




Milling force prediction model based on transfer learning and neural network

Juncheng Wang^{1,2,3} · Bin Zou^{1,2,3}  · Mingfang Liu⁴ · Yishang Li^{1,2,3} · Hongjian Ding^{1,2,3} · Kai Xue^{1,2,3}

Received: 25 September 2019 / Accepted: 20 May 2020 / Published online: 4 June 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

In recent years, the growing popularity of artificial neural networks has urged more and more researchers to try introduce these methods to the machining field, with some of them actually producing good results. The acquisition of cutting data often means higher cost and time, limiting the application of neural network in the machining sector, to a certain extent. In this paper, for the task of cutting force prediction, a “transfer network” was established, based on data obtained by simulation, combined with the theory and method in the field of transfer learning. Compared to “ordinary network”, that is, traditional back-propagation neural network based on experimental samples alone, transfer network exhibits obvious performance advantages. On one hand, this means that, using the same experimental samples, the prediction error of transfer network will be controlled; while on the other hand, when the same prediction error is achieved, the number of experimental samples required by the transfer network will be less.

Keywords Cutting force · Neural network · Transfer learning · Prediction

Introduction

Since the cutting force in the metal cutting process has many influencing factors, it is difficult to establish a perfect cutting force prediction model including all of these factors. Fortunately, the development of neural network technology provides a good solution to this problem. Neural network model can easily determine the implicit relationship between a set of input and output parameters, based on the data set, while effectively capturing the non-linearity between them,

which is very important to the good cutting force prediction in the machining process. In the case of sufficient cutting data, the neural network can be used to predict the cutting force, without considering the influence of various factors, making this an ideal approach.

The most widely used method of neural network is to evolve a comprehensive prediction model between processing information (such as cutting parameters, tool geometry etc.) and processing results (such as cutting force, surface roughness, etc.) (Sharma et al. 2008; Tandon and El-Mounayri 2001). Radhakrishnan and Nandan (2005) developed a regression model to filter out the abnormal samples in the experimental data set. The research shows that using the filtered samples can significantly improve the prediction accuracy of the neural network model. Jurkovic et al. (2018) compared the performance of three machine learning methods: neural network, support vector machine and polynomial regression, using the parameters of surface roughness, cutting force and tool life time in high-speed turning. The research shows that the three methods have advantages and disadvantages, depending on different parameters and ranges. Vaishnav et al. (2019) used the data set, generated by the mechanistic force model, to train the neural network model to predict the instantaneous cutting force in milling, verifying the effectiveness of the method. In addition,

✉ Bin Zou
zb78@sdu.edu.cn

¹ Centre for Advanced Jet Engineering Technologies (CaJET), School of Mechanical Engineering, Shandong University, 17923 Jingshi Road, Jinan 250061, People's Republic of China

² Key Laboratory of High Efficiency and Clean Mechanical Manufacture, Shandong University, Ministry of Education, Jinan, People's Republic of China

³ National Demonstration Center for Experimental Mechanical Engineering Education, Shandong University, Jinan, People's Republic of China

⁴ Shanghai Aerospace Equipments Manufacture Co., Ltd, Shanghai 200245, People's Republic of China

the research on neural network in machining field has led to many achievements (Özel and Nadgir 2002; Rao et al. 2014; Asiltürk and Çunkaş 2011). Yeganefar et al. (2019) compared the performance of four methods, in the task of prediction and optimization of cutting force and surface roughness on aluminum alloy: regression analysis, support vector regression, artificial neural network and multi-objective genetic algorithm. The results show that, in the case of sufficient samples, the neural network model will perform better than other methods. The above review shows that, the application of neural network becomes wider and the usage mode more diverse. However, it is worth noting that neural networks are not always advantageous. The accuracy of the neural network model is mainly dependent on the quantity and quality of the training data set, which is an important disadvantage of this approach. Salimiasl and Özdemir (2016) described the comparative analysis of different methods of tool wear online monitoring, during turning of SAE4140 steel, which included artificial neural network, fuzzy logic and least squares method. According to the results, neural network predicted tool wear more precisely, excluding the case of limited amount of experimental data, where fuzzy logic was more precise. In previous research work, training datasets are mostly generated by conducting machining experiments within the entire range of input parameters. Due to the long processing cycle, high material cost and expensive machine tool maintenance, involved in the actual process, obtaining a large number of data samples often means higher financial and time cost, which limits the popularity of neural networks in the field of machining.

Transfer learning is a branch of the machine learning field. Pan and Yang (2009) proposed the definition of transfer learning as such: given a source domain D_S and learning task T_S , a target domain D_T and learning task T_T , transfer learning aims to help improve the learning of the target predictive function $f_T(\cdot)$ in D_T , using the knowledge in D_S and T_S , where $D_S \neq D_T$, or $T_S \neq T_T$. In real life scenarios, obtaining perfect data samples is an expensive and time-consuming operation. In machine learning, transfer learning is an important tool for solving the basic problem of insufficient training data. In many studies of transfer learning, it is often assumed that the target dataset is significantly smaller than the source domain dataset. As the popularity of deep learning methods increases, more and more researchers use deep neural networks for transfer learning, while the related research field is called deep transfer learning (Tan et al. 2018). Yosinski et al. (2014) took the lead in conducting the study of the transitivity of deep neural networks and proposed the fine-tuning method of neural networks. Nowadays, fine-tuning is almost the most widely used method in the field of neural network transfer. According to this approach, first a neural network model is trained, using the source domain data set, while considering the trained model as the initial value. Next,

the model is trained according to the target domain data, to obtain a model suitable for the target domain. However, the fine-tuning method is not suitable for all transfer tasks. When the similarity between the source domain and the target domain is very low, this method is difficult to achieve satisfactory results. Subsequently, many researchers innovated the structure and loss function of neural networks, based on the work of Yosinski et al. (2014) and maximum mean discrepancy (MMD) (Gretton et al. 2012), thus establishing a series of new transfer methods, such as the ones described in Ghifary et al. (2014), Tzeng et al. (2014) and Long et al. (2015). MMD is almost the most frequently used distance measurement method in transfer learning. It is a framework for analysis and comparison, in order to determine whether two samples are from the same distribution. The MMD was first proposed for the two-sample test problem, to compare the difference between two data distributions (Borgwardt et al. 2006).

Transfer learning provides a possible solution to improve the performance of neural network, by applying the knowledge and skills (in the form of parameters), acquired during the previous tasks with sufficient training data, to the new task with a smaller training data set. On one hand, the research of Zhang et al. (2017) has shown that transfer learning is capable of achieving prominent performance, commensurate with large scale CNNs, using only a small set of training data. On the other hand, transfer learning becomes possible and promising because the layers of the convolutional stages of the convolutional neural network, trained on a large dataset, indeed extract general features of inputs, while the layers of the fully connected stages provide more specific features, as described by Cao et al. (2018). The limitations encountered by the application of neural network, in the field of cutting, are very obvious and well known, while it is often very difficult to obtain large amounts of cutting data in actual machining conditions. In this paper, the cutting force data is used as a research example, in an effort to reduce the dependence of neural network on cutting data, by using the method of deep transfer learning. According to the definition of transfer learning, two cutting force data sets, derived from simulation and its corresponding experimental data, are given. The simulation data represent the source domain, while the experimental data is the target domain. Relevant methods and theories, in the field of transfer learning, combined with different experimental samples, were used to train the neural network, applied to the experimental samples and compared to the ordinary network using only experimental data. The results show that, in most cases, the transfer network is better than the ordinary network.

Experiment and model

According to the transfer learning theory, there is a difference between the source and the target domain data, but at the same time there is a certain correlation. Finite element simulation is often regarded as the prediction and verification means of experiments. Although there may be some deviation between the simulation results and the actual results, this does not prevent the simulation results from retaining a certain reference and guidance significance. At present, as a commonly used research method, the effect of finite element simulation has been confirmed in many studies, such as the research of Zhang et al. (2014) and Wang et al. (2018). Therefore, there may be some differences between the cutting force numerical value, as obtained by simulation and by experiment, under the same machining conditions, but there is still a certain correlation between them. Under this premise, based on the relevant methods and theories of transfer learning, a method for establishing a transfer neural network for cutting force data prediction is proposed.

Experimental and data processing

The integral carbide end milling cutter with a diameter of 12 mm is used in both simulation and cutting experiments. The material of the workpiece is aluminium alloy 2A14, while the processing method is end milling. Aluminum alloy material has low processing difficulty, generates less heat during cutting and lower tool temperature, limiting tool wear. Therefore, cutting parameters can be selected within a wide range. The simulation software uses Third Wave AdvantEdge. Variable parameters and their varying ranges in the experiment are listed in Table 1. The ultimate goal of the two groups of experiments is to extract the cutting force values of each axis, corresponding to the cutting parameters. The simulation experiment includes 467 sets of parameters, while the cutting experiment includes 300 sets. The range of cutting parameters is shown in Table 1. The experiments were performed in a vertical CNC machine centre (DAEWOO ACE-V500), while the experimental setup is shown in Fig. 1. The signals of milling forces are recorded by a dynamometer (Kistler 9257B).

In order to minimize the simulation calculation time, the cutting distance of a single group of simulation experiments

is corresponding to the cutting distance of a tool rotation of 100° . The original cutting force data sets, obtained from simulation and cutting experiments, are all in the form of signals. Prior to developing the neural network model, it is necessary to process the original signal and extract the cutting force value, according to the unified standard. Following, the X-axis force cutting signal, corresponding to the cutting parameters of spindle speed 1500 r/min, feed per tool 0.15 mm, radial depth 1.1 mm and axial depth 1.6 mm, is considered as an example to describe the flow of data processing.

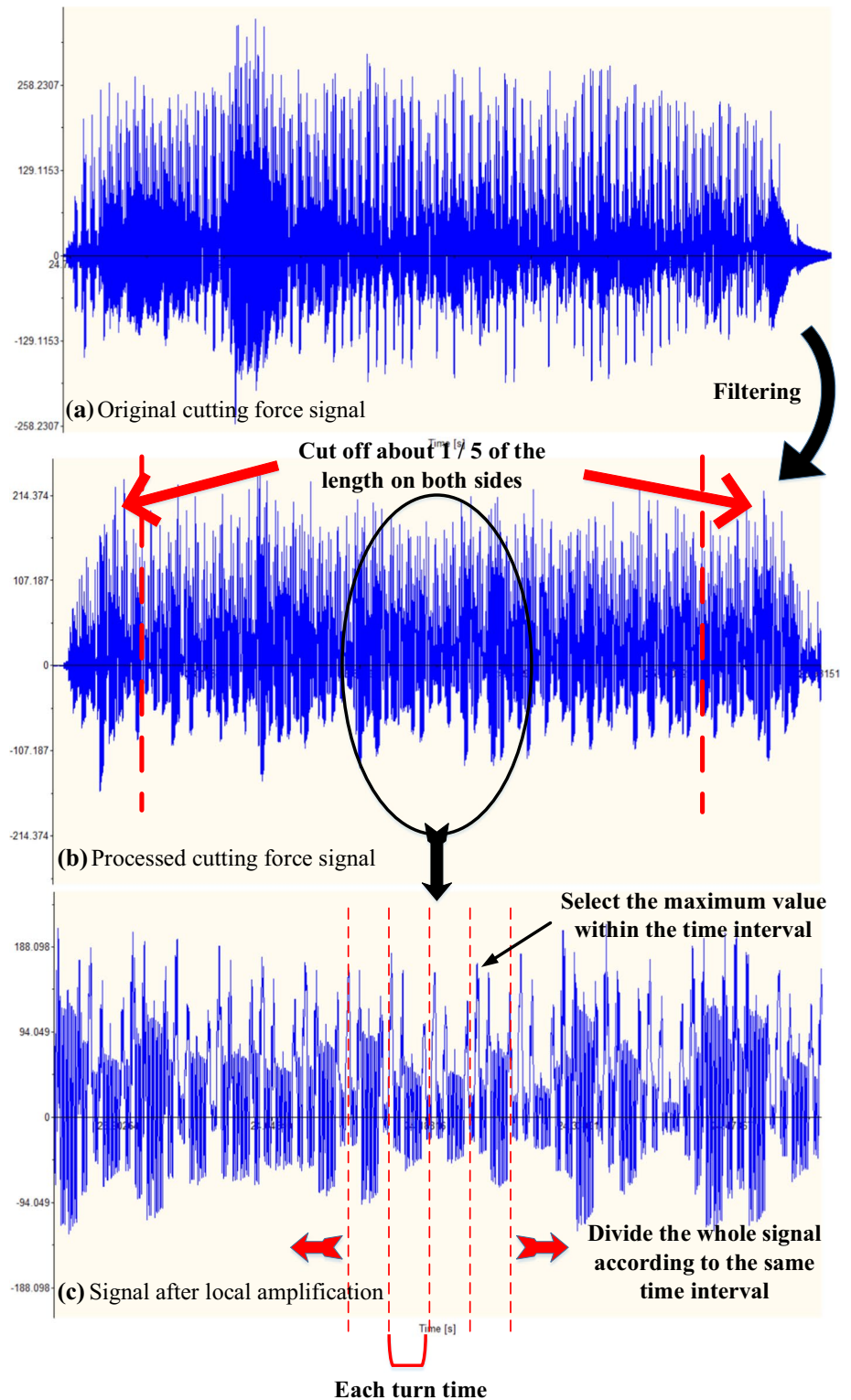
The original cutting force signal, as obtained from the cutting experiment, is illustrated in Fig. 1a. First, it is necessary to use low-pass filter on the original signal. The filter frequency is calculated as: $5 \times z \times n \div 60$ where z represents the number of tool teeth and n represents the spindle speed in r/min. The filter frequency, used here, is 500. As the tool cuts in and out, the cutting force signal will fluctuate to a certain extent. Therefore, about one-fifth of the total length is cut off from each side of the filtered signal, while the remaining part will be segmented, based on the time of each tool rotation. The maximum value of each numerical point, in each small segment, is considered, while the average value of all maximum values is calculated, which then lead to the cutting force value of the X axial force corresponding to the parameter under study.

Figure 2a shows the state at the end of the simulation process. The complete simulation process corresponds to tool rotation of 100° , with two cutter teeth participating in the cutting. The resulting cutting force signal is illustrated in Fig. 2b, where one can see that the original force signal is very disordered, its value shows great fluctuation with a peak value exceeding—500, which is very different from the peak value of the cutting force signal, as obtained from the experiment. Consequently, the cutting force signal from the simulation and the one derived from the experiment cannot be processed using the same method. Figure 2c shows the filtered cutting force signal with a filtering frequency of $8 * n$, that is 12,000 as used here. Compared to the original signal, the filtered signal has greatly improved, while there are still large numerical fluctuations. As shown in Fig. 2c, the peak value of the trough in the circled area exceeds—100, while the average value of the peak section is estimated to be about—85. If the peak value is selected directly, the regularity of the extracted cutting force value will be seriously weakened. Therefore, the tenth degree polynomial is used to fit the filtered signal and the resulting curve is shown in Fig. 2d. The extremum of the two troughs is taken from the fitted curve, while the average value is taken as the value of the X axial force, corresponding to a group of cutting parameters. The cutting force values extracted according to the above process are listed in Table 2.

Table 1 Cutting parameter range

Name of parameter	Range of values
Spindle speed (r/min)	1500–3500
Feed per tooth (mm/r)	0.05–0.25
Radial depth (mm)	0.5–1.7
Axial depth (mm)	0.8–2.4

Fig. 1 Signal processing flow of experimental cutting force



The data processing flow and the results of simulation and experiment show that there are some differences between the cutting force values obtained by simulation and those derived by experiment. This deviation is mainly due to the

difference between the original simulation signal and the different data processing methods. The experimental signal comes from the data measured by the force measuring instrument, while the simulation signal is the calculation

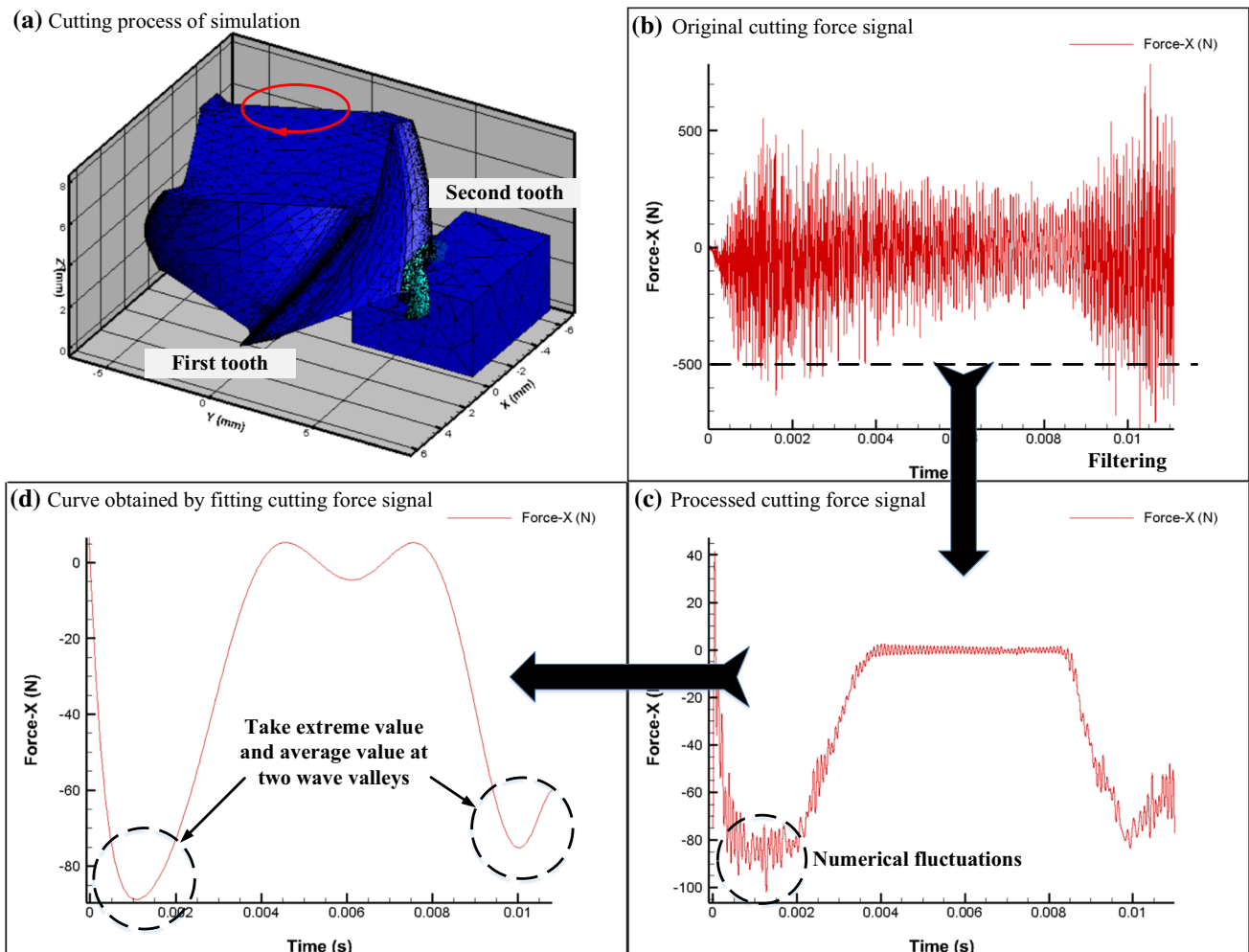


Fig. 2 Signal processing flow of simulation cutting force

Table 2 Some examples of data samples

Spindle speed (r/min)	Feed per tooth (mm)	Radial depth (mm)	Axial depth (mm)	Simulation			Experiment		
				X-axis force (N)	Y-axis force (N)	Z-axis force (N)	X-axis force (N)	Y-axis force (N)	Z-axis force (N)
1500	0.15	1.1	1.6	81.72	92.43	23.80	173.92	191.96	19.998
1500	0.25	1.4	0.8	67.50	60.01	17.24	192.51	190.50	11.53
2000	0.15	1.1	1.2	68.88	72.07	8.08	244.98	172.53	13.46
2000	0.25	1.7	1.2	92.24	94.42	37.39	235.26	225.22	20.46
2500	0.15	0.8	0.8	43.58	43.45	7.29	205.75	125.92	12.198
2500	0.25	0.8	1.2	86.89	72.09	19.88	244.06	176.54	20.26

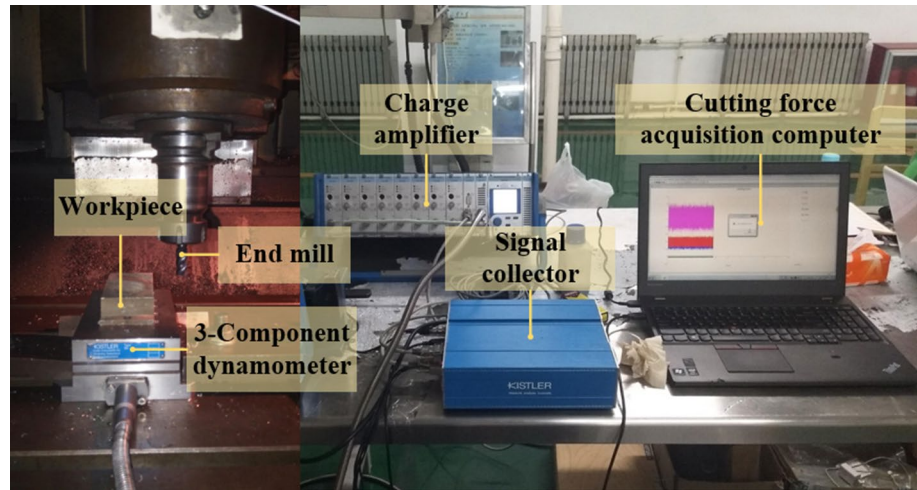
Bold values indicate the cutting force value of the sample used in data processing

The total number of simulation data is 467, while the total number of experimental data is 300

result of the simulation software. The original signal curve, as illustrated in Figs. 1 and 3, indicates that there are very obvious differences between them in the amplitude and change rule of the signal. The difference of the original

signal leads to the fact that it is not appropriate to apply the same processing method. It is necessary to filter and fit the original signal of the simulation, wherein, to a certain extent,

Fig. 3 Set-up and measurements for milling experiments



the extracted simulation cutting force value is significantly lower than the experimental cutting force value.

In this study, the processing method of the simulation signal aims to ensure that the extracted value has as strong regularity as possible, rather than enhancing the consistency between the simulation and experimental cutting force values. In the related research of transfer learning (Pan and Yang 2009; Yosinski et al. 2014; Long et al. 2015), the process of transfer is to use the knowledge in the source domain to improve the performance of the prediction function in the target domain. The laws contained in the data are often important representatives of “knowledge”. Therefore, during data processing, more attention should be paid to the extracted simulation cutting force value, which has more robust regularity.

Modeling

Related work

The statistical test method based on MMD refers to the following sequence of steps: Based on the two distributed samples, by looking for the continuous function f in the sample space, the mean value of the function values of the samples, from different distributions on f , is obtained. The difference between the two values is the mean discrepancy of the two distributions, corresponding to f . Looking for an f makes this mean discrepancy have a maximum, the MMD. Finally, MMD is taken as the test statistic to determine whether the two distributions coincide. If the value is low enough, the two distributions are considered to be the same, otherwise they are considered different. At the same time, this value is also used to determine the degree of similarity between the two distributions. Let $\{X_s^{(i)}\}_i = 1, \dots, n_s$ and $\{X_t^{(j)}\}_j = 1, \dots, n_t$ be data vectors drawn from distributions

D_s and D_t in the data space χ , where the empirical estimate of MMD is:

$$MMD_e(X_s, X_t) = \left\| \frac{1}{n_s} \sum_{i=1}^{n_s} \phi(X_s^{(i)}) - \frac{1}{n_t} \sum_{j=1}^{n_t} \phi(X_t^{(j)}) \right\|_{\mathcal{H}} \quad (1)$$

where $\phi(\cdot): X \rightarrow \mathcal{H}$ is referred to as the feature space map, \mathcal{H} represents Reproducing Kernel Hilbert Space (RKHS). The most important property is that p is equal to q , when $MMD=0$ (Salimiasl and Özdemir 2016). By casting Eq. (1) into a vector–matrix multiplication form, the kernelized equation form of Eq. (1) is as follows:

$$MMD_e(X_s, X_t) = \left(\frac{1}{n_s^2} \sum_{i=1}^{n_s} \sum_{j=1}^{n_s} \kappa(X_s^{(i)}, X_s^{(j)}) + \frac{1}{n_t^2} \sum_{i=1}^{n_t} \sum_{j=1}^{n_t} \kappa(X_t^{(i)}, X_t^{(j)}) - \frac{2}{n_s n_t} \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} \kappa(X_s^{(i)}, X_t^{(j)}) \right) \quad (2)$$

where $[K_{x..}]_{ij} = \kappa(X_s^{(i)}, X_s^{(j)})$ is the gram-matrix of all possible kernels in the data space.

The work of Long et al. (2015) adds MMD distance to the 7th layer of AlexNet network, to reduce the difference between source and target domain. This method is called DDC for short and the idea is to add an adaptation layer to the 7th layer of the network, based on the original AlexNet network. The function of the adaptation layer is to examine the network’s ability to distinguish the source domain from the target domain separately. If this discrimination ability is very poor, it shows that the features learned by the network are not enough to distinguish the two areas of data, so it is helpful to establish the domain-insensitive feature representation.

Based on DDC, the new Deep Adaptation Network (DAN) architecture (Tzeng et al. 2014) was proposed, which offered a good solution to two problems of DDC. First, DDC is a single kernel MMD, while a single fixed

kernel may not be optimal. The DAN method replaces the single-kernel MMD with a multi-kernel MMD (MK-MMD), when calculating the distance between the source and target domain. The MK-MMD method, proposed in Gretton et al. (2012), that is, to construct the total kernel with multiple kernels, provides a better effect than the single-kernel MMD. Furthermore, DDC only adapts to a layer of network, while DAN adapts to the last three layers of the network and adds the calculated distribution distance to the loss function of the neural network, which can be written as:

$$\min_{\Theta} \frac{1}{n_s} \sum_{i=1}^{n_s} J(\theta(X_i^s, y_i^s)) + \lambda \sum_{l=1}^{l_2} d_k^2(D_s^l, D_t^l) \quad (3)$$

where λ is a penalty parameter greater than zero, while the specification between the l_1 and l_2 layer indices is valid. Where, l_1 and l_2 are 6 and 8, respectively, indicating that the network adaptation is from layer 6 to layer 8. $J(\cdot)$ defines a loss function and cross-entropy function is used in DAN. D_s^l is the l th layer hidden representation for the source and target examples, while $d_k^2(D_s^l, D_t^l)$ is the MK-MMD between the source and target domain, evaluated on the l th layer representation.

Cutting force prediction model

In this section, the process of establishing neural network for cutting force data prediction is described, according to the aforementioned related methods and theories of transfer learning. In the established cutting force data set, each sample contains seven dimensions, including rotational speed, feed per tooth, axial depth of cut, radial depth of cut, as inputs; x-axis force, y-axis force and z-axis force, as outputs. The process of building the neural network is shown in Fig. 4.

Since there are not many input and output units in the network, a four-layer hidden layer neural network structure is adapted. First, a neural network is pre-trained based on simulation data, which is used to predict data in simulation mode. Subsequently, the hidden layer of the training network is used as the initial value of the target domain network, while the experimental data is used for training. Experiments in Yeganefar et al. (2019) have proved that the method of transfer arbitrary layers and fine-tuning can achieve better results than ordinary neural networks. In this case, the first four layers are transferred and fine-tuned. In the loss function of the neural network, the MMD distance of the simulation

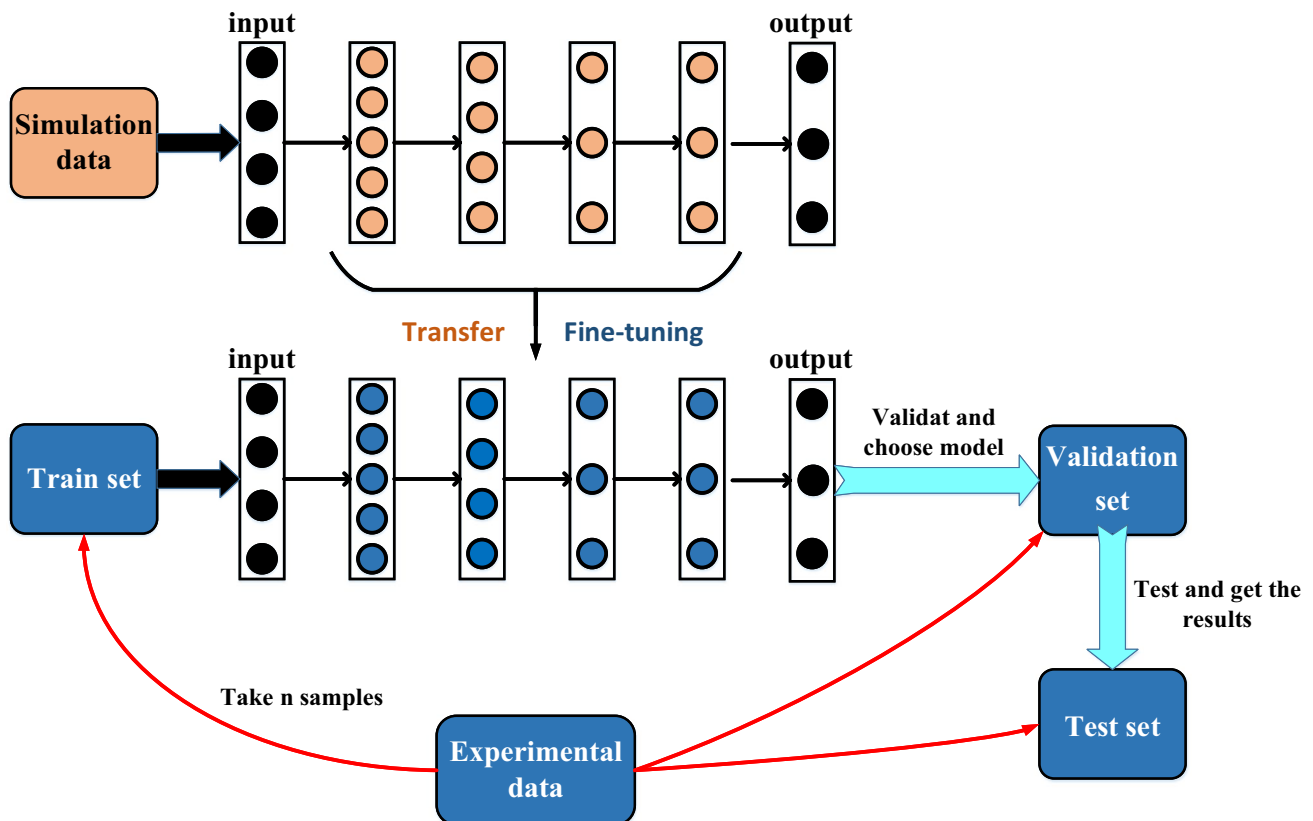


Fig. 4 Transfer network establishment process. The input units of the network include rotational speed, feed/tooth, axial and radial depth of cut, while the output units include X, Y and Z axial forces

and experimental data are additionally considered. The optimization objectives of the whole network include prediction errors in experimental data and discrimination errors in the two domains, which can be expressed as:

$$\min_{\Theta} \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2 + \lambda MKMMD_e(X_s, X_t) \quad (4)$$

Where, \hat{y}_i and y_i represent the real value and the predicted value, respectively, $MKMMD_e(X_s, X_t)$ represents multi-kernel MMD computing method between source and target domain. In this paper, the neural network model, as trained by this method, is called a transfer model. The control group of the transfer model is the traditional BP neural network, trained only by experimental data, which is here considered as the ordinary network, while its optimization objective is:

$$\min_{\Theta} \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2 \quad (5)$$

This experiment establishes prediction models, considering different sample sizes, to explore the effect of transfer learning methods under these variations. The training process of transfer network and ordinary network, with “n” experimental samples as training sets, is as follows:

- (1) Establishing a four hidden layer neural network and initializing randomly. Training is achieved by a training set, consisting of eighty percent of simulation samples, while the performance of the network is tested by the remaining simulation samples.
- (2) Constructing a training set by “n” samples, as extracted from the experimental samples.
- (3) The network, trained by simulation data, is considered as the initial value, as the training set is used to train the network and save the completed training model.
- (4) The above steps are repeated, by considering different values of “n”.
- (5) Using the validation set to evaluate all the models, corresponding to each “n” value; using the established model to evaluate error rate on the test set.

Where $n \in \{5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200\}$. In the training process of transfer network, the extraction of experimental samples is not completely random.

In addition to the different optimization objectives, there are also differences in the training process, between the ordinary network and the transfer network. The training process of ordinary network is relatively simple, as the initial network structure will be trained directly according to experimental samples. In the process of training, the experimental samples and learning rate, iterations and

other parameters used by the ordinary network and the transfer network are identical.

When using n samples to build training data set, the selection of samples should follow certain principles. The first two hundred samples of all experimental samples are used as training set, while the last two hundred samples are divided into verification set and test set. Extracting “n” samples means selecting the first n samples of train set, while all training samples are used, when n is set to 200. In the whole training process, the complete data set will be divided randomly five times. The initial value of the network often has some effect on the convergence results of the network. In order to eliminate the possible influences of the initial value of the network, on the experimental results, in this work, the transfer network and the ordinary network are randomly initialized ten times. Among them, transfer network initialization refers to the initialization process of pre-trained network. That is to say, the transfer and the ordinary network will train 50 models, for each n value. Then, the corresponding test set is used to evaluate the prediction accuracy of the trained model.

Results and discussion

According to the training process, as described in the previous section, the performance of the ordinary network and the transfer network model is evaluated by a test set. According to the number of experimental samples used in training, the two models of each group are evaluated on the test set and the prediction error is averaged. The influence of transfer learning method on the prediction accuracy of neural network model is tested, according to the experimental results. The comparison of the two models is illustrated in Figs. 4 and 5.

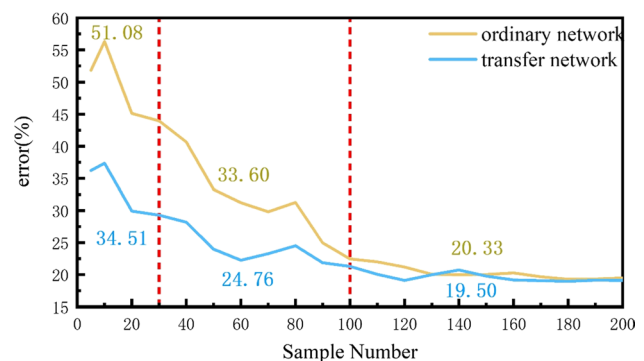


Fig. 5 X-axis force error. The chart is divided into three stages, according to the number of samples, where the numerical value corresponds to the average error of the model in the corresponding stage

Table 3 Performance advantages of transfer network in each stage

Force\sample number	5–20	30–90	5–90	100–200
X	16.57	8.84	11.15	0.83
Y	14.49	5.91	8.49	0.60
Mean	15.53	7.38	9.82	0.72

Each value in the table represents the error rate of the ordinary network minus the error rate of the transfer network, in the corresponding cutting force and sample number area

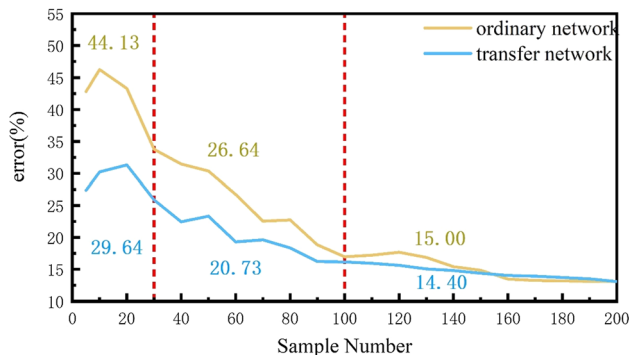


Fig. 6 Y-axis force error. The chart is divided into three stages, according to the number of samples, where the numerical value corresponds to the average error of the model in the corresponding stage

Table 3 includes comparative results regarding the prediction accuracy of the transfer network and the ordinary network, in the range of different sample numbers.

The experimental results illustrated in Figs. 5 and 6, show that the transfer network has different effects in different sample stages. When the number of training samples is less than or equal to 90, the performance of the transfer network exhibits obvious performance advantages, while its average error rate is 11.15% lower than that of the ordinary network; when the number of training samples is higher than 100, there is no significant difference between the performance of the transfer network and the ordinary network.

In the range of 5–90, the average prediction error of transfer network is lower than that of ordinary network, as well as its performance is obviously improved. However, in the sample range of 5–20, the error rate of the transfer network is still high, which may be unacceptable in actual application. Although in this range, the performance improvement effect of the transfer network is the most obvious, its average error is 15.53% lower than that of the ordinary network. Therefore, in the range of 5–20, both transfer network and ordinary network are difficult to meet the actual use requirements. In the range of 30–90 samples, the average error of transfer network is 7.38%, while the average error is 24.76% and 20.73%, respectively, which is a suitable range for transfer networks.

In the range of 100–200 samples, the difference between the prediction error of the transfer network and the ordinary network is less than 1%. This shows that, in this range, the transfer learning method has no effect on the performance of neural network.

In the whole range of samples, the prediction error of the transfer network is, in most cases, lower than that of the ordinary network, while its performance advantage basically decreases gradually and finally disappears. This shows that, in this experiment, the influence of transfer learning method on the performance of neural network will gradually decrease, as the number of samples increases. When the number of samples is higher than 100, the effect of transfer learning becomes very weak, while at a number of samples higher than 140, the method of transfer learning has no effect at all.

Based on the experimental data, in the range of 0–20, the prediction errors of transfer network and ordinary network are relatively high, while the prediction models, established in this range, are not suitable for actual application. In the range of 0–90, the performance of the transfer network has significantly improved, while its performance has improved to a certain extent, compared to the ordinary network. In this range, the transfer learning method is generally the most suitable. In the range of 100–200, the achievable prediction error of the model has basically reached its limit. Even if the training samples continue to increase, it is difficult to improve the performance of the model significantly. In addition, it is still worth noting that, in different sample stages, the transfer network exhibits different performance advantages; but in any sample range, the transfer learning method will not have a negative impact on the performance of the model. In case of uncertainty whether the number of samples can improve the performance of neural network, before building the model, the transfer learning method is a better choice.

Conclusion

In this paper, the transfer network is established by using simulation data as the source domain and experiment data as the target domain, combined with the related methods and theories of transfer learning. Based on the experimental results, the transfer network shows obvious advantages in performance, compared to the ordinary network, which proves that the idea and method of transfer learning can be applied to the field of cutting process, to solve some practical problems, such as the prediction of cutting force. This work mainly includes the following two contributions:

Considering the same number of training samples, the performance of the transfer network exceeds that of the ordinary network. It can also be noted that, the proposed

established method of transfer network requires the lower number of samples in order to provide decent results. To some extent, this can relieve the amount of experimental data needed to predict cutting force, by using neural network. At the same time, this is also in line with one of the problems that transfer learning seeks to solve: the gap between data demand and data volume.

The effect of the transfer learning method will gradually weaken, as the number of samples increases. When the number of samples is sufficient, the performance of the transfer network is basically consistent with that of the ordinary network. Considering any sample number, the transfer learning method will not have a negative impact on the performance of the network. However, when it is uncertain whether the number of samples is enough to train a neural network model of excellent performance, the method of transfer learning is a valid option.

Acknowledgements This project was supported by National Science and Technology Major Project (2018ZX04011001), the Major Program of Shandong Province Natural Science Foundation (ZR2018ZA0401, ZR2018ZB0521) and Shandong Science Fund for Distinguished Young Scholars (JQ201715).

References

- Asiltürk, I., & Çunkaş, M. (2011). Modeling and prediction of surface roughness in turning operations using artificial neural network and multiple regression method. *Expert Systems with Applications*, 38(5), 5826–5832. <https://doi.org/10.1016/j.eswa.2010.11.041>.
- Borgwardt, K. M., Gretton, A., Rasch, M. J., Kriegel, H. P., Schölkopf, B., & Smola, A. J. (2006). Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14), e49–e57. <https://doi.org/10.1093/bioinformatics/btl242>.
- Cao, P., Zhang, S., & Tang, J. (2018). Preprocessing-free gear fault diagnosis using small datasets with deep convolutional neural network-based transfer learning. *IEEE Access*, 6, 26241–26253. <https://doi.org/10.1109/ACCESS.2018.2837621>.
- Ghifary, M., Kleijn, W. B., & Zhang, M. (2014). Domain adaptive neural networks for object recognition. In *Pacific Rim international conference on artificial intelligence* (pp. 898–904). Springer, Cham. https://doi.org/10.1007/978-3-319-13560-1_76.
- Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., & Smola, A. (2012). A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar), 723–773.
- Jurkovic, Z., Cukor, G., Brezocnik, M., & Brajkovic, T. (2018). A comparison of machine learning methods for cutting parameters prediction in high speed turning process. *Journal of Intelligent Manufacturing*, 29(8), 1683–1693. <https://doi.org/10.1007/s10845-016-1206-1>.
- Long, M., Cao, Y., Wang, J., & Jordan, M. I. (2015). Learning transferable features with deep adaptation networks. arXiv preprint [arXiv:1502.02791](https://arxiv.org/abs/1502.02791).
- Özel, T., & Nadgir, A. (2002). Prediction of flank wear by using back propagation neural network modeling when cutting hardened H-13 steel with chamfered and honed CBN tools. *International Journal of Machine Tools and Manufacture*, 42(2), 287–297. [https://doi.org/10.1016/S0890-6955\(01\)00103-1](https://doi.org/10.1016/S0890-6955(01)00103-1).
- Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345–1359. <https://doi.org/10.1109/TKDE.2009.191>.
- Radhakrishnan, T., & Nandan, U. (2005). Milling force prediction using regression and neural networks. *Journal of Intelligent Manufacturing*, 16(1), 93–102. <https://doi.org/10.1007/s10845-005-4826-4>.
- Rao, K. V., Murthy, B. S. N., & Rao, N. M. (2014). Prediction of cutting tool wear, surface roughness and vibration of work piece in boring of AISI 316 steel with artificial neural network. *Measurement*, 51, 63–70. <https://doi.org/10.1016/j.measurement.2014.01.024>.
- Salimiasl, A., & Özdemir, A. (2016). Analyzing the performance of artificial neural network (ANN), fuzzy logic (FL), and least square (LS)-based models for online tool condition monitoring. *The International Journal of Advanced Manufacturing Technology*, 87(1–4), 1145–1158. <https://doi.org/10.1007/s00170-016-8548-x>.
- Sharma, V. S., Dhiman, S., Sehgal, R., & Sharma, S. K. (2008). Estimation of cutting forces and surface roughness for hard turning using neural networks. *Journal of Intelligent Manufacturing*, 19(4), 473–483. <https://doi.org/10.1007/s10845-008-0097-1>.
- Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018). A survey on deep transfer learning. In *International conference on artificial neural networks* (pp. 270–279). Springer, Cham. https://doi.org/10.1007/978-3-030-01424-7_27.
- Tandon, V., & El-Mounayri, H. (2001). A novel artificial neural networks force model for end milling. *The International Journal of Advanced Manufacturing Technology*, 18(10), 693–700. <https://doi.org/10.1007/s001700170011>.
- Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., & Darrell, T. (2014). Deep domain confusion: Maximizing for domain invariance. arXiv preprint [arXiv:1412.3474](https://arxiv.org/abs/1412.3474).
- Vaishnav, S., Agarwal, A., & Desai, K. A. (2019). Machine learning-based instantaneous cutting force model for end milling operation. *Journal of Intelligent Manufacturing*. <https://doi.org/10.1007/s10845-019-01514-8>.
- Wang, L. Y., Huang, H. H., West, R. W., Li, H. J., & Du, J. T. (2018). A model of deformation of thin-wall surface parts during milling machining process. *Journal of Central South University*, 25(5), 1107–1115. <https://doi.org/10.1007/s11771-018-3810-z>.
- Yeganefar, A., Niknam, S. A., & Asadi, R. (2019). The use of support vector machine, neural network, and regression analysis to predict and optimize surface roughness and cutting forces in milling. *The International Journal of Advanced Manufacturing Technology*, 105(1–4), 951–965. <https://doi.org/10.1007/s00170-019-04227-7>.
- Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? In *Advances in neural information processing systems* (pp. 3320–3328).
- Zhang, R., Tao, H., Wu, L., & Guan, Y. (2017). Transfer learning with neural networks for bearing fault diagnosis in changing working conditions. *IEEE Access*, 5, 14347–14357. <https://doi.org/10.1109/ACCESS.2017.2720965>.
- Zhang, F. C., Wang, Q., & Yang, R. (2014). Analysis to milling force base on AdvantEdge finite element analysis. In *Advanced materials research* (Vol. 981, pp. 895–898). Trans Tech Publications Ltd. <https://doi.org/10.4028/www.scientific.net/AMR.981.895>.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.