


Paper Type: Original Article

A Novel Deep Learning Model for Tool Wear Estimation of Cutting Tools

Ahmed Darwish ^{1,*} 

¹ Department of Computer Science, Faculty of Computers and Informatics, Zagazig University, Zagazig, 44519, Egypt; adarwish@fci.zu.edu.eg.

Received: 15 Apr 2024

Revised: 26 Aug 2024

Accepted: 26 Sep 2024

Published: 01 Oct 2024

Abstract

In cutting tools, tool wear may result in low-quality surfaces and precision in dimensional measurement. Tool wear may additionally result in fractures and harm to the workpieces over operations. It's critical to track tool wear in life in order to avoid these situations. This research proposes a data-driven deep learning strategy to improve the accuracy of cutting tool wear estimates. This technique combines a Gated Recurrent Unit (GRU) with a Convolutional Neural Network (CNN). In order to circumvent the problems associated with single-domain features that lack appropriate tool status data, a multi-domain combination technique was used to develop candidate features. We retrieved 4 frequency-domain features, 8 time-frequency domain features, and 12 time-domain features from each channel signal. The CNN eliminates the need for human feature extraction by automatically identifying significant attributes from raw time series data. The model is able to generate more abstract data representations and understand complex temporal patterns thanks to the GRU. Using the Dataset of the 2010 PHM Data Challenge, comparisons with a number of deep learning models were conducted in order to show the effectiveness of the suggested model. The results of the studies indicate that CNN-GRU is a good option for estimating the tool wear of cutting tools because it outperformed all other models that were looked at.

Keywords: Tool Wear Estimation; Prognostics and Health Management; Gated-recurrent Unit Network; Convolutional Neural Network; Remaining Useful Life; Cutting Tools.

1 | Introduction

Since cutting tools are the main component of machining processes, they eventually break down from wear and tear from removing metal components [1]. Because it comes into close contact with the workpiece throughout the production process, the cutting tool is extremely important [2, 3]. Eliminating surplus material in a piece of work in order to get an accurate form is the primary goal of machining operations [4]. It is crucial to switch out cutting tools before they break in order to guarantee both the reliability of the production process and the quality of the workpiece. Workpieces that require high-precision processing are in more demand right now. The workpiece's quality is closely related to the quality of the tool used; wear on the tool is the main cause of the subpar quality of the workpiece and higher reject percentages. There might be two main issues if the measurement of tool wear is not performed accurately. First of all, it can result in



Corresponding Author: adarwish@fci.zu.edu.eg



<https://doi.org/10.61356/SMIJ.2024.9381>



Licensor **Sustainable Machine Intelligence Journal**. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0>).

unwarranted cost increases and tool waste when tools that are still functionally useful are replaced too soon. Second, the quality of the workpiece's surface may deteriorate if the tool is not changed right away, and a defective tool is used for processing the workpiece.

As a result, the quality and accuracy of the finished product will be significantly impacted by the equipment's degradation and attrition. Research indicates that about 20% of machine tool failures are caused by tool problems [1]. Also, tool maintenance expenses account for a significant percentage between 15% and 40% of total production costs [5]. The fact that the quality of machining operations is adversely affected by tool wear is an essential factor to take into account [6]. The prevention of potentially disastrous incidents, such as tooling stoppage and scrapping of the workpiece, can be effectively achieved through the implementation of live machine tool condition tracking and accurate prediction of the remaining useful life (RUL) [7, 8]. As such, it is imperative that manufacturing organizations give top priority to precise live tracking and exact RUL forecast of tool wear. Thus, there has been a growing interest lately in RUL prediction for cutting tools [9, 10]. This is essential to uphold the processing environment's stability, strengthen the workpiece's precision, safeguard the machine tool and processing safety, improve production efficiency, and reduce costs [11]. The demand for Live tool status tracking in contemporary manufacturing enterprises has risen as they are essential for cost reduction and quality improvement [12, 13].

Recently, Scholars have developed various research methodologies for assessing and predicting the state of tools. These strategies fall into three main categories: physics-based methods [14], data-driven methods [15, 16], and hybrid methods [17]. Physics-based approaches focus on developing a theoretical framework that clarifies the functioning of cutting tools. The Physics-based approach looks into the physics to determine how tool conditions and parameters for machining are related to one another. These approaches assert that the wear and tear of the system may be precisely represented by mathematical formulas or methods. A popular adaptive decomposition technique for handling vibration data is empirical mode decomposition (EMD) [18], which depends on modal elements. By using the coefficient of correlation analysis and the intrinsic mode function (IMF) components that are broken down using the EMD approach, Geng et al. [19] were able to effectively accomplish the RUL prediction of bearings. Utilizing stress from contact in line with the fatigue conceptual modeling methodology, Gai et al. [20] presented a method for predicting the fatigue lifespan. The use of the contact fatigue life curve, which is this approach makes it possible to determine the peak contact stress given the provided axial and radial pressures operating on the bearing. This makes it easier to derive the contact fatigue lifetime. By carefully examining the instrument's failing process and creating mathematical models or semi-empirical formulae, the physical model-based forecast illustrates the deterioration of instrument efficiency [21, 22]. Scholars established a number of physical forecasting techniques that utilize the pattern of contact between the instrument and the piece of material for specific cutting settings, as well as the traditional Taylor instrument lifespan formula [23]. Through the clarification of deterioration mechanisms and their effects on the overall system, physics-based approaches aim to give precise RUL estimations that are pertinent to cutting instruments and are based on fundamental physics. However, system complexity, process variability, and uncertainty make it difficult to create realistic physics-based models of machining. Furthermore, creating accurate physical models by trial and error to correspond with experimental data can be a drawn-out procedure that takes years. This problem is addressed by the data-driven technique, which examines sensor data collected during the machining operation. With data-driven approaches, past data from equipment sensors and related measures are used to build prediction models. Generalization may be accomplished with data-driven approaches without the need for specialist knowledge. The association between sensor data and system deterioration may be found using data-driven methodologies [24, 25], indicating a strong capacity to generalize and less dependence on empirical knowledge. By analyzing patterns in temporal data, data-driven techniques for RUL have the ability to produce precise forecasts and reduce the need for traditional technical expertise. This method has the benefit of enabling continuous, real-time equipment tracking, which lowers the possibility of critical faults and enhances future production processes [26].

Data-driven approaches are used in this work to assess the RUL of complex systems. Deep Learning (DL) and Machine Learning (ML) methodologies are included in these methods. With the use of ML techniques, computers are able to acquire knowledge from data by themselves and constantly increase their efficiency without the need for explicit programming, making them an invaluable tool in many facets of daily life. Our way of life, our work habits, and our interactions with technology are all changing as a result of machine learning's ability to autonomously complete tasks and derive information from data. We expect this field to grow and have much more impact on everyone involved. Massive quantities of sensor data, operating characteristics, and previous servicing histories may all be used using machine learning techniques. By using this systematic data-driven process, machine learning algorithms are able to understand complex correlations between many factors that affect the state. According to pertinent research, tool RUL may be reliably predicted from the retrieved signal characteristics using fuzzy neural networks (FNN) [16, 27], support vector regression (SVR) [28, 29], dynamic Bayesian networks (DBN) [30, 31], stochastic processes [32, 33], and relevance vector machines (RVM). In order to forecast RUL, Sohyung et al. [34] employed SVM with the greatest value taken from the cutting force as the eigenvector. Lin et al. [35] used the nonlinear relationship between high-dimensional feature vectors and tools wear based on the evolving connectionist system (ECoS) then the RUL prediction of the tool based on HMM is established. Good experimental results were achieved by Dong et al. [36] after they isolated 16 eigenvalues from the force data and used the ANN model to forecast tool wear.

A prominent subset of machine learning called deep learning has significantly altered many aspects of our culture. It analyzes information by combining multiple-layered artificial neural networks, which imitate the cognitive functions of the human brain. Because deep learning techniques are so flexible, they have made remarkable progress. Models based on deep learning can autonomously derive feature representations, unlike machine learning models, hence they do not require feature engineering. One notable characteristic of deep learning is its intricate network architecture, which makes it easier to identify degrading traits from historical sensor data of equipment under observation [37]. Consequently, deep learning shows greater efficacy in handling high-dimensional, unstructured data types such as pictures and time series data. Interestingly, deep learning techniques outperform conventional machine learning approaches, especially when it comes to predictive maintenance. A multiscale convolution neural network (CNN) based estimation of the RUL approach for bearings was proposed by Zhu et al. [38], and the findings demonstrate that the approach performs better for the RUL estimation of bearings. The characteristics were extracted from the raw data using wavelet packet decomposition methods, and subsequently used as the input of a deep CNN-based model by Belmiloud et al. [39] to predict the health stage. In order to anticipate the RUL, Li et al. [40] proposed using a deep convolution neural network (DCNN), which incorporates the original sensor information into the model and allows DCNN to automatically create the features. Liu et al. [41] introduced a novel approach using adaptive variational mode decomposition (VMD) and a deep learning model combining one-dimensional convolutional LSTM with an attention mechanism. Wan et al. [42] used the complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) to decompose vibration signals into intrinsic mode functions (IMFs). High-frequency and low-frequency IMFs are reconstructed using improved fine-to-coarse reconstruction (IFTC) and phase space reconstruction (PSR), then input into bi-directional long short-term memory (BiLSTM) and CNN models to construct an ensemble RUL prediction model.

Recent research has integrated data-driven and physics-based models to solve the issues of data quality and distributional changes that significantly lower the accuracy of data-driven models. First, a physics-based approach is used to build the degradation system model. This approach is predicated on a thorough understanding of the behavior and degradation mechanisms of the system. This model incorporates the physical factors that lead to degradation. The RUL is then estimated using a data-driven approach using the created model and available data. The most effective elements of both strategies are combined in hybrid techniques to provide more accurate and consistent RUL predictions. The hybrid method optimizes its potential by combining the advantages of both the model-based approach, which incorporates fundamental physics ideas into the cutting tool system, and the data-driven strategy, which uses real data and statistical

approaches to improve the accuracy of RUL prediction. By combining these techniques, it is possible to gain a better knowledge of the system's performance and degradation patterns, which will improve the estimation of RUL. By using the empirical wear-time model and updating the state of the regularized particle filtering (RPF) approach, Hanachi et al. [43] made improvements. A data-driven approach utilizing an automated machine learning strategy that made use of the adaptive neuro-fuzzy inference system (ANFIS) was used by Liu et al. [44]. They also combined a semi-empirical degradation method with an adaptive centerless Kalman filter (ACKF) methodology to anticipate the remaining operating life. Within the discipline of PHM, the Least Squares SVM (LSSVM) is used in many different contexts. Yang et al. [45] created a hybrid strategy for tool wear prediction that combines model-informed and data-driven techniques. On bi-directional gate recurrent unit (Bi-GRU) networks, Wang et al. [46] investigated the use of feature fusion and data augmentation.

In this work, we provide a unique data-driven deep learning technique designed to forecast cutting tool wear prediction. In order to anticipate tool wear estimation, a unique method called a convolution neural network paired with a gated recurrent unit (CNN-GRU) network is proposed. This method takes sensor data, extracts spatial properties, and finds temporal relationships. The GRU architecture has been combined with a CNN to assign weight values to the extracted features. This helps to highlight important information and improves the model's RUL forecast. The well-known IEEE PHM 2010 Challenge Dataset [43] is used to evaluate the effectiveness of the CNN-GRU model for forecasting cutting tool wear. Our experimental results show that the suggested method outperforms other current models in terms of accuracy and reduces uncertainty in multi-step prediction tasks. The significant contributions of the study may be summed up as follows:

- An innovative framework for data-driven deep learning that is intended to comprehend temporal correlations in time series data. This is accomplished by combining a gated recurrent unit (GRU), which effectively detects degradation patterns in the cutting instruments, with a convolutional neural network (CNN).
- In order to circumvent the problems associated with single-domain features that lack appropriate tool status data, a multi-domain combination technique was used to develop candidate features.
- By using the dataset from the IEEE 2010 PHM data challenge, the validity of the model was confirmed [47], showing that it performed better than other models in estimating the tool wear.

The portions of this paper are arranged as follows. A description of the suggested DL model is provided in Section 2. Section 3 then goes into great depth on the experimental settings. Section 4 explores the Discussion and Results. Finally, the study's conclusions are outlined in Section 5.

2 | The Proposed DL Model

2.1 | CNN-GRU Components

2.1.1 | Convolutional Neural Network

It seems improbable that Convolutional Neural Networks would be able to automatically identify important qualities from unprocessed time series data, hence removing the need for manual feature extraction. This enables the model to reveal patterns that people might not be able to recognize and does away with the need for domain-specific expertise when choosing features. CNNs can recognize high-level characteristics like temporal patterns and intricate relationships in deeper layers and low-level features like local trends as well as patterns in early layers, much to how they are used in picture data. CNNs are capable of capturing both global and local features of time series data, such as local patterns and dependencies, because of their hierarchical feature learning capability. CNNs are able to recognize trends such as spikes, recurrent themes, or sudden changes in the time series by applying convolutional filters with narrow receptive fields. The ability to identify local patterns is essential for identifying pertinent aspects that may point to noteworthy occurrences or abnormalities in the data. By using the same set of weights (filters) at different temporal points in the time

series, CNNs use parameter sharing. By reducing the number of trainable parameters in the network, this parameter sharing improves generalization and learning efficiency. Additionally, methods like dropout regularization and pooling can strengthen the network's resistance to noise and overfitting. In some cases, CNNs can provide faster training on time series data since they can process data at the same time and use GPU capabilities to extract features more effectively.

2.1.2 | Gated-Recurrent Unit

One kind of recurrent neural network (RNN) architecture called the Gated Recurrent Unit (GRU) is intended to alleviate the vanishing gradient problem and identify relationships in sequential input. It is a less complicated and more computationally effective substitute for the widely used Long Short-Term Memory (LSTM) network type of RNN. GRUs are faster to train and more computationally efficient than LSTMs because of their more straightforward architecture and fewer gates. GRUs are very helpful for jobs involving sequences, such as speech recognition, language modeling, and time series forecasting. The GRU consists of two main gates the update gate z_t , and reset gate r_t . The update gate z_t regulates how much of the present concealed state gets filtered through from the preceding hidden state and determines the amount of historical data that must be kept. The update gate z_t calculated from Eq. (1). The reset gate r_t establishes how much of the prior concealed state should be forgotten and how much historical data should be disregarded. The reset gate r_t calculated from Eq. (2). The content that has to be added to the existing hidden state is represented by the candidate's hidden state h'_t . It is in the running to become the next secret state. The candidate's hidden state is calculated by Eq. (3). The current hidden state h_t , which is determined by the update gate, is the combination of the candidate hidden state and the previous hidden state for the current time step. The current hidden state is calculated by Eq. (4).

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (1)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (2)$$

$$h'_t = \tanh(W_a x_t + U_a \odot h_{t-1} + b_a) \quad (3)$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot h'_t \quad (4)$$

Where the symbol σ denotes the sigmoid function, t represents the time step, x_t signifies the input feature at time t , h_{t-1} denotes the output hidden state from the previous time sample, the operator \odot indicates element-wise multiplication, the parameters $W_z, W_r, W_a, U_z, U_r, U_a, b_z, b_r, b_a$ are optimized during the training process.

2.2 | CNN-GRU Model

Since cutting tool wear estimate is considered a supervised regression problem, deep learning models must be trained and evaluated using data from the challenge of the 2010 PHM Data dataset. For tool wear assessment of cutting tools, this work proposes a novel data-driven deep learning technique called CNN-GRU, which combines CNN and GRU, as shown in Figure 1. To extract features from the input features, the CNN component is used. Time-domain, frequency-domain, and time-frequency-domain characteristics are among the input features; they will be covered in more detail in section 3.3. There are three convolution blocks in the CNN-GRU. A BatchNormalization layer is placed after three Conv1D layers with 64 filters and three kernel sizes in the first block. In order to lower the dimensionality and preserve the most important characteristics, the second block consists of two Conv1D layers with 128 filters and three kernel sizes, followed by a BatchNormalization layer and MaxPooling1D with a pool size of equal to two. Two Conv1D layers with 256 filters and three kernel sizes are included in the third block. A BatchNormalization layer and MaxPooling1D with a pool size of two are next. The GRU, which consists of three GRU layers, receives the CNN's output after that. A GRU layer consists of 32 units. Through memory cells and gates, each GRU layer in the sequence transmits information after processing the input in a sequential fashion. The model can more easily capture hierarchical representations of the input data since the output of one GRU layer serves as the

input for the subsequent layer. The model can capture intricate temporal correlations and provide more abstract data representations because of the addition of additional GRU layers. Subsequently, the GlobalAveragePooling1D layer processes the GRU layer's output, thereby reducing it so that the dense layer can handle it. This facilitates the development of a model that is more effective and maybe more generalizable. For the ultimate tool wear estimation, the combined and processed data from the hidden layer is sent to the fully connected (FC) network. The FC network is composed of a 32-unit dense layer and a dropout layer with a 0.1 dropout rate. Algorithm 1 concludes by outlining the suggested model's pseudocode.

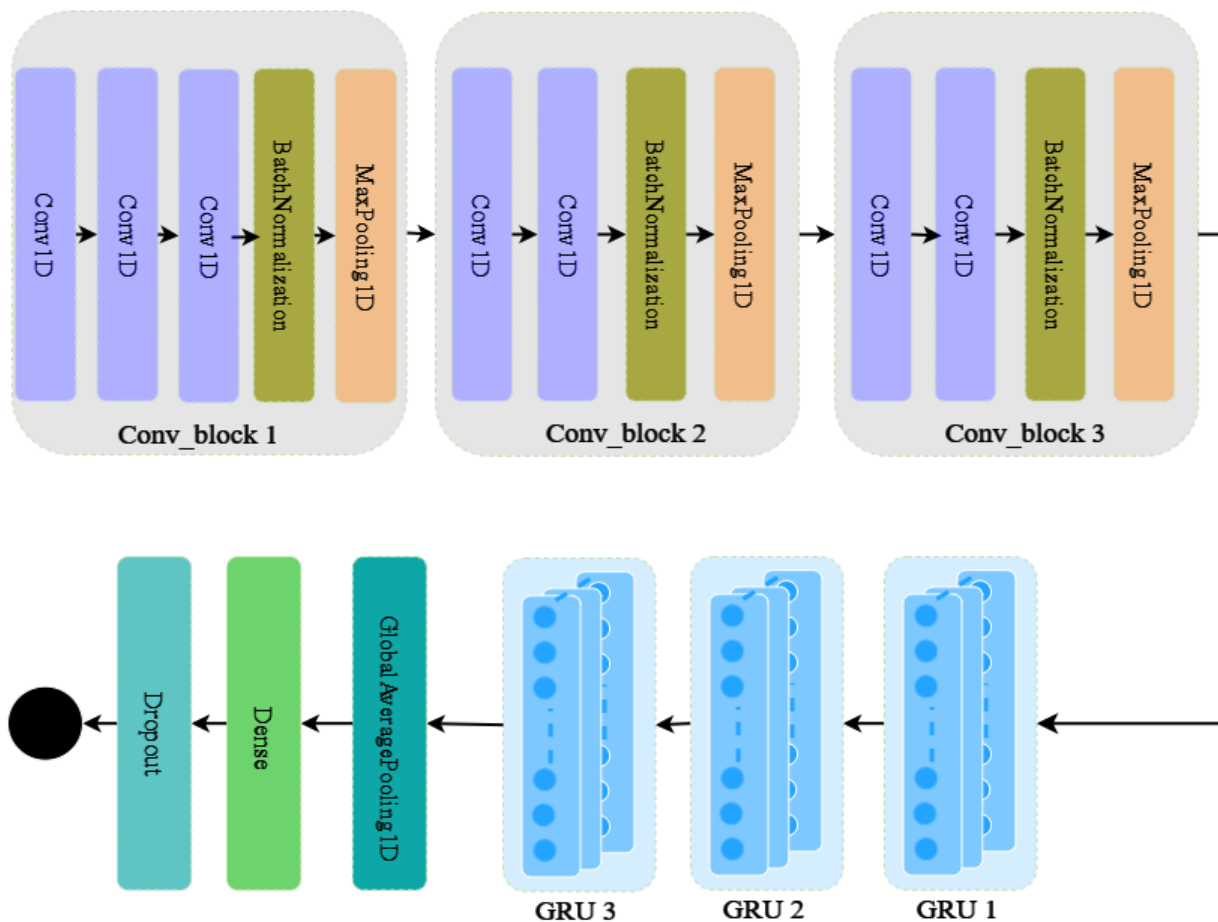


Figure 1. Flowchart of the proposed CNN-LSTM.

Algorithm 1 Pseudo-code of CNN-GRU	
Input:	Input data (D), batch size (B_s), maximum epoch (T), and learning rate (lr)
Output:	$loss$ (MSE), $RMSE$
1:	Conducting the preprocessing step /* Create the proposed CNN-GRU model */
2:	Input: Construct an input layer to receive the input data /* feature extraction based on the LSTM */ /* first convolution block*/
3:	x: Create a Conv1D layer with 64 filters, 3 kernel sizes, the same padding, and a Relu activation function to take the data from the input layer.
4:	x: Add a Conv1D layer with 64 filters, 3 kernel sizes, the same padding, and a Relu activation function to x .
5:	x: Add a Conv1D layer with 64 filters, 3 kernel sizes, the same padding, and a Relu activation function to x .
6:	x: Add a BatchNormalization to x . /* second convolution block*/
7:	x: Add a Conv1D layer with 128 filters, 3 kernel sizes, the same padding, and a Relu activation function to x .
8:	x: Add a Conv1D layer with 128 filters, 3 kernel sizes, the same padding, and a Relu activation function to x .

```

9:  x: Add a BatchNormalization to x.
10: x: Add a MaxPooling1D with pool size equals 2 to x.
    /* third convolution block*/
11: x: Add a Conv1D layer with 256 filters, 3 kernel sizes, the same padding, and a Relu activation function to x.
12: x: Add a Conv1D layer with 256 filters, 3 kernel sizes, the same padding, and a Relu activation function to x.
13: x: Add a BatchNormalization to x.
14: x: Add a MaxPooling1D with pool size equals 2 to x.
    /* GRU block*/
15: x: Add GRU layer with 32 units, 0.1 dropouts, and Tanh activation function to x.
16: x: Add GRU layer with 32 units, 0.1 dropouts, and Tanh activation function to x.
17: x: Add GRU layer with 32 units, 0.1 dropouts, and Tanh activation function to x.
    /* Prediction Block */
18: x: Add a GlobalAveragePooling1D to x.
19: x: Add a dense layer with 32 nodes, and a Relu activation function to x.
20: x: Add a Dropout layer with 0.1 value as a dropout rate to x.
21: x: Add a dense layer with 1 node to x.
    /* Optimization process */
22:  $N = \text{Size}(D)/B_s$  /* Estimate the number of batches */
23:  $t = 0$ , Current epoch
24: while  $t < T$ 
25:     |    $i = 0$ , the current batch size.
26:     |   while  $i < N$ 
27:     |       |   Optimize the MSE function by updating the weights according to Adam.
28:     |       |    $i = i + 1$ 
29:     |       |   end while.
30:     |    $t = t + 1$ 
31: end while

```

3 | Experimental Settings

3.1 | PHM 2010 Dataset

The IEEE PHM 2010 dataset [47] is used as a benchmark in this part to show how the created monitoring system is applied. Data on experimental samples were acquired at the Roders Tech RFM760 milling facility. Figure 2 shows a schematic illustration of the experimental setup. Throughout the procedure, none of the tools' cutting settings are altered. Three sets of tool run-to-failure tests are included in the dataset. A stainless-steel block measuring 108 mm in the tool feed direction served as the workpiece. The cutting process was done sporadically while the tool was set to mill 108 mm of the workpiece in each cycle. The feed rate was set to 1555 mm/min, the spindle speed was set to 10,400 rpm, the radial depth was set to 0.125 mm, and the axial depth was set to 0.2 mm. These characteristics were unchanged for the duration of the instruments' life. On the machining table were the dynamometers for measuring the cutting force in three directions. To monitor the vibration of the tool in three different directions, three Kistler Piezo accelerometers were mounted on the workpiece. Sensor data was saved using the DAQ NI PCI1200 at a sampling frequency of 50 Hz. The sensor data comprises three types of vibration in three directions (V_x , V_y , and V_z), cutting force (F_x , F_y , and F_z), and acoustic emission (AE). The goal output is the VB value of three flutes measured offline with the LEICA MZ12 microscope after each surface is finished. The objective is to forecast the real VB value. The test instruments that comprised our dataset, c1, c4, and c6, were documented. The three tools are identified as C1, C4, and C6 in the IEEE PHM 2010 dataset. Figure 3 displays the measures of tool wear. There are 315 data samples in each tool, and each data sample has an associated wear value.

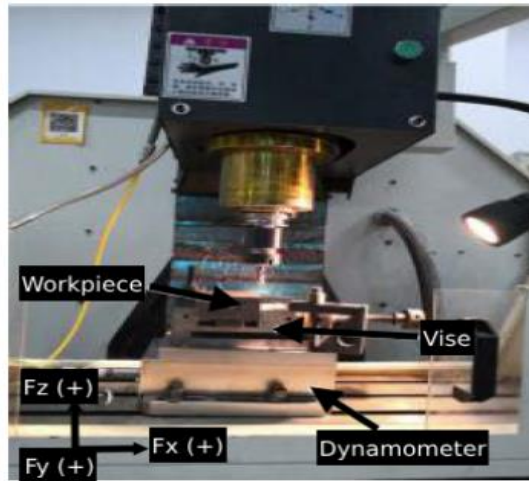


Figure 2. PHM 2010 competition operating platform.

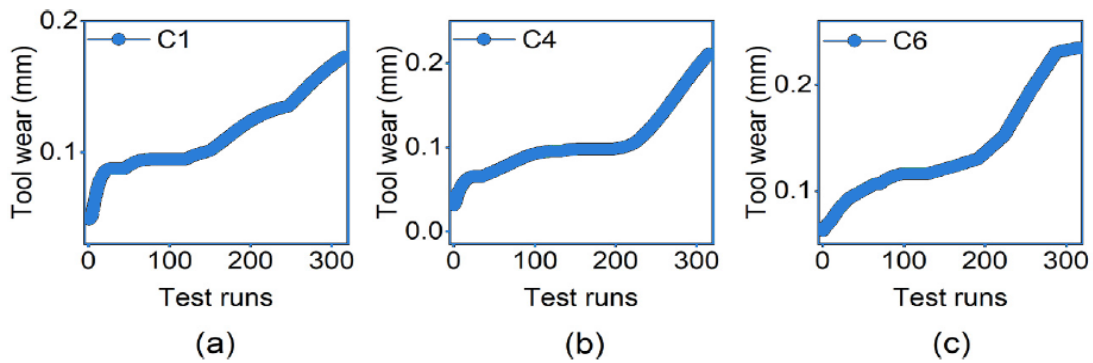


Figure 3. Measurements of tool wear are (a) C1, (b) C4, and (c) C6.

3.2 | The Measurement of the Tool Wear

The total length for every pass is 108 mm, and every tool requires full 315 processing. Every cutting's tool wear value is calculated and noted. Tools 1, 4, and 6's VB wear curves, which illustrate the correlation between the number of cuts and the VB value, are displayed in Figure 4 after the study's full-time wear data of Tools was reviewed. Three phases of the tool VB value change are depicted in Figure 4. Tool wear increases quickly in the first stage because of the new cutting-edge sharpness. Wear moves into a steady state and begins to rise gradually as the tool's contact area with the workpiece increases. Tool wear moves into the Sharp stage once the threshold is reached.

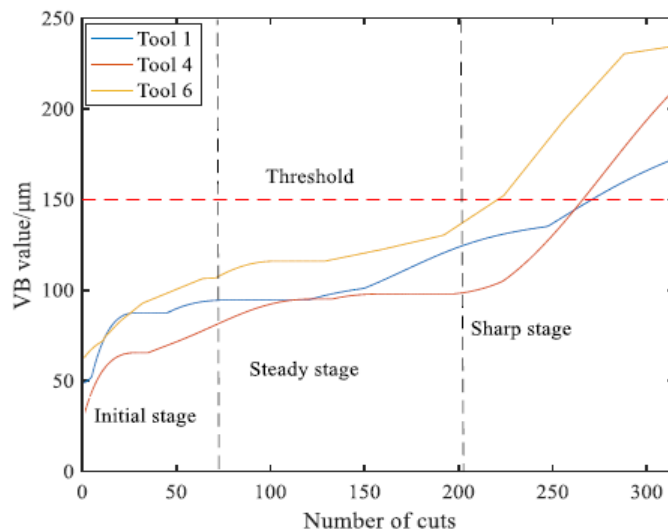


Figure 4. The number of cuts and tool wear (VB) relationship.

3.3 | Signal Analysis

Figure 5 displays a visualization and examination of the cutting force signals along the x, y, and z axes using the complete lifespan data of Tool 1, which was obtained by analysis of the experiment's full-life sensors data of Tool 1, Tool 4, and Tool 6 involving 315 cuts. The vibration signal is especially susceptible to acute wear of the tool, as seen in Figure 6, which displays the original acceleration vibration signals in x, y, and z directions. Figure 7 displays the first few data points of the cutting force signals along the x, y, and z axes using the complete lifespan data of Tool 1.

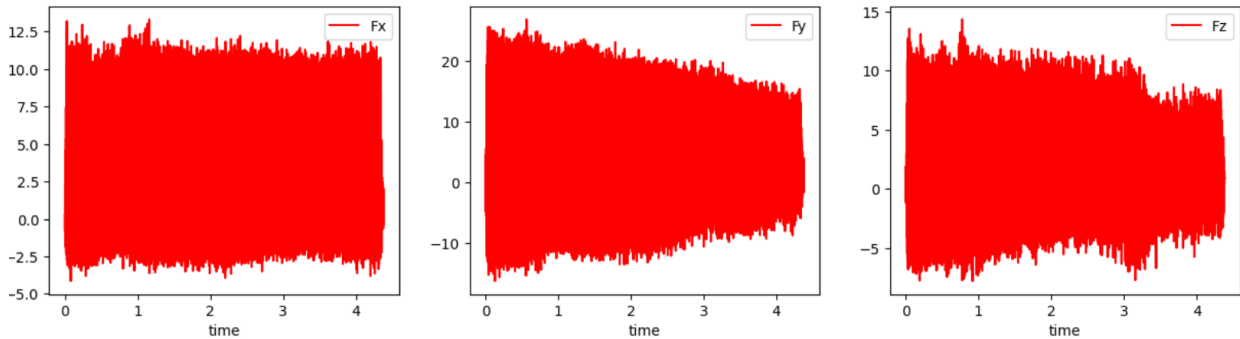


Figure 5. The cutting force signals along the x, y, and z axes using the complete lifespan data of Tool 1.

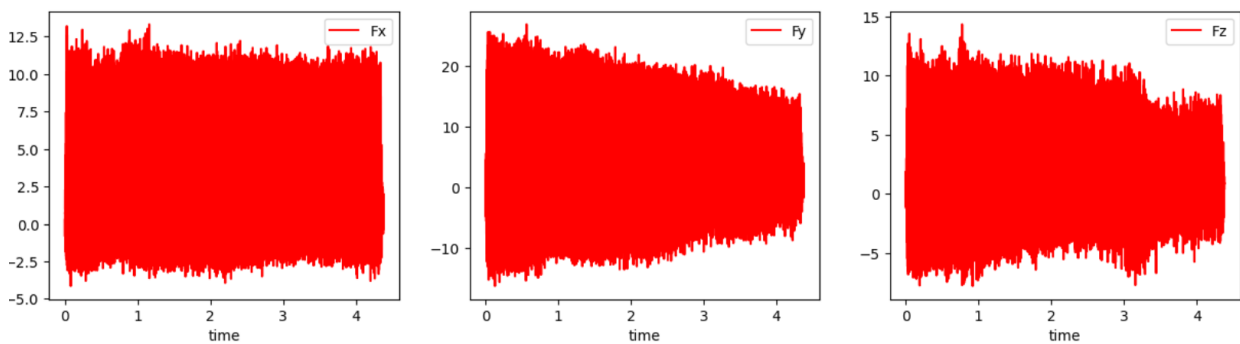


Figure 6. The original acceleration vibration signals in x, y, and z directions of Tool 1.

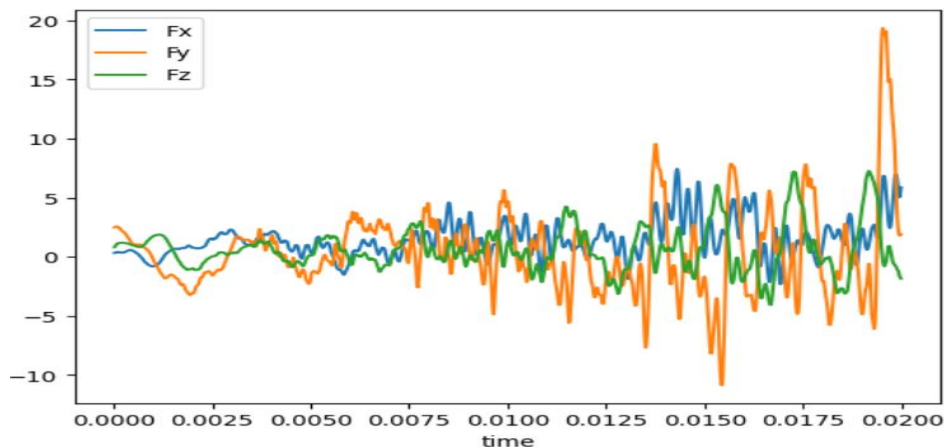


Figure 7. The first few data points of the cutting force signals using the complete lifespan data of Tool 1.

A multi-domain combination approach was utilized to create candidate features, avoiding the drawbacks of single-domain features with insufficient tool status data. From every channel signal, we have extracted a few characteristics: 12 time-domain features, 4 frequency-domain features, and 8 time-frequency domain features. The time-domain features include absolute mean, max, root mean square, square root amplitude, skewness, kurtosis, shape factor, pulse factor, skewness factor, crest factor, clearance factor, and kurtosis factor. The equations of these features are calculated from Eq. (5) to Eq. (16). Figure 8 displays the time-domain features

for Tool 1. The frequency-domain features include the center of gravity frequency (FC), mean square frequency (MSF), root mean square frequency (RMSF), and variance Frequency (VF). The time-frequency domain features are generated using Wavelet Packet Transform (WPT) by calculating the norms of the resulting wavelet packet coefficients. Figure 9 displays the time-frequency domain features for Tool 1.

$$\text{Absolute mean value} = \frac{\sum_{i=1}^N |x_i|}{N} \quad (5)$$

$$\text{Max} = \max_{i=1,2,\dots,N} (x_i) \quad (6)$$

$$\text{Root mean square value} = \sqrt{\frac{\sum_{i=1}^N x_i^2}{N}} \quad (7)$$

$$\text{Square root amplitude} = \left(\frac{\sum_{i=1}^N \sqrt{|x_i|}}{N} \right)^2 \quad (8)$$

$$\text{Skewness} = \frac{1}{N} \sum_{i=1}^N (|x_i| \quad (9)$$

– X)³ Where X is the mean of the absolute values

$$\text{Kurtosis} = \frac{1}{N} \sum_{i=1}^N x_i^4 \quad (10)$$

$$\text{Shape factor} = \frac{\text{Root mean square value}}{\text{Absolute mean value}} \quad (11)$$

$$\text{Pulse factor} = \frac{\text{Max}}{\text{Absolute mean value}} \quad (12)$$

$$\text{Skewness factor} = \frac{\text{Skewness}}{(\text{Root mean square value})^3} \quad (13)$$

$$\text{Crest factor} = \frac{\text{Max}}{\text{Root mean square value}} \quad (14)$$

$$\text{Clearance factor} = \frac{\text{Max}}{\text{Square root amplitude}} \quad (15)$$

$$\text{Kurtosis factor} = \frac{\text{Kurtosis}}{(\text{Root mean square value})^4} \quad (16)$$

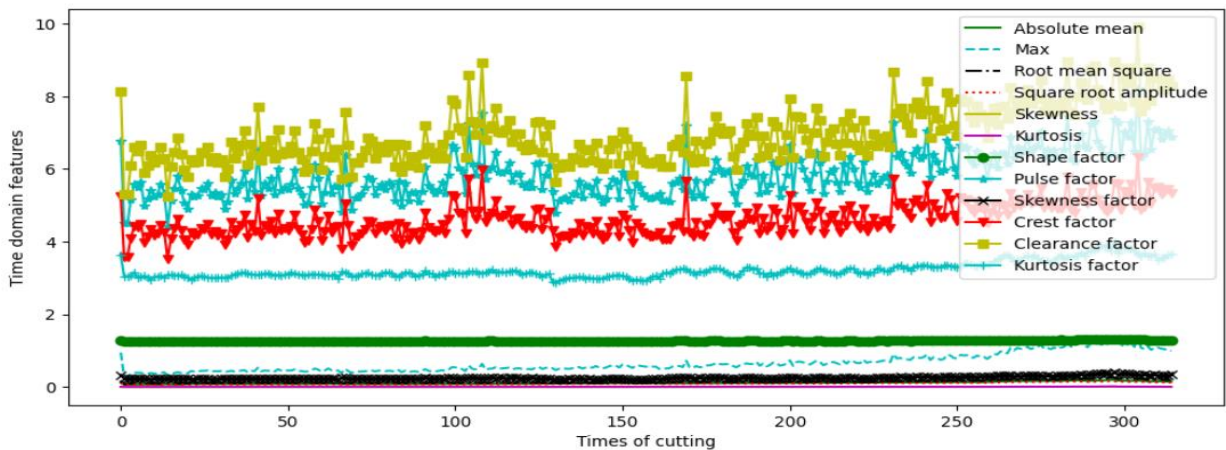


Figure 8. The time-domain features for C1 (Tool 1).

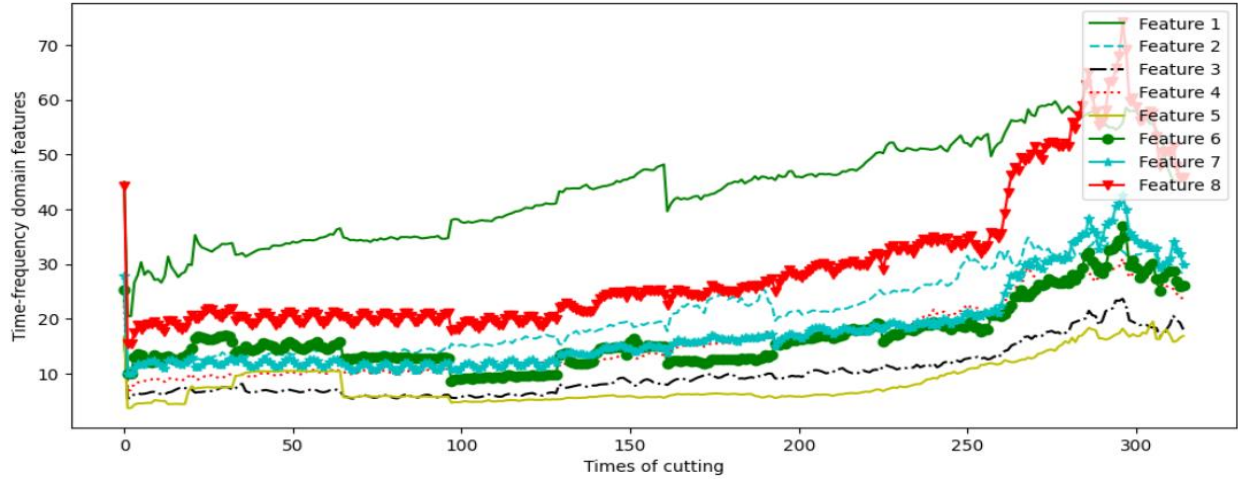


Figure 9. The time-frequency domain features for C1 (Tool 1).

3.4 | Data Normalization

The uniformity of the inputs is crucial for the deep learning model since any significant discrepancy may negatively impact the model's performance. Therefore, before being included in the model, the vibration signals collected at each time period should be normalized. Z-score normalization, also known as standardization, is a popular method for this that standardizes the values of various parameters. This scaling technique is very important in the field of machine learning, particularly when dealing with characteristics that have different scales or units. This method guarantees that the features are rescaled to provide a mean of 0 and a standard deviation of 1 by transforming the data to a standardized range. The mathematical formula for z-score scaling is presented as Eq. (17). where μ_j , and σ_j donate to the mean and standard deviation of the j th feature, respectively.

$$x'_{i,j} = \frac{x_{i,j} - \mu_j}{\sigma_j} \quad (17)$$

3.5 | Evaluation Metrics

The network parameters are improved in this study by utilizing the mean square error (MSE) loss, which can be found using Eq. (18), and the Adam optimization [41] approach.

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - y'_i)^2 \quad (18)$$

The true and predicted labels of the i th sample are represented y_i and y'_i , respectively, while N stands for the total sample size. The suggested model has been evaluated using the root mean square error (RMSE) evaluation measure, which compares the RMSE values between each dataset's real and projected labels for each data point. Eq. (19) provides the mathematical method used to determine the RMSE value.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - y'_i)^2} \quad (19)$$

The reduction of both MSE and RMSE is essential for enhancing the accuracy in predicting the tool wear estimation of cutting tools.

4 | Experiment Results and Discussion

4.1 | Hyperparameter Tuning

Several hyper-parameters are included in the CNN-GRU model suggested in this paper, including the number of units inside each GRU layer, the number of filters and kernel size within each convolutional layer, the GRU dropout rate, and the learning rate. To improve the model's efficiency and lower the RMSE, these parameters must be carefully defined. Thus, Table 1 shows that several experiments were carried out in this research to investigate various configurations for each parameter in order to get the most ideal values that result in a significant improvement in the model's performance. Figures 10 through 18 show how these hyperparameters affect the CNN-GRU approach's effectiveness. For example, the number of filters in each convolutional layer has a big impact on the model's performance. As a result, several tests were run, ranging from 64 to 256, to determine the optimal number of filters for each convolutional layer. Figures 10, 11, and 12 illustrate how the number of filters in each convolutional layer affects the results.

Table 1. Experimental analysis of the influence of Hyperparameter on RMSE of the estimation value in mm.

Hyper-parameter	Value	C1	C4	C6	Average
Number of filters in Conv_block_1	64	0.0017	0.0011	0.0012	0.0013
	128	0.0527	0.0511	0.0515	0.0517
	256	0.0736	0.0741	0.0755	0.0744
Number of filters in Conv_block_2	64	0.0167	0.0159	0.0162	0.0162
	128	0.0017	0.0011	0.0012	0.0013
	256	0.0215	0.0199	0.0201	0.0205
Number of filters in Conv_block_3	64	0.0264	0.0259	0.0261	0.0261
	128	0.0140	0.0149	0.0155	0.0148
	256	0.0017	0.0011	0.0012	0.0013
Number of units of GRU	32	0.0017	0.0011	0.0012	0.0013
	64	0.0025	0.0020	0.0021	0.0022
	128	0.0084	0.0075	0.0074	0.0077
Kernel size	2	0.0023	0.0020	0.0018	0.0020
	3	0.0017	0.0011	0.0012	0.0013
	4	0.0031	0.0028	0.0029	0.0029
Optimizer	Adam	0.0017	0.0011	0.0012	0.0013
	SGD	0.0214	0.0220	0.0236	0.0223
	RMSprop	0.0157	0.0174	0.0201	0.0177
Batch size	64	0.0049	0.0041	0.0045	0.0045
	128	0.0017	0.0011	0.0012	0.0013
	256	0.0035	0.0029	0.0031	0.0031
Learning rate	0.001	0.0021	0.0019	0.0020	0.0017
	0.002	0.0017	0.0011	0.0012	0.0013
	0.006	0.0019	0.0026	0.0018	0.0021
	0.01	0.0159	0.0161	0.0124	0.0148
Dropout rate	0.1	0.0017	0.0011	0.0012	0.0013
	0.2	0.0039	0.0036	0.0041	0.0038
	0.3	0.0057	0.0049	0.0054	0.0053

During neural network training, the dropout rate is a crucial hyperparameter that helps to prevent overfitting. Overfitting is the phenomenon when a model adds noise and complexities that are ineffective when applied to fresh, unknown data, on top of the underlying patterns seen in the training data. To determine the ideal dropout rate, several studies are carried out. Figure 18 illustrates how various dropout rates affect the situation. A critical hyperparameter in deep learning model training is the learning rate, which determines how much of the optimization process's modifications are made to steer the method's parameters in the direction of minimizing the loss function. From the alternatives of 0.0001, 0.001, 0.006, and 0.01, several tests were conducted to determine the most appropriate learning rate. The findings indicate that 0.006 is the ideal

learning rate. Figure 17 depicts the impact of various learning rate values. The final hyperparameters of the proposed model are detailed in Table 2.

Table 2. The CNN-GRU hyperparameters.

Parameter	value
Number of filters in Conv_block_1	64
Number of filters in Conv_block_2	128
Number of filters in Conv_block_3	256
Number of units of GRU	32
Kernel size	128
Learning rate	0.002
Batch size	128
Max no. of epoch	500
Dropout rate	0.1
Loss	MSE
Optimizer	Adam

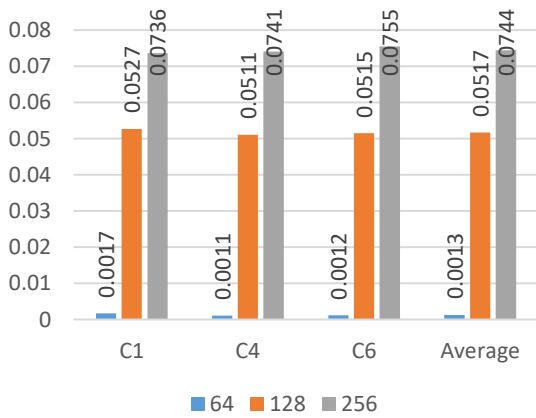


Figure 10. The impact of conv_block_1 filter number through experimentation.

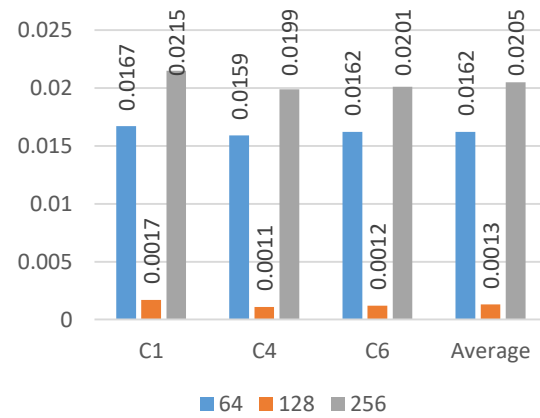


Figure 11. The impact of conv_block_2 filter number through experimentation.

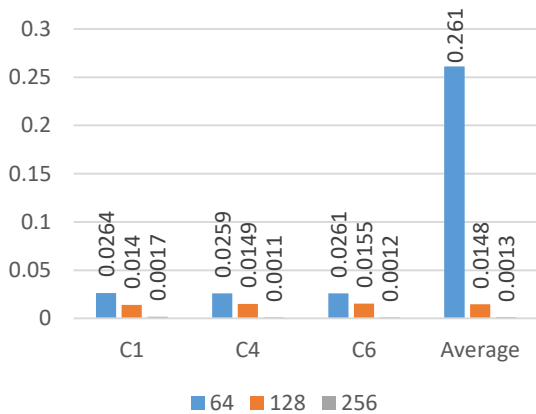


Figure 12. The impact of conv_block_3 filter number through experimentation.

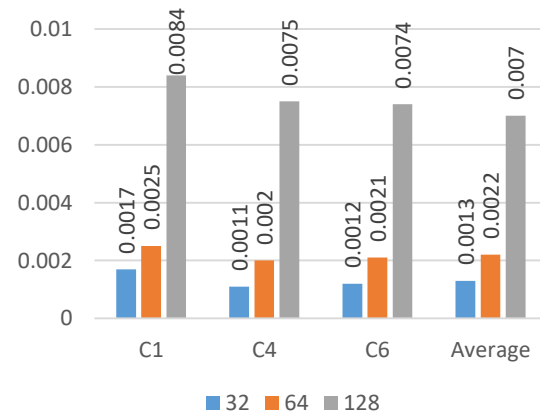


Figure 13. The impact of GRU unit number through experimentation.

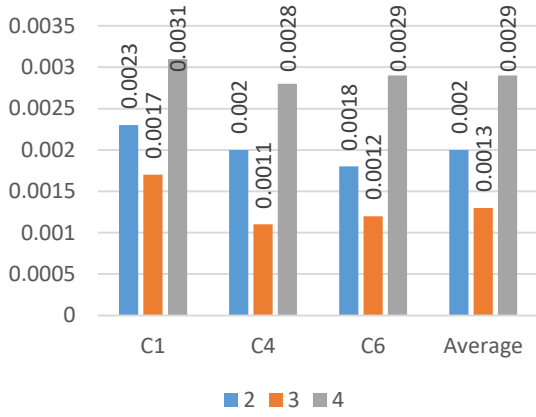


Figure 14. The influence of kernel size through experiments.

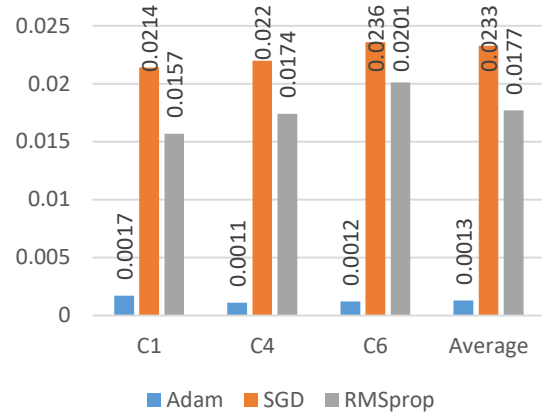


Figure 15. The influence of optimizer through experiments.

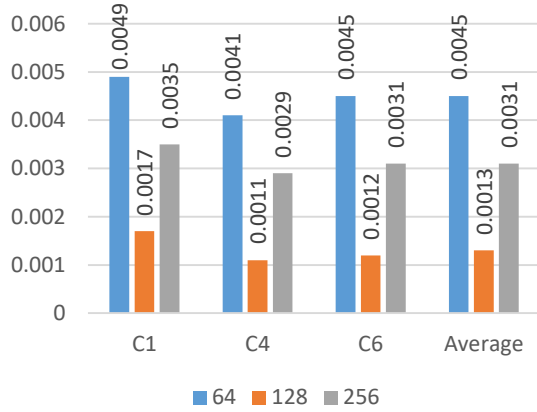


Figure 16. The influence of batch size through experiments.

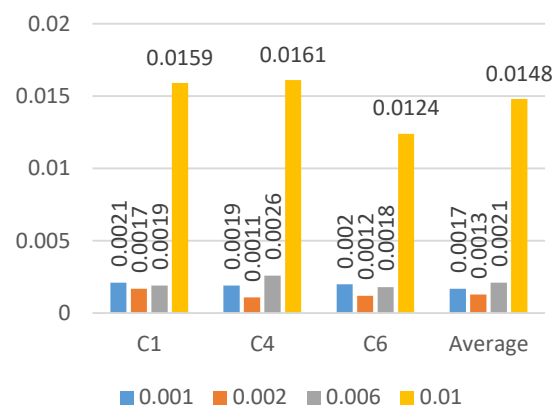


Figure 17. The influence of learning rate through experiments.

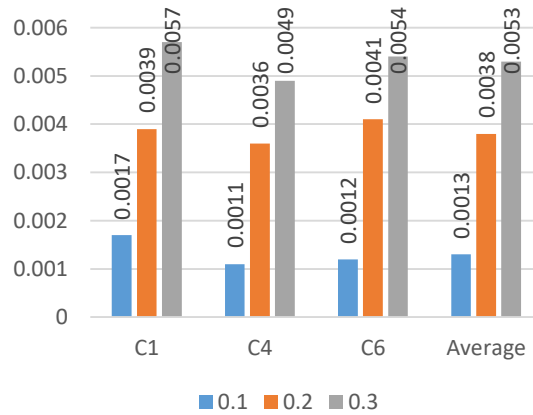


Figure 18. The influence of dropout rate through experiments.

4.2 | Results for Cutting Tools

The CNN-GRU model's outcomes over the three tools (C1, C4, and C6) are shown in this section. Using C4 and C6 tools, the CNN-GRU model was trained for C1. The tool wear estimation's root mean square error (RMSE) was 0.0017 when the CNN-GRU was evaluated using the C1 tool. C1 and C6 tools were used to train the CNN-GRU model for C4. The tool wear estimation's root mean square error (RMSE) was 0.0011 when the CNN-GRU was evaluated using the C4 tool. Using C1 and C4 tools, the CNN-GRU model was trained for C6. The tool wear estimation's root mean square error (RMSE) was 0.0012 when the CNN-GRU was evaluated using the C6 tool. Then, for three tools (C1, C4, and C6) from the PHM 2010 dataset, the

average RMSE for tool wear estimation was 0.0013. The CNN-GRU model's total findings are shown in Table 3, and Figure 19 provides an illustration of it.

Table 3. CNN-GRU performance (tool wear estimation error (mm)) for 3 tools based on RMSE.

	C1	C2	C3	Average
CNN-GRU	0.0017	0.0011	0.0012	0.0013

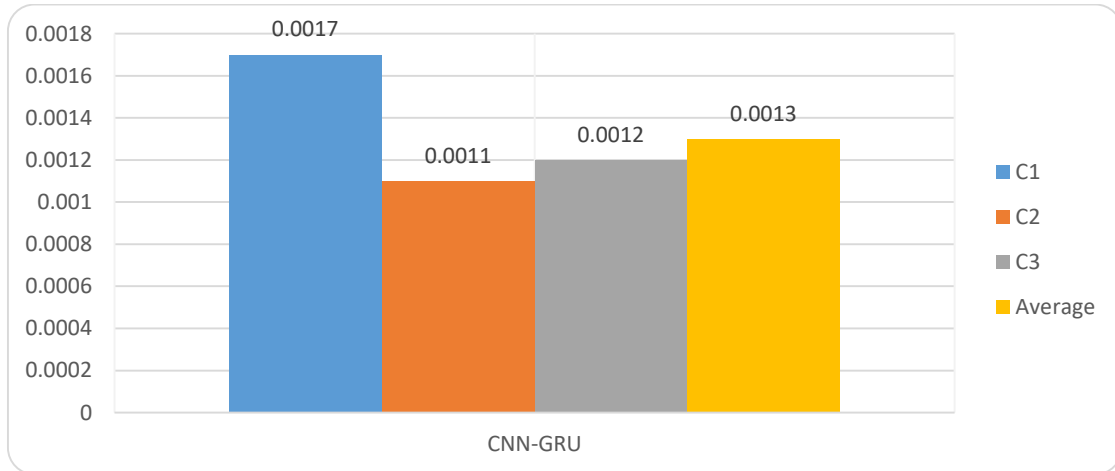


Figure 19. CNN-GRU performance for C1, C2, and C3.

4.3 | Comparison Results

In this part, CNN-GRU's results for the average of the three tools (C1, C4, and C6) in the 2010 PHM Data Challenge Dataset are compared with competing models to demonstrate their performance. The RMSE metric is used to quantify these results in order to show how successfully the models close the difference between the intended and predicted tool wear estimation. The model presented in the 2010 PHM Data Challenge Dataset is compared to a number of rival models, including CNN-LSTM [48], CTNN [49], Bi-LSTM [50], ANN, [35] HSMM [35], Probabilistic model [51], Approach_3 IMM [52], and Approach_5 IMM [52]. The results are presented in a way that demonstrates the system's higher efficacy, as demonstrated by the RMSE values listed in Table 4.

A bold style is used to highlight the major outcomes. With RMSE values of 0.0013, Table 4 shows that CNN-GRU beats all other models examined when compared to the average of the three tools (C1, C4, and C6). When our results are compared to the best results obtained by several cited models, our proposed model exhibits a 13.3% reduction in RMSE for the 2010 PHM Data Challenge Dataset. Because it can function most effectively based on the RMSE measure, which gives equal weight to predictions made sooner and later, the suggested model is thought to be a strong option for handling this situation. Figure 20 shows the RMSE values for the average of the three tools (C1, C4, and C6) that were derived using different algorithms in order to graphically demonstrate the effectiveness of the suggested model.

Table 4. Performance (tool wear estimation error (mm)) comparison with competing models based on the average of 3 tools.

Models	RMSE
CNN-LSTM	0.0026
CTNN	0.0070
Bi-LSTM	0.0155
ANN	0.0413
HSMM	0.0148
Probabilistic model	0.0020
Approach_3 IMM	0.0019
Approach_5 IMM	0.0015
Proposed model	0.0013

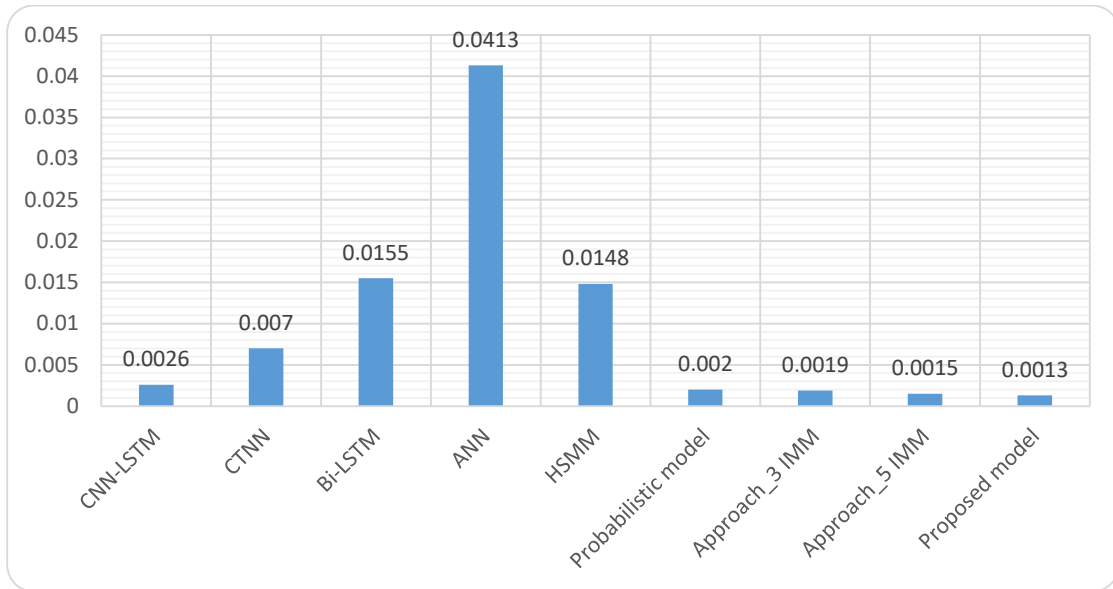


Figure 20. Shows how RMSE values obtained from various models are shown.

4.4 | Ablation Study Results

Using the suggested technique, ablation experiments are carried out to evaluate the impact of each CNN-GRU component separately. Three experiments that are carried out under the same circumstances as the CNN-GRU are described in this section. Several neural network topologies are used in the experiments: a CNN model without a GRU, a GRU model without a CNN, and a CNN-GRU model that combines both CNN and GRU. In addition, Table 5 presents the results of the studies in detail, and Figure 21 provides a visual representation of the results. Over the course of the three tools dataset, the CNN-GRU outperforms CNN and GRU in the RMSE statistic. For C1 (Tool 1), the CNN-GRU approach's RMSE value is recorded at 0.0017, demonstrating improvements of 98.8% and 98.1% over CNN and GRU, respectively. Additionally, the suggested CNN-GRU technique for C4 (Tool 4) is reported at 0.0011, suggesting improvements over CNN and GRU of 99.1% and 98.1%, respectively. The suggested CNN-GRU strategy for C6 (Tool 6) is registered at 0.0012, suggesting improvements over CNN and GRU of 99.2% and 98.7%, respectively. Additionally, the suggested method CNN-GRU is registered at 0.0013 across the average of the three tools (C1, C4, and C6), demonstrating improvements of 99.1% and 98.4% over CNN and GRU, respectively.

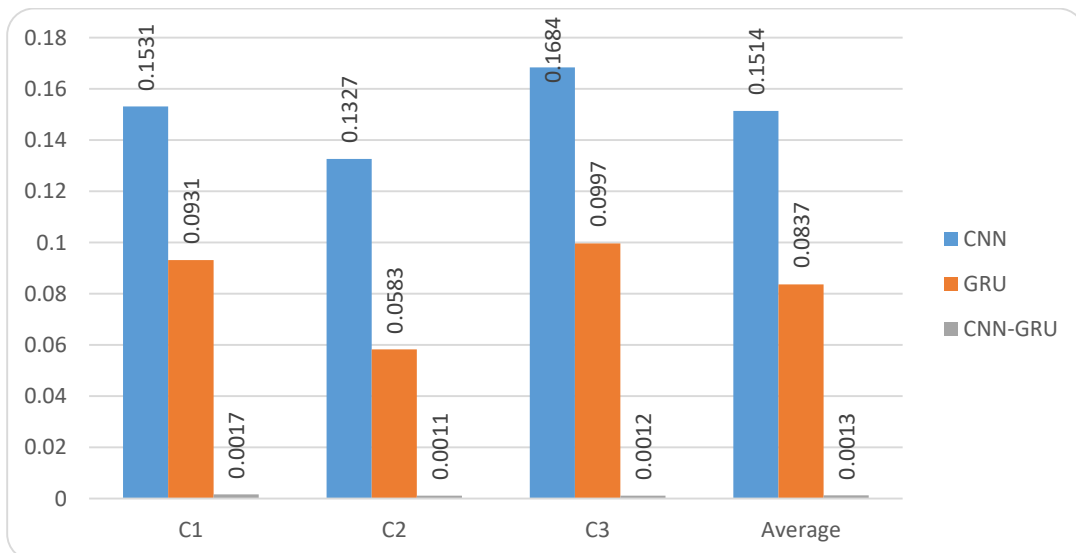


Figure 21. The representation of RMSE values acquired through ablation experiments.

Table 5. The results of the ablation study performance (tool wear estimation error (mm)) for 3 tools based on RMSE.

	C1	C2	C3	Average
CNN	0.1531	0.1327	0.1684	0.1514
GRU	0.0931	0.0583	0.0997	0.0837
CNN-GRU	0.0017	0.0011	0.0012	0.0013

5 | Conclusion

This paper introduces CNN-GRU, a novel data-driven deep learning technique intended for cutting tool wear estimation. CNN and GRU are combined in the CNN-GRU paradigm. CNN was employed for feature extraction from multi-domain input data, circumventing the limitations of single-domain features with inadequate tool status data, by employing a multi-domain combination strategy to generate candidate features. GRUs are useful for collecting complicated temporal correlations in time series data because they are used to capture and maintain patterns throughout lengthy durations. The projections for tool wear are produced via a Fully Connected (FC) network. The PHM 2010 Challenge dataset was used to conduct the experimental evaluation. Our suggested model, when compared to other well-known models in the field, reduced the root mean square error (RMSE) for the average of the three tools (C1, C4, and C6) by 13.3%.

Acknowledgments

The author is grateful to the editorial and reviewers, as well as the correspondent author, who offered assistance in the form of advice, assessment, and checking during the study period.

Funding

This research was conducted without external funding support.

Data Availability

The datasets generated during and/or analyzed during the current study are not publicly available due to the privacy-preserving nature of the data but are available from the corresponding author upon reasonable request.

Conflicts of Interest

The author declares that there is no conflict of interest in the research.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors.

References

- [1] X. Li, X. Liu, C. Yue, S. Y. Liang, and L. Wang, "Systematic review on tool breakage monitoring techniques in machining operations," *International Journal of Machine Tools and Manufacture*, vol. 176, p. 103882, May 2022, doi: 10.1016/j.ijmachtools.2022.103882.
- [2] G. Li, S. Du, D. Huang, C. Zhao, and Y. Deng, "Dynamics Modeling-Based Optimization of Process Parameters in Face Milling of Workpieces With Discontinuous Surfaces," *Journal of Manufacturing Science and Engineering*, vol. 141, no. 101009, Sep. 2019, doi: 10.1115/1.4044421.
- [3] C. Zhou et al., "Vibration singularity analysis for milling tool condition monitoring," *International Journal of Mechanical Sciences*, vol. 166, p. 105254, Jan. 2020, doi: 10.1016/j.ijmecsci.2019.105254.

- [4] Y.-P. Liu et al., "Bioactive monoterpene indole alkaloids from *Nauclea officinalis*," *Bioorganic Chemistry*, vol. 83, pp. 1–5, Mar. 2019, doi: 10.1016/j.bioorg.2018.10.013.
- [5] H. Guo, Y. Zhang, and K. Zhu, "Interpretable deep learning approach for tool wear monitoring in high-speed milling," *Computers in Industry*, vol. 138, p. 103638, Jun. 2022, doi: 10.1016/j.compind.2022.103638.
- [6] Y. Zhou, C. Liu, X. Yu, B. Liu, and Y. Quan, "Tool wear mechanism, monitoring and remaining useful life (RUL) technology based on big data: a review," *SN Applied Sciences*, vol. 4, no. 8, p. 232, 2022.
- [7] C. Zhou, K. Guo, and J. Sun, "Sound singularity analysis for milling tool condition monitoring towards sustainable manufacturing," *Mechanical Systems and Signal Processing*, vol. 157, p. 107738, Aug. 2021, doi: 10.1016/j.ymsp.2021.107738.
- [8] G. Li, S. Du, B. Wang, J. Lv, and Y. Deng, "High Definition Metrology-Based Quality Improvement of Surface Texture in Face Milling of Workpieces With Discontinuous Surfaces," *Journal of Manufacturing Science and Engineering*, vol. 144, no. 031001, Aug. 2021, doi: 10.1115/1.4051883.
- [9] Y. Zhu, J. Wu, J. Wu, and S. Liu, "Dimensionality reduce-based for remaining useful life prediction of machining tools with multisensor fusion," *Reliability Engineering & System Safety*, vol. 218, p. 108179, Feb. 2022, doi: 10.1016/j.res.2021.108179.
- [10] K. Zhang et al., "Tool wear monitoring using a novel parallel BiLSTM model with multi-domain features for robotic milling Al7050-T7451 workpiece," *The International Journal of Advanced Manufacturing Technology*, vol. 129, no. 3, pp. 1883–1899, Nov. 2023, doi: 10.1007/s00170-023-12322-z.
- [11] A. H. Musfirah, J. A. Ghani, and C. H. C. Haron, "Tool wear and surface integrity of inconel 718 in dry and cryogenic coolant at high cutting speed," *Wear*, vol. 376–377, pp. 125–133, Apr. 2017, doi: 10.1016/j.wear.2017.01.031.
- [12] T. Mohanraj, S. Shankar, R. Rajasekar, N. R. Sakthivel, and A. Pramanik, "Tool condition monitoring techniques in milling process — a review," *Journal of Materials Research and Technology*, vol. 9, no. 1, pp. 1032–1042, Jan. 2020, doi: 10.1016/j.jmrt.2019.10.031.
- [13] C. Zhou, K. Guo, and J. Sun, "An integrated wireless vibration sensing tool holder for milling tool condition monitoring with singularity analysis," *Measurement*, vol. 174, p. 109038, Apr. 2021, doi: 10.1016/j.measurement.2021.109038.
- [14] J. Montalvo-Urquiza, C. Niebuhr, A. Schmidt, and M. G. Villarreal-Marroquín, "Reducing deformation, stress, and tool wear during milling processes using simulation-based multiobjective optimization," *The International Journal of Advanced Manufacturing Technology*, vol. 96, no. 5, pp. 1859–1873, May 2018, doi: 10.1007/s00170-018-1681-y.
- [15] Z. Li, R. Liu, and D. Wu, "Data-driven smart manufacturing: Tool wear monitoring with audio signals and machine learning," *Journal of Manufacturing Processes*, vol. 48, pp. 66–76, Dec. 2019, doi: 10.1016/j.jmapro.2019.10.020.
- [16] C. Zhou, K. Guo, B. Yang, H. Wang, J. Sun, and L. Lu, "Singularity Analysis of Cutting Force and Vibration for Tool Condition Monitoring in Milling," *IEEE Access*, vol. 7, pp. 134113–134124, 2019, doi: 10.1109/ACCESS.2019.2941287.
- [17] S. Cheng, M. H. Azarian, and M. G. Pecht, "Sensor Systems for Prognostics and Health Management," *Sensors*, vol. 10, no. 6, pp. 5774–5797, 2010, doi: 10.3390/s100605774.
- [18] N. E. Huang et al., "The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences*, vol. 454, no. 1971, pp. 903–995, 1998, doi: 10.1098/rspa.1998.0193.
- [19] Z. Geng, X. Zhang, Y. Han, C. Zhang, K. Chen, and F. Xie, "Bearing Health Monitoring Based on the Improved BiLSTM-CRF," in *2021 IEEE 10th Data Driven Control and Learning Systems Conference (DDCLS)*, May 2021, pp. 35–40. doi: 10.1109/DDCLS52934.2021.9455471.
- [20] S. Cai and G. Zhang, "Fatigue life prediction of high-speed railway bearing based on contact stress," presented at the 2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), IEEE, 2017, pp. 650-653.
- [21] Q. Wang, D. Zhang, K. Tang, and Y. Zhang, "A mechanics based prediction model for tool wear and power consumption in drilling operations and its applications," *Journal of Cleaner Production*, vol. 234, pp. 171–184, Oct. 2019, doi: 10.1016/j.jclepro.2019.06.148.
- [22] J. Dou, S. Jiao, C. Xu, F. Luo, L. Tang, and X. Xu, "Unsupervised online prediction of tool wear values using force model coefficients in milling," *The International Journal of Advanced Manufacturing Technology*, vol. 109, no. 3, pp. 1153–1166, Jul. 2020, doi: 10.1007/s00170-020-05684-1.
- [23] D. Liu, Z. Liu, J. Zhao, Q. Song, X. Ren, and H. Ma, "Tool wear monitoring through online measured cutting force and cutting temperature during face milling Inconel 718," *The International Journal of Advanced Manufacturing Technology*, vol. 122, no. 2, pp. 729–740, Sep. 2022, doi: 10.1007/s00170-022-09950-2.
- [24] J. Wu, K. Hu, Y. Cheng, H. Zhu, X. Shao, and Y. Wang, "Data-driven remaining useful life prediction via multiple sensor signals and deep long short-term memory neural network," *ISA Transactions*, vol. 97, pp. 241–250, Feb. 2020, doi: 10.1016/j.isatra.2019.07.004.
- [25] S. Fu, Y. Zhang, L. Lin, M. Zhao, and S. Zhong, "Deep residual LSTM with domain-invariance for remaining useful life prediction across domains," *Reliability Engineering & System Safety*, vol. 216, p. 108012, Dec. 2021, doi: 10.1016/j.res.2021.108012.
- [26] W. Zhang, D. Yang, and H. Wang, "Data-Driven Methods for Predictive Maintenance of Industrial Equipment: A Survey," *IEEE Systems Journal*, vol. 13, no. 3, pp. 2213–2227, Sep. 2019, doi: 10.1109/JSYST.2019.2905565.

- [27] X. Han, Y. He, Z. Wang, Y. Cai, and W. Dai, "Remaining useful life prediction of manufacturing system based on fuzzy Quality State Task Network," *Journal of Manufacturing Systems*, vol. 65, pp. 233–243, Oct. 2022, doi: 10.1016/j.jmsy.2022.09.008.
- [28] T. Benkedjouh, K. Medjaher, N. Zerhouni, and S. Rechak, "Health assessment and life prediction of cutting tools based on support vector regression," *Journal of Intelligent Manufacturing*, vol. 26, no. 2, pp. 213–223, Apr. 2015, doi: 10.1007/s10845-013-0774-6.
- [29] R. Khelif, B. Chebel-Morello, S. Malinowski, E. Laajili, F. Fnaiech, and N. Zerhouni, "Direct Remaining Useful Life Estimation Based on Support Vector Regression," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 3, pp. 2276–2285, Mar. 2017, doi: 10.1109/TIE.2016.2623260.
- [30] D. A. Tobon-Mejia, K. Medjaher, and N. Zerhouni, "CNC machine tool's wear diagnostic and prognostic by using dynamic Bayesian networks," *Mechanical Systems and Signal Processing*, vol. 28, pp. 167–182, Apr. 2012, doi: 10.1016/j.ymssp.2011.10.018.
- [31] A. Mosallam, K. Medjaher, and N. Zerhouni, "Data-driven prognostic method based on Bayesian approaches for direct remaining useful life prediction," *Journal of Intelligent Manufacturing*, vol. 27, no. 5, pp. 1037–1048, Oct. 2016, doi: 10.1007/s10845-014-0933-4.
- [32] J. Ben Ali, B. Chebel-Morello, L. Saidi, S. Malinowski, and F. Fnaiech, "Accurate bearing remaining useful life prediction based on Weibull distribution and artificial neural network," *Mechanical Systems and Signal Processing*, vol. 56–57, pp. 150–172, May 2015, doi: 10.1016/j.ymssp.2014.10.014.
- [33] H. Sun, D. Cao, Z. Zhao, and X. Kang, "A Hybrid Approach to Cutting Tool Remaining Useful Life Prediction Based on the Wiener Process," *IEEE Transactions on Reliability*, vol. 67, no. 3, pp. 1294–1303, Sep. 2018, doi: 10.1109/TR.2018.2831256.
- [34] S. Cho, S. Asfour, A. Onar, and N. Kaundinya, "Tool breakage detection using support vector machine learning in a milling process," *International Journal of Machine Tools and Manufacture*, vol. 45, no. 3, pp. 241–249, Mar. 2005, doi: 10.1016/j.ijmachtools.2004.08.016.
- [35] M. Lin, S. Wanqing, D. Chen, and E. Zio, "Evolving Connectionist System and Hidden Semi-Markov Model for Learning-Based Tool Wear Monitoring and Remaining Useful Life Prediction," *IEEE Access*, vol. 10, pp. 82469–82482, 2022, doi: 10.1109/ACCESS.2022.3196016.
- [36] J. Dong, K. V. R. Subrahmanyam, Y. S. Wong, G. S. Hong, and A. R. Mohanty, "Bayesian-inference-based neural networks for tool wear estimation," *The International Journal of Advanced Manufacturing Technology*, vol. 30, no. 9, pp. 797–807, Oct. 2006, doi: 10.1007/s00170-005-0124-8.
- [37] J. -Y. Wu, M. Wu, Z. Chen, X. -L. Li, and R. Yan, "Degradation-Aware Remaining Useful Life Prediction With LSTM Autoencoder," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–10, 2021, doi: 10.1109/TIM.2021.3055788.
- [38] J. Zhu, N. Chen, and W. Peng, "Estimation of Bearing Remaining Useful Life Based on Multiscale Convolutional Neural Network," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 4, pp. 3208–3216, Apr. 2019, doi: 10.1109/TIE.2018.2844856.
- [39] D. Belmoud, T. Benkedjouh, M. Lachi, A. Laggoun, and J. Dron, "Deep convolutional neural networks for Bearings failure prediction and temperature correlation," *Journal of Vibroengineering*, vol. 20, no. 8, pp. 2878–2891, 2018, doi: 10.21595/jve.2018.19637.
- [40] X. Li, Q. Ding, and J.-Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering & System Safety*, vol. 172, pp. 1–11, Apr. 2018, doi: 10.1016/j.res.2017.11.021.
- [41] C. Liu, L. Zhang, J. Niu, R. Yao, and C. Wu, "Intelligent prognostics of machining tools based on adaptive variational mode decomposition and deep learning method with attention mechanism," *Neurocomputing*, vol. 417, pp. 239–254, Dec. 2020, doi: 10.1016/j.neucom.2020.06.116.
- [42] L. Wan, K. Chen, Y. Li, Y. Wu, Z. Wang, and C. Li, "A Novel Remaining Useful Life Prediction Method Based on CEEMDAN-IFTC-PSR and Ensemble CNN/BiLSTM Model for Cutting Tool," *IEEE Access*, vol. 10, pp. 2182–2195, 2022, doi: 10.1109/ACCESS.2021.3140165.
- [43] H. Hanachi, W. Yu, I. Y. Kim, J. Liu, and C. K. Mechefske, "Hybrid data-driven physics-based model fusion framework for tool wear prediction," *The International Journal of Advanced Manufacturing Technology*, vol. 101, no. 9, pp. 2861–2872, Apr. 2019, doi: 10.1007/s00170-018-3157-5.
- [44] H. Liu, J. Chen, D. Hissel, and H. Su, "Remaining useful life estimation for proton exchange membrane fuel cells using a hybrid method," *Applied Energy*, vol. 237, pp. 910–919, Mar. 2019, doi: 10.1016/j.apenergy.2019.01.023.
- [45] Q. Yang, K. R. Pattipati, U. Awasthi, and G. M. Bollas, "Hybrid data-driven and model-informed online tool wear detection in milling machines," *Journal of Manufacturing Systems*, vol. 63, pp. 329–343, Apr. 2022, doi: 10.1016/j.jmsy.2022.04.001.
- [46] J. Wang, Y. Li, R. Zhao, and R. X. Gao, "Physics guided neural network for machining tool wear prediction," *Journal of Manufacturing Systems*, vol. 57, pp. 298–310, Oct. 2020, doi: 10.1016/j.jmsy.2020.09.005.
- [47] "2010 PHM Data Challenge." 2010. [Online]. Available: <https://www.phmsociety.org/competition/phm/10>
- [48] H. Sun, J. Zhang, R. Mo, and X. Zhang, "In-process tool condition forecasting based on a deep learning method," *Robotics and Computer-Integrated Manufacturing*, vol. 64, p. 101924, Aug. 2020, doi: 10.1016/j.rcim.2019.101924.
- [49] H. Liu, Z. Liu, W. Jia, X. Lin, and S. Zhang, "A novel transformer-based neural network model for tool wear estimation," *Measurement Science and Technology*, vol. 31, no. 6, p. 065106, Apr. 2020, doi: 10.1088/1361-6501/ab7282.

- [50] Y.-W. Chan, T.-C. Kang, C.-T. Yang, C.-H. Chang, S.-M. Huang, and Y.-T. Tsai, "Tool wear prediction using convolutional bidirectional LSTM networks," *The Journal of Supercomputing*, vol. 78, no. 1, pp. 810–832, Jan. 2022, doi: 10.1007/s11227-021-03903-4.
- [51] Z. Ma, M. Zhao, X. Dai, and Y. Chen, "A hybrid-driven probabilistic state space model for tool wear monitoring," *Mechanical Systems and Signal Processing*, vol. 200, p. 110599, Oct. 2023, doi: 10.1016/j.ymssp.2023.110599.
- [52] Q. Yang, D. Mishra, U. Awasthi, G. M. Bolas, and K. R. Pattipati, "Tool wear and remaining useful life estimation in precision machining using interacting multiple model," *Journal of Manufacturing Systems*, vol. 74, pp. 367–386, Jun. 2024, doi: 10.1016/j.jmsy.2024.04.001.

Disclaimer/Publisher's Note: The perspectives, opinions, and data shared in all publications are the sole responsibility of the individual authors and contributors, and do not necessarily reflect the views of Sciences Force or the editorial team. Sciences Force and the editorial team disclaim any liability for potential harm to individuals or property resulting from the ideas, methods, instructions, or products referenced in the content.