Int J Adv Manuf Technol
DOI 10.1007/s00170-005-0114-x

ORIGINAL ARTICLE

Gang Yu · Hai Qiu · Dragan Djurdjanovic · Jay Lee

Feature signature prediction of a boring process using neural network modeling with confidence bounds

Received: 16 March 2004 / Accepted: 4 April 2005

Abstract Prediction of machine tool failure has been very important in modern metal cutting operations in order to meet the growing demand for product quality and cost reduction. This paper presents the study of building a neural network model for predicting the behavior of a boring process during its full life cycle. This prediction is achieved by the fusion of the predictions of three principal components extracted as features from the joint time–frequency distributions of energy of the spindle loads observed during the boring process. Furthermore, prediction uncertainty is assessed using nonlinear regression in order to quantify the errors associated with the prediction. The results show that the implemented Elman recurrent neural network is a viable method for the prediction of the feature behavior of the boring process, and that the constructed confidence bounds provide information crucial for subsequent maintenance decision making based on the predicted cutting tool degradation.

Keywords Prediction · Neural networks · Prediction confidence bounds · Boring process · Degradation

NSF Industry/University Cooperative Research Center (NSF I/UCRC) for Intelligent Maintenance Systems (IMS).

G. Yu (✉)
Department of Mechanical Engineering and Automation,
Harbin Institute of Technology (HIT)
Shenzhen Graduate School,
Xili Shenzhen University Town HIT Campus,
Shenzhen, Guangdong 518055, P.R. China
e-mail: gangyu@hitsz.edu.cn
Tel.: 86-755-26032696
Fax: 86-755-26033774

H. Qiu · J. Lee
Department of Industrial and Manufacturing Engineering,
University of Wisconsin-Milwaukee,
Milwaukee, WI 53224, USA

D. Djurdjanovic
Department of Mechanical Engineering,
University of Michigan,
Ann Arbor, MI 48109, USA

1 Introduction

In metal cutting operations, the condition of the cutting tool plays a significant role in achieving consistent quality and controlling the overall cost of the manufacturing process. Considering the damage that tool failure can cause to a machine tool and its peripheral components, it becomes increasingly important in contemporary manufacturing to predict and prevent machine failures, instead of allowing the machine to fail and then react to the failure [1]. The need to achieve near-zero-downtime performance has been driving the shift from the traditional "fail and fix" (FAF) practice to the "predict and prevent" (PAP) paradigm [2].

Since the paradigm of machine degradation assessment was first introduced by Lee [3, 4] based on a quantitative confidence value (CV) index of machine degradation, research focus has been placed on the ability to predict the future condition of a system by observing available performance-related features extracted from various sensors on the machine. This advanced capability of observing past outcomes of a phenomenon in order to anticipate its future behavior enables a more proactive approach to condition-based maintenance (CBM).

The feature domain prediction approach used in this paper is adopted from Engel et al. [5], where an implicit assumption is made that signatures indicative of process performance are available. Figure 1 illustrates this approach based on capturing the internal dynamics of performance-related signatures $F(t)$ using the past and present realizations of feature vectors $F(0)$, F(1),..., $F(N)$ in order to predict its future realizations $F_N(1)$ (one step ahead of realization), $F_N(2)$ (two steps ahead of realization), etc.

In addition, each prediction is associated with an inherent uncertainty of prediction, which is crucial for proper interpretation of feature predictions and the use of those predictions for intelligent and more proactive maintenance decision making [6]. If a region (or probability distribution) of features describing normal process behavior is known, then an overlap between the predicted feature's distribution and the normal behavior region indicates the predicted probability that the underlying

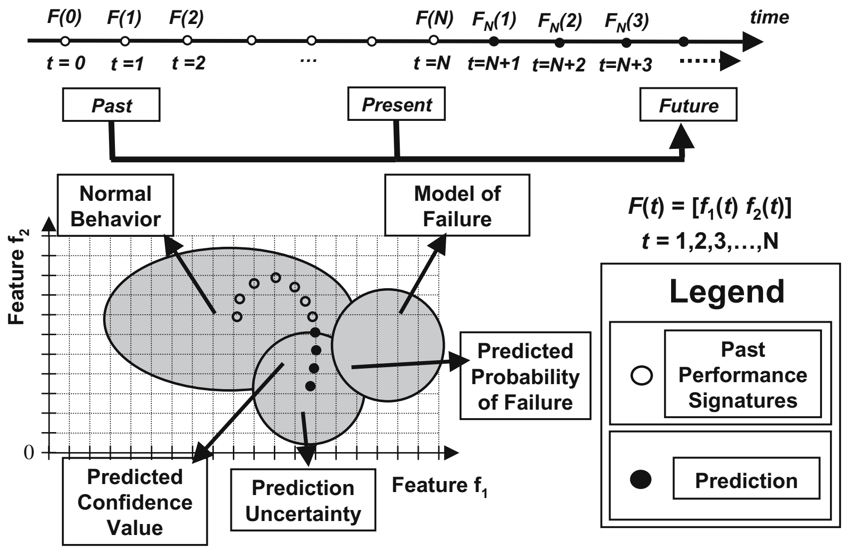


Fig. 1 Illustration of feature domain prediction of process performance

process will behave normally (denoted as the "predicted confidence value" in Fig. 1). Furthermore, if a region (or probability distribution) of features corresponding to a particular failure mode is known, then a measure of the overlap between that feature region with the predicted distribution of process-related features would signify the predicted probability of occurrence of that particular fault.

Traditionally, the prediction of the future condition of a system is based on statistical methods such as autoregressive moving average (ARMA) models [7]. More recently, neural networks have been applied and found more effective than statistical methods [8, 9] for prediction in the cases of nonstationary problems because of neural networks' freedom from restrictive assumptions, such as linearity, which are often needed to make the traditional mathematical models analytically tractable.

However, in developing neural network techniques for real-world applications such as providing the functionalities for key embedded components in mission-critical systems [10], it is noted that it is also necessary to assess the uncertainties inherent in the predictions [11]. Therefore, it is desirable to construct prediction intervals over point predictions generated by the neural network predictor. Methods of computing prediction intervals based on bootstrapping have been developed by Efron and Tibshirani [12], but suffer from a high computational burden. On the other hand, Huang and Ding [13] proved that asymptotically valid prediction intervals for neural networks can be constructed using nonlinear regression.

The objective of this paper is to propose an Elman recurrent neural network for predicting feature signatures describing a manufacturing process, to investigate the forecasting performance of the proposed neural network model on real data from a boring process, and to examine the construction of prediction confidence intervals caused by prediction uncertainty using nonlinear regression.

2 Neural networks for prediction

Since Lapedes and Farber [14] first applied multi-layer feedforward neural networks for forecasting purpose, extensive research work has been devoted to using neural networks for prediction in a variety of application areas. Most studies have been based on the straightforward multi-layer feedforward neural networks. However, recurrent neural networks have been found to be very powerful in forecasting because they are capable of storing the previous states of the system through the recurrent connections [15–18]. Consequently, several researchers have confirmed the superiority of recurrent neural networks over the feedforward neural networks for nonlinear time series prediction [19–23]. The Elman recurrent neural network, which is one type of recurrent neural network, has shown promising

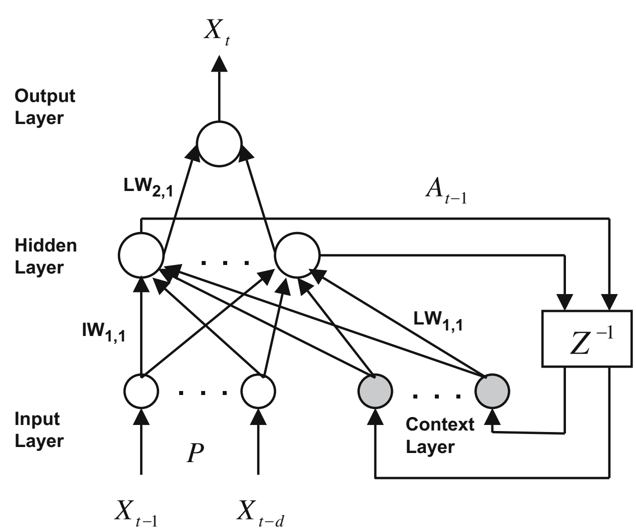


Fig. 2 Structure of an Elman recurrent neural network (ERNN) model

potential for the prediction of polymer product quality [24], dissolution profiles in pharmaceutical product development [25], and chaotic time series [26, 27]. In this paper, an Elman recurrent neural network is implemented for the feature signature prediction of a boring process.

2.1 Elman recurrent neural network (ERNN)

An ERNN, proposed by Elman, is one kind of globally feedforward locally recurrent neural network model. The structure of an ERNN is illustrated in Fig. 2. It consists of four layers: input layer, hidden layer, context layer, and output layer. An ERNN has certain unique dynamic characteristics not displayed by static neural networks, such as back propagation and radial basis function neural networks, because of a set of context nodes storing the previous realization of the internal network states.

For example, an ERNN has a sigmoid activation function in its hidden layer and a linear activation function in its output layer. Then, the internal network states A_t at time t are:

$$A_t = \tan sig(LW_{1,1}A_{t-1} + IW_{1,1}P + b_1) \tag{1}$$

where A_{t-1} is the internal network state at time t−1, P is the time series data from X_{t-1} to X_{t-d}, $LW_{1,\ 1}$ and $IW_{1,\ 1}$ are network weights, and b_1 is the network bias, tan sig is a sigmoid activation function.

Then, the network prediction output X_t at time t is:

$$X_t = purelin(LW_{2,1}A_t + b_2) \tag{2}$$

where $LW_{2,\ 1}$ is a network weight, b_2 is the network bias, and $purelin$ is a linear activation function.

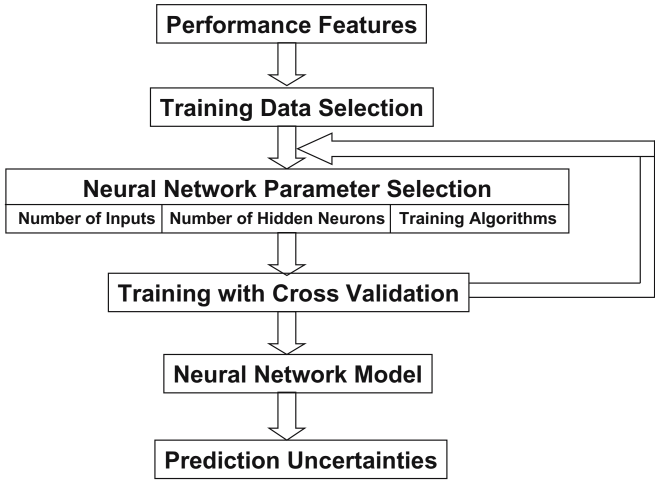


Fig. 4 Flowchart for building a neural network model for forecasting

These distinct recurrent connections of A_{t-1} through context nodes allow the ERNN to both detect and generate time-varying patterns, which is essentially useful in prediction.

3 Quantifying prediction uncertainties by prediction intervals

As an empirical modeling tool of a physical system, neural networks have been used widely in control and optimization applications. However, a model of a physical system has error associated with its predictions due to the dependence of the physical system's output on unobservable quantities. Therefore, a method to quantify the accuracy of the predictions from neural network models is desired. A

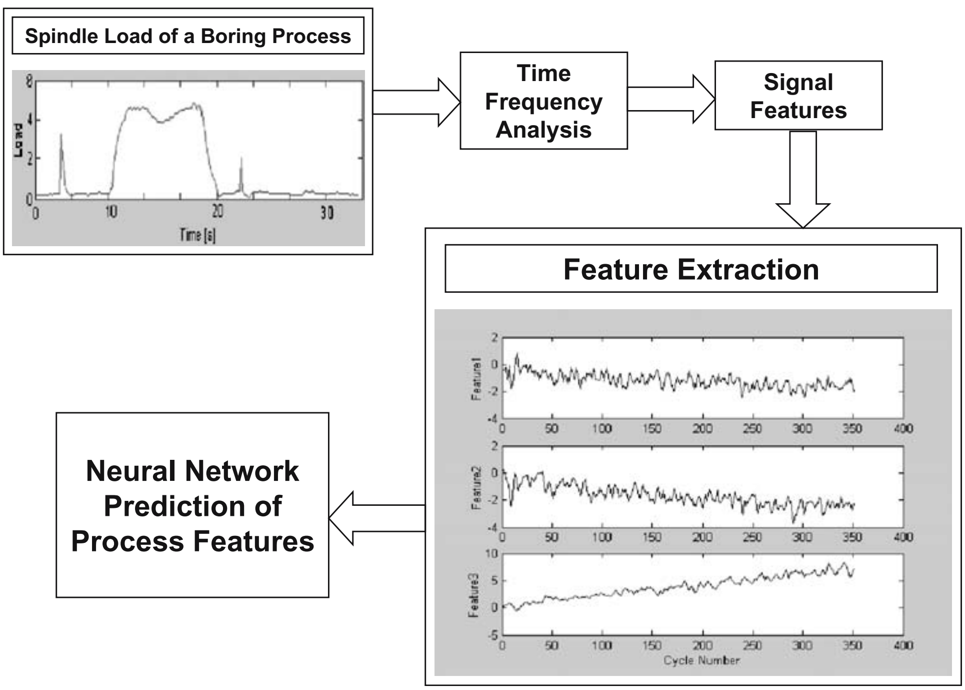
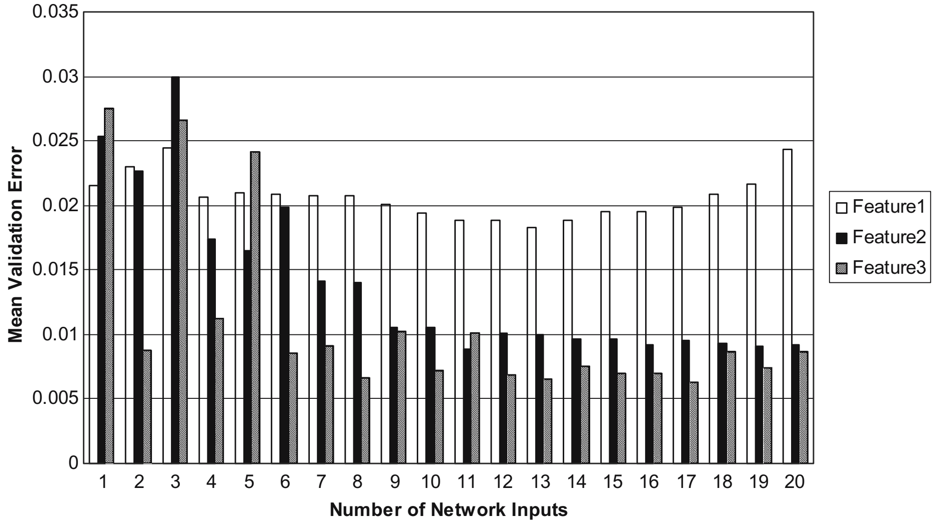


Fig. 3 Flowchart for feature extraction of spindle load of the boring process

Fig. 5 Mean validation errors of trained ERNN models for three features



neural network model can be simply viewed as a nonlinear regression model:

$$y_i = f(x_i; \theta^*) + \varepsilon_i \quad (i = 1, \ldots . n) \tag{3}$$

where x_i are the inputs, y_i are the outputs, θ* are the true values of the set of parameters θ, and the errors ε_i are assumed to be independently and identically distributed following a Gaussian distribution $N(0, \sigma^2)$ with mean zero and variance σ^2.

Therefore, standard asymptotic theory from nonlinear regression can be applied to derive prediction intervals for neural network models [13]. Let $\widehat{\theta}$ be the least squares estimate of θ*, obtained by minimizing the error function:

$$S(\theta) = \sum_{i=1}^{n} (y_i - f(x_i; \theta))^2 \tag{4}$$

for a training set (y_i, x_i), i=1,..., n. The predicted output of the neural network, for the input x_0, is $\widehat{y}_0 = f\left(x_0; \widehat{\theta}\right)$. Then, asymptotic linearization by applying the well known Taylor series expansion to the first order can now be used to approximate $f\left(x_0; \widehat{\theta}\right)$ in terms of $f(x_0; \theta^*)$:

$$f\left(x_0; \widehat{\theta}\right) \approx f(x_0; \theta^*) + f_0'\left(\widehat{\theta} - \theta^*\right) \tag{5}$$

Table 1 Summary of the ERNN models applied for the three features

Feature	Number of inputs	Number of hidden neurons	Training algorithm	Number of training epochs
1	13	7	Traincgb	21
2	11	6	Traincgb	17
3	17	9	Traincgb	36

where:

$$f_0' = \left(\frac{\partial f(x_0; \theta^*)}{\partial \theta_1^*}, \frac{\partial f(x_0; \theta^*)}{\partial \theta_2^*}, \ldots, \frac{\partial f(x_0; \theta^*)}{\partial \theta_p^*} \right) \tag{6}$$

Then, an approximate 100*(1−α) percent prediction interval for $\widehat{y}_0$ can be described by Seber and Wild [28] as:

$$\widehat{y}_0 \pm t_{\alpha/2, n-p} s \left[1 + f_0' \left(F' F \right)^{-1} f_0 \right]^{1/2} \tag{7}$$

where $t_{\alpha/2,n-p}$ is the t distribution with $n-p$ degrees of freedom, and:

$$F = \left[\left(\frac{\partial f\left(x_i; \widehat{\theta}\right)}{\partial \widehat{\theta}_j} \right)_{i=1:n} \right]_{j=1:p} \tag{8}$$

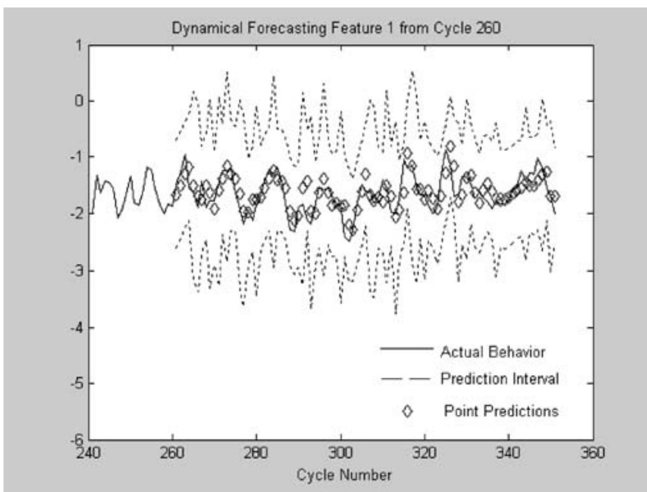


Fig. 6 Forecasting the behavior of feature 1 with 95% prediction intervals

F is the Jacobian matrix of neural network outputs with respect to its parameters. The matrix F has dimensions $n \times p$, where n is the number of samples used to obtain $\widehat{\theta}$, and p is the number of parameters θ_j which composes $\widehat{\theta}$. The unbiased estimator of σ^2 is:

$$s^2 = \frac{\sum_{i=1}^{n} \left(y_i - f\left(x_i; \widehat{\theta}\right)\right)^2}{n - p} \tag{9}$$

4 Building a neural network model for the feature behavior prediction of the boring process

4.1 Feature extraction of spindle load signal of the boring process

Due to the non-linearity of the boring process, the spindle load signal during the operation is highly non-stationary. Therefore, non-stationary signal analysis tools are needed for the prediction of signatures extracted from sensor readings obtained during the boring process. In this paper, joint time–frequency signal analysis is used to simultaneously decompose signal energy in the time and frequency directions [29]. Following Djurdjanovic et al. [1], the principal components of moments of the joint time–frequency signal energy distribution are used as features describing the spindle load readings. Signatures from 351 boring process cycles are obtained from the load signal time–frequency distribution, and their behavior is predicted using an ERNN prediction tool, as shown in Fig. 3.

4.2 Neural network model construction

Despite many sfatisfactory characteristics of neural networks, building a neural network model for a forecasting problem is not a trivial task. Modeling issues that affect the performance of a neural network must be considered carefully. One critical decision is to determine the neural network type and appropriate architecture, such as the number of input nodes (the number of past observations) and the number of hidden neurons. Other network design decisions include the selection of the training algorithm and the training data. Figure 4 shows a flowchart of the implementation for building a neural network model for forecasting. The process starts with the extraction of features associated with system performance. Subsequently, one needs to select data for network training and validation, as well as network parameters. The selection of network parameters is achieved by repeating the training of the neural network with different sets of parameters and keeping the neural network architecture with minimum validation errors. Then, it is possible to accomplish a prediction based on the constructed neural network model, and to construct prediction intervals to address prediction uncertainties.

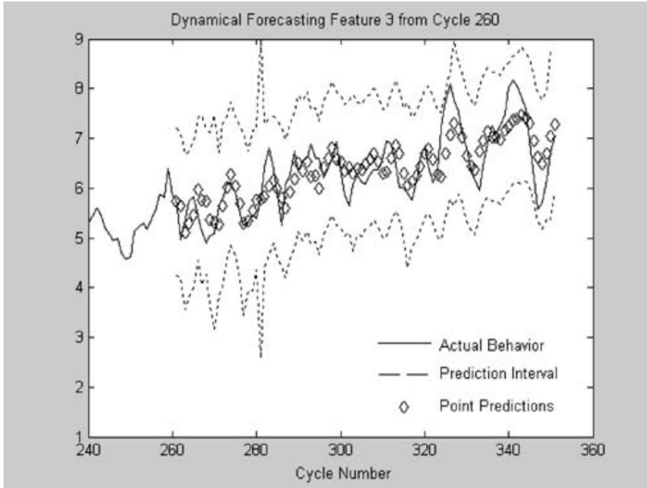


Fig. 8 Forecasting the behavior of feature 3 with 95% prediction intervals

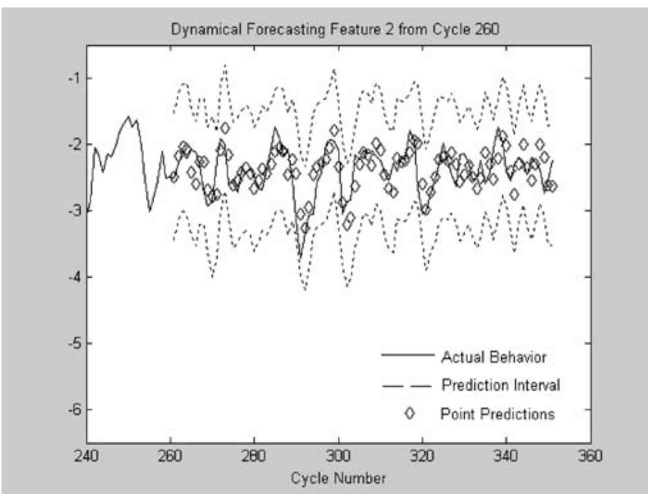


Fig. 7 Forecasting the behavior of feature 2 with 95% prediction intervals

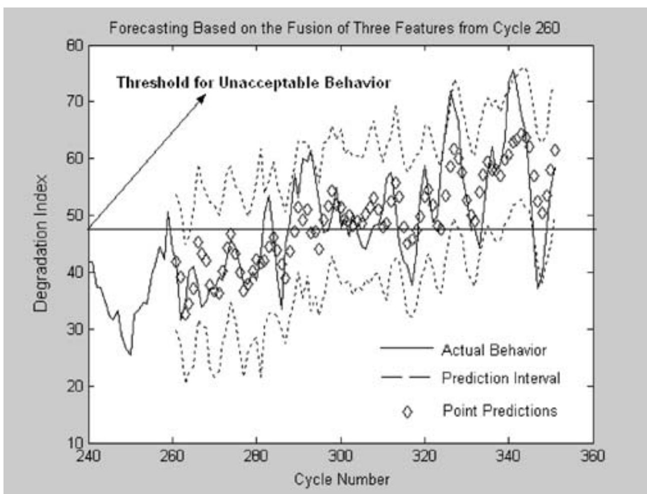


Fig. 9 Prediction based on the fusion of three features starting from cycle 260

In this research, samples of the first 200 cycles from each of three features of the boring process are used to construct a separate ERNN model for each feature. An early stopping approach for network training is employed in order to prevent network overfitting. By using 180 samples for training and 20 samples for validation, a series of ERNNs for each feature are trained using different training algorithms with a different number of network inputs (the number of past observations) ranging from 1 to 20 and a different number of hidden neurons. Each network configuration is trained 100 times by considering different random initializations of the network. The number of network inputs, the number of hidden neurons, and the type of

Table 2 Prediction results (1)

Cycle number	Actual DI	Predicted DI	95% confidence bounds	
			Upper limit	Lower limit
261	42.0863	41.7309	53.7539	29.7079
262	31.5007	39.2274	51.1566	27.2982
263	34.4041	32.6044	44.8402	20.3687
264	40.187	34.3846	46.3908	22.3785
265	40.9384	37.1955	51.0299	23.3611
266	37.4946	45.2371	58.9773	31.4969
267	33.8383	43.2372	55.9561	30.5183
268	34.8946	42.0145	53.94	30.0889
269	37.1822	37.6474	52.3599	22.9348
270	37.1155	36.6612	51.7681	21.5544
271	39.163	36.3179	50.0977	22.538
272	38.6096	40.1328	52.7276	27.5379
273	42.7581	44.2991	58.0634	30.5348
274	45.6621	46.673	58.6073	34.7388
275	44.3999	43.3444	55.0592	31.6296
276	41.1114	40.0143	54.1323	25.8963
277	38.1466	36.6573	51.7743	21.5404
278	38.0088	37.9792	49.7141	26.2442
279	40.0028	40.0885	52.4745	27.7024
280	38.5904	42.166	55.8885	28.4434
281	43.3461	41.633	61.6194	21.6466
282	50.332	42.0928	53.8846	30.3011
283	53.507	44.5142	56.1063	32.9221
284	46.4232	46.1164	59.4981	32.7346
285	39.2422	43.7818	55.3942	32.1695
286	33.3432	41.3186	52.948	29.6891
287	44.8267	39.0504	50.6571	27.4436
288	49.0183	43.5738	55.174	31.9737
289	57.4023	47.0696	58.696	35.4433
290	53.0911	51.5201	63.0503	39.99
291	60.1373	49.2564	62.975	35.5377
292	59.3805	51.0205	62.6327	39.4083
293	61.4471	46.8477	60.4759	33.2194
294	56.3283	47.177	58.6684	35.6855
295	52.6013	44.0912	55.7346	32.4477
296	46.9969	49.1898	62.736	35.6436
297	47.3913	51.6608	63.1736	40.148
298	49.725	54.3799	65.8452	42.9147
299	54.9923	51.7577	63.2754	40.24
300	47.8511	51.5355	65.0593	38.0117
301	49.6557	49.3972	60.8065	37.9878
302	46.2872	50.0967	61.6948	38.4986
303	50.4502	48.2663	60.4295	36.1032
304	47.0045	49.0157	60.4972	37.5342
305	44.494	48.5252	60.1295	36.9209
306	44.0939	50.188	61.6195	38.7564

Table 3 Prediction results (2)

Cycle number	Actual DI	Predicted DI	95% confidence bounds	
			Upper limit	Lower limit
307	46.4574	51.4029	64.8624	37.9434
308	48.0932	53.0564	66.5296	39.5832
309	48.3422	51.1145	62.4584	39.7706
310	50.998	48.0484	59.3465	36.7503
311	56.5719	48.5814	61.9908	35.1721
312	57.5732	52.5484	63.8721	41.2248
313	51.2115	55.771	69.1947	42.3473
314	45.15	53.2618	64.551	41.9725
315	41.9297	47.8806	59.2059	36.5552
316	41.2642	45.0188	57.5422	32.4953
317	37.6464	45.7067	59.1134	32.3001
318	43.6825	47.349	60.5932	34.1049
319	53.5345	49.8498	61.0934	38.6061
320	58.8295	53.2709	65.7642	40.7777
321	54.0596	54.4583	65.7093	43.2072
322	48.8836	51.7598	62.9993	40.5203
323	50.1117	48.1718	59.7312	36.6124
324	59.1134	47.6017	58.893	36.3103
325	65.248	53.4181	64.633	42.2032
326	71.9634	58.5659	69.8538	47.278
327	69.2204	61.6574	73.9806	49.3341
328	66.545	60.0542	72.6015	47.5069
329	57.2256	57.5272	68.7996	46.2549
330	53.0145	52.5902	65.135	40.0454
331	47.3412	50.075	61.3216	38.8284
332	47.7338	48.9754	60.4085	37.5422
333	44.263	54.0793	65.3307	42.8279
334	49.5054	57.1261	68.4632	45.7889
335	57.6773	59.5197	70.7716	48.2678
336	62.1344	58.0196	69.2695	46.7696
337	57.8437	57.8975	70.2175	45.5774
338	59.0143	56.9231	68.134	45.7122
339	63.2263	59.783	71.0408	48.5251
340	73.2911	60.6197	71.8235	49.4159
341	75.6948	62.964	74.1598	51.7683
342	72.676	63.3706	74.5837	52.1575
343	67.7013	64.3349	75.7608	52.9089
344	64.5438	63.6456	75.9467	51.3446
345	55.971	62.0984	73.2698	50.9269
346	44.9633	57.0759	68.236	45.9158
347	37.2216	52.4884	63.7023	41.2744
348	39.6497	50.462	62.7712	38.1528
349	47.9792	53.4242	64.649	42.1994
350	54.5708	58.0131	71.4747	44.5514
351	58.7098	61.4125	72.7827	50.0423

training algorithm for all the three features are selected based on the mean network validation error, as shown in Fig. 5.

Table 1 shows a summary of the network models found for the three features extracted from the time–frequency distribution of the spindle load signal observed during the boring process.

5 Results and discussion

For feature 1, we have constructed an ERNN model with 13 input neurons and 7 hidden neurons. The inputs are 13 past observations X_{t-1}, X_{t-2},..., X_{t-13} of feature 1. The objective is to predict the next realization X_t of feature 1. Figure 6 shows the predicted feature realizations from cycle 260 to cycle 351, based on the constructed neural network model, along with the 95% prediction intervals.

Similar analysis was carried out for features 2 and 3. Figures 7 and 8 show the predicted feature realizations, actual feature realizations, and the 95% prediction intervals for features 2 and 3, respectively.

After the behavior of each of the three features has been predicted separately, they are combined to predict the degradation behavior of the boring process, which is represented by the degradation index (DI), as shown in Fig. 9. The sum of squares of the predicted features are used to depict the degradation level because the three principal components of the time–frequency moments are asymptotically Gaussian, meaning that the sum of squares of those components will asymptotically follow a Chi-square distribution with 3 degrees of freedom and that the statistical significance of the drift of the DI could easily be assessed. Figure 9 shows the predictions of DI based on the fusion of the three features. The sum of squares of the actual realization of each feature is the solid line, the sum of squares of the predictions from each feature is shown as diamonds, and the corresponding prediction intervals are represented as dashed lines.

Figure 9 also shows that the boring process crosses the threshold of unacceptable behavior after 288 cycles. This threshold is set in the fused principal component's domain based on the consultations with the automotive manufacturer who uses this boring process in their powertrain manufacturing. The degradation behavior of the boring process near the limit of unacceptable behavior is modeled and predicted using the ERNN in order to initiate appropriate actions to prevent boring tool failure. In addition, the quantification of the variability associated with predictions by prediction intervals has provided additional information on the confidence in making the decisions for remedial actions.

Tables 2 and 3 summarize the numerical prediction results and the 95% confidence bounds.

6 Conclusions

In this paper, we present a study of using an Elman recurrent neural network (ERNN) in modeling and predicting the behavior of a boring process for better maintenance decision making. The results of this study show that an ERNN is a viable alternative for feature signature prediction. The use of prediction intervals are demonstrated by constructing the intervals via nonlinear regression techniques. This forecasting capability provides additional predictive information crucial for subsequent intelligent maintenance decision making so that the early detection of process degradation is made possible and timely preventive maintenance actions can be initiated.

Acknowledgement This research was supported by the National Science Foundation Industry/University Cooperative Research Center (NSF I/UCRC) for Intelligent Maintenance Systems (IMS) under grant no. 0117518.

References

1. Djurdjanovic D, Ni J, Lee J (2002) Time–frequency based sensor fusion in the assessment and monitoring of machine performance degradation. ASME DSC Division 71:15–22
2. NSF I/UCRC Center for Intelligent Maintenance Systems (2002) Home page at: http://www.imscenter.net/
3. Lee J (1995) Machine performance monitoring and proactive maintenance in computer-integrated manufacturing: review and perspective. Int J Comput Integ Manuf 8:370–380
4. Lee J (1996) Measurement of machine performance degradation using a neural network model. Comput Ind 30:193–209
5. Engel SJ, Gilmartin BJ, Bongort K, Hess A (2000) Prognostics, the real issues involved with predicting life remaining. In: Proceedings of the IEEE Aerospace Conference, Big Sky, Montana, March 2000, vol 6, pp 457–469
6. Yang Z, Djurdjanovic D, Mayor R, Ni J, Lee J (2004) Maintenance scheduling in production systems based on predicted machine degradation. Automat Sci Eng (submitted, paper no V2004–067)
7. Zhang SF, Liu RJ (2000) A rapid algorithm for on-line and real-time ARMA modeling. In: Proceedings of the 5th International Conference on Signal Processing (WCCC-ICSP 2000), Beijing, China, August 2000, pp 230–233
8. Weigend AS, Huberman BA, Rumelhart DE (1990) Predicting the future: a connectionist approach. Int J Neural Syst 1:193–209
9. Rape R, Fefer D, Jeglic (1995) Comparison of neural networks to statistical techniques for prediction of time series generated by nonlinear dynamic systems. In: Proceedings of the Instrumentation and Measurement Technology Conference (IMTC/95), Waltham, Massachusetts, April 1995, pp 300–304
10. Lowe D, Zapart C (1999) Point-wise confidence interval estimation by neural networks: a comparative study based on automotive engine calibration. Neural Comput Appl 8:77–85
11. De Veaux RD, Schumi J, Schweinsberg J, Unger LH (1998) Prediction intervals for neural networks via nonlinear regression. Technometrics 40:273–282
12. Efron B, Tibshirani RJ (1993) An introduction to the bootstrap. Chapman and Hall, New York
13. Huang JTG, Ding AA (1997) Prediction intervals for artificial neural networks. J Am Stat Assoc 92:748–757
14. Lapedes A, Farber R (1987) Nonlinear signal processing using neural networks: prediction and system modeling. Los Alamos National Laboratory Report, Los Alamos, NM, technical report no. LA-UR-87-2662
15. Lee KY, Choi TI, Ku CC, Park JH (1994) Neural network architectures for short-term load forecasting. In: Neural Networks, proceedings of the IEEE World Congress on Computational Intelligence, Orlando, Florida, June/July 1994, pp 4724–4729

16. Senjyu T, Takara H, Uezato K, Funabashi T (2002) One-hour-ahead load forecasting using neural network. IEEE T Power Syst 17:113–118
17. Wulff NH, Hertz JA (1992) Prediction with recurrent networks. In: Proceedings of the 1992 IEEE-SP Workshop on Neural Networks for Signal Processing, Copenhagen, Denmark, August/September 1992, pp 464–473
18. Khotanzad A, Abaye A, Maratukulam D (1994) An adaptive recurrent neural network system for multi-step-ahead hourly prediction of power system loads. In: Neural Networks, proceedings of the IEEE World Congress on Computational Intelligence, Orlando, Florida, June/July 1994, pp 3393–3397
19. Logar AM, Corwin EM, Oldham WJB (1993) A comparison of recurrent neural network learning algorithms. In: Proceedings of the IEEE International Conference on Neural Networks, San Francisco, California, March 1993, pp 1129–1134
20. Rao SS, Sethuraman S, Ramamurti V (1992) A recurrent neural network for nonlinear time series prediction–a comparative study. In: Proceedings of the 1992 IEEE-SP Workshop on Neural Networks for Signal Processing, Copenhagen, Denmark, August/September 1992, pp 531–539
21. Chen CH, Yu L (1997) A learning algorithm for improved recurrent neural networks. In: Proceedings of the International Conference on Neural Networks (ICNN'97), Houston, Texas, June 1997, pp 2198–2202
22. Connor J, Atlas L (1991) Recurrent neural networks and time series prediction. In: Proceedings of the Seattle International Joint Conference on Neural Networks (IJCNN'91–Seattle), Seattle, Washington, July 1991, pp 301–306
23. Jenq-Neng Hwang, Little E (1996) Real time recurrent neural networks for time series prediction and confidence estimation. In: Proceedings of the IEEE International Conference on Neural Networks (ICNN'96), Washington, DC, June 1996, pp 1889–1894
24. Barton RS, Himmelblau DM (1997) Online prediction of polymer product quality in an industrial reactor using recurrent neural networks. In: Proceedings of the International Conference on Neural Networks (ICNN'97), Houston, Texas, June 1997, pp 111–114
25. Goh WY, Lim CP, Peh KK, Subari K (2000) A neural-network-based intelligent system for time-series prediction problems in product development. In: Proceedings of TENCON 2000, Kuala Lumpur, Malaysia, September 2000, pp 151–155
26. Jun Zhang, Tang KS, Man KF (1997) Recurrent NN model for chaotic time series prediction. In: Proceedings of the 23rd International Conference on Industrial Electronics, Control and Instrumentation (IECON'97), New Orleans, Louisiana, November 1997, pp 1108–1112
27. Zhang J, Man KF (1998) Time series prediction using RNN in multi-dimension embedding phase space. In: Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC), San Diego, California, October 1998, pp 1868–1873
28. Seber GAF, Wild CJ (1989) Nonlinear regression. Wiley, New York
29. Cohen L (1995) Time-frequency analysis. Prentice-Hall, Englewood Cliffs, New Jersey