


Article

ConvLSTM-Att: An Attention-Based Composite Deep Neural Network for Tool Wear Prediction

Renwang Li ^{1,*}, Xiaolei Ye ¹, Fangqing Yang ¹ and Ke-Lin Du ² ¹ School of Mechanical Engineering, Zhejiang Sci-Tech University, Hangzhou 310018, China² Department of Electrical and Computer Engineering, Concordia University, Montreal, QC H3G 1M8, Canada

* Correspondence: renwangli@zstu.edu.cn

Abstract: In order to improve the accuracy of tool wear prediction, an attention-based composite neural network, referred to as the ConvLSTM-Att model (1DCNN-LSTM-Attention), is proposed. Firstly, local multidimensional feature vectors are extracted with the help of a one-dimensional convolutional neural network (1D-CNN), which avoids the loss of wear features caused by manual feature extraction. Then the temporal relationship learning between multidimensional feature vectors is performed by introducing a long short-term memory (LSTM) network to make up for the lack of long-short distance dependence of the captured sequence of the CNN network. Finally, an attention mechanism is applied to strengthen the ability to extract key information from tool-wearing temporal features. The proposed ConvLSTM-Att model is trained with the measured tool wear data and then performs as a tool wear predictor. The model is compared with several state-of-the-art models on the PHM tool wear data sets. It significantly outperforms the other models in terms of prediction accuracy, but with similar computational complexity.

Keywords: tool wear prediction; feature extraction; attention; LSTM; metrology



Citation: Li, R.; Ye, X.; Yang, F.; Du, K.-L. ConvLSTM-Att: An Attention-Based Composite Deep Neural Network for Tool Wear Prediction. *Machines* **2023**, *11*, 297. <https://doi.org/10.3390/machines11020297>

Academic Editor: Mark J. Jackson

Received: 17 January 2023

Revised: 12 February 2023

Accepted: 15 February 2023

Published: 16 February 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

As a result of China's vigorous promotion of the Manufacturing Industry 2025, the machinery manufacturing industry has increasingly higher requirements for intelligence [1,2]. As an important part of machinery production and processing, the degree of wear and tear on tools severely affects the accuracy of workpieces and the manufacturing costs of enterprises. Most of the traditional tool changing methods are based on experience, determining the timing of tool stopping and tool changing. Changing a tool too early will cause wastage of the tool [3], whereas changing a tool too late will reduce the quality of a workpiece and lead to scrapping. During the machining process, timely and accurate prediction of tool wear is beneficial to both improving the machining accuracy of products and reducing the manufacturing and labor costs of enterprises. Therefore, intelligent and accurate prediction of tool wear has become an important topic.

Direct and indirect measurements are the two main approaches to tool wear prediction [4]. The direct measurement approach requires off-line measurement of the tool between machining intervals, which greatly affects the coherence of machining and is difficult to apply in production practice. The indirect measurement method is primarily used to predict tool wear by mining and analyzing the relationship between the data taken during machining and the tool wear data. However, the data acquired during machining is subject to noise in the industrial environment, which reduces the validity of the data [5].

Traditional machine learning approaches, such as artificial neural networks [6], fuzzy logic [7], the hidden Markov model and support vector machine, as well as metaheuristic approaches [8,9], can be implemented for tool wear prediction, but the prediction accuracy is generally low [10]. It is difficult for traditional machine learning approaches to predict the true data directly from the measured data [11]. Therefore, some preprocessing methods

such as time-frequency domain analysis [12], principal component analysis (PCA) [13], and empirical modal decomposition (EMD) [14] are usually required for the data. However, extracted features are easy to underweight in the data that reflect tool wear, and the tool wear data are essentially time series, so the above approaches cannot tap the time-series features between different data samples.

In recent years, deep learning [6], with its excellent ability to automatically learn features, has provided a new idea for dealing with large-scale mechanical data [15]. It is widely used in the fields of computer vision [16], speech recognition [17], natural language processing [18], and mechanical fault diagnosis [19]. The tool wear prediction model based on deep learning has significantly improved the accuracy of tool wear prediction by virtue of its powerful data processing as well as feature extraction capability [19].

Convolutional neural networks (CNN) have excellent performance in processing data, text, and image recognition. They can analyze the original data and extract high-dimensional hidden features, effectively circumventing the problems that arise from hand-crafted feature extraction. Kothuru et al. [20] established a CNN-based tool condition prediction model for tool wear prediction by analyzing the spectral characteristics of the acoustic signals during machining. Xu et al. [21] used CNN to extract features from the collected vibration data during machining and designed an extended convolutional residual block and fully connected layer to achieve effective prediction of tool wear. The long short-term memory (LSTM) network is a type of recurrent neural network that effectively avoids the vanishing gradient problem in the recurrent neural network and better mines the temporal features between different temporal data sets [22]. In order to capture the long-term dependence of tool wear data, Zhao et al. [23] presented a deep LSTM model to predict tool wear by regression, confirming that LSTM has certain advantages compared with conventional recurrent neural networks in processing temporal data. Cai et al. [24] proposed a hybrid model based on LSTM to extract the temporal features from the raw sequence data through a designed stacked LSTM and finally in a nonlinear regression model to obtain the predicted tool wear. Chan et al. [25] combined CNN and LSTM and proposed an LSTM-CNN model. It uses CNN to extract tool wear features, and then mines the temporal features of tool wear by LSTM, to achieve effective prediction of tool wear. Schwendemann et al. [26] propose a deep learning method based on transfer learning. This method uses the windowed envelope, de-noising, and normalization processing of low-frequency sensor data to form the intermediate domain image and uses CNN and LSTM to perform the estimation of remaining useful life and realize effective transfer learning between different bearing types. Qiao et al. [27] collected vibration and current signals as well as tool wear as input to construct a training data set and input the features extracted by the multi-scale convolutional LSTM model to a bidirectional LSTM model to predict tool wear. It meets the requirements of high accuracy and low latency. Attention is a weight assignment mechanism that improves learning accuracy by learning to continuously update the attention weights corresponding to different features and to ignore unimportant signals [28]. Huang et al. [29] combined CNN with attention and proposed a multiscale CNN based on attention fusion, which will improve the accuracy of tool wear prediction by extracting tool wear data through multi-layer convolution as input and passing them to a multilayer attention mechanism.

The above methods consider, to some extent, the feature extraction of tool wear data in spatial and temporal dimensions and the contribution of different data to tool wear in the spatial dimension, respectively. However, they do not consider the different specifics of different data features fused with spatial and temporal dimensions for tool wear prediction. In order to enhance the extraction of key information while fusing different dimensional features of the measured data, we propose a composite neural network model based on an attention mechanism. The model uses a 1D-CNN neural network to extract multidimensional feature data. Then, in order to fuse the features in the temporal dimension, an LSTM network is introduced to extract sequential features by learning the temporal relationship between multidimensional features through an LSTM layer. Finally, the ability of the model

to obtain key wear data features is improved with the help of the attention mechanism, resulting in an efficient and accurate prediction of tool wear.

2. Composition of the Prediction Model

2.1. One-Dimensional Convolutional Neural Network (1D-CNN)

The CNNs are feedforward neural networks with special structures that excel at image and speech recognition. Among them, the one-dimensional convolutional neural network (1D-CNN) has excellent performance in processing text data [30]. The 1D-CNN structure is shown in Figure 1, which is composed of three major layers, namely, the convolutional layer, the pooling layer, and the fully connected layer [31]. The convolutional layer slides with the elements in the perceptual field at a prespecified step and makes a nonlinear mapping through the activation function to extract the feature vectors. It is the core layer of the whole network. The pooling layer reduces the dimensions of the feature vectors outputted from the convolutional layer and keeps the most significant ones locally to reduce operations. The fully connected layer is a fully connected neural network, which does a nonlinear transformation of the feature vectors to generate the specified dimensions and passes them on to generate the classification result [32].

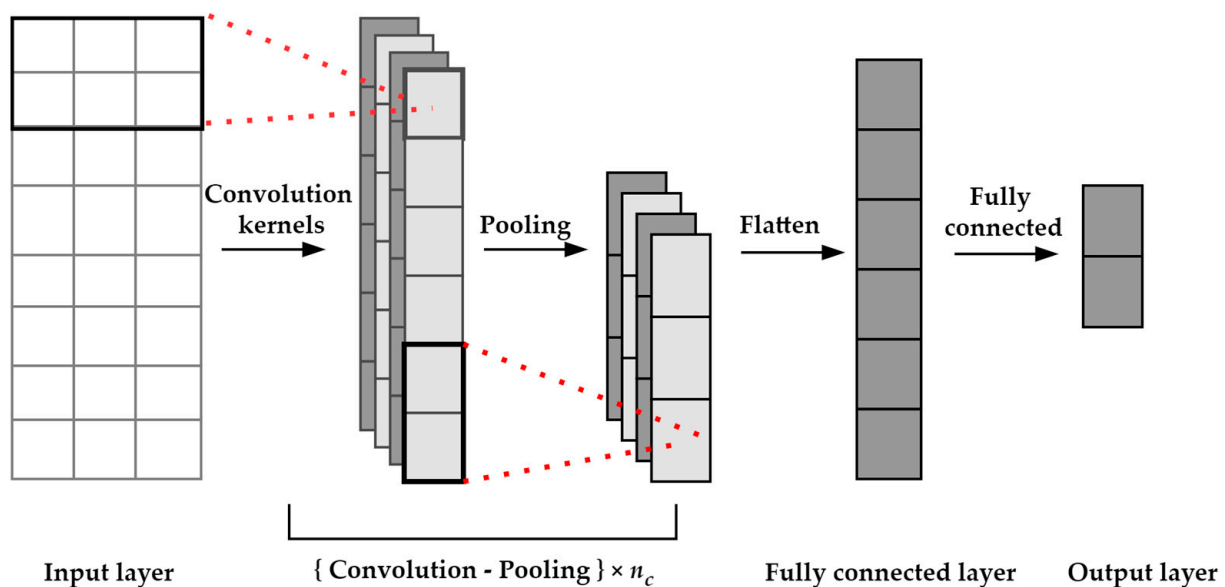


Figure 1. Network structure of the 1D-CNN. n_c is the number of the convolution-pooling layers.

2.2. Long Short-Term Memory (LSTM) Network

The LSTM is a recurrent neural network with long-term and short-term memories between feature vectors, and the memories can be dynamically adjusted with the input, which greatly solves problems such as memory degradation caused by too long sequences [33]. At the same time, LSTM can also solve the problem of vanishing gradients that exists in conventional recurrent neural networks and can better exploit the temporal relationship between tool and feature vectors. The core idea of LSTM is to introduce input gates, forgetting gates, and output gates for each memory unit [34]. The structure of LSTM is shown in Figure 2.

The forget gate decides what is deleted from the memory cell according to the previous hidden layer state $\mathbf{h}_{t-1} \in R^n$, n is the LSTM number of hidden layer neurons, with the current input $\mathbf{d}_t \in R^k$, k being the number of convolution kernels, by the sigmoid function. The input gate decides what to save from the previous hidden state \mathbf{h}_{t-1} with the current input \mathbf{d}_t according to the sigmoid function and gets a candidate parameter $\mathbf{c}'_t \in R^n$ according to the tanh function. Combining the forget gate with the input gate, the current state $\mathbf{c}_t \in R^n$ of the memory cell is updated. The output gate uses the sigmoid function to determine what is output from \mathbf{h}_{t-1} and \mathbf{d}_t . The outputs of the output gate are combined

with \mathbf{c}_t which is processed by the tanh function to selectively output the hidden layer state \mathbf{h}_t at the current time. The LSTM model is given by

$$\mathbf{f}_t = \sigma_1(\mathbf{V}_f \mathbf{h}_{t-1} + \mathbf{W}_f \mathbf{d}_t + \mathbf{b}_f), \tag{1}$$

$$\mathbf{i}_t = \sigma_1(\mathbf{V}_i \mathbf{h}_{t-1} + \mathbf{W}_i \mathbf{d}_t + \mathbf{b}_i), \tag{2}$$

$$\mathbf{c}'_t = \sigma_2(\mathbf{V}_c \mathbf{h}_{t-1} + \mathbf{W}_c \mathbf{d}_t + \mathbf{b}_c), \tag{3}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{c}'_t, \tag{4}$$

$$\mathbf{o}_t = \sigma_1(\mathbf{V}_o \mathbf{h}_{t-1} + \mathbf{W}_o \mathbf{d}_t + \mathbf{b}_o), \tag{5}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \sigma_2(\mathbf{c}_t), \tag{6}$$

where $\mathbf{f}_t \in R^n$ is the forget gate output at the time t , $\mathbf{i}_t \in R^n$ is the input gate output at time t , \mathbf{c}_t is the state of the memory cell at the time t , $\mathbf{o}_t \in R^n$ is the output gate output at the time t , $\sigma_1(\cdot)$ is the sigmoid function, $\sigma_2(\cdot)$ is usually selected as the tanh function, \mathbf{h}_t is the hidden layer state at time t , \mathbf{d}_t is the input at time t , $\mathbf{V}_f \in R^{n \times n}$, $\mathbf{V}_i \in R^{n \times n}$, $\mathbf{V}_c \in R^{n \times n}$ and $\mathbf{V}_o \in R^{n \times n}$ are the weight matrices of \mathbf{h}_t , $\mathbf{W}_f \in R^{k \times n}$, $\mathbf{W}_i \in R^{k \times n}$, $\mathbf{W}_c \in R^{k \times n}$ and $\mathbf{W}_o \in R^{k \times n}$ are the weight matrices of \mathbf{d}_t , $\mathbf{b}_f \in R^n$, $\mathbf{b}_i \in R^n$, $\mathbf{b}_c \in R^n$, $\mathbf{b}_o \in R^n$ are the bias vectors, and \odot is point-wise multiplication of two vectors.

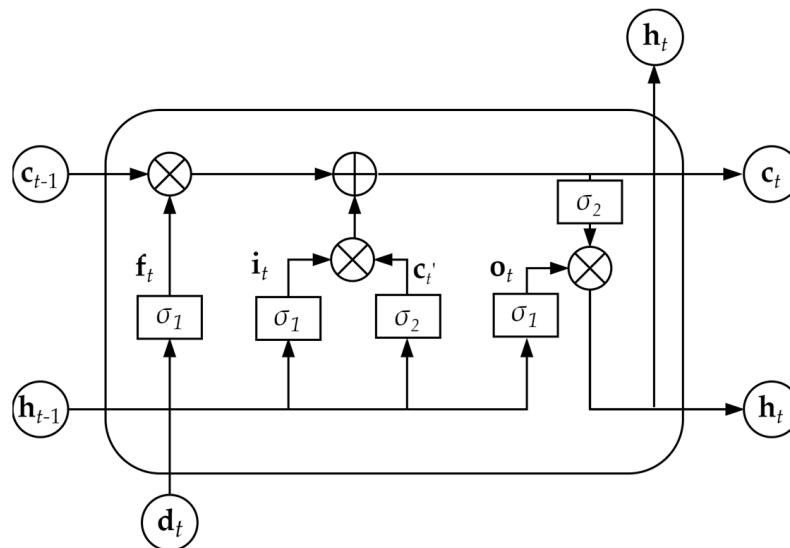


Figure 2. LSTM network structure.

2.3. Attention Mechanism

The attention mechanism is a bionic mechanism in deep learning, whose essence is to give more weight to the parts that need to be focused on [35]. The essential idea of the attention mechanism is shown in Figure 3.

The calculation of the attention mechanism can be divided into two steps. First, the degree of correlation between the state output \mathbf{h}_t of the hidden layer of the LSTM layer and the query vector $\mathbf{q} \in R^n$ is calculated using the attention score function $s(\cdot)$, and the corresponding attention weight α_t is obtained using the softmax function. Second, the output $\hat{\mathbf{y}}_t \in R^t$ of the attention layer is obtained by weighted summation based on the attention weights. The steps are given by

$$\alpha_t = \text{softmax}(s(\mathbf{h}_t, \mathbf{q})) = \frac{\exp(s(\mathbf{h}_t, \mathbf{q}))}{\sum_{i=0}^m \exp(s(\mathbf{h}_t, \mathbf{q}))}, \tag{7}$$

$$\hat{y}_t' = \sum_{i=0}^m \alpha_i h_i, \tag{8}$$

where m is the number of time steps.

Then, input \hat{y}_t' into the regression layer to get the predicted tool wear at the current time.

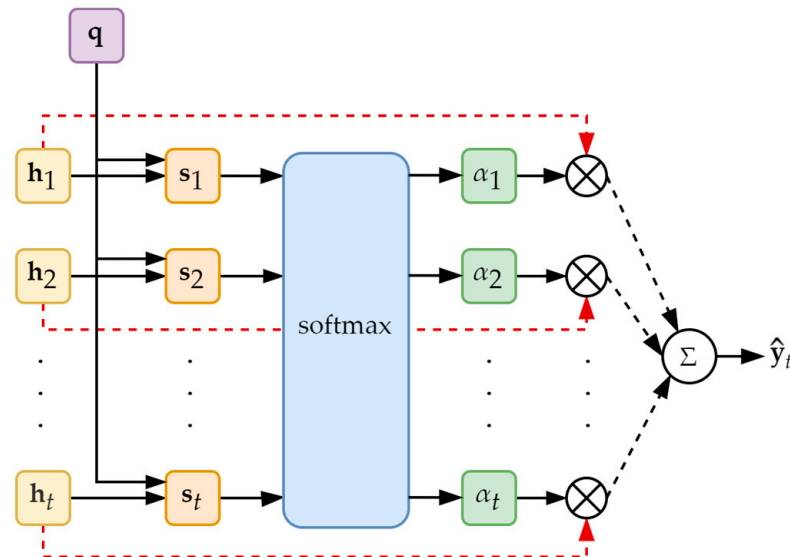


Figure 3. The attention mechanism.

3. Predicting Model

3.1. ConvLSTM-Att Model Construction

According to Zhou et al. [36], the most frequently used sensors in the study of milling process sensor configurations are force, vibration, and acoustic sensors. Therefore, in this paper, the cutting force and vibration along the x -, y -, and z -axes, and the acoustic emission at time t , $x_t^i \in R^{1 \times 7}, i \in \{1, 2, \dots, m\}$ being the tap index, are selected as the inputs of the model for training the ConvLSTM-Att model, $X_t = (x_t^1, \dots, x_t^m), X_t \in R^{m \times 7}$.

A one-dimensional CNN is first used to extract the high-dimensional features in the measured time-domain data through multiple convolutional layers. The weights of the CNN are updated by backpropagating the error calculated by the loss function. As the number of network layers increases, the problem of the vanishing gradient becomes increasingly obvious. To reduce the impact of vanishing gradients, each convolutional layer is followed by a connection layer, which includes batch normalization, ReLU, and max-pooling layers. The structure is shown in Figure 4.

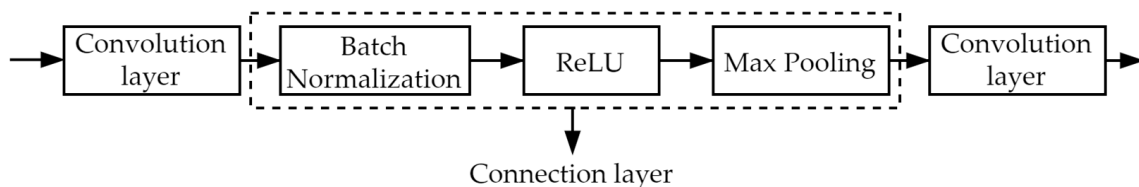


Figure 4. A connection layer following each convolutional layer.

As shown in Figure 5, the ConvLSTM-Att model first automatically mines the features in the tool wear data x_t through the 1D-CNN layer and uses them as the input of the LSTM layer. Then, through the LSTM layer, it learns the temporal relationship between the multidimensional vectors and gets the corresponding hidden layer state vector h_t . Finally, through the attention layer, the attention weight α_t of each input is calculated, and the predicted tool wear $\hat{y}_t' \in R^t$ is obtained through the nonlinear regression layer.

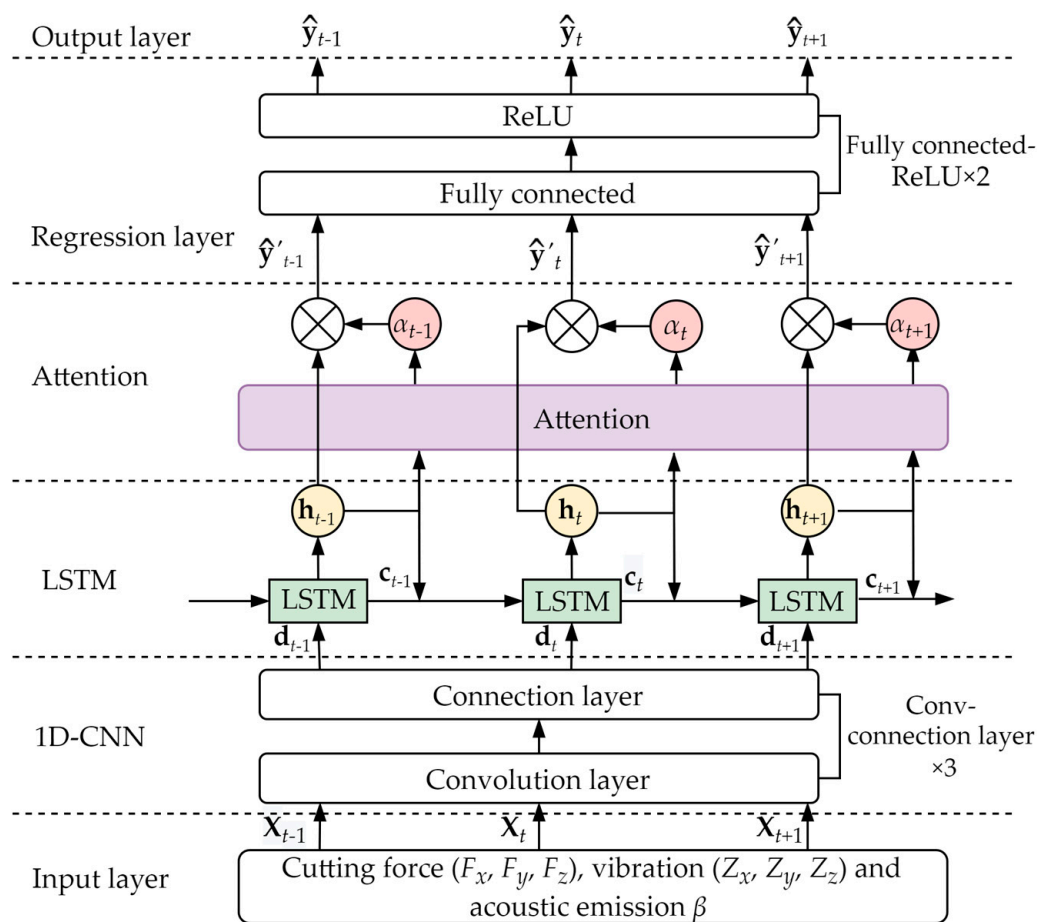


Figure 5. ConvLSTM-Att model.

3.2. The ConvLSTM-Att Model for Predicting the Tool Wear Process

All data are first normalized and processed, and then divided into different data sets. Train the ConvLSTM-Att model using the training set data. The accuracy of the model is improved by adjusting the model parameters and structure according to the validation results from the validation set. Apply the optimal ConvLSTM-Att model to the test set and output the predicted wear to evaluate the model’s performance. The specific operations are given in Algorithm 1.

Algorithm 1. Training of the ConvLSTM-Att model

Input: Tool wear data set \mathbf{D} , learning rate η , epoch T , LSTM neuron number n , maximum error ϵ .

1. All data are processed by max-min normalization.
2. Divide \mathbf{D} into the training set \mathbf{D}_{tr} , validation set \mathbf{D}_v and test set \mathbf{D}_{te} .
3. Perform training with \mathbf{D}_{tr} :

do

Adjust η, T, n, ϵ .

Train ConvLSTM-Att model with \mathbf{D}_{tr} .

Validate the currently trained model using \mathbf{D}_v .

while $(Loss(\hat{\mathbf{y}}, \mathbf{y}))$ does not converges && $Loss > \epsilon$

return the current training model.

End

4. Applying the optimal model to \mathbf{D}_{te} .

Output: $\hat{\mathbf{y}}$ for \mathbf{D}_{te}

The loss function of the model is the mean square error (MSE) function,

$$Loss(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{t} \sum_{i=0}^t (\hat{\mathbf{y}}_i - \mathbf{y}_i)^2, \quad (9)$$

where $\hat{\mathbf{y}}_i$ is the predicted wear and $\mathbf{y}_i \in R^t$ is the measured wear.

The accuracy indices are the mean absolute error (MAE), root mean square error (RMSE) and coefficient of determination R2, given by

$$MAE = \frac{1}{m} \sum_{i=1}^m |\hat{\mathbf{y}}_i - \mathbf{y}_i|, \quad (10)$$

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^m (\mathbf{y}_i - \hat{\mathbf{y}}_i)^2}, \quad (11)$$

$$R2 = 1 - \frac{\sum_{i=1}^m (\mathbf{y}_i - \hat{\mathbf{y}}_i)^2}{\sum_{i=1}^m (\mathbf{y}_i - \bar{\mathbf{y}})^2}, \quad (12)$$

where $\bar{\mathbf{y}}$ is the average of all measured wear.

4. Experimental Study and Results

4.1. Experimental Setting

The proposed method was validated using the tool wear data set published by the PHM Association in 2010 [37]. The experimental parameters are shown in Table 1.

Table 1. Experimental parameters.

Cutting Conditions	Spindle Speed/r/min	Feeding Speed/mm/min	Axial Depth of Cut/mm	Radial depth of Cut/mm	Feed Amount/mm	Sampling Fre-quency/kHz
Parameters	10,400	1555	0.2	0.125	0.001	50

The tool used for the experiments was a three-tooth end mill, and the material machined was HRC52 stainless steel. The experiments were conducted at room temperature with dry cutting—that is, no cutting fluid was used. The force and vibration data in the x -, y - and z -axes during each machining process and the acoustic emission data during the tool move process were collected using a Kistler force gauge, an acceleration sensor, and an acoustic emission sensor, respectively. During each machining process, the tool cut in the x -direction had a length of 108 mm. After each tool move, the wear of the rear face of each tool cutter flute was measured with a LEICA MZ12 microscope to get the wear of each tool move. The sensor setup is shown in Figure 6.

A total of three tools (C1, C4, and C6) were used to collect data for the experiments, each acquiring 315 samples for a total of 945 samples. Each sample contains seven components: cutting force (F_x, F_y, F_z), vibration (Z_x, Z_y, Z_z), acoustic emission β as input, and the wear of three cutter flutes as output. According to the recommendations of ISO 8688-2 (1989), the average of the wears of the three cutter flutes was taken as the measured wear of that sample, which was used as the output in the data set. Take the tool C1 as an example, as shown in Figure 7. According to the trend of tool wear, 0~50 tool moves (wear < 90 μm) were classified as the initial wear stage, 51~190 tool moves (wear < 120 μm) as the normal wear stage, and 191~315 tool moves (wear < 173 μm) as the severe wear stage.

The experiments were conducted at a sampling frequency of 50 kHz, resulting in up to 200,000 data points per sample. In order to exclude the interference of the incoming and outgoing cutters and simplify the computation, three segments of 5000 data-points were selected, as shown in Figure 8. It can be seen from Figure 8 that the maximum and minimum values of the data in each segment are similar. In order to retain the data characteristics

of each segment, the average value of the three segments was taken as the model input for training, so that each wear corresponded to tool wear data with a size of 5000×7 . In order to obtain a reliable and stable model, cross validation was used, and data sets were divided. Among the C1, C4, and C6 data sets, two of the data sets are divided into a training set and a validation set for model training and model parameter adjustment. The ratio of the training set to the verification set is 8:2. The third data set was used as a test set for model evaluation.

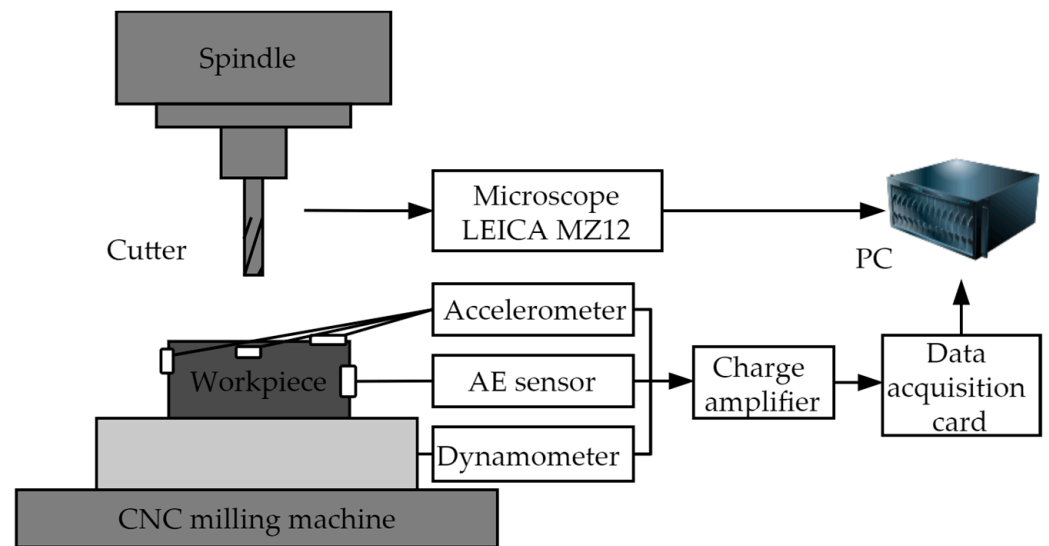


Figure 6. Experimental setup.

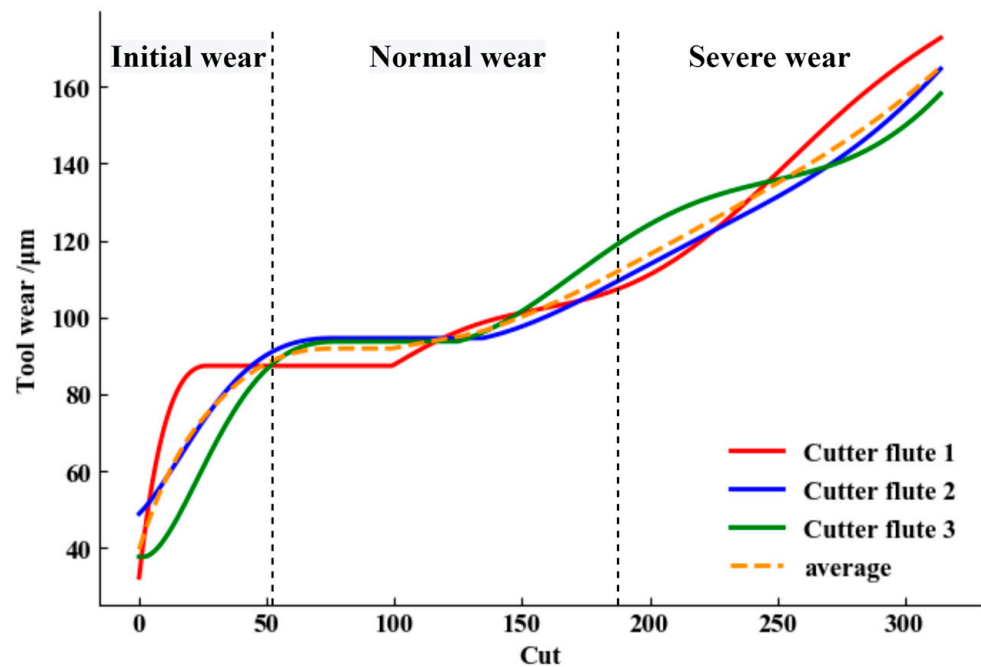


Figure 7. Division of tool wear stages.

The model is built with the PyTorch deep learning library. It is coded in python and runs on Python 3.7, CUDA 11.4, and Windows with an Intel Core i7 CPU and an NVIDIA GeForce Mx350 GPU.

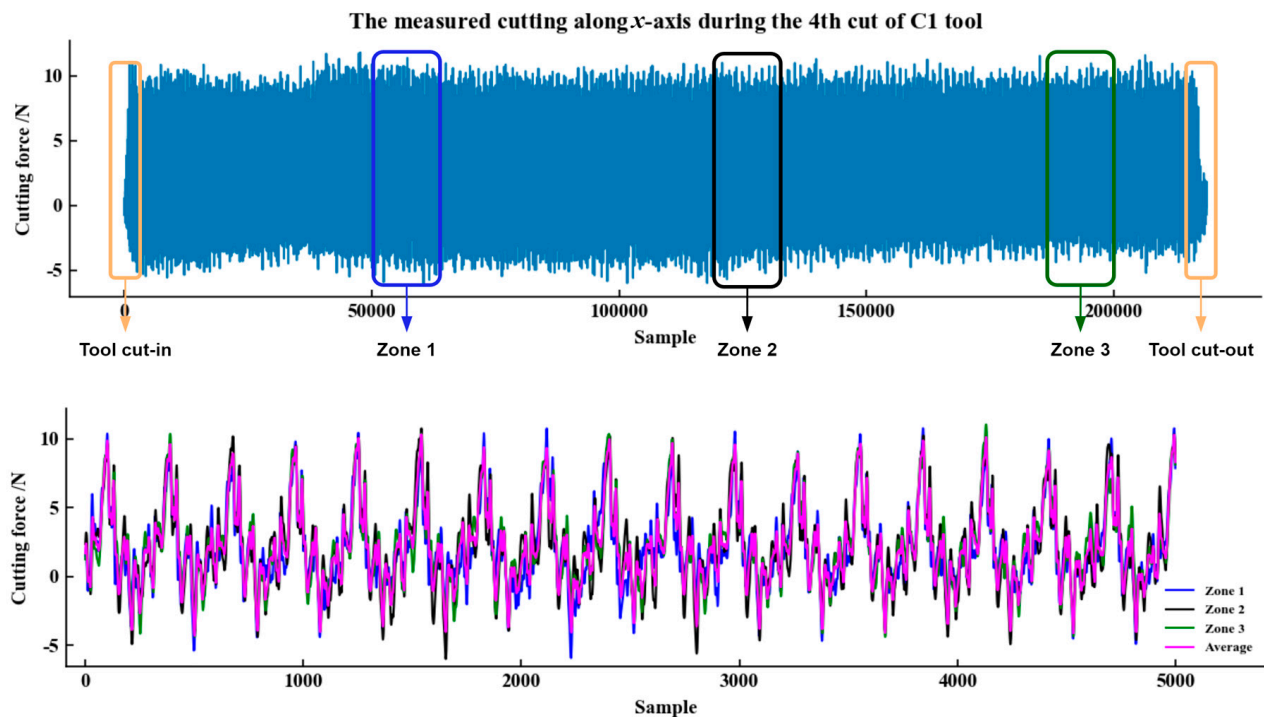


Figure 8. A close-up of the measured data.

4.2. Experimental Design and Parameter Setting

In order to verify the accuracy as well as the effectiveness of the tool wear prediction based on the proposed ConvLSTM-Att model, experiments were conducted using the same CNN, CNN-LSTM, and ConvLSTM-Att data set.

In the convolutional model, a small convolutional kernel can reduce the complexity of the model computation, and a small convolutional step size can improve the accuracy of the feature extraction results. Therefore, the size of the convolution kernel is set to (3, 7), the step length is set to 2, and the boundary fill is 1. To avoid losing too many features in the downscaling process, the pooling layer is selected as max-pooling, and the pooling size is set (4, 4). The number of epochs is 500, the initial parameter of the learning rate is 0.001, which decays as the epoch increases, and the optimizer is Adam. To improve the robustness and generalization ability of the model, a dropout layer is added to the regression layer, and the retention rate is set to 0.5. The number of LSTM hidden layer neurons is 128. The maximum error is 6.3.

4.3. Comparison and Analysis of Experimental Results

The models are trained on the training set, the trained model parameters are adjusted on the validation set, and the models are evaluated on the test set. For each model, RMSE, MAE, and R2 are calculated on each test set separately. If the error is large, it means that the model is not properly fitted, and we need to readjust the parameters or train the model structure. The smaller the error, the higher the accuracy of the model. The prediction errors of tool wear for different models on different test sets are shown in Table 2. The calculation times of the 1D-CNN, CNN-LSTM, and ConvLSTM-Att models are shown in Table 3. The performance results of different models for tool wear prediction on different test sets are shown in Figures 9–11.

Table 2. Error performance of different models on different test sets.

Models	Test Set								
	C1			C4			C6		
	RMSE	MAE	R2	RMSE	MAE	R2	RMSE	MAE	R2
RNN [23]	15.6	13.1	/	19.7	16.7	/	32.9	25.5	/
1D-CNN	10.849	8.411	0.749	19.074	16.089	0.702	15.748	13.002	0.827
Deep LSTMs [23]	12.1	8.3	/	10.2	8.7	/	18.9	15.2	/
LSTM [24]	11.4	8.5	/	11.7	8.5	/	21.2	14.6	/
HLLSTM [25]	8.0	6.6	/	7.5	6.0	/	8.8	7.1	/
CNN-LSTM	6.480	5.336	0.929	9.897	7.389	0.925	10.646	7.749	0.914
TDCovLSTM [38]	8.33	6.99	/	8.39	6.96	/	10.22	7.50	/
ConvLSTM-Att	4.251	3.218	0.976	6.224	4.610	0.970	5.716	4.056	0.971

Table 3. The calculation time of the 1D-CNN, CNN-LSTM and ConvL-Att models.

Models	Time/s		
	C1	C4	C6
1D-CNN	356.6	362.3	349.7
CNN-LSTM	589.75	598.25	593.25
ConvLSTM-Att	679.8	684.9	681.4

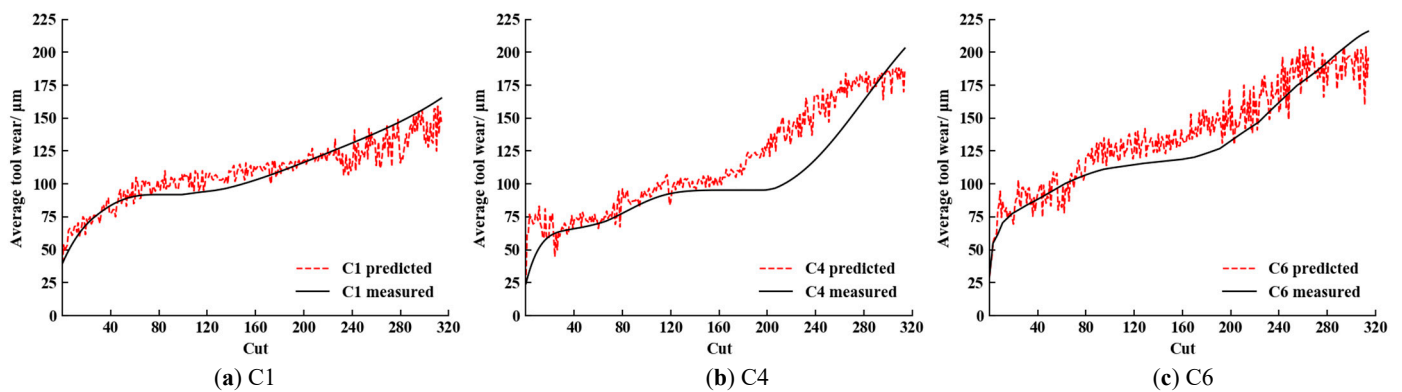


Figure 9. Predicted results of 1D-CNN model on the C1, C4 and C6 test sets.

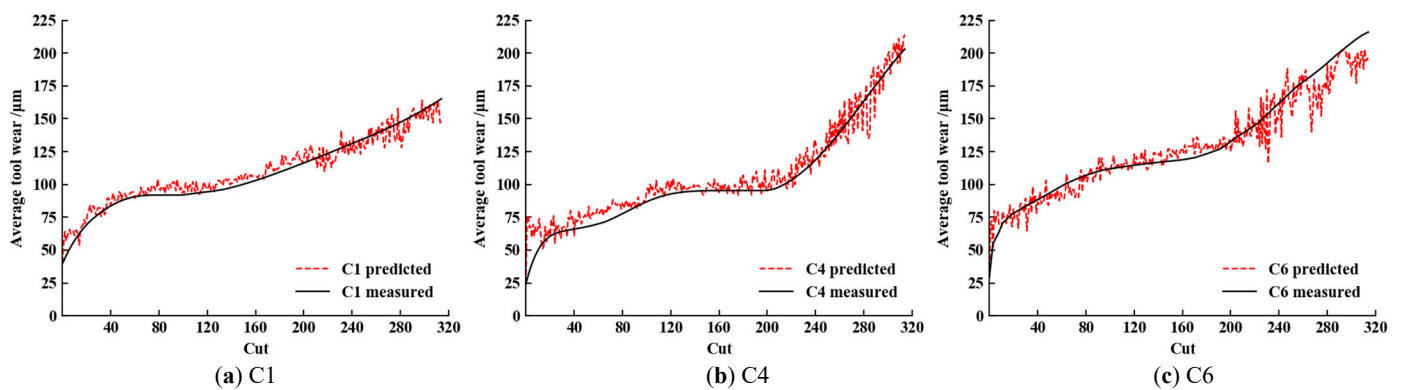


Figure 10. Predicted results of CNN-LSTM model on the C1, C4 and C6 test sets.

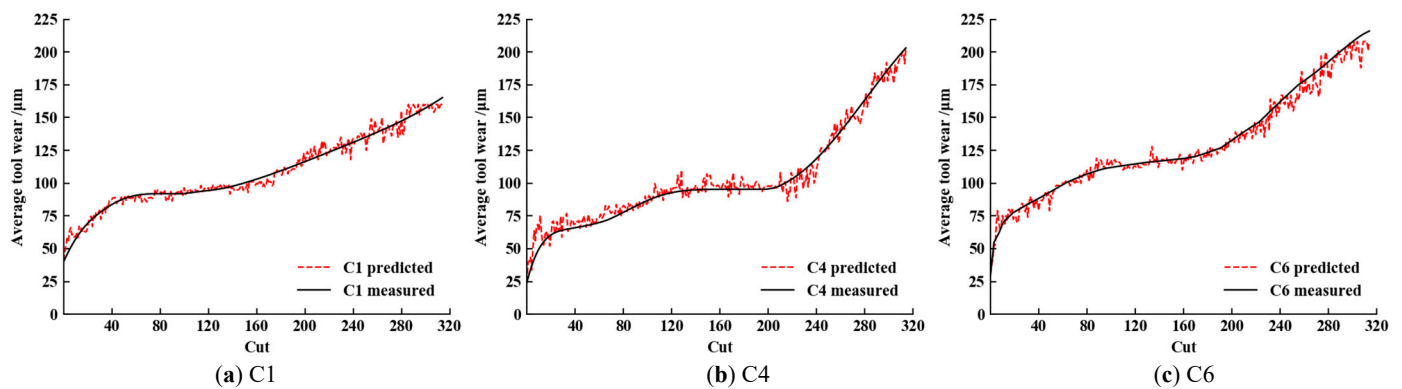


Figure 11. Predicted results of ConvLSTM-Att model on the C1, C4 and C6 test sets.

It can be seen that the 1D-CNN model has the worst overall performance, which is shown in Figure 9. The model-fitting results are poor, and the predicted results of this model at the initial and severe wear stages of the tool have a large gap with the measured wear, which leads to model errors that are generally large. Then, as shown in Figure 10, the accuracy of the CNN-LSTM model is substantially improved compared with that of the 1D-CNN model, and the predicted wear of the model basically fits the measured wear. Due to the fact that the measured time-domain data of tool wear are essentially sequential data, adding the learning of temporal features to the 1D-CNN can improve the accuracy of the model's prediction. However, there is still a large error in predicting the wear at the initial and severe wear stages of the tool, especially at the severe wear stage for tool C6. From Figure 11, the tool wear predicted using the ConvLSTM-Att model is closer to the measured one. The model improves the ability to extract the key features of tool wear and thus has achieved a higher level of accuracy in predicting the normal and severe wear stages of the tool. Although the prediction of the initial tool wear phase still has some errors, it has been improved compared to the CNN-LSTM model.

From Table 3, the CNN-LSTM model has a complexity that is slightly higher than 1D-CNN and CNN-LSTM due to the extra structural components in the model. Its complexity is higher than 1D-CNN and CNN-LSTM by 91.6% and 14.8%, respectively. RNN and 1D-CNN models have similar complexity, but the 1D-CNN model has better prediction accuracy. The CNN-LSTM and TDconvLSTM models have similar complexity and prediction accuracy. Compared with the CNN-LSTM model, the ConvLSTM-Att model has similar complexity but better prediction accuracy.

On the three different test sets, the ConvLSTM-Att model reduced the RMSE by 60.8%, 67.4%, and 63.7%, and the MAE by 61.7%, 71.3%, and 68.8%, respectively, compared with the 1D-CNN model. When compared with the CNN-LSTM model, the RMSE was reduced by 34.4%, 37.1%, and 46.3%, respectively, and the MAE decreased by 39.6%, 37.6%, and 47.7%, respectively. The results of the ConvLSTM-Att model are also compared with several state-of-the-art models using the same data sets, as shown in Table 2. Compared with these models, the ConvLSTM-Att model has the smallest RMSE and MAE in predicting tool wear, which further confirms the effectiveness and superiority of the ConvLSTM-Att model.

5. Conclusions

In this paper, a tool wear prediction model based on an attentional composite neural network is proposed. Based on a large amount of tool wear data, the original time-domain data in the tool machining state is used as input, and the wear features are mined by 1D-CNN. Using LSTM and attention mechanisms to learn the important tool wear features, the measured wear is approximated by regression. By comparing the predicted tool wear and the measured tool wear, it is confirmed that the ConvLSTM-Att model can well reflect the trend of aggravating tool wear during machining. Under the same test set, the error rate in the tool wear prediction using the ConvLSTM-Att model has decreased significantly compared with other state-of-the-art models. For different test sets, the prediction

curves derived from the ConvLSTM-Att model can accurately approximate the measured wear. In sum, the ConvLSTM-Att model provides higher prediction accuracy and better generalization ability compared with the existing models.

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

Funding: This work was funded in part by the Zhejiang Provincial 2023 Annual Leading Research and Development Program under Grant no. 2022C01SA111123 and in part by the National Natural Science Foundation of China under Grant no. 51475434.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: PHM Society: 2010 PHM Society Conference Data Challenge. Available online: <https://www.phmsociety.org/competition/phm/10>.

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

References

1. Salierno, G.; Leonardi, L.; Cabri, G. The Future of Factories: Different Trends. *Appl. Sci.* **2021**, *11*, 9980. [CrossRef]
2. Lu, J.; Chen, R.; Liang, H.; Yan, Q. The Influence of Concentration of Hydroxyl Radical on the Chemical Mechanical Polishing of SiC Wafer Based on the Fenton Reaction. *Precis. Eng.* **2018**, *52*, 221–226. [CrossRef]
3. Chen, N.; Hao, B.; Guo, Y.; Li, L.; Khan, M.A.; He, N. Research on Tool Wear Monitoring in Drilling Process Based on APSO-LS-SVM Approach. *Int. J. Adv. Manuf. Technol.* **2020**, *108*, 2091–2101. [CrossRef]
4. Yang, X.; Yuan, R.; Lv, Y.; Li, L.; Song, H. A Novel Multivariate Cutting Force-Based Tool Wear Monitoring Method Using One-Dimensional Convolutional Neural Network. *Sensors* **2022**, *22*, 8343. [CrossRef]
5. García-Ordás, M.T.; Alegre-Gutiérrez, E.; Alaiz-Rodríguez, R.; González-Castro, V. Tool Wear Monitoring Using an Online, Automatic and Low Cost System Based on Local Texture. *Mech. Syst. Signal Process.* **2018**, *112*, 98–112. [CrossRef]
6. Du, K.-L.; Swamy, M.N.S. *Neural Networks and Statistical Learning*; Springer London: London, UK, 2019; ISBN 978-1-4471-7451-6.
7. Du, K.-L.; Swamy, M.N.S. *Neural Networks in a Softcomputing Framework*; Springer-Verlag: London, UK, 2006; ISBN 978-1-84628-302-4.
8. Du, K.-L.; Swamy, M.N.S. *Search and Optimization by Metaheuristics*; Springer International Publishing: Cham, Switzerland, 2016; ISBN 978-3-319-41191-0.
9. Li, J.; Wang, Y.; Du, K.-L. Distribution Path Optimization by an Improved Genetic Algorithm Combined with a Divide-and-Conquer Strategy. *Technologies* **2022**, *10*, 81. [CrossRef]
10. Kuntoğlu, M.; Aslan, A.; Pimenov, D.Y.; Usca, Ü.A.; Salur, E.; Gupta, M.K.; Mikolajczyk, T.; Giasin, K.; Kapłonek, W.; Sharma, S. A Review of Indirect Tool Condition Monitoring Systems and Decision-Making Methods in Turning: Critical Analysis and Trends. *Sensors* **2020**, *21*, 108. [CrossRef] [PubMed]
11. Mao, W.; Feng, W.; Liu, Y.; Zhang, D.; Liang, X. A New Deep Auto-Encoder Method with Fusing Discriminant Information for Bearing Fault Diagnosis. *Mech. Syst. Signal Process.* **2021**, *150*, 107233. [CrossRef]
12. Wang, J.; Xie, J.; Zhao, R.; Zhang, L.; Duan, L. Multisensory Fusion Based Virtual Tool Wear Sensing for Ubiquitous Manufacturing. *Robot. Comput.-Integr. Manuf.* **2017**, *45*, 47–58. [CrossRef]
13. Qiu, J.; Wang, H.; Lu, J.; Zhang, B.; Du, K.-L. Neural Network Implementations for PCA and Its Extensions. *ISRN Artif. Intell.* **2012**, *2012*, 847305. [CrossRef]
14. Feng, T.; Guo, L.; Gao, H.; Chen, T.; Yu, Y.; Li, C. A New Time-Space Attention Mechanism Driven Multi-Feature Fusion Method for Tool Wear Monitoring. *Int. J. Adv. Manuf. Technol.* **2022**, *120*, 5633–5648. [CrossRef]
15. Zhao, R.; Yan, R.; Chen, Z.; Mao, K.; Wang, P.; Gao, R.X. Deep Learning and Its Applications to Machine Health Monitoring. *Mech. Syst. Signal Process.* **2019**, *115*, 213–237. [CrossRef]
16. Voulodimos, A.; Doulamis, N.; Doulamis, A.; Protopapadakis, E. Deep Learning for Computer Vision: A Brief Review. *Comput. Intell. Neurosci.* **2018**, *2018*, 7068349. [CrossRef]
17. Song, Z. English Speech Recognition Based on Deep Learning with Multiple Features. *Computing* **2020**, *102*, 663–682. [CrossRef]
18. Otter, D.W.; Medina, J.R.; Kalita, J.K. A Survey of the Usages of Deep Learning for Natural Language Processing. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 604–624. [CrossRef]
19. Luo, B.; Wang, H.; Liu, H.; Li, B.; Peng, F. Early Fault Detection of Machine Tools Based on Deep Learning and Dynamic Identification. *IEEE Trans. Ind. Electron.* **2019**, *66*, 509–518. [CrossRef]

20. Kothuru, A.; Nooka, S.P.; Liu, R. Application of Deep Visualization in CNN-Based Tool Condition Monitoring for End Milling. *Procedia Manuf.* **2019**, *34*, 995–1004. [[CrossRef](#)]
21. Xu, X.; Wang, J.; Ming, W.; Chen, M.; An, Q. In-Process Tap Tool Wear Monitoring and Prediction Using a Novel Model Based on Deep Learning. *Int. J. Adv. Manuf. Technol.* **2021**, *112*, 453–466. [[CrossRef](#)]
22. Chen, Q.; Xie, Q.; Yuan, Q.; Huang, H.; Li, Y. Research on a Real-Time Monitoring Method for the Wear State of a Tool Based on a Convolutional Bidirectional LSTM Model. *Symmetry* **2019**, *11*, 1233. [[CrossRef](#)]
23. Zhao, R.; Wang, J.; Yan, R.; Mao, K. Machine Health Monitoring with LSTM Networks. In Proceedings of the 2016 10th International Conference on Sensing Technology (ICST), Nanjing, China, 11–13 November 2016; pp. 1–6.
24. Cai, W.; Zhang, W.; Hu, X.; Liu, Y. A Hybrid Information Model Based on Long Short-Term Memory Network for Tool Condition Monitoring. *J. Intell. Manuf.* **2020**, *31*, 1497–1510. [[CrossRef](#)]
25. Chan, Y.-W.; Kang, T.-C.; Yang, C.-T.; Chang, C.-H.; Huang, S.-M.; Tsai, Y.-T. Tool Wear Prediction Using Convolutional Bidirectional LSTM Networks. *J. Supercomput.* **2022**, *78*, 810–832. [[CrossRef](#)]
26. Schwendemann, S.; Sikora, A. Transfer-Learning-Based Estimation of the Remaining Useful Life of Heterogeneous Bearing Types Using Low-Frequency Accelerometers. *J. Imaging* **2023**, *9*, 34. [[CrossRef](#)]
27. Qiao, H.; Wang, T.; Wang, P. A Tool Wear Monitoring and Prediction System Based on Multiscale Deep Learning Models and Fog Computing. *Int. J. Adv. Manuf. Technol.* **2020**, *108*, 2367–2384. [[CrossRef](#)]
28. Wang, G.; Zhang, F. A Sequence-to-Sequence Model With Attention and Monotonicity Loss for Tool Wear Monitoring and Prediction. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–11. [[CrossRef](#)]
29. Huang, Q.; Wu, D.; Huang, H.; Zhang, Y.; Han, Y. Tool Wear Prediction Based on a Multi-Scale Convolutional Neural Network with Attention Fusion. *Information* **2022**, *13*, 504. [[CrossRef](#)]
30. Li, Z.; Liu, F.; Yang, W.; Peng, S.; Zhou, J. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 6999–7019. [[CrossRef](#)]
31. Huang, Z.; Zhu, J.; Lei, J.; Li, X.; Tian, F. Tool Wear Predicting Based on Multi-Domain Feature Fusion by Deep Convolutional Neural Network in Milling Operations. *J. Intell. Manuf.* **2020**, *31*, 953–966. [[CrossRef](#)]
32. Kiranyaz, S.; Avci, O.; Abdeljaber, O.; Ince, T.; Gabbouj, M.; Inman, D.J. 1D Convolutional Neural Networks and Applications: A Survey. *Mech. Syst. Signal Process.* **2021**, *151*, 107398. [[CrossRef](#)]
33. Van Houdt, G.; Mosquera, C.; Nápoles, G. A Review on the Long Short-Term Memory Model. *Artif. Intell. Rev.* **2020**, *53*, 5929–5955. [[CrossRef](#)]
34. Sherstinsky, A. Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network. *Phys. D Nonlinear Phenom.* **2020**, *404*, 132306. [[CrossRef](#)]
35. Niu, Z.; Zhong, G.; Yu, H. A Review on the Attention Mechanism of Deep Learning. *Neurocomputing* **2021**, *452*, 48–62. [[CrossRef](#)]
36. Zhou, Y.; Xue, W. Review of Tool Condition Monitoring Methods in Milling Processes. *Int. J. Adv. Manuf. Technol.* **2018**, *96*, 2509–2523. [[CrossRef](#)]
37. PHM Society: 2010 PHM Society Conference Data Challenge. Available online: <https://www.phmsociety.org/competition/phm/10> (accessed on 13 January 2023).
38. Qiao, H.; Wang, T.; Wang, P.; Qiao, S.; Zhang, L. A Time-Distributed Spatiotemporal Feature Learning Method for Machine Health Monitoring with Multi-Sensor Time Series. *Sensors* **2018**, *18*, 2932. [[CrossRef](#)]

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.