actuators

MDPI

Article

Examining Attribution Reliability Through Cutting-Speed Classification as a Benchmark Task: Comparing 1D-CNN, LSTM, and Fully Connected Networks on 17-4PH Milling Force Signals

Shih-Feng Chen, Bo-Xuan Wu, Sin-Wei Chen, Chung-Chen Tsao * and Ting-Hsueh Chuang *

Department of Mechanical Engineering, Lunghwa University of Science and Technology, Taoyuan 33326, Taiwan; dave2esc@gm.lhu.edu.tw (S.-F.C.)

* Correspondence: aetcc@mail.lhu.edu.tw (C.-C.T.); tinghsueh@gm.lhu.edu.tw or tinghsueh2017@gmail.com (T.-H.C.)

Abstract

Deep learning has achieved high accuracy in machining condition monitoring, yet accuracy alone cannot reveal whether a model relies on physically meaningful evidence or on dataset shortcuts. The difficulty is that common tasks such as tool-wear and surface-roughness estimation lack a known ground-truth basis for discrimination, so attribution outputs cannot be validated. This study therefore constructs a benchmark task—three-class cutting-speed classification (Vc = 60, 80, 100 m/min)—in which, for a fixed tool diameter, the spindle-speed-dependent spectral content is an a priori known correct basis, enabling the reliability of explainable AI (XAI) methods to be examined under controlled conditions; the task is deliberately simple and serves as a validation framework for attribution methods rather than as a demanding monitoring problem. Cutting force signals were acquired from 3 × 3 full-factorial milling experiments on 17-4PH stainless steel using a Kistler 9170A rotating dynamometer; the torque Mz and resultant force Fa were segmented by sliding windows into 2970 single-channel samples (15 records × 99 windows × 2 channels), each represented by 50 time segments × 6 amplitude-independent statistics. A 1D-CNN, an LSTM, and a fully connected network (NN) were evaluated under five-fold StratifiedGroupKFold cross-validation. The 1D-CNN achieved 93.97% accuracy with roughly one-seventh of the parameters (NN 90.24%, LSTM 89.12%), the smallest fold-to-fold standard deviation, and 84.5% on the most difficult fold; as a single random seed was used, performance conclusions are descriptive only. Holding out whole cutting conditions reduced the mean accuracy to 83–86% (one feed level: 54–57%), so the accuracies describe generalization across records of the studied conditions, not to unseen conditions. Fold-wise application of LRP, LIME, and SHAP showed that the 1D-CNN and NN distributed importance across the entire signal (normalized entropy ≥ 0.97), whereas for the LSTM the gradient × input attribution used as the LRP implementation placed 9.4 times the uniform share on the final time segment, consistently across all five folds, a concentration that SHAP (expected gradients referenced to background training samples) and the perturbation-based LIME did not reproduce. Gradient-free occlusion and resampling reproduced it, and retraining showed that it follows the last input position while removing the final segments did not reduce accuracy; it therefore reflects positional sensitivity rather than reliance on the sequence end. In this run, cross-fold Spearman correlations of LIME rankings lay within −0.05 to 0.05, versus 0.20–0.70 for LRP and SHAP; because exploratory recalculations with alternative aggregation choices gave markedly different LIME correlations, this is reported as a run-specific observation rather than as a property of LIME. All three architectures relied predominantly on frequency-domain features; because the tooth-passing frequencies (159.2–265.3 Hz) exceed the 100 Hz Nyquist frequency of the 200 Hz sampling rate yet lie

Check for updates

Academic Editor: Zhuming Bi

Received: 3 September 2026
Revised: 8 October 2026
Accepted: 12 October 2026
Published: 15 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

below the 300 Hz low-pass cutoff of the signal conditioner, they are aliased into the analyzed band, and the dominant-frequency feature reflects the spindle rotation frequency and aliased tooth-passing components, both proportional to spindle speed and hence consistent with the physical definition of cutting speed. The contributions are: a controlled validation framework that cross-validates attribution credibility through importance concentration, inter-method agreement, and cross-fold reproducibility; and the observation that the end-of-sequence concentration of the LSTM, revealed by gradient × input but not by SHAP or LIME, reflects positional sensitivity rather than the basis of classification, highlighting the need to verify attribution concentration with independent analyses rather than relying on a single attribution method.

Keywords: explainable AI (XAI); cutting force signal; positional sensitivity; attribution reproducibility; layer-wise relevance propagation (LRP); 1D-CNN; 17-4PH milling

1. Introduction

This study applies explainable AI (XAI) to dynamometer force signals acquired during milling of 17-4 PH stainless steel, and examines the reliability of attribution methods through a benchmark task whose correct basis for discrimination is known a priori. The work spans three areas: the interpretability of deep learning models, the characteristics of the workpiece material, and cutting force measurement. To establish the research foundation and identify existing gaps, this section reviews the relevant literature in sequence. Section 1.1 surveys the principal XAI methods and their application to sequential data [1–6], outlining the development of post-hoc and intrinsically interpretable approaches together with their limitations, including disagreement among explainers and the lack of means to validate attribution outputs. Section 1.2 examines the heat-treatment mechanisms, additively manufactured microstructures, and corrosion behavior of 17-4 PH precipitation-hardening stainless steel [7–11], clarifying its properties and machining difficulty as an aerospace precision-component material. Section 1.3 reviews advances in cutting force sensor design, experimental modeling, and online monitoring systems [12–16], establishing the value of cutting force signals as indicators of process state. Finally, Section 1.4 synthesizes the three strands, showing that existing studies lack a benchmark task for verifying attribution credibility and have not examined whether models rely on dataset shortcuts rather than physical features, and positions the objectives and contributions of this study accordingly.

1.1. Explainable Artificial Intelligence (XAI)

As opaque models such as deep neural networks are increasingly deployed for decision-making in critical domains, the need for interpretability of model behavior has become widely accepted. Aysel et al. [1] provided a structured survey of XAI methods, covering explanation techniques at different levels of abstraction and their limitations. Bennetot et al. [2] presented a practical tutorial applying post-hoc techniques such as SHAP, LIME, and Grad-CAM to common models and datasets, with readily adaptable Python 3.10 examples, as tools for validating and certifying model outputs and conferring trustworthiness and transparency. For sequential data, Bento et al. [3] proposed TimeSHAP, which extends KernelSHAP to recurrent models, computes feature-, timestep-, and cell-level attributions, and employs a pruning scheme to reduce the computational cost and attribution variance for long sequences. Borujeni et al. [4] extended layer-wise relevance propagation (LRP) to a sequence-to-sequence model with GRU layers for pollution forecasting, showing that the key meteorological and temporal features identified by the relevance heatmaps

were consistent with domain knowledge. In contrast to post-hoc explanation, Yang et al. [5] advocated architectural constraints—sparse additive subnetworks, orthogonal projection pursuit, and smooth function approximation—to construct an intrinsically explainable neural network (ExNN) that balances predictive accuracy and interpretability. XAI has also been extended to continual learning: Proietti et al. [6] reviewed methods that use explanation techniques to mitigate catastrophic forgetting, organized them into a taxonomy according to what is explained (data, features, or network components), and outlined future research directions. Notably, most of these studies report results from a single explainer or provide only qualitative corroboration against domain knowledge; when different explainers yield divergent attributions for the same model, which one should be trusted remains undeterminable in tasks lacking a known correct basis [1,2].

1.2. 17-4 PH Precipitation-Hardening Stainless Steel

17-4 PH stainless steel is widely used in aerospace, ordnance, and precision components because it combines high strength with corrosion resistance. Its properties derive from a martensitic matrix strengthened by Cu-rich precipitates and can be tailored by heat treatment [7,8]; the processing route, including additive manufacturing [9,10], and surface passivation [11] further affect its microstructure and corrosion behavior. This high-strength microstructure makes 17-4 PH difficult to machine and renders its cutting force signals sensitive to changes in cutting parameters, making it a suitable subject for identifying process states from force signals.

1.3. Cutting Force Measurement and Monitoring

Cutting force is a key physical quantity reflecting tool wear, machining quality, and process stability. Daniyan et al. [12] combined a Kistler 9257A dynamometer with response surface methodology to establish a predictive model relating cutting force in Ti6Al4V milling to cutting speed, feed per tooth, and depth of cut. To overcome the cost and mounting constraints of commercial dynamometers, several studies have pursued in-house sensor development: Liang et al. [13] designed a six-component sensing system with a monolithic elastic element that simultaneously measures three-axis cutting forces and moments, verified for linearity and low coupling error on a five-axis parallel kinematic machining center; Subasi et al. [14] developed a triaxial optoelectronic dynamometer using photo-interrupters with a monolithic flexure structure, validated through static and dynamic characterization and milling tests; and Li et al. [15] proposed a spindle-integrated three-axis cutting force measurement system for ultra-precision diamond milling. On the signal-processing side, Zhao [16] constructed an online cutting force monitoring system for CNC lathes, acquiring signals with piezoelectric sensors and converting them via a data acquisition card into digital form for real-time monitoring and subsequent analysis. These studies indicate that, beyond reflecting process state, the periodic structure of cutting force signals is governed by the spindle rotation frequency and the tooth-passing frequency, both determined by spindle speed (the latter also by the number of cutting edges); for a fixed tool diameter, a definite physical correspondence therefore exists between cutting speed and these spindle-speed-dependent spectral components. This property enables cutting-speed classification to serve as a benchmark task for verifying the reliability of attribution methods.

1.4. Research Gaps and Positioning of This Study

Taken together, the literature reveals clear disconnects among the three strands. First, cutting force research has concentrated on sensor design or statistical modeling [12–16]; few studies have applied deep learning directly to time-series force signals for process-state identification, and none have examined whether such models rely on physical features

or on dataset shortcuts. Second, applications of XAI to sequential data have focused on finance and environmental forecasting [3,4], typically reporting attributions from a single explainer with only qualitative corroboration against domain knowledge; yet different post-hoc explainers frequently produce systematically divergent attributions for the same model [1,2], and in tasks such as tool-wear or surface-roughness estimation—where no correct basis for discrimination is known—such divergence cannot be adjudicated, nor can the reliability of the attribution methods themselves be examined. Third, research on 17-4 PH has largely addressed microstructure and properties [7–11], leaving data-driven analysis of its cutting force signals unexplored. Accordingly, this study exploits the definite physical correspondence between cutting speed and the spindle-speed-dependent spectral components described in Section 1.3 to construct a three-class cutting-speed benchmark task from Kistler 9170A rotating-dynamometer force signals acquired during 17-4 PH milling, so that the correct basis for discrimination is known a priori. On this foundation, 1D-CNN, LSTM, and fully connected architectures are built, and LRP, LIME, and SHAP are applied fold by fold; attribution credibility is cross-validated through three indicators—importance concentration, inter-method agreement, and cross-fold reproducibility—to examine whether the models rely on physical features rather than shortcuts, and whether each attribution method can detect such shortcuts. The study thereby provides a controlled validation framework for attribution methods—one in which the correct basis for discrimination is known a priori—rather than general evidence that XAI methods are reliable in complex monitoring tasks such as tool-wear or surface-roughness prediction.

2. System Architecture

2.1. Workpiece Material and Cutting Force Measurement Platform

The workpiece material, 17-4PH (AISI 630/UNS S17400), is a martensitic precipitation-hardening stainless steel (approximately 17 wt% Cr, 4 wt% Ni, and 3–4 wt% Cu) strengthened mainly by nanoscale Cu-rich precipitates formed during aging, and is the most widely used grade of its family in aerospace structural components. Such high-strength, difficult-to-machine materials generate relatively large cutting forces during milling, providing force signals rich enough to discriminate among machining conditions.

Cutting force signals were acquired from milling experiments on 17-4PH (H1025) specimens. The experiments were performed on a Tongtai TMV-850QII (Tongtai Machine & Tool Co., Ltd., Kaohsiung, Taiwan) vertical machining center using a Kistler 9170A rotating piezoelectric cutting force dynamometer (RCD) (Kistler Instrumente AG, Winterthur, Switzerland) with a 5236B stator. The sensor is mounted between the spindle and the tool holder and rotates synchronously with the tool, acquiring four components in situ near the cutting edge via non-contact inductive telemetry: the feed-direction force Fx, radial force Fy, axial thrust Fz, and cutting torque Mz; the resultant in-plane cutting force F_a was computed by DynoWare 3.3.2 from the two horizontal force components as

$$F_a = \sqrt{{F_x}^2 + {F_y}^2} \quad (1)$$

and was verified against the exported F_x, F_y, and F_a channels. The cutting tool was a Tungaloy AH715 series ∅8 mm modular (exchangeable-head) end mill with a PVD nano-multilayer coating, and the specimen dimensions were 80 × 80 × 8 mm^3; details of the tool and specimen are given in the following subsections. The cutting parameters followed a 3 × 3 full-factorial design in which the cutting speed Vc (60, 80, 100 m/min) and feed per tooth fz (0.02, 0.035, 0.05 mm/tooth) were varied crosswise, while the axial depth of cut ap = 1.0 mm and radial width of cut ae = 0.05 mm were held constant, yielding cutting

force signals under nine conditions of equal material removal volume as the data source for the classification task. The rotating cutting force dynamometer used is shown in Figure 1.



Figure 1. Kistler Type 9170A rotating cutting force dynamometer.

The workpiece material was 17-4PH stainless steel in the H1025 precipitation-hardened condition, with dimensions of 80 × 80 × 8 mm^3. The specimen was clamped in a machine vise fixed to the worktable; the clamping arrangement and the machined surface appearance are shown in Figure 2.

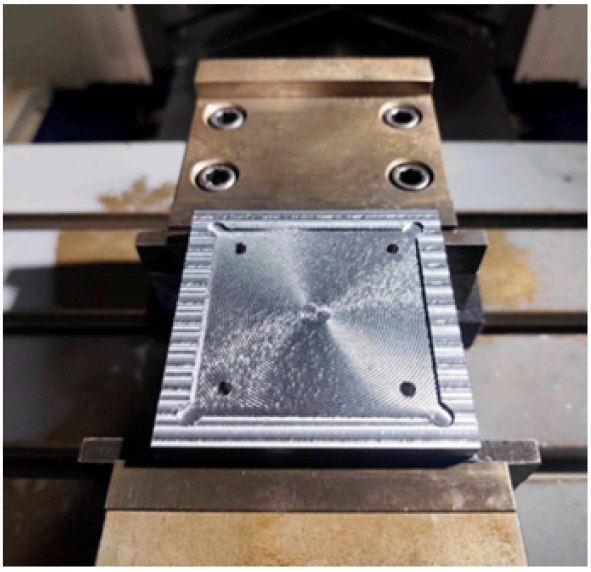


Figure 2. Clamping of the 17-4PH (H1025) specimen and surface appearance after machining.

The cutting tool was a Tungaloy AH715 series modular (two-piece) end mill, in which the holder and the milling head are separate components joined by a threaded connection. Only the milling head needs to be replaced to restore the cutting-edge condition, which substantially reduces tool material consumption and improves tool-change efficiency in long-duration cutting experiments. A ∅8 mm milling head was used in this study; its detailed dimensions are shown in Figure 3, and the main specifications are listed in Table 1.

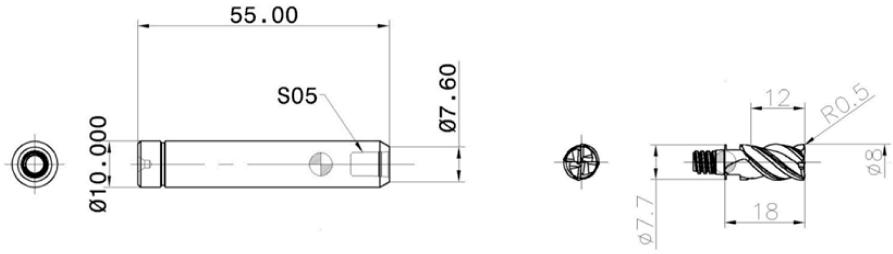


Figure 3. Detailed dimensions of the ∅8 mm milling head and holder.

Table 1. Tool specifications.

Item	Specification
Tool system	Tungaloy AH715 series, modular (two-piece) exchangeable-head end mill
Coating	PVD nano-multilayer coating
Holder overall length	55.00 mm
Holder shank diameter	∅10.000 mm
Holder front-end diameter	∅7.60 mm
Holder designation	S05
Head diameter D	∅8 mm
Head overall length	18 mm
Effective flute length	12 mm
Neck diameter	∅7.7 mm
Corner radius	R0.5 mm
Number of flutes z	4

This tool configuration has two implications for the experimental design of this study. The first concerns the control of confounding factors: the modular design allows a new milling head to be fitted for each of the nine experiments while the same holder is retained, so that the tool condition remains consistent and tool wear does not accumulate over the course of the experiments and become confounded with cutting speed and feed per tooth. This is particularly important here, since the study rests on the premise that the differences among the three classes arise solely from cutting speed; if tool wear varied monotonically with time, the model could discriminate on wear-related rather than speed-related features, constituting a typical case of shortcut learning. The second concerns the cutting geometry: the axial depth of cut ap = 1.0 mm is twice the corner radius R0.5 mm, so the cutting action involves both the corner arc and the straight cutting edge; only about 8% of the 12 mm effective flute length participates in cutting, and wear is concentrated in a narrow band near the tool tip.

The head diameter ∅8 mm is also the value of D used in the subsequent frequency calculations. For a tool with z flutes, the cutting speed V_c (m/min), spindle speed N (rpm), spindle rotation frequency f_s, and tooth-passing frequency f_t are related by

$$V_c = \frac{\pi D N}{1000} \tag{2}$$

$$f_s = \frac{N}{60} \tag{3}$$

$$f_t = z f_s \tag{4}$$

The complete measurement chain comprises the rotating sensor, measuring stator, signal conditioner, and data acquisition hardware connected in series; the composition and function of each subsystem are given in Table 2. Dynamometer signals were acquired using Kistler DynoWare software to configure the amplifier and measurement parameters; the complete settings are listed in Table 3. With z = 4 flutes, the tooth-passing frequencies

corresponding to the three cutting speeds are 159.2, 212.2, and 265.3 Hz. These tooth-passing frequencies must be distinguished from the spindle rotation frequencies f_s (39.8, 53.1, and 66.3 Hz), which are four times lower; the consequences of the 200 Hz sampling rate for both are discussed after.

Table 2. Subsystems of the cutting force measurement system.

Subsystem	Model/Specification	Composition and Function
Rotating sensor	Kistler Type 9170A	Integrates quartz piezoelectric sensing elements, charge amplifier circuitry, a wireless telemetry transmitter, and a standard tool clamping interface; mounted between the spindle and tool holder and rotates with the tool
Measuring stator	Kistler Type 5236B	Fixed to the spindle head housing coaxially with the sensor; contains the telemetry receiving antenna, demodulation circuitry, inductive power-supply coil, and pneumatic cooling channel
Signal conditioner	Kistler Type 5238B	Performs low-pass filtering, gain amplification, and per-channel sensitivity calibration, outputting standardized voltage signals
Data acquisition hardware (DAQ)	16-bit multi-channel A/D module (up to 10 kHz)	Converts analog voltages to digital signals; provides a TTL synchronization trigger for integrated acquisition with vibration acceleration signals. The sampling rate actually used is given in Table 3

Table 3. Measurement parameter settings.

Item	Setting
Rotor model (RCD)	Type 9170A
Signal conditioner	Type 5238B
Low-pass filter	300 Hz (output filter setting of the signal conditioner; above the 100 Hz Nyquist frequency, so it does not act as an anti-aliasing filter—see Section 2.1)
Measuring time	1500 s per record (1200 s for run L9), from the DynoWare file header
Sample rate	200 Hz (confirmed from the DynoWare file header; sampling interval 0.005 s)
Number of cycles	1 (Continuous not selected)
Trigger delay time	0 s
Measuring range per axis	Fx, Fy, Fz, Mz set individually in software according to the sensor's axial sensitivity

In terms of the signal path, the cutting force is converted into a charge signal by the rotating sensor immediately adjacent to the cutting edge and amplified on the spot, then transmitted to the stator by wireless telemetry, thereby avoiding the noise and phase delay associated with long-distance analog transmission; an inductive power supply and pneumatic cooling allow the sensor to operate continuously at high rotational speeds. The TTL synchronization trigger of the DAQ enables the cutting force signals to share a common time base with vibration acceleration signals, reserving an interface for subsequent multi-sensor fusion, although only the cutting force channels were used in this study.

The on-site configuration of this measurement chain is shown in Figure 4. The figure shows the machining setup inside the machine tool: the rotating sensor is mounted between the spindle and the tool holder and rotates with the tool, with the stator behind it fixed to the spindle head housing and its signals routed out by cable; the specimen is clamped in a machine vise fixture at the center of the worktable, and coolant nozzles on both sides are directed at the cutting edge.

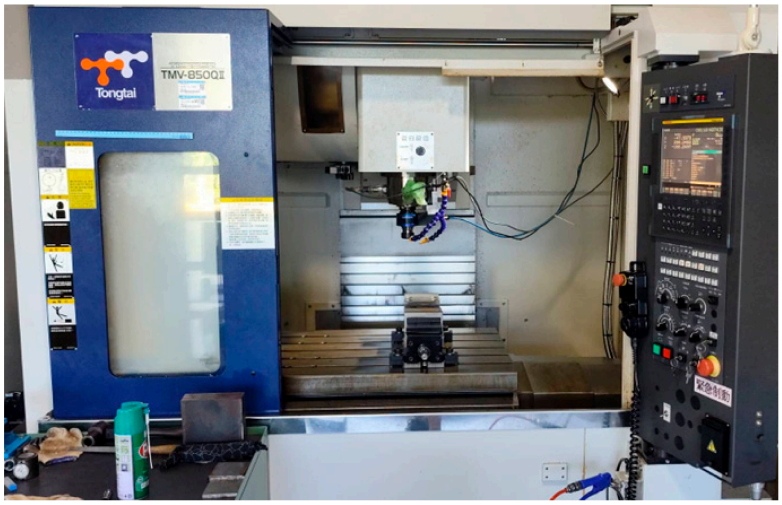


Figure 4. On-site configuration inside the machine tool for the milling experiments.

At a sampling rate of 200 Hz, each 5000-point window adopted in this study corresponds to a duration of 25.0 s; 99 windows are taken per record and channel at equally spaced start positions, so that the step between consecutive windows varies with record length (about 6.7–13.2 s; Section 2.4). Each window is further divided into 50 time segments of 100 points (0.5 s) each, for which the discrete Fourier transform yields a frequency resolution of 2 Hz. The spindle speeds and characteristic frequencies corresponding to the three cutting speeds at a tool diameter of D = 8 mm are listed in Table 4.

Table 4. Characteristic frequencies corresponding to the three cutting speeds (D = 8 mm, z = 4, sampling rate 200 Hz).

Cutting Speed Vc (m/min)	Spindle Speed (rpm)	Spindle Rotation Frequency fs (Hz)	Tooth-Passing Frequency ft = z·N/60 (Hz)	Apparent (Aliased) Frequency f_{app} at 200 Hz Sampling, Equation (5) (Hz)
60	2387	39.8	159.2	40.8
80	3183	53.1	212.2	12.2
100	3979	66.3	265.3	65.3

In terms of temporal resolution, each 0.5 s time segment spans 20 to 33 complete spindle revolutions, and the 2 Hz frequency resolution is finer than the 13.3 Hz spacing between the three spindle rotation frequencies (39.8, 53.1, and 66.3 Hz), all of which lie below the 100 Hz Nyquist frequency. The tooth-passing frequencies (159.2, 212.2, and 265.3 Hz), however, exceed the Nyquist frequency, and the 300 Hz low-pass filter of the signal conditioner does not act as an anti-aliasing filter at this sampling rate. The tooth-passing components are therefore aliased to the apparent frequencies

$$f_{\mathrm{app}} = \left| f_{\mathrm{t}} - f_{\mathrm{samp}} \right| \tag{5}$$

where $f_{\mathrm{samp}} = 200$ Hz is the sampling rate; this form applies because all three tooth-passing frequencies lie between 100 and 300 Hz. The resulting apparent frequencies are 40.8, 12.2, and 65.3 Hz, respectively (Table 4). High-resolution long-record spectra (65,536 points, Hann window) confirm that both the spindle rotation component and the aliased tooth-passing component are present, with a channel-dependent balance: in Mz the aliased tooth-passing component is the stronger of the two at all three cutting speeds (spindle-to-alias peak ratio 0.01–0.73), whereas in Fa the spindle rotation component is stronger at Vc = 80 m/min (ratio 1.18–2.10) and in three of the four Vc = 100 m/min records (ratio

1.31–3.71), while the aliased component remains stronger at Vc = 60 m/min (ratio 0.02–0.97). When the dominant frequency is computed over each 0.5 s segment (mean removed, 0 Hz bin excluded, 2 Hz bins), it falls predominantly in the 40 Hz bin for Vc = 60 (67–72% of segments), the 12 Hz bin for Vc = 80 (78–86%), and the 66 Hz bin for Vc = 100 (55–74%). For Vc = 60 and 100, the spindle and aliased components (39.8/40.8 Hz and 66.3/65.3 Hz) fall into the same 2 Hz bin and cannot be separated at the segment level; for Vc = 80, the segment-level dominant frequency is the aliased tooth-passing component (12.2 Hz) in both channels. The four quantities are thus linked as follows: the spindle speed N sets the spindle rotation frequency f_s and the tooth-passing frequency f_t through Equations (3) and (4); the 200 Hz sampling rate retains f_s but folds f_t to f_{app} according to Equation (5); and the dominant-frequency feature selects whichever of these components is strongest in a given 0.5 s segment. Because every candidate is a fixed function of N and the candidates remain well separated across the three classes, the one-to-one correspondence between dominant frequency and cutting speed on which the benchmark relies is preserved, even though the physical identity of the feature varies with channel and cutting speed. It also means that cutting speed is encoded by a strong, low-dimensional spectral cue, which is why the task serves as a controlled test of attribution methods rather than as a demanding monitoring problem (Section 4.4). The waveform at the tooth-passing scale is not resolved; a sampling rate of at least 2 kHz with an anti-aliasing filter below the Nyquist frequency is recommended for future experiments.

Each of the nine conditions was machined in ten passes. During each pass, DynoWare recorded one to three consecutive records of 1500 s each (1200 s for L9) at 200 Hz. The records of the fifth pass of each condition were used, yielding 15 records in total—6 for Vc = 60 (L1: 3, L2: 2, L3: 1), 5 for Vc = 80 (L4: 2, L5: 2, L6: 1), and 4 for Vc = 100 (L7: 2, L8: 1, L9: 1)—which accounts for the unequal class sizes. Each record was truncated after its last non-zero sample (the end of cutting), giving effective durations of 735–1359 s before the removal of non-cutting intervals described in Section 2.4 (Table 5).

Table 5. Dynamometer records used in this study (fifth pass of each condition).

Run	Cutting Speed Vc (m/min)	Feed per Tooth fz (mm/tooth)	Records (DynoWare File ID)	Effective Duration per Record (s)
L1	60	0.02	test_251/252/253	1103.9/1100.2/1109.5
L2	60	0.035	test_226/227	959.0/971.3
L3	60	0.05	test_211	1358.8
L4	80	0.02	test_195/196	1235.5/1291.2
L5	80	0.035	test_175/176	735.3/740.7
L6	80	0.05	test_160	1048.1
L7	100	0.02	test_144/145	1001.1/1019.4
L8	100	0.035	test_130	1195.6
L9	100	0.05	test_118	872.7

With respect to the measurement system, Kistler piezoelectric dynamometers are the widely adopted reference standard for cutting force measurement in CNC machining, combining high frequency response, high sensitivity, and high rigidity; they can detect small fluctuations in cutting force and respond rapidly to subtle changes in tool geometry and coating, and their high natural frequency makes them applicable to force detection in both low- and high-speed machining [12]. Owing to this benchmark status, Kistler dynamometers are frequently used as the reference for validating new cutting force sensors: an optoelectronic triaxial dynamometer was validated against a Kistler 9256C1 [14], and a spindle-integrated piezoelectric measurement mechanism was substantiated by its good correlation with a commercial standard dynamometer [15]; in online monitoring applications, Kistler piezoelectric sensors convert cutting force into voltage signals that

are digitized by a data acquisition board for real-time monitoring [16]. It should be noted, however, that piezoelectric sensors have inherent limitations in static force measurement, and the structural dynamics of the sensor housing may distort force measurements at high rotational speeds [13]; the spindle-integrated rotating dynamometer adopted in this study is designed precisely to mitigate such transmission-path distortion and to obtain high-fidelity cutting force signals under high-speed milling.

2.2. Milling Experiment Results: Cutting Force and Surface Roughness

Before proceeding to the classification task, the mean resultant cutting force Fa and mean surface roughness Ra obtained under the nine cutting conditions were first examined to characterize the distribution of the data. The values for the nine conditions are listed in Table 6, and the trends are shown in Figures 5 and 6, respectively.

Table 6. Mean Fa and Ra for the nine experimental conditions.

Run	Cutting Speed Vc (m/min)	Feed per Tooth fz (mm/tooth)	Mean Fa (N)	Mean Ra (μm)
L1	60	0.02	17.80	0.11
L2	60	0.035	18.22	0.11
L3	60	0.05	24.34	0.09
L4	80	0.02	18.27	0.09
L5	80	0.035	18.98	0.09
L6	80	0.05	18.25	0.08
L7	100	0.02	16.52	0.06
L8	100	0.035	17.31	0.11
L9	100	0.05	15.42	0.09

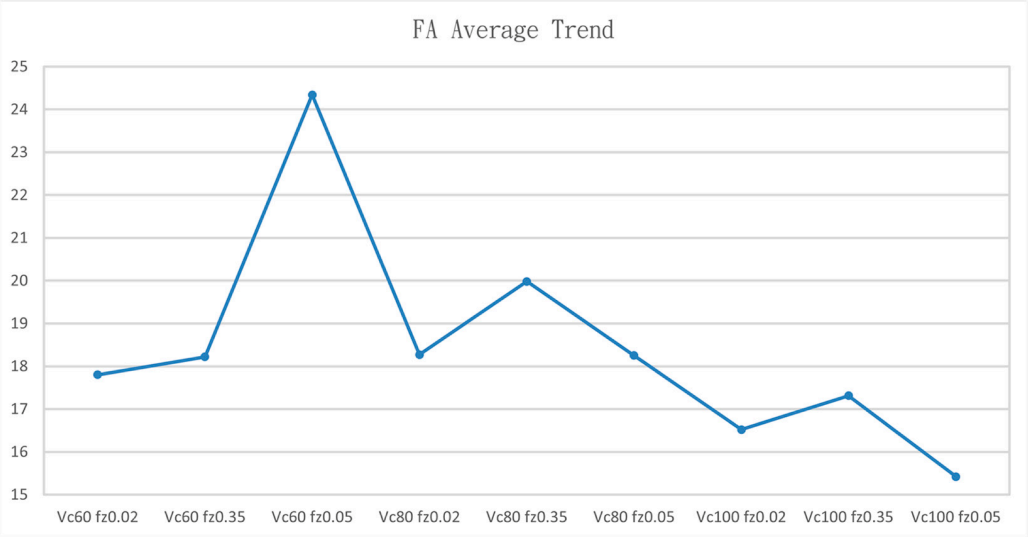


Figure 5. Trend of mean Fa across the nine experimental conditions.

The trend of the mean resultant cutting force Fa across the nine experimental conditions is shown in Figure 5. The Vc = 60 m/min group exhibits a peak of 24.34 N at fz = 0.05, the highest value overall; the three conditions at Vc = 80 m/min are stable, ranging from 18.25 to 18.98 N; and the Vc = 100 m/min group is the lowest overall, ranging from 15.42 to 17.31 N.

The trend of mean surface roughness Ra is shown in Figure 6 and differs markedly from that of Fa. In the Vc = 60 m/min group, Ra decreases monotonically with increasing fz, reaching 0.089 μm at fz = 0.05; the three conditions at Vc = 80 m/min almost coincide

at approximately 0.085 μm; and the Vc = 100 m/min group jumps from 0.063 to 0.112 μm before falling back, showing the greatest dispersion of the three groups.

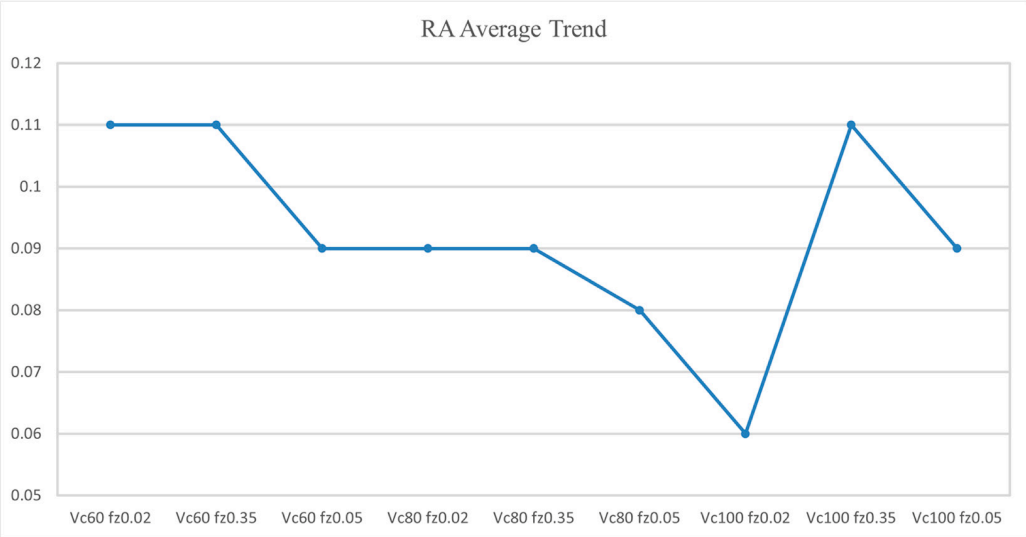


Figure 6. Trend of mean Ra across the nine experimental conditions.

For the classification task, these results carry a methodological implication: the magnitude of the resultant cutting force alone is insufficient to discriminate among the three cutting speeds. Fa in the Vc = 60 group spans 17.80 to 24.34 N and in the Vc = 80 group 18.25 to 18.98 N, overlapping substantially around 18 N; although the Vc = 100 group is lower overall (15.42–17.31 N), its upper bound lies adjacent to the low-feed condition of the Vc = 60 group (17.80 N). In other words, if force magnitude were used as the discriminative feature, the boundaries between the three classes would be unclear. This is precisely why the study adopts amplitude-independent statistics—the dimensionless kurtosis, skewness, crest factor, and spectral entropy, together with the dominant frequency and spectral centroid (in Hz) (see Section 2.3)—rather than the absolute magnitude of force: classification relies on the shape and frequency structure of the signal, not on its amplitude.

Table 6 also helps explain the difficulty of Fold 5 described in Section 4.1.3. Among the three test records in that fold, L3 has the highest Fa overall (24.34 N) and L9 the lowest (15.42 N), the two extremes of the nine conditions. Taking Vc = 60 as an example, during training the model sees only samples of this class at fz = 0.02 and 0.035 (Fa = 17.80 and 18.22 N), yet at test time it must handle samples at fz = 0.05 (Fa = 24.34 N)—a force level roughly 35% higher, with different chip thickness and vibration characteristics.

As for Ra, its variation is not synchronized with Fa: in the Vc = 60 group, Ra decreases as Fa increases with fz, and in the Vc = 100 group Ra fluctuates non-monotonically. Surface roughness is not the target variable of the classification task and is reported here only for completeness of the experimental conditions; nevertheless, its decoupling from cutting force indirectly supports the problem statement in Section 1—when quality indicators such as surface roughness are the prediction target, no known correct basis for discrimination exists against which attribution outputs can be checked, which is why this study instead constructs a benchmark task from cutting-speed classification.

2.3. Input Features and Controlled Variables

The three architectures differ solely in how the network processes the same set of numbers; their input features are identical, consisting of 50 time segments × 6 statistics, namely kurtosis, skewness, crest factor, dominant frequency f_{dom} (Equation (9)), spectral centroid, and spectral entropy. The 1D-CNN and LSTM receive a (50, 6) sequence input,

whereas the NN receives the flattened (300,) vector. The three architectures embody different design orientations: the 1D-CNN slides convolution kernels along the time axis to capture local trends across adjacent time segments; the LSTM reads the sequence recurrently and classifies from the final hidden state; and the NN treats the 300 values as mutually independent tabular features.

Each 5000-point window of a single channel (Mz or Fa) is divided into 50 non-overlapping segments of n = 100 points (0.5 s), and six statistics are computed per segment. With the segment mean μ and standard deviation σ computed from population moments, the excess kurtosis K, skewness S, and crest factor C of a segment $x_1, \ldots, x_n$ are

$$K = \frac{1}{n}\sum_{i=1}^{n}\frac{(x_i-\mu)^4}{\sigma^4} - 3 \tag{6}$$

$$S = \frac{1}{n}\sum_{i=1}^{n}\frac{(x_i-\mu)^3}{\sigma^3} \tag{7}$$

$$C = \frac{\max_i |x_i|}{x_{\mathrm{rms}}}, x_{\mathrm{rms}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} {x_i}^2} \tag{8}$$

where x_{rms} is the root-mean-square value of the segment. The frequency-domain statistics are computed from the one-sided power spectrum $P(f_k)$ of the segment's discrete Fourier transform. The segment mean is removed before the transform, no window function is applied, and the 0 Hz bin is excluded, so the statistics are computed over the 50 bins $f_k = 2k$ Hz ($k = 1, \ldots, 50$). The dominant frequency f_{dom}, spectral centroid f_{c}, and normalized spectral entropy H_{s} are

$$f_{\mathrm{dom}} = \arg\max_{f_k} P(f_k) \tag{9}$$

$$f_{\mathrm{c}} = \frac{\sum_{k=1}^{50} f_k P(f_k)}{\sum_{k=1}^{50} P(f_k)} \tag{10}$$

$$H_{\mathrm{s}} = -\frac{1}{\ln 50}\sum_{k=1}^{50} p_k \ln p_k, p_k = \frac{P(f_k)}{\sum_{m=1}^{50} P(f_m)} \tag{11}$$

where normalization by ln 50 bounds H_{s} between 0 and 1. Kurtosis, skewness, crest factor, and spectral entropy are dimensionless, whereas the dominant frequency and spectral centroid carry units of Hz; all six are independent of the absolute force amplitude.

Before training, the 300 input values of each sample are scaled with z-score standardization (StandardScaler) applied separately to each of the 300 features, and the scaling parameters are estimated on the training data of each fold only and then applied to that fold's validation and test data.

2.4. Training and Evaluation Settings

The dataset comprises 2970 samples (1188, 990, and 792 per class) derived from the Mz and Fa signals of the 15 records of the 3 × 3 full-factorial milling experiments (Table 5), each represented by 50 time segments × 6 amplitude-independent statistics. To ensure that differences in results can be attributed to the architectures themselves rather than to training techniques, L2 regularization and batch normalization were removed from all three models, retaining only dropout, and the optimizer, learning rate, batch size, maximum number of epochs, EarlyStopping, ReduceLROnPlateau, and class_weight settings were kept identical; this is referred to herein as the "identical training configuration". Identical hyperparameters do not guarantee an equally favorable setting for every architecture, because the

optimal configuration may differ between architectures; the comparison therefore reflects performance under a common configuration rather than the best achievable performance of each architecture. Evaluation used five-fold StratifiedGroupKFold cross-validation with the record (milling experiment file) as the grouping unit, ensuring that samples from the same record never appear in both the training and test sets and thereby avoiding data leakage from overlapping windows. Grouping by record does not, however, ensure independence at the level of cutting conditions: records of the same condition, some of which were acquired consecutively within the same machining pass, can be assigned to different folds. A condition-level validation is therefore reported in addition (Sections 3.4 and 4.7). The complete settings are summarized in Table 7.

Table 7. Validation scheme and experimental configuration.

Item	Setting
Cross-validation	Five-fold StratifiedGroupKFold
Grouping unit	Record (milling experiment file); samples from the same record never split across training/test sets
Random seed	Single seed (seed = 42)
Total number of models	15 (1 seed × 5 folds × 3 architectures)
Model retention	Retained per fold for XAI analysis of that fold
XAI application	LRP, LIME, and SHAP applied to every fold, architecture, and class (implementation details in Section 3.3)
Regularization	L2 and batch normalization removed throughout; dropout only

The Mz and Fa signals of each record are segmented independently into 99 windows of 5000 points (25 s) whose start positions are equally spaced from the beginning to the end of the record, so that every record contributes the same number of samples (99 per channel, 198 in total) irrespective of its length and longer records do not dominate training. Because the records differ in length (Table 5), the step between consecutive windows ranges from about 1340 to 2630 points (6.7–13.2 s), corresponding to an overlap of about 47–73% between adjacent windows. Before segmentation, non-cutting intervals (air cuts and pauses, about 3–12% of each record) were removed from the Mz and Fa channels; no segmentation beyond the 99 windows and the 50 time segments is applied.

Mz and Fa windows are treated as independent single-channel samples rather than combined into a two-channel input: each sample is the (50, 6) feature matrix of one channel, labeled with the cutting speed of its record. Because grouping in the cross-validation is by record, the Mz and Fa windows of the same record—and all mutually overlapping windows—always fall on the same side of the train/test split.

Within each training fold, about one quarter of the training records, selected by an inner StratifiedGroupKFold split that again groups by record, is held out as the validation set that drives EarlyStopping and ReduceLROnPlateau; the test records of the fold are never used for model selection.

2.5. XAI Sample Coverage

Fold-wise attribution analysis requires sampling correctly predicted instances from the test set of each fold. The following fold/architecture/class combinations yielded no attribution results, either because the corresponding fold's test set contained no record of that class or because none of its test samples of that class were predicted correctly; they are listed in Table 8.

The absence of Vc100 across all three architectures in Fold 2 reflects the absence of a Vc100 record from the test set of that fold (Table 11) rather than a deficiency of any single architecture: with three test records of 198 samples each, a Vc100 record that was entirely misclassified would limit the fold accuracy to about 67%, whereas all three architectures exceed 93% on Fold 2 (Table 12). All cross-fold means in the following sections are computed over the number of folds actually available; missing cells are not entered as zero. In other

words, not every class has data from all five folds, a point to bear in mind when interpreting the cross-fold statistics in Section 4.

Table 8. Fold/architecture/class combinations without XAI results.

Fold	Architecture	Missing Class
Fold 2	1D-CNN/LSTM/NN (all three)	Vc100
Fold 5	LSTM	Vc60

Because the 15 records are distributed over five folds with 6, 5, and 4 records per class, StratifiedGroupKFold cannot place every class in every test fold: with only four Vc100 records and three test records per fold, one of the five folds necessarily contains no Vc100 record, which is Fold 2 in this study.

It should be noted that the two missing cases have different origins. The Fold 2 gap is structural, arising from the fold composition described above, whereas the Fold 5 gap (LSTM, Vc60) arises from misclassification: the analysis of the Fold 5 test-set composition in Section 4.1.3 shows that the test records for the three classes in that fold were not drawn from the same feed-per-tooth conditions; such within-fold feed asymmetry is a plausible cause of the difficulty in recognizing a particular class in a particular fold. The record and feed composition of each fold should therefore be regarded as essential background for interpreting all cross-fold statistics in this study.

3. Experimental Method

3.1. Layer Structure and Parameter Counts

The layer structure and per-layer parameter counts of the three architectures are given in Table 9. The 1D-CNN has 3027 parameters in total, the LSTM 20,355, and the NN 21,443, a ratio of approximately 1:6.7:7.1 (CNN:LSTM:NN); that is, the 1D-CNN has only about one-seventh the parameters of the other two.

This parameter disparity stems mainly from the weight-sharing mechanism of convolution: the kernels share the same set of weights across all 50 time segments, and GlobalAveragePooling1D is used in place of Flatten to compress the 25×32 feature map directly into 32 dimensions—had Flatten been used, the subsequent Dense layer would require roughly 25,000 additional parameters. The factor of 4 in the LSTM parameter count arises from its four gates: input, forget, output, and candidate.

Table 9. Layer structure and parameter counts of the three architectures (each cell: "layer·parameters").

1D-CNN	LSTM	NN (Dense)
Conv1D(16, k = 3)·304	LSTM(64)·18,176	Dense(64, relu)·19,264
MaxPooling1D(2)·0	Dropout(0.3)·0	Dropout(0.3)·0
Dropout(0.2)·0	Dense(32, relu)·2080	Dense(32, relu)·2080
Conv1D(32, k = 3)·1568	Dense(3, softmax)·99	Dropout(0.2)·0
Dropout(0.2)·0	—	Dense(3, softmax)·99
GlobalAveragePooling1D·0	—	—
Dense(32, relu)·1056	—	—
Dropout(0.3)·0	—	—
Dense(3, softmax)·99	—	—
Total 3027	Total 20,355	Total 21,443

3.2. Training Hyperparameters

The three architectures share identical training hyperparameters and differ only in network structure. For control of randomness, the three architectures within each fold share the same seed, set via tf.keras.utils.set_random_seed(42). The detailed hyperparameter settings are given in Table 10.

Table 10. Detailed training parameters and evaluation conditions.

Item	Setting
Optimizer	Adam, learning_rate = 1×10^{-3}
Loss	categorical_crossentropy
Batch size	8
Max epochs	120
EarlyStopping	val_loss, patience 12, restore_best_weights
ReduceLROnPlateau	val_loss, factor 0.5, patience 5, min_lr 1×10^{-6}
class_weight	balanced (recomputed per fold from that fold's training data)
Dropout	0.2–0.3 after hidden layers and in the classification head, as listed per layer in Table 9
L2/BatchNorm	None (all removed)

3.3. XAI Implementation

Attribution was computed separately for each fold with the model trained on that fold, using correctly predicted test samples of each class. For each fold, architecture, and class, ten correctly predicted test samples are drawn at random (seed 42 plus the fold index); when fewer than ten are available, all are used. The explained output is the softmax probability of the true class, which for correctly predicted samples is also the predicted class.

(a) Software. All models were implemented in TensorFlow/Keras. LRP was computed with a custom TensorFlow implementation based on automatic differentiation (tf.GradientTape); SHAP values were obtained with the shap package (GradientExplainer) and LIME explanations with the lime package (LimeTabularExplainer); data handling and cross-validation used scikit-learn.

(b) LRP propagation rule. Relevance is computed as gradient × input of the explained output with respect to the standardized (50, 6) input. For networks composed of ReLU and linear layers, this corresponds to the LRP-0 rule; for the LSTM, whose gates are not propagated with dedicated LRP-LSTM rules, it serves as an approximation of LRP. Pooling and dropout layers (inactive at inference) require no separate treatment. Throughout this paper, "LRP" denotes this gradient × input implementation; for the LSTM, the result is a gradient × input attribution rather than a full LRP-LSTM propagation, and it is referred to as such where the LSTM is discussed. Its end-of-sequence concentration was therefore examined with gradient-free analyses and ablation (Sections 3.4 and 4.6).

(c) SHAP background. The background distribution consists of 50 samples drawn at random from the standardized training data of the same fold; the GradientExplainer (expected gradients) is used with its default settings.

(d) LIME perturbation settings. Each explanation uses 5000 perturbation samples drawn from the training-data distribution of the same fold, the default kernel width of $0.75 \cdot \sqrt{300} \approx 13.0$, quartile discretization of continuous features (discretize_continuous = True), and weights returned for all 300 features. LIME is applied to the flattened 300-dimensional input.

(e) From signed attributions to importance shares. For each explained sample, the signed attribution values $a_{t,j}$ (time segment $t = 1, \ldots, 50$; statistic $j = 1, \ldots, 6$) are converted to magnitudes $|a_{t,j}|$. The time-axis share of segment t and the feature share of statistic j are

$$s_t = \frac{\sum_{j=1}^{6} |a_{t,j}|}{\sum_{t=1}^{50} \sum_{j=1}^{6} |a_{t,j}|} \tag{12}$$

$$s_j = \frac{\sum_{t=1}^{50} |a_{t,j}|}{\sum_{t=1}^{50} \sum_{j=1}^{6} |a_{t,j}|} \tag{13}$$

so that each set of shares sums to 1.

(f) Averaging over samples, classes, and folds. Shares are first averaged over the explained samples of each class, then over the classes, and finally over the folds actually available; missing fold–class cells (Table 8) are excluded rather than entered as zero. Class averages are unweighted.

(g) Normalized entropy. For a time-axis share vector $s = (s_1, \ldots, s_{50})$, the normalized entropy is

$$H_{\text{norm}} = -\frac{1}{\ln 50} \sum_{t=1}^{50} s_t \ln s_t \tag{14}$$

which equals 1 for a uniform distribution and 0 when all importance falls on a single segment.

(h) Cross-fold reproducibility (Section 4.3) is the mean Spearman rank correlation over all pairs of folds in which the combination is available (10 pairs when all five folds are available, 6 when one fold is missing) of the importance rankings of the 50 time segments (time-axis share vectors) for each architecture–method–class combination.

3.4. Additional Verification Analyses

Two sets of additional analyses were performed to examine the end-of-sequence concentration of the LSTM independently of the gradient × input approximation and to assess generalization at the level of cutting conditions. They used the same feature extraction, the same five record-grouped folds, the identical training configuration, and the same inner validation split as the main experiments, but were run in a separate computing environment. Because software differences change individual fold results, the record-grouped baseline was rerun in the same environment, and all comparisons in Sections 4.6 and 4.7 are made against this rerun baseline rather than against Tables 11 and 12. The LSTM, whose results vary most between training runs, was trained with three random seeds (42, 7, and 2026), except for the repeated attribution analyses after retraining in (b), which used seed 42; the 1D-CNN and NN were trained with seed 42 in the record-grouped analyses and with all three seeds in the condition-level analyses.

Table 11. Five-fold OOF accuracy statistics of the three architectures.

Architecture	OOF Accuracy (%)	95% CI (%)	Fold-to-Fold SD (%)	Min (%)	Max (%)	Parameters
1D-CNN	93.97	86.7–100.0	5.83	84.51	100.00	3027
NN (Dense)	90.24	76.8–100.0	10.86	71.38	97.81	21,443
LSTM	89.12	70.8–100.0	14.75	63.47	98.99	20,355

Note: 95% confidence intervals (CI) of the mean accuracy are computed from the five per-fold accuracies (Table 12) with the t-distribution (df = 4); upper bounds exceeding 100% (101.2%, 103.7%, and 107.4%, respectively) are truncated at 100%.

Table 12. Per-fold accuracy of the three architectures.

Architecture	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
1D-CNN	100.0%	93.4%	94.9%	97.0%	84.5%
NN (Dense)	97.8%	96.1%	90.7%	95.1%	71.4%
LSTM	99.0%	93.8%	98.7%	90.7%	63.5%

(a) Gradient-free sensitivity analysis. For each fold, architecture, and class, up to 30 correctly predicted test samples were explained. Besides gradient × input, two perturbation measures that use only the forward predictions of the model were computed for each time segment t: occlusion, in which all six statistics of segment t are replaced by the training mean (zero after standardization), and resampling, in which they are

replaced by the values of the same segment in randomly drawn training samples (five draws averaged), so that the perturbed input remains within the data distribution. The resulting changes in the true-class probability were converted to time-axis shares in the same way as the attributions (Section 3.3; (e,f)), and the unnormalized mean absolute probability change per segment was also recorded. In addition, the trained models were evaluated with the first, middle, or last k = 1, 5, or 10 segments occluded or resampled at inference time.

(b) Ablation by retraining. Models were retrained and re-evaluated on inputs from which the last segment or the last five segments had been removed, on inputs from which the first five segments had been removed (control), and on time-reversed inputs. For the LSTM retrained without the last five segments and on time-reversed inputs, the analyses of (a) were repeated to examine whether the concentration moves with the end of the input sequence.

(c) Condition-level validation. Two schemes with the cutting condition as the grouping unit were applied: leave-one-condition-out validation (nine folds, each testing all records of one condition) and leave-one-feed-level-out validation (three folds, each testing the three conditions that share one feed per tooth, i.e., one condition per class). The inner validation split was again grouped by record. As a training-free physical reference, each sample was also assigned to the class whose apparent tooth-passing frequency (Table 4: 40.8, 12.2, and 65.3 Hz) is closest to the most frequent segment-level dominant frequency of that sample.

4. Experimental Results

4.1. Classification Performance Comparison

4.1.1. Out-of-Fold Accuracy

The out-of-fold (OOF) accuracy statistics of the three architectures are summarized in Table 11, the per-fold details in Table 12, and the fold-to-fold variation with error bars in Figure 7. The 1D-CNN achieved a mean accuracy of 93.97%, higher than the NN (90.24%) and the LSTM (89.12%), and did so with 3027 parameters—about one-seventh of the other two. The OOF accuracy reported for each architecture is the unweighted mean of its five per-fold test accuracies (Table 12), and the fold-to-fold SD is the sample standard deviation ($n - 1$) of those five values. Because every record contributes the same number of samples (198) and each fold contains three test records (594 samples), this unweighted mean coincides with the pooled OOF accuracy computed over all 2970 samples.

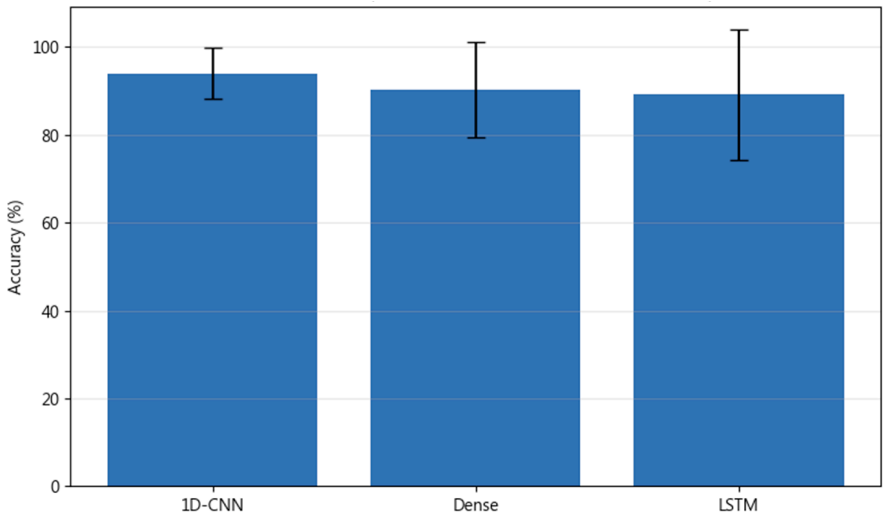


Figure 7. Five-fold OOF accuracy of the three architectures (error bars = fold-to-fold SD).

4.1.2. Fold-to-Fold Variation and Robustness

The mean accuracies rank as 1D-CNN > NN > LSTM, but the more noteworthy difference lies in the fold-to-fold variation. The standard deviations of the NN and LSTM reach 10.86 and 14.75 percentage points, respectively, both exceeding 10 pp, whereas that of the 1D-CNN is only 5.83 pp. This gap stems mainly from Fold 5, where the NN drops to 71.4% and the LSTM to 63.5%, while the 1D-CNN maintains 84.5%. Excluding Fold 5, the three architectures perform comparably over the remaining four folds (1D-CNN 93.4–100.0%, NN 90.7–97.8%, LSTM 90.7–99.0%).

Because grouping is based on records, the differences across folds reflect generalization when testing on previously unseen records of the studied cutting conditions; generalization to unseen conditions is examined in Section 4.7. This supports the inference that the 1D-CNN is the least sensitive to the data partition, sustaining a usable recognition level even under the most unfavorable split, whereas the NN and LSTM degrade markedly in that case. For practical machining condition monitoring, such worst-case performance is often more informative than mean accuracy.

The nature of the reported standard deviations should also be clarified. Since seed repetition was omitted in this run, the standard deviations in Table 11 represent fold-to-fold variation—differences arising from partitioning the data differently under the same settings—and do not include the training stochasticity of obtaining different results from training twice on the same data. The two types of variation have distinct sources: the former reflects the model's sensitivity to the data partition, the latter the uncertainty of the training process itself. Only the former is reported in this study.

The performance conclusions of this study are descriptive in nature. Because only a single random seed was used, only five paired samples are available, and the five folds share the same body of data rather than being independent samples, the power of statistical tests is inherently low; this paper therefore makes no claim of statistical significance for the differences among architectures. The corresponding paired t-test results are provided in Appendix A for reference. The 95% confidence intervals of the mean accuracies (Table 11) and of the paired differences (Appendix A) are wide, and all paired intervals include zero, reflecting the small number of folds and of independent experimental units. Formal statistical inference on architectural differences would require repeating the full procedure with 2–3 random seeds, raising the number of paired samples to 10–15 and allowing the standard deviations to encompass both fold-to-fold variation and training stochasticity.

A further limitation concerns the number of independent experimental units. The 2970 samples are overlapping windows drawn from only 15 records, and these records come from nine cutting conditions—three per class—with the multiple records of a condition acquired consecutively during the same machining pass. The effective number of independent units is therefore closer to nine conditions than to 2970 samples; moreover, records of the same condition can be assigned to different folds, so the folds are not fully independent. Accuracy estimates and fold-to-fold variation should be interpreted with this small number of conditions in mind, and the robustness of the architecture ranking cannot be established without additional conditions and repeated experiments. The condition-level validation in Section 4.7 quantifies the effect of this dependence on the accuracy estimates.

4.1.3. Fold 5 Test-Set Composition and Feed Asymmetry

The composition of the Fold 5 test set is given in Table 13. The fold comprises three records, one per class, each contributing 99 cutting torque (Mz) windows and 99 resultant force (Fa) windows, for a total of 594 samples. Run L4 comprises two records (Table 5), only one of which is in the test set of this fold.

Table 13. Composition of the Fold 5 test set.

Class	Test Record	fz of the Record (mm/tooth)	Samples (Mz + Fa Windows)
Vc60	L3	0.05	99 + 99 = 198
Vc80	L4	0.02	99 + 99 = 198
Vc100	L9	0.05	99 + 99 = 198
Total	3 records	—	594

This table reveals why Fold 5 is particularly difficult: the three classes are not tested under the same feed condition in this fold. The Vc80 test record of L4 has a feed per tooth of 0.02 mm/tooth, whereas Vc60 (L3) and Vc100 (L9) both have 0.05 mm/tooth. Since the statistical features of cutting force signals are governed jointly by cutting speed and feed per tooth, this feed asymmetry transforms the task of the fold from simply discriminating three cutting speeds into discriminating cutting speeds under inconsistent feed conditions. A model that learned its representation of Vc80 mainly from other feed conditions during training must, in this fold, confront signals with altered chip thickness—their time-domain statistics (kurtosis, skewness, crest factor) and spectral morphology both shift, constituting an out-of-distribution test scenario. This observation also echoes the coverage gaps in Section 2.5: in Fold 5 the LSTM has no correctly predicted samples for Vc60, and in Fold 2 the test set contains no Vc100 record.

The differing degrees of degradation across the three architectures in this fold (1D-CNN 84.5%, NN 71.4%, LSTM 63.5%) may be related to their architectural characteristics. The 1D-CNN combines convolutional weight sharing with global average pooling, so its output depends on the averaged response over the entire signal and is less sensitive to local waveform changes, which is consistent with its smallest degradation. For the LSTM, the ablation experiments in Section 4.6 do not support an explanation based on the end-of-sequence attribution concentration discussed in Section 4.5: removing the final segments did not lower its Fold 5 accuracy on average, so the degradation is more plausibly attributed to the feed-condition shift itself. More generally, because most of the difference between architectures arises in this fold, the architecture ranking may partly reflect how the nine conditions are distributed across folds rather than intrinsic architectural capability.

4.2. Distribution of Attributions Along the Time Axis

To quantify how evenly each architecture–method combination distributes importance along the time axis, two indicators are computed from the cross-fold mean time-axis shares (Section 3.3): the normalized entropy and the share of the final (50th) time segment, compared with the uniform value of 0.020. For the 1D-CNN and NN, all three methods yield normalized entropies above 0.97 (NN 0.979–0.985), and their final-segment shares stay close to the uniform value (0.0090–0.0230). The LSTM under SHAP and LIME behaves similarly (final-segment shares 0.0182 and 0.0193). The only exception is the LSTM under LRP (gradient × input), whose normalized entropy of 0.864 is markedly lower than that of every other combination (all above 0.959) and whose final-segment share of 0.1878 is 9.4 times the uniform value; this pattern is examined in Section 4.5. The cross-fold mean distributions are shown in Figure 8.

As Figure 8 shows, all three methods for the 1D-CNN and NN fluctuate mildly around the uniform line with no surges; only the LSTM's LRP (gradient × input) curve rises sharply from about the 40th time segment, exceeding nine times the uniform value at the 50th, while the SHAP and LIME curves of the same model hug the uniform line throughout. The interpretation of this pattern is given in Section 4.5.

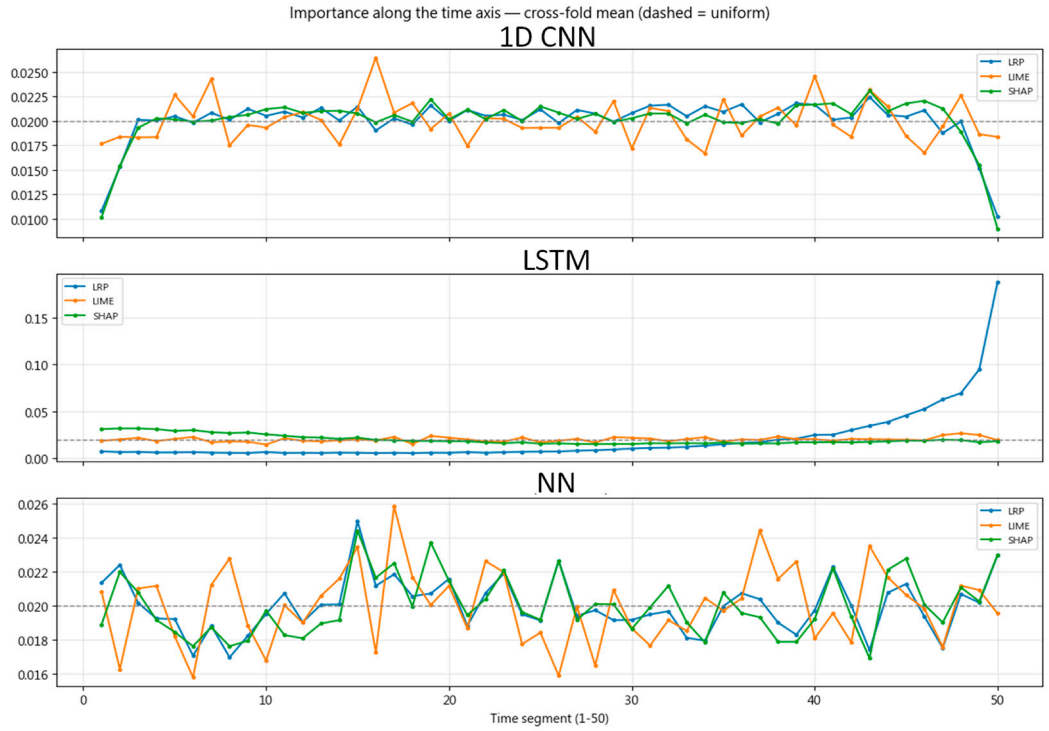


Figure 8. Cross-fold mean importance distribution along the time axis (time segments 1–50) for the three architectures (dashed line = uniform-allocation baseline).

4.3. Cross-Fold Reproducibility of Attributions

This section computes, for each (architecture, method, class) combination, the mean pairwise Spearman correlation of importance rankings across the five folds: the closer the value is to 1, the more consistent the explanatory basis of the five independently trained models. The three-class averages per architecture are shown in Table 14.

Table 14. Cross-fold Spearman correlation of attribution rankings (three-class average).

Architecture	LRP	LIME	SHAP
1D-CNN	0.506	0.007	0.602
NN (Dense)	0.203	0.006	0.235
LSTM	0.699	−0.002	0.534

The first observation concerns LIME and must be read with particular caution. Across all nine (architecture × class) combinations, LIME's cross-fold Spearman correlations fall between −0.05 and 0.05 in this run, that is, essentially uncorrelated: with the same architecture and the same training conditions, exchanging one fold of training data produced LIME rankings that did not align. Within the conditions of this study, this result restricts the use of LIME outputs: its importance rankings should not serve as quantitative evidence, and at most a single-case plot may illustrate what LIME's explanation of a given sample looks like—no claim that an architecture stably relies on certain features can be based on it. Methodologically, the phenomenon accords with LIME's sampling nature: its explanations are built on random perturbations sampled from the training-data distribution and weighted by their proximity to the explained sample, and in the present task, with limited samples and a grouped data structure, the variance of this perturbation sampling may be large enough to mask true differences between models. This finding is specific to the present setting and should not be generalized to LIME as such: the reproducibility of LIME rankings may depend on the number of explained samples per cell (ten here), the number

of perturbation samples (5000), the discretization of continuous features, the kernel width, the number of features returned, the ranking target (the 50 time-segment shares), and the way signed weights are aggregated (absolute versus signed values, class-averaged versus sample-level comparison). Exploratory recalculations in which these aggregation choices were varied yielded markedly different, and generally higher, cross-fold correlations for LIME. The near-zero values in Table 14 should therefore be regarded as an observation of this particular run and analysis pipeline rather than as an established property of LIME, even within this benchmark; a systematic sensitivity analysis over these factors is left to future work.

The second finding is that the cross-fold consistency of LRP and SHAP is markedly higher, though with large gaps between architectures. The 1D-CNN attains 0.506 (LRP) and 0.602 (SHAP), the LSTM 0.699 and 0.534, but the NN only 0.203 and 0.235. The NN's correlations across the three classes range from 0.044 to 0.452, indicating weaker explanatory stability than the other two architectures—even though its time-axis importance distribution is the closest to uniform (entropies of 0.979–0.985, Section 4.2, the highest of the three), its ranking of which positions matter swings substantially from fold to fold. This contrast shows that a uniform distribution does not equal a stable explanation; the two are independent properties.

The third finding is that the LSTM's LRP cross-fold consistency is the highest of the three architectures (0.699). This value must be read together with the concentration results of Section 4.2: the LSTM's LRP importance is highly concentrated on the final time segments, and that pattern reappears stably in the models of all five independently trained folds, which drives the ranking correlation upward. In other words, the high consistency here reflects not "a dispersed and stable explanation" but "the same end-of-sequence concentration recurring"—five independently trained models converge on the sequence end, indicating that the attribution pattern is reproducible across training runs rather than the accident of a single run. The example also illustrates that a consistency metric alone cannot judge attribution quality; only in combination with the concentration metric can one know what it is that is consistent.

Combining the performance stability of Section 4.1 with the two explainability indicators above, the 1D-CNN is the only architecture that performs well simultaneously in robustness of performance, dispersion of importance, and cross-fold reproducibility, and it is therefore the most suitable for supporting claims about what the model bases its decisions on.

4.4. Statistical Features Relied upon by Each Architecture

Beyond the distribution over time, this section examines which statistical features each architecture relies upon. Under uniform allocation, each of the six statistics should account for 0.167 (1/6); cross-fold averages are shown in Table 15 and Figure 9. The shares in Table 15 and Figure 9 are averaged over LRP, LIME, and SHAP (Section 3.3). Because the cross-fold reproducibility of LIME rankings was low in this run and sensitive to the aggregation procedure (Section 4.3), the LIME component is the least reliable part of these averages; the discussion below therefore relies on the qualitative ordering of the statistics rather than on their exact values.

The three architectures exhibit a highly consistent pattern: the frequency-domain features (dominant frequency, spectral entropy, spectral centroid) all lie above or level with the uniform line, whereas the three time-domain features (kurtosis, skewness, crest factor) all fall below it. For the dominant frequency, the LSTM allocates 0.264, the 1D-CNN 0.230, and the NN 0.217—about 1.3 to 1.6 times the uniform value. This cross-architecture consistency is a positive signal: three architectures of different principles, each trained

independently, converge on the same physically plausible set of frequency-domain features, which is less suggestive of individual architectures guessing blindly or taking shortcuts.

Table 15. Importance shares of the six statistical features (cross-fold averages; uniform allocation = 0.167).

Statistic	1D-CNN	NN (Dense)	LSTM
Kurtosis	0.120	0.137	0.110
Skewness	0.132	0.133	0.124
Crest factor	0.112	0.141	0.117
Dominant frequency	0.230	0.217	0.264
Spectral centroid	0.176	0.164	0.169
Spectral entropy	0.231	0.208	0.215

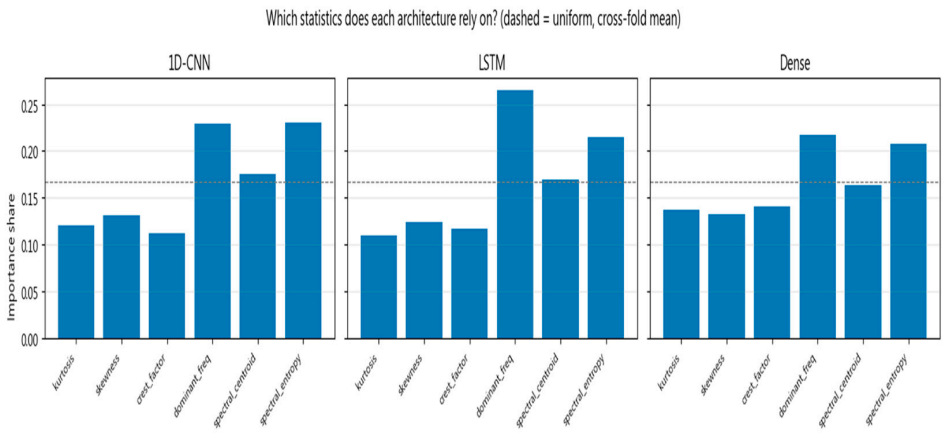


Figure 9. Importance shares allocated to the six statistics by the three architectures (cross-fold averages; dashed line = uniform-allocation baseline).

It must be noted that the 1D-CNN's spectral-entropy share (0.231) and dominant-frequency share (0.230) are effectively tied, differing by only 0.001—within the range of cross-fold variation—so the architecture should not be claimed to rely "primarily on the dominant frequency"; the safer statement is that the 1D-CNN relies on the dominant frequency and spectral entropy jointly as its principal basis. The LSTM and NN clearly place the dominant frequency first.

These results carry clear physical meaning. The dominant frequency is determined by the spindle speed N—through the spindle rotation frequency and the aliased tooth-passing frequency (Section 2.1)—and cutting speed is defined by Equation (2); with the tool diameter D fixed, the dominant frequency and Vc therefore stand in a one-to-one relationship. That all three architectures, given no physical prior whatsoever, independently learned the feature most directly tied to the definition of cutting speed is consistent with the models relying on physically meaningful features. The condition-level validation in Section 4.7 qualifies this statement: the models do not rely on this frequency alone. It should be borne in mind, however, that precisely because D is fixed and the dominant frequency maps almost one-to-one onto Vc, the difficulty of this classification task is inherently low; the results of this section should be positioned as a validation of explainability—confirming that the learned representations accord with the physical definition—rather than as a demonstration of predictive capability.

4.5. End-of-Sequence Attribution Concentration in the LSTM

As shown in Section 4.2, the LSTM under LRP is the sole anomaly among the nine architecture–method combinations. The pattern is not the accident of a single training run: the models trained independently on all five folds reproduce the same distribution, and

the cross-fold Spearman correlation of their LRP rankings reaches 0.699, the highest of the three architectures.

Because the LSTM attribution is a gradient × input approximation rather than a full LRP-LSTM propagation (Section 3.3; (b)), it must first be established whether this concentration reflects the model or the attribution method; the gradient-free sensitivity analysis and the ablation experiments performed for this purpose are reported in Section 4.6. In the attribution itself, the evidence for the decision is placed at the end of the signal, a position bearing no stable causal relationship to cutting speed, whereas the physical basis (the spindle-speed-related frequency content) is present throughout the signal. Attribution concentration alone does not establish shortcut learning: it shows where the explanation places the evidence, not that the model's decisions depend on these positions. Architecturally, a plausible cause is that the LSTM classifies from its final hidden state, and the influence of time segments near the sequence end on that state is not attenuated by subsequent gating, so the model may develop a strategy of relying mainly on the tail of the sequence. By contrast, the 1D-CNN spreads weight evenly across time segments through global average pooling, and the NN flattens the 50 time segments into tabular features of equal standing; neither has this structural preference—and indeed their final-segment shares fall between 0.0090 and 0.0230, hugging the uniform value.

A further clarification is needed: Section 4.4 shows that the LSTM's dominant-frequency share (0.264) is the highest of the three architectures, which does not conflict with the judgment of this section, because the two concern different dimensions. Section 4.4 addresses the allocation of weight among the six statistics (the feature dimension), whereas this section addresses the allocation among the 50 time segments (the time dimension). A model can simultaneously "rely most on the dominant frequency among features" and "read only the sequence end in time"—that is, what it would read is the dominant-frequency value of the final time segments, not the frequency structure of the whole signal. Section 4.6 shows, however, that such a narrowing does not hold for the classification itself: the concentration describes where the output of the LSTM is most sensitive, whereas removing the final segments does not reduce its accuracy.

As for why SHAP (0.0182) and LIME (0.0193) do not exhibit the concentration, the two methods must be considered separately, because they rest on different principles (Section 3.3). The LRP variant used here (gradient × input) measures the local sensitivity of the explained output at the actual input relative to an all-zero reference, which after standardization corresponds to the training mean. SHAP, as implemented by GradientExplainer (expected gradients), is also gradient-based, but it averages gradients along paths from 50 background samples of the training data to the input, so its attributions are referenced to realistic training signals rather than to a single mean reference. If the final hidden state of the LSTM is locally highly sensitive to the last time segments while those segments differ little between the explained sample and typical training samples, gradient × input can assign them large relevance whereas expected gradients assign them little. LIME, in contrast, is a perturbation-based method that fits a local surrogate model to randomly perturbed, discretized copies of the input and presupposes that input features can be perturbed independently; perturbing the value of a single time segment may not remove the sequence-position information carried by the recurrent hidden states, so the importance it estimates for the final segments may be low. These explanations remain hypotheses: the discrepancy could also reflect differences in the explained output, the reference or background data, or the perturbation settings (Section 3.3), and it should therefore be read as a disagreement among methods to be resolved rather than as evidence that one method is correct.

Whether this concentration is related to the LSTM's drop on Fold 5 (Section 4.1.3) was examined by ablation (Section 4.6): averaged over three seeds, removing the last five segments did not lower the LSTM's Fold 5 accuracy (90.7% versus 78.2% for the rerun baseline), so the drop is not explained by reliance on the sequence end.

As a matter of reporting practice, this study recommends presenting the results of both LRP and SHAP side by side with the discrepancy explicitly noted, and never concluding that a model is free of positional dependence on the strength of a single attribution method alone—this case shows that had only LIME or SHAP been used, the concentration phenomenon would have gone entirely undetected. Conversely, Section 4.6 shows that an attribution concentration detected by one method should not be interpreted as a decision basis without independent verification.

The direct verification of this possible dependence—a gradient-free sensitivity analysis and an ablation experiment in which the final segments are removed, and the LSTM is retrained and re-evaluated—is reported in Section 4.6.

4.6. Verification of the End-of-Sequence Concentration

Table 16 compares the final-segment share and the normalized entropy obtained with gradient × input and with the two gradient-free measures of Section 3.4 (a). For the 1D-CNN and NN, all three measures remain close to uniform (final-segment share 0.41–1.16 times the uniform value; normalized entropy ≥ 0.988). For the LSTM, occlusion and resampling reproduce the concentration found with gradient × input: the final-segment share is 9.0 and 9.3 times the uniform value, respectively, compared with 8.0 times for gradient × input in the rerun (9.4 times in Section 4.2), and the normalized entropy is 0.865–0.873. The concentration is therefore not an artifact of the gradient × input approximation: it also appears when segments are perturbed, and only the forward predictions are compared. Its magnitude varied considerably between folds and seeds (2.6–21.3 times the uniform value across the 15 fold–seed runs). The rerun baseline accuracies (1D-CNN 97.7%, LSTM 91.2%, NN 95.9%) differ from Table 11 because of the different computing environment and are used only as the reference for the comparisons below.

Table 16. Final-segment share (multiple of the uniform value) and normalized entropy H of the time-axis distributions obtained with gradient × input and two gradient-free sensitivity measures.

Architecture/Input	Gradient × Input	Occlusion (Training Mean)	Resampling (Training Samples)
1D-CNN	0.49 (H = 0.998)	0.43 (H = 0.996)	0.41 (H = 0.995)
NN	1.15 (H = 0.999)	1.16 (H = 0.994)	1.10 (H = 0.988)
LSTM	8.03 (H = 0.873)	9.02 (H = 0.865)	9.26 (H = 0.869)
LSTM, retrained without last 5 segments [1]	6.76 (H = 0.890)	7.70 (H = 0.877)	7.30 (H = 0.887)
LSTM, retrained on time-reversed input [2]	7.48 (H = 0.883)	7.95 (H = 0.876)	8.31 (H = 0.869)

Note: Rerun with the record-grouped five folds; LSTM rows 1–3: three seeds × five folds; 1D-CNN, NN, and the retrained LSTM: seed 42. Uniform value = 1/50 (1/45 for the input without the last five segments). [1] Share of the new final segment (segment 45). [2] Share of the final position, which now contains the original first segment.

Ablation shows that the concentration follows the position rather than the content of the final segment (Table 16). When the LSTM was retrained without the last five segments, the concentration moved to the new final segment (segment 45; 6.8–7.7 times the uniform value). When it was retrained on time-reversed inputs, the concentration again appeared at the final position (7.5–8.3 times), which now contains the original first segment, whereas the original final segment, now in the first position, received less than the uniform share (0.43–0.89 times). The concentration thus reflects the sensitivity of the LSTM's final hidden

state to its most recent inputs (Section 4.5) rather than information specific to the end of the signal.

The concentration does not, however, indicate that the classification depends on the final segments (Table 17). Occluding the last five segments at inference time changed the LSTM's accuracy by +0.5 percentage points on average (resampling: −0.9 pp), whereas occluding the first five changed it by −1.2 pp (resampling: −3.8 pp); with ten segments, removing the first segments again caused the larger loss (−6.1 versus −1.2 pp; resampling: −11.4 versus −6.8 pp). Retraining without the last segment or the last five segments did not reduce the LSTM's accuracy (94.7% and 95.6% versus 91.2% for the rerun baseline, three-seed means), and the same held for the 1D-CNN and NN. The unnormalized probability changes point in the same direction: for the LSTM, occluding the first segment changed the true-class probability by 0.0071 on average (resampling: 0.0148), compared with 0.0019 (0.0023) for the last segment. The large final-segment share in Table 16 is thus a relative quantity, normalized within each sample; it is consistent with the true-class probability being nearly saturated for most explained samples, so that their small residual changes occur mainly at the last positions, while larger changes in the prediction are produced mainly by early segments.

In summary, the end-of-sequence concentration of the LSTM is reproduced by gradient-free measures and is therefore not attributable to the gradient × input approximation; it is a positional sensitivity of the LSTM's output that follows the last input position. It is not evidence that the LSTM relies on the final segment for classification, nor a property of LRP in general, and the interpretation as a possible positional dependence of the decisions (Section 4.5) is not supported by the ablation. The discrepancy between gradient × input and SHAP or LIME (Section 4.5) therefore concerns the local sensitivity at the sequence end rather than the evidence that determines the classification.

Table 17. Accuracy (%) of the rerun models under inference-time masking and retraining ablation (record-grouped five-fold cross-validation).

Condition	1D-CNN	LSTM	NN
Rerun baseline	97.7	91.2	95.9
Occlusion, last 5 segments (Δ, pp)	+0.1	+0.5	−0.4
Occlusion, first 5 segments (Δ, pp)	+0.0	−1.2	−0.7
Occlusion, last 10 segments (Δ, pp)	+0.4	−1.2	−0.9
Occlusion, first 10 segments (Δ, pp)	+0.2	−6.1	−1.3
Resampling, last 5 segments (Δ, pp)	+0.1	−0.9	−1.4
Resampling, first 5 segments (Δ, pp)	+0.1	−3.8	−0.6
Resampling, last 10 segments (Δ, pp)	−2.1	−6.8	−6.4
Resampling, first 10 segments (Δ, pp)	−1.5	−11.4	−5.2
Retrained without last segment	98.1	94.7	95.3
Retrained without last 5 segments	96.6	95.6	94.8
Retrained without first 5 segments (control)	98.0	93.0	94.9
Retrained on time-reversed input	97.7	91.4	93.9

Note: Mean over the five folds; LSTM: mean of three seeds (42, 7, 2026); 1D-CNN and NN: seed 42. Δ = change relative to the rerun baseline of the same model when the segments are replaced at inference time without retraining.

4.7. Condition-Level Validation

Table 18 reports the accuracy when entire cutting conditions are held out (Section 3.4; (c)). Under leave-one-condition-out validation, the mean accuracy over the nine conditions was 86.5% for the 1D-CNN, 83.5% for the LSTM, and 84.6% for the NN (three-seed means), compared with 97.7%, 91.2%, and 95.9% under record-level grouping in the same environment. Most of this difference arises from condition L2 (Vc = 60 m/min, fz = 0.035 mm/tooth), which was misclassified almost entirely as Vc100 by all three architectures (0–17% correct); L4 (1D-

CNN, 71.5%) and L5 (LSTM, 81.9%) were also less stable, whereas the other conditions were recognized at 84.5–100%. Under leave-one-feed-level-out validation, in which each class is represented in training only by its two conditions at the other feed levels, the mean accuracy fell to 54–57%: holding out fz = 0.05 mm/tooth still gave 95–99%, but holding out fz = 0.02 or 0.035 mm/tooth gave only 24–45%.

These results show that the record-level accuracies in Tables 11 and 12 overestimate performance on unseen cutting conditions: the models also use condition-specific characteristics that do not transfer to other conditions of the same cutting speed. The classification results of this study should therefore be read as generalization across records of the studied conditions, not as generalization to unseen machining conditions. The condition-level schemes have their own limitations: with three conditions per class, holding out one condition leaves only two conditions of that class for training, and holding out one feed level removes a third of the training conditions and requires extrapolation in feed per tooth, so these accuracies represent a stringent lower bound rather than an estimate of performance under a fuller experimental design.

The failure on L2 is not caused by the absence of the physical cue in that condition. The training-free rule based on the apparent tooth-passing frequency classified 99.9% of all samples correctly and at least 99.0% in every condition, including L2. The networks' errors on L2 therefore indicate that their decisions are not determined by the dominant frequency alone but also draw on other statistics whose values differ between conditions of the same cutting speed. This qualifies the interpretation in Section 4.4: the predominance of frequency-domain features in the attributions is consistent with the physical definition of cutting speed, but it does not imply that the models rely exclusively on the physically determined frequency. Within the benchmark, the condition-level results thus mark where the trained models depart from the known basis for discrimination.

Table 18. Accuracy (%) under condition-level validation (three-seed means) and of the training-free dominant-frequency rule.

Held-Out Test Set	Vc (m/min)/fz (mm/tooth)	1D-CNN	LSTM	NN	Frequency Rule
L1	60/0.02	99.7	99.3	87.6	100.0
L2	60/0.035	17.2	0.0	0.8	100.0
L3	60/0.05	92.4	89.1	84.5	100.0
L4	80/0.02	71.5	87.6	100.0	100.0
L5	80/0.035	98.4	81.9	95.0	99.0
L6	80/0.05	100.0	100.0	100.0	100.0
L7	100/0.02	99.0	95.6	95.9	100.0
L8	100/0.035	100.0	97.8	97.8	100.0
L9	100/0.05	100.0	100.0	99.3	100.0
Leave-one-condition-out, mean	—	86.5	83.5	84.6	99.9
fz = 0.02 held out (L1, L4, L7)	—	44.6	38.2	45.0	100.0
fz = 0.035 held out (L2, L5, L8)	—	26.9	29.0	23.5	99.6
fz = 0.05 held out (L3, L6, L9)	—	98.5	95.1	94.9	100.0
Leave-one-feed-level-out, mean	—	56.7	54.1	54.5	99.9
Record-grouped five folds (rerun)	—	97.7	91.2	95.9	99.9

Note: Seed-to-seed standard deviation of the mean accuracy: leave-one-condition-out 5.5, 4.2, and 0.4 pp, leave-one-feed-level-out 3.6, 6.8, and 2.3 pp (1D-CNN, LSTM, NN). Record-grouped values: LSTM three seeds, 1D-CNN and NN seed 42. Frequency rule: each sample is assigned to the class whose apparent tooth-passing frequency (Table 4) is closest to its most frequent segment-level dominant frequency; it involves no training.

5. Conclusions

This study performed three-class cutting-speed classification on cutting force signals from 17-4PH milling, comparing the 1D-CNN, LSTM, and NN architectures under an

identical training configuration, and—by retaining the model of each fold—applied LRP, LIME, and SHAP attribution analyses to every fold individually.

In terms of performance, the 1D-CNN achieved an out-of-fold accuracy of 93.97% with roughly one-seventh the parameters, higher than the NN (90.24%) and the LSTM (89.12%), and its fold-to-fold standard deviation of 5.83 percentage points was the smallest of the three; on the most difficult fold, Fold 5, its 84.5% likewise clearly surpassed the other two at 71.4% and 63.5%. However, because this run used a single random seed and only five paired folds were available, the performance conclusions are descriptive only—the 1D-CNN is the highest on average and the most robust—and no claim of statistically significant superiority is made (paired *t*-test results are given in Appendix A). Moreover, since most of the mean difference is contributed by Fold 5, whose three test records have inconsistent feed conditions (see Table 13), the advantage of the 1D-CNN is better understood as smaller degradation under feed-condition shift rather than higher accuracy in general settings. These accuracies refer to record-level grouping; when whole cutting conditions were held out (Section 4.7), the mean accuracy fell to 83–86% (leave-one-condition-out) and 54–57% (leave-one-feed-level-out), with condition L2 misclassified almost entirely, so the results describe generalization across records of the studied conditions rather than to unseen machining conditions.

Regarding explainability, four conclusions are drawn. First, the discriminative bases of the 1D-CNN and NN are distributed across the entire signal (normalized entropies all above 0.97), with no abnormal concentration of weight. Second, in this run LIME's importance rankings showed near-zero correlation across folds (-0.05 to 0.05), whereas the cross-fold consistency of LRP and SHAP was markedly higher; because this observation proved sensitive to how the attributions were aggregated, it is not claimed as a general property of LIME, and LIME rankings are not used here as quantitative evidence. Third, the LSTM exhibits end-of-sequence attribution concentration under gradient $\times$ input (the LRP implementation used here): its final-segment importance share reaches 0.1878 (9.4 times the uniform value), its normalized entropy of 0.864 is the lowest of all combinations, and the pattern reappears stably in the independently trained models of all five folds (cross-fold LRP correlation 0.699). Gradient-free occlusion and resampling reproduced the concentration, and ablation showed that it follows the last input position rather than the content of the final segment, while removing the final segments did not reduce accuracy, including on Fold 5. The concentration is therefore not attributable to the gradient $\times$ input approximation; it is a positional sensitivity of the LSTM output rather than a reliance of its classification on the sequence end. That SHAP and LIME did not show the concentration may reflect their different reference points—SHAP averages gradients relative to background training samples, and LIME perturbs individual features, which may not remove the sequence-position information carried by a recurrent network—but these explanations have not yet been tested. In addition, dispersion of importance and stability of explanation are two mutually independent properties: the NN's time-axis distribution is the closest to uniform, yet its cross-fold ranking consistency is the lowest (LRP 0.203, SHAP 0.235). The indicators must be interpreted jointly; any one of them used alone may mislead. Fourth, all three architectures allocate importance markedly to frequency-domain rather than time-domain features, with the dominant-frequency share at roughly 1.3 to 1.6 times the uniform value, consistent with the physical definition of cutting speed (Equation (2)), since the dominant frequency reflects the spindle rotation frequency and the aliased tooth-passing frequency, both proportional to N; this is consistent with the physical definition, although the condition-level validation (Section 4.7) shows that the models do not rely on this frequency alone. Finally, the 200 Hz sampling rate does not resolve the tooth-passing

frequencies (159.2–265.3 Hz), which are aliased into the analyzed band; future experiments should use a sampling rate of at least 2 kHz with anti-aliasing filtering.

These conclusions are subject to four limitations. First, the data comprise 15 records from nine cutting conditions (three per class), so the number of independent experimental units is small and fold-level statistics depend on how conditions are assigned to folds; record-level grouping does not ensure independence at the condition level, and condition-level validation gave markedly lower accuracy, so the classification results are restricted to generalization across records of the studied conditions. Second, because the tool diameter is fixed, cutting speed is encoded by a strong spectral cue; the benchmark should therefore be regarded as a controlled validation framework for attribution methods rather than as general evidence that XAI methods are reliable in more complex monitoring tasks such as tool-wear or surface-roughness prediction. Third, the LIME observations are specific to the explanation settings and aggregation procedure of this run and changed markedly under alternative aggregation choices. Fourth, the verification analyses (Sections 4.6 and 4.7) were run in a separate computing environment with a rerun baseline; they used three seeds for the LSTM but a single seed for the 1D-CNN and NN in the record-grouped experiments. Future work should extend the experimental design to additional cutting-speed and feed levels and to other tool diameters, so that the benchmark rests on more independent conditions and its frequency cue is not tied to a single tool geometry.

Author Contributions: Conceptualization, T.-H.C. and C.-C.T.; methodology, T.-H.C. and S.-F.C.; software, B.-X.W. and S.-W.C.; validation, B.-X.W. and S.-W.C.; investigation, S.-F.C., B.-X.W. and S.-W.C.; data curation, B.-X.W. and S.-W.C.; writing—original draft preparation, T.-H.C.; writing—review and editing, C.-C.T. and S.-F.C.; supervision, T.-H.C. and C.-C.T.; funding acquisition, T.-H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Science and Technology Council (NSTC), Taiwan, grant number NSTC 114-2222-E-262-001.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding authors.

Acknowledgments: The authors thank the National Science and Technology Council, Taiwan, for supporting this research under Project No. NSTC 114-2222-E-262-001.

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

Appendix A. Paired *t*-Test Results

Paired *t*-tests (two-sided, df = 4) on the five per-fold accuracies in Table 12 give the following results: 1D-CNN vs. NN, mean difference 3.74 percentage points, t = 1.44, $p = 0.223$; 1D-CNN vs. LSTM, mean difference 4.82 percentage points, t = 1.11, $p = 0.331$; NN vs. LSTM, mean difference 1.08 percentage points, t = 0.40, $p = 0.710$. None of the differences is statistically significant at the 0.05 level, consistent with the descriptive interpretation in Section 4.1.2. The corresponding 95% confidence intervals of the paired differences are [−3.47, 10.95] (1D-CNN vs. NN), [−7.28, 16.92] (1D-CNN vs. LSTM), and [−6.44, 8.60] (NN vs. LSTM) percentage points; all three include zero, consistent with the paired *t*-test results.

References

1. Aysel, H.I.; Cai, X.; Prugel-Bennett, A. Explainable artificial intelligence: Advancements and limitations. *Appl. Sci.* **2025**, *15*, 7261. [CrossRef]
2. Bennetot, A.; Donadello, I.; Haouari, A.E.Q.E.; Dragoni, M.; Frossard, T.; Wagner, B.; Sarranti, A.; Tulli, S.; Trocan, M.; Chatila, R.; et al. A practical tutorial on explainable AI techniques. *ACM Comput. Surv.* **2025**, *57*, 1–44. [CrossRef]
3. Bento, J.; Saleiro, P.; Cruz, A.F.; Figueiredo, M.A.T.; Bizarro, P. TimeSHAP: Explaining recurrent models through sequence perturbations. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), Virtual, 14–18 August 2021; pp. 2565–2573. [CrossRef]
4. Borujeni, S.M.; Arras, L.; Srinivasan, V.; Samek, W. Explainable sequence-to-sequence GRU neural network for pollution forecasting. *Sci. Rep.* **2023**, *13*, 9940. [CrossRef] [PubMed]
5. Yang, Z.; Zhang, A.; Sudjianto, A. Enhancing explainability of neural networks through architecture constraints. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 2610–2621. [CrossRef] [PubMed]
6. Proietti, M.; Ragno, A.; Capobianco, R. XAI-guided continual learning: Rationale, methods, and future directions. *WIREs Data Min. Knowl. Discov.* **2025**, *15*, e70046. [CrossRef]
7. Hsieh, P.-Y.; Wu, B.-D.; Hung, F.-Y. 17-4 precipitation-hardening stainless steel: Soft-tough heat treatment mechanism and thermal fatigue characteristics. *Materials* **2024**, *17*, 5851. [CrossRef] [PubMed]
8. Villa, M.; Grumsen, F.B.; Niessen, F.; Dahmen, T.; Cao, L.; Reich, M.; Kessler, O.; Huang, X.; Somers, M.A. Aging 17-4 PH martensitic stainless steel prior to hardening: Effects on martensitic transformation, microstructure and properties. *Materialia* **2023**, *32*, 101882. [CrossRef]
9. Rafi, H.K.; Pal, D.; Patil, N.; Starr, T.L.; Stucker, B.E. Microstructure and mechanical behavior of 17-4 precipitation hardenable steel processed by selective laser melting. *J. Mater. Eng. Perform.* **2014**, *23*, 4421–4428. [CrossRef]
10. Sabzi, H.E.; Lim, S.-H.; Della Crociata, D.; Castellote-Alvarez, R.; Simonelli, M.; San-Martín, D.; Hao, X.; Choi, P.-P.; Rivera-Díaz-Del-Castillo, P.E. Genetic design of precipitation-hardening stainless steels for additive manufacturing. *Acta Mater.* **2024**, *274*, 120018. [CrossRef]
11. Villegas-Tovar, J.; Gaona-Tiburcio, C.; Lara-Banda, M.; Maldonado-Bandala, E.; Baltazar-Zamora, M.A.; Cabral-Miramontes, J.; Nieves-Mendoza, D.; Olguin-Coca, J.; Estupiñan-Lopez, F.; Almeraya-Calderón, F. Electrochemical corrosion behavior of passivated precipitation hardening stainless steels for aerospace applications. *Metals* **2023**, *13*, 835. [CrossRef]
12. Daniyan, I.A.; Tlhabadira, I.; Phokobye, S.N.; Siviwe, M.; Mpofu, K. Modelling and optimization of the cutting forces during Ti6Al4V milling process using the response surface methodology and dynamometer. *MM Sci. J.* **2019**, *2019*, 3353–3363. [CrossRef]
13. Liang, Q.; Zhang, D.; Coppola, G.; Mao, J.; Sun, W.; Wang, Y.; Ge, Y. Design and analysis of a sensor system for cutting force measurement in machining processes. *Sensors* **2016**, *16*, 70. [CrossRef] [PubMed]
14. Subasi, O.; Yazgi, S.G.; Lazoglu, I. A novel triaxial optoelectronic based dynamometer for machining processes. *Sens. Actuators A Phys.* **2018**, *279*, 168–177. [CrossRef]
15. Li, Z.; An, L.; Ding, Y.; Lin, H.; Chen, Y.-L. Spindle-integrated three-axis cutting force measurement system for ultra-precision diamond milling. *Sensors* **2026**, *26*, 1817. [CrossRef] [PubMed]
16. Zhao, J. The design of CNC lathe cutting force signal on-line monitoring. *Appl. Mech. Mater.* **2012**, *155–156*, 526–530. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.