardening function. We have previously conducted experiments and published articles on the subject (Ref 45-48). However, the limited number of experiments does not satisfy the quantitative requirements of machine learning. Therefore, in this study the hardening function is considered to be the J-C constitutive model. But, in real applications, it is still necessary to obtain raw data through experiments for a new material. After establishing the input and output variables for the machine learning model, datasets are generated using the methodology mentioned above. Subsequently, optimal hyperparameters for ANN and SVR models are determined. The ANN and SVR models are then trained. Finally, finite element comparison and validation are conducted, and the results are analyzed to draw conclusions.

6. Results and Conclusion

To evaluate the accuracy of the ANN and SVR constitutive models, the trained parameters from both models were implemented into the UMAT subroutine of ABAQUS for finite element analysis. A 3D cell was selected for uniaxial extrusion in ABAQUS, with boundary conditions set to ensure that the strain did not exceed 0.1. The analysis was conducted over 500 steps, with each strain increment set to 0.002. For further details on specific parameter settings, please refer to the previous document (Ref 49).

Under different conditions, the stress–strain curves predicted by ANN and SVR and the real stress–strain curves, as well as the absolute percentage errors between them, are shown in Fig. 9. And the R^2 of ANN and SVR is shown in Table 5.

Through a systematic analysis of experimental results and finite element validation results of the machine learning

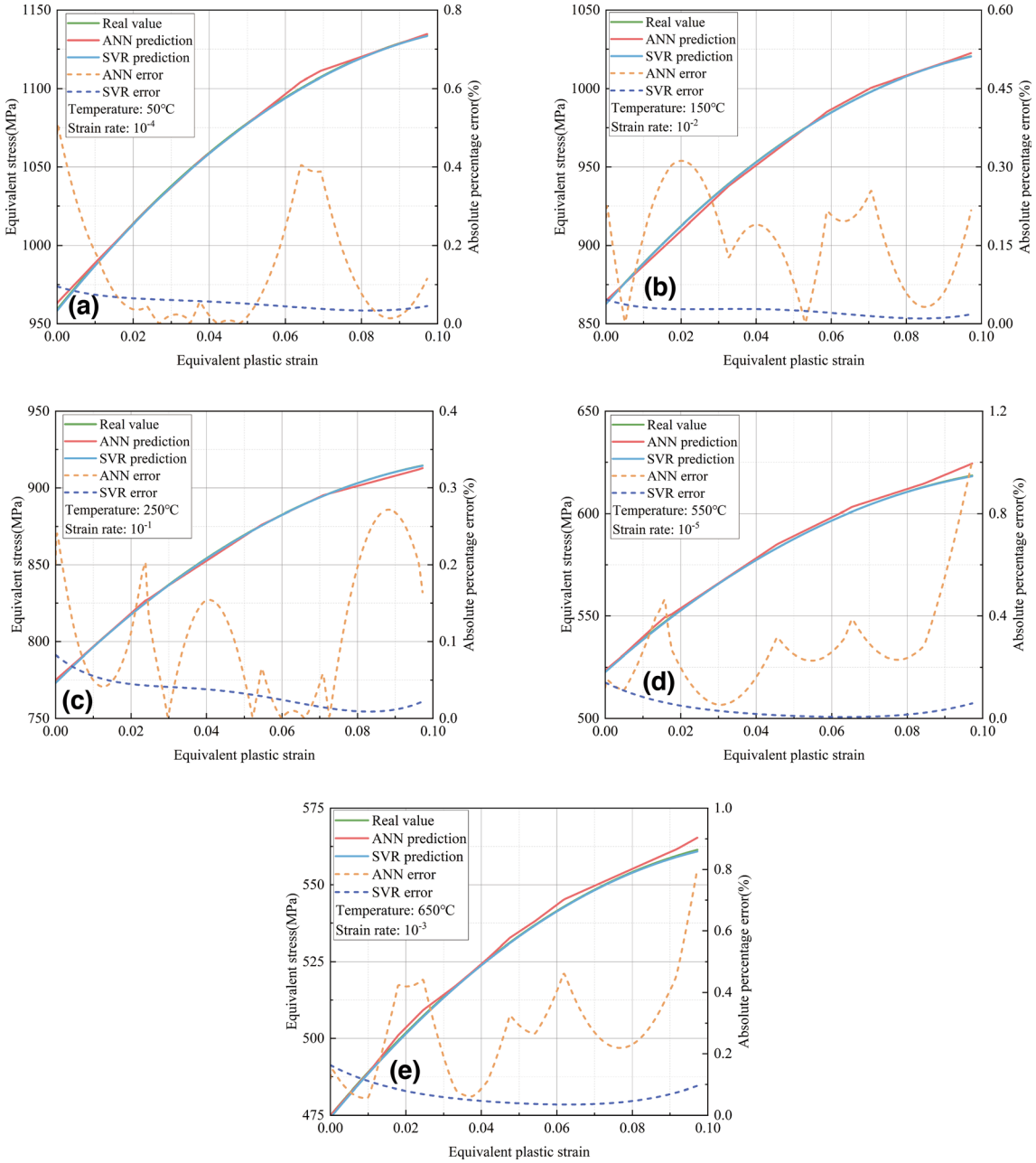

Fig. 8 The overall technical process route of the study.

constitutive model under various temperatures and strain rates, several conclusions can be drawn as follows:

(1) Overall, SVR significantly outperforms ANN on each set of test sets. This indicates that both generalization and fitting abilities of SVR model are better than ANN model. But at the same time, the effect of ANN model is good enough. Its maximum absolute percentage error is only 1%, and minimum value of R^2 also reaches 0.9951. Though SVR models are a litter better, ANN model is acceptable too.

(2) For ANN, it is obvious to find that there is a relationship between the effect of the model and the temperature, while there is no significant relationship with the strain rate. As the temperature increases, the effect of the ANN model first becomes better and then worse. When the temperature is 50°C, the absolute percent error of ANN stays near 0.5. When the temperature increases to 150°Cand 250°C, the absolute percentage error of ANN becomes smaller. At 150°C, it is maintained near 0.3, and at 250°C, it is maintained below 0.3. When the temperature is further increased to 550°Cand 650°C, the maximum absolute percentage error is greater than 0.8. Meanwhile, with the increase in temperature, R^2 also shows a trend of becoming larger and then smaller, as shown in Table 5.

(3) In addition, according to Fig. 9, when the temperature is low, the ANN does not work well in the small strain state. However, when the temperature is high, the ANN does not work well in the large strain state. This indi-

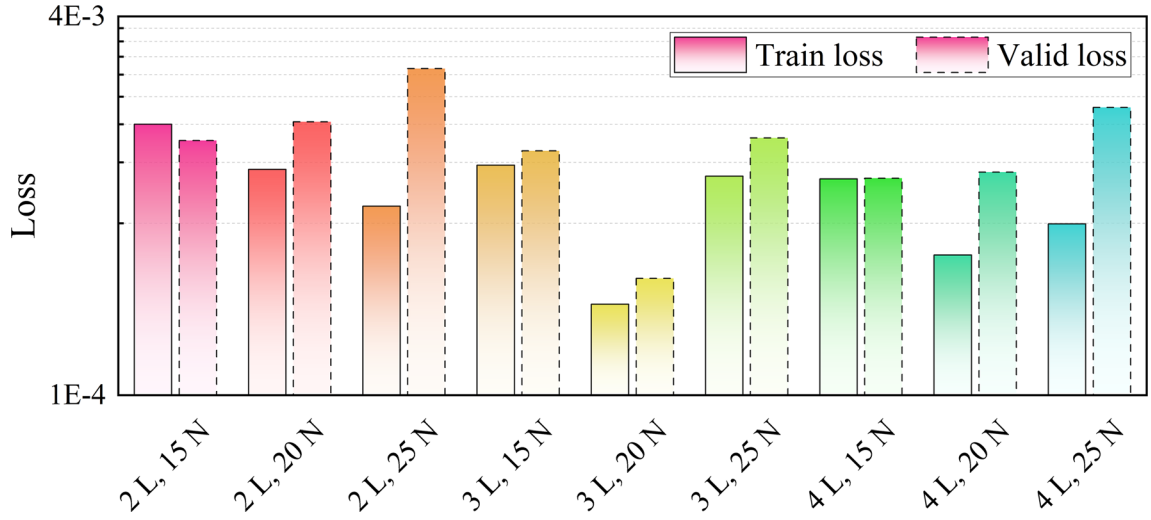

Fig. 9 Comparison of stress–strain curves for different conditions and absolute percentage errors between them. **a** Temperature is 50°C, and strain rate is 10^{-4}. **b** Temperature is 150°C, and strain rate is 10^{-2}. **c** Temperature is 250°C, and strain rate is 10^{-1}. **d** Temperature is 550°C, and strain rate is 10^{-5}. **e** Temperature is 650°C, and strain rate is 10^{-3}

Table 5 R^2 of ANN and SVR under different conditions

	***T*: 50°C*S*: 10^{-4}**	***T*: 150°C*S*: 10^{-2}**	***T*: 250°C*S*: 10^{-1}**	***T*: 550°C*S*: 10^{-5}**	***T*: 650°C*S*: 10^{-3}**
ANN R^2	0.9986	0.9987	0.9992	0.9956	0.9954
SVR R^2	0.9999	1.0000	0.9999	0.9999	0.9998

cates that the ANN model is not good enough for the critical conditions. The effect of ANN is the worst when it is in the large strain condition of the high-temperature condition, and its absolute percentage error reaches the maximum value, 1%. Moreover, as the strain rate increases, there is no significant trend in either Fig. 9 or R^2.

(4) For SVR, no significant relationship is found between its effect and temperature and strain rate. This is because SVR works exceptionally well at any strain rate and temperature and its R^2 can be considered to be 1. What can be found is that the model becomes more effective and then less effective as the strain increases, as shown in Fig. 9. This indicates that the model does not handle the strain critical conditions well.

(5) ANN excels at capturing nonlinear relationships within data. In the context of material behavior, temperature variations exhibit nonlinear characteristics that ANN can initially model effectively. However, as temperature exceeds a certain threshold, the material's behavior may become increasingly complex and unpredictable, leading to a decline in the model's performance. At elevated temperatures, the ANN model may start to overfit specific patterns within the training data, neglecting broader trends. This overfitting issue becomes more pronounced when the temperature moves beyond a critical range, impairing the model's ability to generalize to new data points and resulting in decreased performance. In this study, the effect of strain rate on the ANN model is found to be less significant compared to temperature, as the model does not strongly capture variations in strain rate. Consequently, strain rate has a minimal impact on the ANN model's predictions. On the other hand, SVR demonstrates strong robustness and generalization capabilities, particularly in handling high-dimensional data without succumbing to overfitting. By selecting an appropriate kernel function, SVR can effectively capture nonlinear relationships. However, both ANN and SVR models exhibit limitations in modeling the transition from elasticity to plasticity, leading to poor performance in predicting material behavior at small strain levels. Additionally, in large strain states, both models suffer from error accumulation during the incremental process, contributing to their diminished accuracy.

(6) SVR requires fewer hyperparameters than ANN. SVR has only 3 hyperparameters, but ANN has optimizer, learning rate, number of hidden layers and many other hyperparameters. SVR requires less time and effort to optimize the hyperparameters of the model than ANN. However, ANN is more versatile and flexible than SVR. For example, the interpretability of the model can be enhanced by adding physical information constraints, or pre-trained models can be introduced to cope with a variety of different downstream tasks.

7. Outlook

The success of machine learning in developing a temperature- and strain rate-dependent plastic hardening model is a significant step in material mechanics. It uses AI to understand the complex interactions of variables crucial for material

behavior. The application of machine learning in material modeling opens new research and engineering paths. Its success in capturing these relationships paves the way for similar studies in other materials.

Machine learning provides a dynamic and flexible framework that can adapt to new data and conditions, expanding engineering design and optimization. Ongoing advancements offer potential for model improvement. Future research may include more material characteristics, algorithm refinement, and new methodologies to enhance predictions.

Acknowledgments

The work was supported by the National Natural Science Foundation of China (No. 12272245, No. 11832007, No.12332012). The authors are sincerely grateful and supported by the fund of State Key Laboratory of Clean and Efficient Turbomachinery Power Equipment (DEC8300CG202417229A1228117).

References

1. E.W. Hart, A Phenomenological Theory for Plastic Deformation of Polycrystalline Metals, *Acta Metall.*, 1970, **18**(6), p 599–610.
2. G. Li and S. Cui, A Review on Theory and Application of Plastic Meso-Damage Mechanics, *Theoret. Appl. Fract. Mech.*, 2020, **109**, p 102686.
3. Y. Tang et al., A Review on the Dynamic-Mechanical Behaviors of High-Entropy Alloys, *Prog. Mater. Sci.*, 2023, **135**, p 101090.
4. Z. Savaedi, R. Motallebi and H. Mirzadeh, A review of Hot Deformation Behavior and Constitutive Models to Predict Flow Stress of High-Entropy Alloys, *J. Alloy. Compd.*, 2022, **903**, p 163964.
5. X. Zhang, X. Lu, J. Zhao, Q. Kan, Z. Li and G. Kang, Temperature Effect on Tensile Behavior of an Interstitial High Entropy Alloy: Crystal Plasticity Modeling, *Int. J. Plast.*, 2022, **150**, p 103201.
6. B.K. Roy, Y.P. Korkolis, Y. Arai, W. Araki, T. Iijima and J. Kouyama, Plastic Deformation of AA6061-T6 at Elevated Temperatures: Experiments and Modeling, *Int. J. Mech. Sci.*, 2022, **216**, p 106943.
7. C.B. Finfrock, M.M. Thrun, D. Bhattacharya, T.J. Ballard, A.J. Clarke and K.D. Clarke, Strain Rate Dependent Ductility and Strain Hardening in Q&P Steels, *Metall. and Mater. Trans. A.*, 2021, **52**, p 928–942.
8. J.-Y. Zhang, P. Jiang, Z.-L. Zhu, Q. Chen, J. Zhou and Y. Meng, Tensile Properties and Strain Hardening Mechanism of Cr-Mn-Si-Ni Alloyed Ultra-Strength Steel at Different Temperatures and Strain Rates, *J. Alloy. Compd.*, 2020, **842**, p 155856.
9. T. Matsunaga, H. Hongo, M. Tabuchi and R. Sahara, Suppression of Grain Refinement in Heat-Affected zone of 9Cr–3W–3Co–VNb Steels, *Mater. Sci. Eng. A*, 2016, **655**, p 168–174.
10. P. Morris, P. Morgan and K. Ridal, Metals fit for the Twenty-First Century: Processing for High Performance Applications, *Ironmaking Steelmaking*, 2012, **39**(4), p 236–243.
11. M.I. Jordan and T.M. Mitchell, Machine Learning: Trends, Perspectives, and Prospects, *Science*, 2015, **349**(6245), p 255–260.
12. Y. LeCun et al., Backpropagation Applied to Handwritten zip code Recognition, *Neural Comput.*, 1989, **1**(4), p 541–551.
13. I.H. Sarker, Machine Learning: algorithms, Real-World Applications and Research Directions, *SN comput sci.*, 2021, **2**(3), p 160.
14. D.P. Jang, P. Fazily and J.W. Yoon, Machine Learning-Based Constitutive Model for J2-Plasticity, *Int. J. Plast.*, 2021, **138**, 102919
15. H. Shang, P. Wu, Y. Lou, J. Wang and Q. Chen, Machine Learning-Based Modeling of the Coupling Effect of Strain Rate and Temperature on Strain Hardening for 5182-O Aluminum Alloy, *J. Mater. Process. Technol.*, 2022, **302**, 117501
16. N.N. Vlassis and W. Sun, Sobolev Training of Thermodynamic-Informed Neural Networks for Interpretable Elasto-Plasticity Models with Level set Hardening, *Comput. Methods Appl. Mech. Eng.*, 2021, **377**, 113695
17. A. Zhang and D. Mohr, Using Neural Networks to Represent von Mises Plasticity with Isotropic Hardening, *Int. J. Plast.*, 2020, **132**, 102732
18. I. ISO, "6892-2: 2018 Metallic Materials—Tensile Testing—Part 2: Method of Test at Elevated Temperature," *International Organization for Standardization (ISO): Geneva, Switzerland,* 2018
19. E. ISO, "6892-1: 2016 Metallic materials—Tensile testing—Part 1: Method of test at room temperature," *International Organization for Standardization (ISO): Geneva, Switzerland,* 2016
20. R.V. Mises, Mechanik der Festen Körper im Plastisch-Deformablen Zustand, *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, 1913, **1913**, p 582–592.
21. E. Madenci and S. Oterkus, Ordinary State-Based Peridynamics for Plastic Deformation According to von Mises Yield Criteria with Isotropic Hardening, *J. Mech. Phys. Solids*, 2016, **86**, p 192–219.
22. J. Sun, X. Cao, H. Liang, W. Huang, Z. Chen and Z. Li, New Interpretations of Normalization Methods in Deep Learning, *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, **34**(04), p 5875–5882.
23. C. Garbin, X. Zhu and O. Marques, Dropout Versus Batch Normalization: an Empirical Study of Their Impact to Deep Learning, *Multimedia tools appl.*, 2020, **79**(19), p 12777–12815.
24. Y. Fan, W. Wang, Z. Hao and C. Zhan, Work Hardening Mechanism Based on Molecular Dynamics Simulation in Cutting Ni–Fe–Cr Series of Ni-Based Alloy, *J. Alloy. Compd.*, 2020, **819**, 153331
25. Z. Wang, C. Jiang, B. Wei and Y. Wang, Analysis of the High Temperature Plastic Deformation Characteristics of 18CrNi4A Steel and Establishment of a Modified Johnson-Cook Constitutive Model, *Coatings*, 2023, **13**(10), p 1697.
26. N. Kotkunde, H.N. Krishnamurthy, P. Puranik, A.K. Gupta and S.K. Singh, Microstructure Study and Constitutive Modeling of Ti–6Al–4V Alloy at Elevated Temperatures, *Mater. Des.*, 2014, **1980–2015**(54), p 96–103.
27. M. Safaei, M.-G. Lee and W. De Waele, Evaluation of Stress Integration Algorithms for Elastic–Plastic Constitutive Models Based on Associated and Non-Associated Flow Rules, *Comput. Methods Appl. Mech. Eng.*, 2015, **295**, p 414–445. https://doi.org/10.1016/j.cma.2015.07.014
28. P. Ray, S.S. Reddy and T. Banerjee, Various Dimension Reduction Techniques for High Dimensional Data Analysis: a Review, *Artif. Intell. Rev.*, 2021, **54**(5), p 3473–3515. https://doi.org/10.1007/s10462-020-09928-0
29. J.O.B. Lira, H.G. Riella, N. Padoin and C. Soares, Computational Fluid Dynamics (CFD), Artificial Neural Network (ANN) and Genetic Algorithm (GA) as a Hybrid Method for the Analysis and Optimization of Micro-Photocatalytic Reactors: NOx Abatement as a case Study, *Chem. Eng. J.*, 2022, **431**, p 133771. https://doi.org/10.1016/j.cej.2021.133771
30. A. Tuan Hoang et al., A Review on Application of Artificial Neural Network (ANN) for Performance and Emission Characteristics of Diesel Engine Fueled with Biodiesel-Based Fuels, *Sustain. Energy Technol. Assessments*, 2021, **47**, p 101416. https://doi.org/10.1016/j.seta.2021.101416
31. D.A. Otchere, T.O. Arbi Ganat, R. Gholami and S. Ridha, Application of Supervised Machine Learning Paradigms in the Prediction of Petroleum Reservoir Properties: Comparative Analysis of ANN and SVM Models, *J. Petroleum Sci. Eng.*, 2021, **200**, p 108182. https://doi.org/10.1016/j.petrol.2020.108182
32. A. Apicella, F. Donnarumma, F. Isgrò and R. Prevete, A Survey on Modern Trainable Activation Functions, *Neural Netw.*, 2021, **138**, p 14–32.
33. S.R. Dubey, S.K. Singh and B.B. Chaudhuri, Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark, *Neurocomputing*, 2022, **503**, p 92–108. https://doi.org/10.1016/j.neucom.2022.06.111
34. C. Banerjee, T. Mukherjee, and E. Pasiliao, "An Empirical Study on Generalizations of the ReLU Activation Function," presented at the Proceedings of the 2019 ACM Southeast Conference, Kennesaw, GA, USA, 2019. [Online]. Available: https://doi.org/10.1145/3299815.3314450
35. H. Saad Hikmat and A. Adnan Mohsin, "COMPARISON OF OPTIMIZATION TECHNIQUES BASED ON GRADIENT DESCENT ALGORITHM: A REVIEW," *PalArch's Journal of Archaeol-*

ogy of Egypt / Egyptology, vol. 18, no. 4, pp. 2715-2743, 02/18 2021. [Online]. Available: https://archives.palarch.nl/index.php/jae/article/view/6705
36. D. A. Pisner and D. M. Schnyer, "Chapter 6 - Support vector machine," In, A. Mechelli and S. Vieira (Eds). *Machine Learning* Academic Press, 2020, pp. 101-121
37. M. Awad and R. Khanna, Support Vector Regression, *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*. M. Awad, R. Khanna Ed., Apress, Berkeley, CA, 2015, p 67–80
38. C. Cortes and V. Vapnik, Support-Vector Networks, *Mach. Learn.*, 1995, **20**(3), p 273–297. https://doi.org/10.1007/BF00994018
39. J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua and A. Lopez, A Comprehensive Survey on Support Vector Machine Classification: Applications, Challenges and Trends, *Neurocomputing*, 2020, **408**, p 189–215. https://doi.org/10.1016/j.neucom.2019.10.118
40. M. Sheykhmousa, M. Mahdianpari, H. Ghanbari, F. Mohammadimanesh, P. Ghamisi and S. Homayouni, Support Vector Machine Versus Random Forest for Remote Sensing Image Classification: a Meta-Analysis and Systematic Review, *IEEE J. Select. Topics in Appl. Earth Observ. Remote Sensing*, 2020, **13**, p 6308–6325. https://doi.org/10.1109/JSTARS.2020.3026724
41. B. Gaye, D. Zhang and A. Wulamu, Improvement of Support Vector Machine Algorithm in Big Data Background, *Math. Problems Eng.*, 2021, **2021**, p 5594899. https://doi.org/10.1155/2021/5594899
42. C. Campbell and Y. Ying, *Learning with support vector machines*. Springer Nature, 2022
43. V. D. Sánchez A, Advanced Support Vector Machines and Kernel Methods, *Neurocomputing*, 2003, **55**(1), p 5–20. https://doi.org/10.1016/S0925-2312(03)00373-4
44. M. Panahi, N. Sadhasivam, H.R. Pourghasemi, F. Rezaie and S. Lee, Spatial Prediction of Groundwater Potential Mapping Based on Convolutional Neural Network (CNN) and Support Vector Regression (SVR), *J. Hydrol.*, 2020, **588**, p 125033. https://doi.org/10.1016/j.jhydrol.2020.125033
45. T. Zou et al., Effect of Temperature on Tensile Behavior, Fracture Morphology, and Deformation Mechanisms of Nickel-Based Additive Manufacturing 939 Superalloy, *J. Alloy. Compd.*, 2023, **959**, 170559
46. M. Liu et al., Microstructure Evolution, Failure MECHANISM AND Life Prediction of Additively MANUFACTURED INconel 625 superalloy with Comparable Low Cycle Fatigue Performance, *Int. J. Fatigue*, 2024, **181**, 108142
47. Y. Jiang et al., Numerical Simulation and High Cycle Fatigue Behaviour Study on Shot Peening of MAR-M247 Nickel-Based Alloy, *Int. J. Fatigue*, 2024, **182**, 108161
48. Q. Wang et al., The Cyclic Deformation Behavior And Microstructural Evolution of 304L Steel Manufactured By Selective Laser Melting Under Various Temperatures, *Mater. Sci. Eng. A*, 2024, **891**, 145949
49. Q. Chen, N. D. Georganas, and E. M. Petriu, "Real-time Vision-based Hand Gesture Recognition Using Haar-like Features," in *2007 IEEE Instrumentation & Measurement Technology Conference IMTC 2007*, 1-3 May 2007 2007, pp. 1-6, https://doi.org/10.1109/IMTC.2007.379068