processes

MDPI

Article

Hybrid Cloud-Based Information and Control System Using LSTM-DNN Neural Networks for Optimization of Metallurgical Production

Kuldashbay Avazov [1], Jasur Sevinov [2], Barnokhon Temerbekova [3], Gulnora Bekimbetova [4], Ulugbek Mamanazarov [3], Akmalbek Abdusalomov [1,5,6] and Young Im Cho [1,*]

[1] Department of Computer Engineering, Gachon University Sujeong-Gu, Seongnam-si 13120, Republic of Korea; kuldoshbay@gachon.ac.kr (K.A.); akmaljon@gachon.ac.kr (A.A.)
[2] Department of Information Processing and Management Systems, Tashkent State Technical University, Tashkent 100095, Uzbekistan; j.sevinov@tdtu.uz
[3] Department of Information Technologies and Automation of Technological Processes and Production, Almalyk Branch of the National University of Science and Technology MISIS, Almalyk 110100, Uzbekistan; misis_temerbekova@mail.ru (B.T.); m67811@mail.ru (U.M.)
[4] Department of Economics and Management, Tashkent State University of Economics, Tashkent 100007, Uzbekistan; gmbbkd@gmail.com
[5] Department of Computer Systems, Tashkent University of Information Technologies Named After Muhammad Al-Khwarizmi, Tashkent 100200, Uzbekistan
[6] Department of International Scientific Journals and Rankings, Alfraganus University, Yukori Karakamish Street 2a, Tashkent 100190, Uzbekistan
* Correspondence: yicho@gachon.ac.kr

Abstract

A methodology for detecting systematic errors in sets of equally accurate, uncorrelated, aggregate measurements is proposed and applied within the automatic real-time dispatch control system of a copper concentrator plant (CCP) to refine the technical and economic performance indicators (EPIs) computed by the system. This work addresses and solves the problem of selecting and obtaining reliable measurement data by exploiting the redundant measurements of process streams together with the balance equations linking those streams. This study formulates an approach for integrating cloud technologies, machine learning methods, and forecasting into information control systems (ICSs) via predictive analytics to optimize CCP production processes. A method for combining the hybrid cloud infrastructure with an LSTM-DNN neural network model has been developed, yielding a marked improvement in TEP for copper concentration operations. The forecasting accuracy for the key process parameters rose from 75% to 95%. Predictive control reduced energy consumption by 10% through more efficient resource use, while the copper losses to tailings fell by 15–20% thanks to optimized reagent dosing and the stabilization of the flotation process. Equipment failure prediction cut the amount of unplanned downtime by 30%. As a result, the control system became adaptive, automatically correcting the parameters in real time and lessening the reliance on operator decisions. The architectural model of an ICS for metallurgical production based on the hybrid cloud and the LSTM-DNN model was devised to enhance forecasting accuracy and optimize the EPIs of the CCP. The proposed model was experimentally evaluated against alternative neural network architectures (DNN, GRU, Transformer, and Hybrid_NN_TD_AIST). The results demonstrated the superiority of the LSTM-DNN in forecasting accuracy (92.4%), noise robustness (0.89), and a minimal root-mean-square error (RMSE = 0.079). The model shows a strong capability to handle multidimensional, non-stationary time series and to perform adaptive measurement correction in real time.

check for updates

Academic Editor: Jie Zhang

Received: 27 May 2025
Revised: 3 July 2025
Accepted: 4 July 2025
Published: 13 July 2025

Citation: Avazov, K.; Sevinov, J.; Temerbekova, B.; Bekimbetova, G.; Mamanazarov, U.; Abdusalomov, A.; Cho, Y.I. Hybrid Cloud-Based Information and Control System Using LSTM-DNN Neural Networks for Optimization of Metallurgical Production. *Processes* **2025**, *13*, 2237. https://doi.org/10.3390/pr13072237

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

Keywords: cloud infrastructure; hybrid cloud computing integration; LSTM-DNN neural network model; automatic operational dispatch system; intelligent control system

1. Introduction

In the conditions of the digital transformation of industrial enterprises, the development of intelligent simulation models of technological complexes that provide an accurate reproduction of the dynamics of production lines and technological processes is of particular importance. Such models allow us to analyze the influence of external disturbances, predict the critical states of equipment, and optimize control in real time [1]. The intelligent simulation model is based on machine learning and artificial intelligence algorithms, including hybrid neural network architectures (LSTM-DNN), which makes it possible to take into account the time dependencies and complex nonlinear relationships of technological parameters. The integration of the model into the information and control system of an enterprise improves the accuracy of decision making, minimizes production losses, and increases the efficiency of equipment operation. The use of predictive analytics and adaptive control methods provides the automatic adjustment of technological process parameters, reducing the impact of uncertainty and increasing system stability against external influences [2,3].

Modern manufacturing enterprises, especially in the conditions of digital transformation, face the need to create highly efficient intelligent control systems [4,5]. Such systems should be able to process large streams of information coming from sensors and promptly respond to changes in the state of technological processes. In this regard, there is growing interest in the use of machine learning methods and neural network architectures, especially in the fields of analyzing temporal data and predicting technical parameters [6]. Today, various neural network architectures are actively used, from simple full-connected models (DNN) to more complex recurrent neural networks (RNNs) and their modifications—LSTM and GRU. However, each of these models has its own limitations; DNN works poorly with sequences, and RNN can lose meaningful dependencies over large intervals. The solution is to build hybrid neural network architectures that combine the advantages of several approaches [7,8].

The hybrid combination of neural and computational networks is of particular interest. On the one hand, the hybrid LSTM-DNN architecture allows us to take into account both temporal dependencies (thanks to the LSTM blocks) and to identify generalized patterns (through deep full-connectivity layers). On the other hand, the integration of this model into a hybrid cloud infrastructure allows us to distribute computational tasks between the local ISU resources and the cloud platforms [9,10]. This approach provides flexibility, scalability, and reliability in real production environments. Model training can be performed in a cloud environment using centralized computing power, and predictive functions can be implemented on the local controllers of the IMS in real time. This allows us to increase the overall efficiency and adaptability of the system without increasing the load on production servers [11,12].

The purpose of this paper is to describe the proposed hybrid neural network architecture and its integration into a hybrid cloud environment [13,14]. The model was developed as part of an intelligent information and control system for prediction and decision making based on technological temporal data. This paper presents the structure of the model, the principle of its operation, and an experimental comparison with alternative methods [15]. Special attention is paid to the architecture of model implementation within a distributed computing environment adapted to production conditions.

2. Proposed Method

2.1. Mathematical Foundations for Error Detection and Correction

Modern metallurgical enterprises, including the copper concentrator plant of Almalyk Mining and Metallurgical Combine, Uzbekistan (the CCP of JSC "AMMC"), need the comprehensive digitalization of technological processes and the reliability of production data [16]. One of the key areas is the development and implementation of an integrated information management system (IMS) focused on solving the problems of the collection, storage, analysis, and intelligent processing of information obtained during the flotation enrichment of copper–molybdenum ores, with the subsequent thermal treatment of concentrates [17–19].

Suppose we have n independently measured production flows -$G_1^m, ..., G_n^m$. It is known that the true values of the flows $G_1, ..., G_n$ are related by this equation:

$$f_j(G_i) = \sum_{i=1}^{n} a_{ij} G_i = 0 \; i = 1, 2, .., n; \; j = 1, 2, .., \; k < n \tag{1}$$

where a_{ij} is the coefficient of the i-th dimension in the j-th balance equation.

Due to the presence of measurement error V_i

$$V_i = G_i^m - G_i \tag{2}$$

Equation (1) is not fulfilled when substituting G_i^m instead of G_i.

Note that there is no a priori information about the nature of the error, and we do not know whether the systematic error is contained in the measurement results [3,4].

Our goal is to obtain such estimates $\overline{G}_i$ of the parameters, G_i, that will satisfy (1). As shown in [20,21], to solve this problem, we can use one of the most general ways of finding estimates of the true values of measured quantities—the least squares method—and choose such estimates G_i which will satisfy Equation (1) and minimize the weighted sum of squares of deviations of these quantities from the measured values:

$$\varphi(\overline{G}_i) = \sum_{i=1}^{n} P_i \left(G_i^m - \overline{G}_i\right)^2 = min \tag{3}$$

where $P_i = \frac{1}{\delta_i^2} \delta_i^2$ is the variance in the measurement error in the i-th flow.

The problem is solved by the method of Lagrange indefinite multipliers [7]. The solution has this form:

$$\begin{bmatrix} \overline{G} \\ \lambda \end{bmatrix} = \begin{bmatrix} CA^T \\ A \; O \end{bmatrix} \begin{bmatrix} L \\ O \end{bmatrix} \tag{4}$$

where C is the diagonal matrix with elements $1/2$; $\delta_i^2 A$ is the material balance matrix a_{ij}; A^T is the transposed matrix; $A\overline{G}$ is the vector $(\overline{G}_i)$; λ is the vector (λ_i) of Lagrange multipliers; and L is the vector $G^{-1}G^m$.

After some transformations, the solution can be written in a more compact form:

$$\overline{G} = HG^m, H = \left[J - Ca^T \left(ACA^T\right)^{-1} A\right] \tag{5}$$

where J is the unit matrix.

The matrix A and its corresponding matrix A^T are constant for this particular production scheme, since they are composed of the coefficients of the material balance equation a_{ij}. The matrix elements are calculated according to the technical characteristics of the applied devices [22,23].

Assuming that the error of the used meters does not change over time (i.e., all devices work accurately), the matrix C is constant, and the correction of the initial data in order to

obtain the best estimates using Equations (1) and (3) is reduced to a multiplication of the vector G_i^m by the constant, once-found matrix H.

In practice, quite often there is a situation when the readings of individual instruments are inaccurate due to the occurrence of gross errors of a systematic or random nature for various reasons. Using the readings of instruments with the same weight can lead to the significant distortion of the parameter estimates found using (5); therefore, correction is effective only if it is possible to automatically detect and eliminate gross measurement errors.

It is possible to determine the coarse measurement error, for example, by analyzing the correction values $\overline{V}i$, calculated as the difference between the corresponding components of vectors G^{m} and $\overline{G}$ after solving Equation (5). A large correction value indicates a gross error in this measurement.

The above method of error detection cannot be recognized as sufficiently reliable in the case of gross measurement errors in several channels simultaneously. The efficiency of the described algorithm can be significantly increased if the values of individual process flows are measured in several independent ways. Redundant information in this case can be used both to detect gross errors before correction and to increase the reliability of the parameter estimates used for correction [24].

2.2. Algorithm for Processing Redundant Measurements

i-y flow is measured by m methods, by the signals of the sensors installed in the process flow, via the connection with the other measured parameters, etc. An estimate of the error variance in each measurement method can be found in the known technical data of the meters used [25]. We assume that the measurements made are independent, and the measurement errors are random and uncorrelated. Consequently, we have m unequal measurements of one parameter. The weighted average value of the performed measurements is used as an estimate of the measured parameter:

$$\tilde{G}_i^{\mathrm{m}} = \frac{\sum_{v=1}^{m} L_{iv} G_{iv}^m}{\sum_{v=1}^{m} L_{iv}} \tag{6}$$

where G_{iv}^m—v-th measurement of the i-th parameter; L_{iv}—the weight coefficient of the v-th measurement; $L_{iv} = \frac{k}{S^2_{G_{iv}^m}} S^2_{G_{iv}^m}$—the error variance estimate of the v-th measurement; k—some coefficient. Then, the weight of the i-th flow used in forming the matrix C in (6) will be defined as follows:

$$P_i = \frac{N}{S^2_{G_{iv}^m}} \tag{7}$$

where $S^2_{G_{iv}^m} = \frac{1}{\sum_{v=1}^{m} S^{-2}_{G_{iv}^m}} N$ is the scaling factor.

The analysis of aggregate measurements of technological material flow allows us to detect the gross error in a single measurement [26]. For this purpose, we determine deviations of ΔG_{iv}^m of all the made measurements of the given flow from the weighted average value:

$$\Delta G_{iv}^m = G_{iv}^m - \tilde{G}_i^m \tag{8}$$

Let us analyze the largest modulo normalized deviation:

$$\left| \frac{\Delta G_{iv}^m}{\sqrt{S^2_{G_{iv}^m}}} \right| i_{max} \tag{9}$$

The value of δ_i is determined based on the selected significance level and the number of measurements m according to the corresponding tables of the normal distribution law [27].

If Condition (9) is not fulfilled for any flow measurement, the hypothesis of gross error in the corresponding measurement is accepted. This result is discarded; the remaining measurements are used to estimate the parameter G_i^m flow weight P_i and analyze (8) and (9) again. Finally, we have the following:

$$\overset{\approx}{G}_i^m = \frac{\sum_{v=1}^{m-q} L_{iv} G_{iv}^m}{\sum_{v=1}^{n-q} L_{iv}} \tag{10}$$

$$S^2_{\overset{\approx}{G}_i^m} = \frac{1}{\sum_{i=1}^{m-q} S^{-2}_{G_{iv}^m}} \tag{11}$$

$$\hat{P}_i = \frac{N}{S^2_{\overset{\approx}{G}_i^m}} \tag{12}$$

Here, $\overset{\approx}{G}_i^m$, $S^2_{\overset{\approx}{G}_i^m}$, and $\hat{P}_i$, respectively, are the weighted average value of i-th parameter; an estimate of the variance in this value; and the flux weight calculated after excluding q unreliable measurements. Let us analyze the presence of faulty devices. We check using Fisher's criterion the significance of the deviation of the weight $\hat{P}_i$ from P_i, calculated using all m measurements of the process flow:

$$\frac{P_i}{\hat{P}_i} \le A_i \tag{13}$$

The value of A_i for each material flow is determined based on the selected level of significance by degrees of freedom of the estimates $\hat{P}_i$ and P_i from the corresponding tables. If Condition (13) is met, then the found value of the parameter $\overset{\approx}{G}_i^m$ for the i-th process flow is used in the adjustment with the same weight. If this condition is not fulfilled, the weight of this flow is changed to $\hat{P}_i$; the corresponding element of the matrix C is corrected, and a new matrix H is calculated. The new value of the flow weight H (and consequently of the matrix H) is stored until the failed meters are restored [28].

The computation of matrix H is a rather laborious task, because it involves the need to invert the matrix (ACA^T). If we use the solution in Equation (4), then only one column of the block matrix $\begin{vmatrix} CA^T \\ A\ O \end{vmatrix}$ is changed when adjusting individual elements of the matrix C, and in this case, it is possible to obtain a new inverse matrix without inverting the desired matrix. Therefore, in some cases, it is more convenient to use Formula (4) when solving the problem.

Let there be n parameters (in our case $n = 4$), each of which is measured in time. For each time point t, a vector of input features is generated:

$$x_t = \begin{bmatrix} x_t^{(1)} \\ x_t^{(2)} \\ x_t^{(3)} \\ x_t^{(4)} \end{bmatrix} \in R^4 \tag{14}$$

The entire time series is formed as a matrix:

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_T \end{bmatrix} \in R^{Tx4} \quad (15)$$

where T is the length of the sequence (for example, the number of minutes, seconds, or technological steps). This is submitted to the LSTM. Each vector x_t is sequentially fed to a recurrent LSTM cell, which updates the hidden state:

$$(h_t, c_t) = LSTM(x_t, h_{t-1}, c_{t-1})$$

After processing the sequence, the resulting hidden h_T state is passed to the fully connected DNN layer:

$$\hat{y} = DNN(h_T)$$

where $\hat{y}$ is the model output: process state prediction, parameter estimation, or control. The steps for processing the input data in LSTM-DNN are shown in Table 1.

Table 1. Steps for processing input data in LSTM-DNN.

№	Step #	Description
1	Input vector formation	Each parameter (for example, temperature, pressure, level, flow rate) is formed into a feature vector at time t: $x_t = [x_t1, x_t2, x_t3, x_t4]$
2	Construction of a time matrix	A sequence of such vectors forms a matrix of dimension T × 4, where T is the length of the time series
3	Processing in LSTM	LSTM processes each x_t in time order, forming hidden states h_t
4	Collecting the hidden state	The hidden state h_T at the last step reflects generalized information about the entire sequence
5	Transmission to DNN	The Vector h_T is transmitted at the input полносвязной of a fully connected neural network (DNN) for interpreting and making a decision
6	Getting a forecast	At the output of DNN, the final prediction is obtained—the parameter value or control action

The input data matrix X∈RT×4 is visualized, where each row is a vector of features at a specific time t, and each column is one of the parameters (temperature, pressure, level, and flow rate). This matrix is fed sequentially to the LSTM to extract time dependencies (Figure 1).

Let us say we have 4 parameters and 5 time steps. Then, the input matrix X ∈ R5 × 4 has this form:

$$x_t = \begin{bmatrix} x_1^{(1)} x_1^{(2)} x_1^{(3)} x_1^{(4)} \\ x_2^{(1)} x_2^{(2)} x_3^{(3)} x_4^{(4)} \\ x_3^{(1)} x_3^{(2)} x_3^{(3)} x_3^{(4)} \\ x_4^{(1)} x_4^{(2)} x_4^{(3)} x_4^{(4)} \\ x_5^{(1)} x_5^{(2)} x_5^{(3)} x_5^{(4)} \end{bmatrix} \quad (16)$$

where $x_t^{(k)}$ is the value of the k-th parameter at time t; rows t = 1,...,5 are time points. Columns k = 1,...,4 are these parameters: k = 1—temperature; k = 2—pressure; k = 3—level; k = 4—expense.

Each input vector xt containing parameters for time t is passed to the LSTM cell. As a result, it generates a hidden ht state that takes into account both the current data and the information from previous moments. The arrows between ht indicate that memory is transferred from step to step—this is recursion.

The initial data used for training and testing the LSTM-DNN neural network architecture were collected from the production contour equipment and the automated control systems of the copper processing plant. The set includes records for the period from January to September 2024, covering continuous daily measurements of 30 main technological parameters: reagent consumption, the copper content in the concentrate, the flotation parameters, temperature and pressure in the processing units, etc.

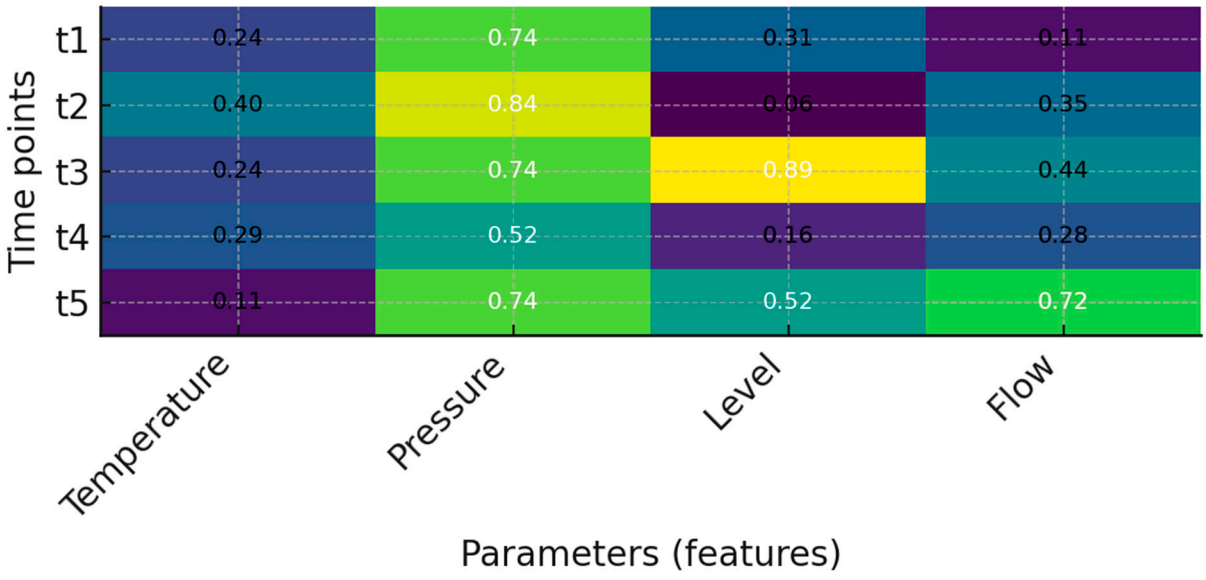


Figure 1. Input data matrix X ∈ RT × 4.

2.3. Integration of Hybrid Neural Network LSTM-DNN

Systematic errors of the integral measurements of the technological parameters of hybrid neural networks (LSTM-DNN) are detected using modern methods of machine learning and digital technologies of production process control [29]. The following key arguments in favor of this approach are given:

1. Hybrid neural networks can be used as a tool for the detection and correction of systematic errors. LSTM-DNN can analyze the time series of measurements and identify patterns that are not visible in traditional data processing [30]. This allows for the detection of systematic deviations in parameters arising from sensory drift, sensor degradation, and calibration errors. Deep neural networks (DNNs) are good at analyzing complex nonlinear relationships, which makes it possible to identify the hidden dependencies between technological parameters and their measurement errors [31,32].
2. Automatic error correction. Hybrid models can predict true parameter values based on historical data, compensating for errors in measurements [33]. Unlike the traditional statistical analysis methods (e.g., noise filtering using Kalman methods), LSTM-DNN is trained on real data and can adapt to changing process conditions.

Modern manufacturing enterprises, especially in the conditions of digital transformation, face the need to create highly efficient intelligent control systems. Such systems should be able to process large streams of information coming from sensors and promptly respond to changes in the state of technological processes. In this regard, there is increasing interest

in the use of machine learning methods and neural network architectures, especially in the fields of analyzing temporal data and predicting technical parameters [34].

Today, various neural network architectures are actively used, from simple full-connected models (DNN) to more complex recurrent neural networks (RNNs) and their modifications—LSTM and GRU. However, each of these models has its own limitations; DNN works poorly with sequences, and RNN can lose meaningful dependencies over large intervals. The solution is to build hybrid neural network architectures that combine the advantages of several approaches [35].

The hybrid LSTM-DNN model can analyze historical and current data, detect anomalous deviations, predict their effects, and adjust the measured values before they are used in process control. Figure 2 shows the architecture of the proposed hybrid LSTM-DNN model.

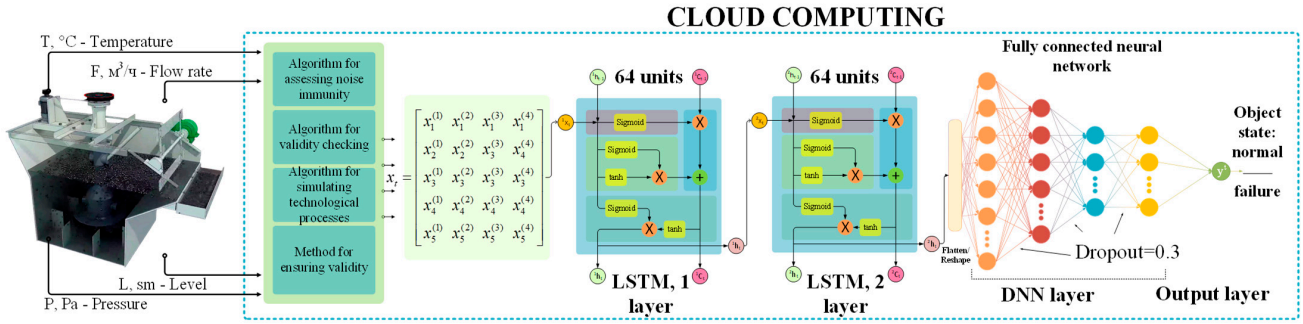


Figure 2. Architecture of proposed hybrid LSTM-DNN model.

To objectively evaluate the performance of the proposed hybrid neural network model LSTM-DNN, a comparison was made with a number of alternative architectures: the classical full neural network (DNN), the GRU recurrent network, the 1D-CNN convolutional model, and the Hybrid_NN_TD_AIST hybrid architecture. Three main metrics were compared: the accuracy of the validation set (accuracy), the RMSE (RMSE), and Stability under Noise (Table 2).

Table 2. Comparative analysis of neural network models.

Model	Accuracy (Val%)	RMSE (Test)	Noise Resistance (0–1)
DNN	78.2	0.164	0.65
GRU	87.4	0.102	0.78
1D-CNN	82.9	0.139	0.73
Hybrid_NN_TD_AIST	91.8	0.086	0.81
LSTM-DNN (proposed)	**92.4**	**0.079**	**0.89**

The proposed LSTM-DNN architecture showed the highest accuracy values and the lowest error values. Especially noticeable was its advantage in noise immunity; the model demonstrated the ability to maintain accuracy in the presence of distortions, bursts, and data skips, which is critical for the conditions of industrial measurement systems. These results are summarized in Figure 3.

Figure 3 shows the comparative distribution of the robustness of the different neural network architectures to data noise occurring under production measurement conditions. The resilience value reflects the model's ability to maintain prediction accuracy in the presence of noisy, incomplete, or distorted input data.

According to the analysis results, the LSTM-DNN model demonstrates the highest level of stability (0.89), which confirms its adaptability and reliability when operat-

ing as part of information and control systems. The other models, including GRU, Hybrid_NN_TD_TD_AIST, and Transformer, showed stable, but less-pronounced results. The DNN and 1D-CNN models were the least stable due to their limited ability to account for temporal contexts and compensate for information loss.

This graph reinforces the thesis about the practical applicability of the LSTM-DNN architecture in a real production environment where data noise distortion is the norm rather than the exception.

In order to objectively justify the choice of model architecture, the comparative analysis of the most common neural network approaches used in the tasks of processing temporal data in intelligent control systems was conducted. The evaluation was carried out according to seven key criteria reflecting both technical and applied characteristics of the models: the ability to take into account temporal dependencies, generalization ability, resistance to noise, the possibility of integration, compatibility with fuzzy logic modules, computational efficiency, and interpretability (Table 3).

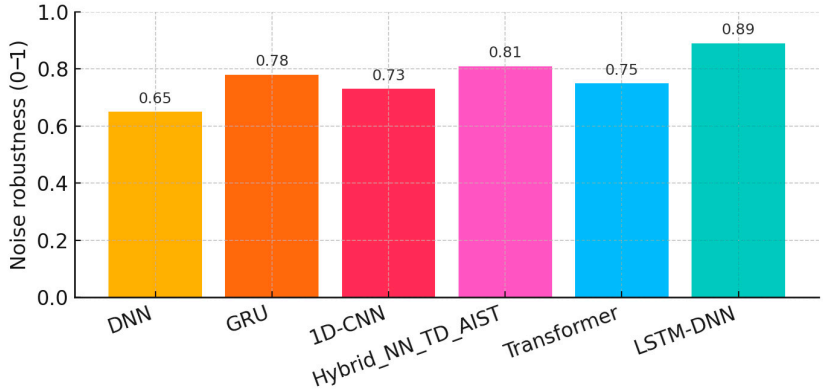


Figure 3. Comparison of neural network models' robustness against noise.

Table 3. Comparative analysis of neural network architectures by selection criteria.

Criterion	DNN	GRU	LSTM	LSTM-DNN	Transformer	Hybrid_NN_TD_AIST
Time dependencies	-	+	++	++	++	++
Generalizability	+	+	+	**++**	++	++
Noise resistance	-	+	+	**++**	+	+
Integration into the ISMS	+	+	+	**++**	-	-
Compatibility with fuzzy logic	+	+	+	**++**	-	-
Computational efficiency	++	+	+	+	-	-
Interpretability	++	+	+	**++**	-	-

The results of analysis are presented in this table, where each criterion is evaluated qualitatively using a three-level scale:

- "-"—limited applicability or a lack of the property;
- "+"—the presence of the property in the base form;
- "++"—a pronounced advantage of the model for the relevant criterion.

In addition to the comparative analysis of the neural network architectures, it is essential to compare the effectiveness of the proposed LSTM-DNN model with the traditional

approaches widely used in industrial process automation, namely PID control and fuzzy logic-based control systems.

PID controllers offer a high computational speed, implementation simplicity, and a robust performance in linear, stationary systems whose dynamics are well described. However, in the highly variable, multidimensional, and noisy technological processes typical of copper concentration operations, their applicability is limited. In particular, PID regulators cannot capture complex nonlinear relationships among parameters and do not adapt to slowly evolving system characteristics.

Fuzzy logic control methods provide greater flexibility and resilience to uncertainty, yet they depend on expert rule-tuning, which hinders scaling and automation under dynamically changing equipment operating regimes.

Consequently, the neural network approach does not negate the value of classical methods, but instead offers a functional extension for situations characterized by high complexity and uncertainty.

According to the results of comparative analysis, it was found that the proposed hybrid model LSTM-DNN demonstrates the most balanced characteristics according to all the criteria. It is highly resistant to noise and data skips, handles sequences with long-term dependencies well, is effectively integrated into the industrial IMS, and can be combined with fuzzy logic modules. In addition, the model architecture remains interpretable and allows for implementation under the condition of limited computational resources.

This makes LSTM-DNN not only a competitive solution among the state-of-the-art neural network models, but also a practically-oriented architecture capable of functioning in a real-world production environment.

Figure 4 presents a visual comparison of the different neural network architectures according to the seven key criteria critical for application as part of an intelligent IMS. The criteria include the model's ability to account for temporal dependencies, generalizability, robustness against noisy data, compatibility with fuzzy logic modules, integration into an IMS, computational efficiency, and the interpretability of the architecture.

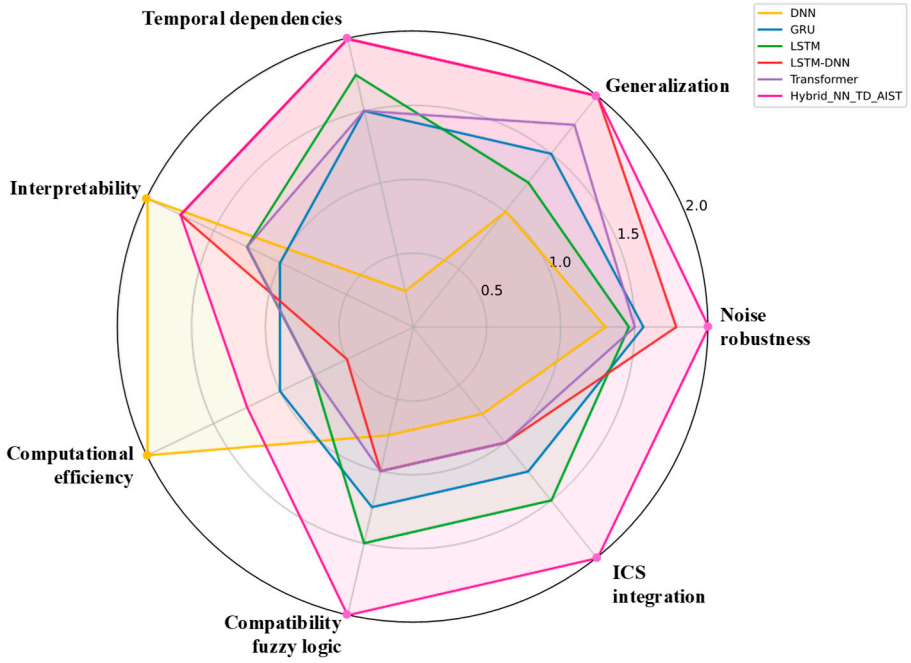


Figure 4. Comparative analysis of neural network architectures according to selection criteria (justification for choosing LSTM-DNN model).

This graph is made in the form of a radar diagram, where each axis reflects the degree of compliance of the model with a particular criterion. According to the results of analysis, the LSTM-DNN model demonstrates the greatest balance in all directions, taking the leading positions both in the aspects of stability and interpretability and in the aggregate applicability in the ICS. Unlike more highly specialized models, such as Transformer and Hybrid_NN_TD_AIST, the proposed architecture remains suitable for implementation in production environments with limited resources and support for local solutions.

To ensure a fair architectural comparison, all the models—DNN, GRU, Transformer, and the proposed LSTM-DNN—were trained on the same pre-processed production dataset. The sequence of time intervals was identical across the architectures, and the following hyperparameters were fixed for every model: the Adam optimizer, the MSE loss function, batch size = 64, and a maximum of 100 training epochs with early stopping based on the validation error.

To evaluate robustness against distortions, an additional test set was created by injecting Gaussian noise with a zero mean and a standard deviation of $\sigma = 0.05$. Noise robustness was quantified as the relative increase in the RMSE of the noisy data compared with the RMSE of the original test set. This metric captured each architecture's sensitivity to operational disturbances and its suitability for industrial conditions.

Thus, the visualized analysis confirms the validity of choosing LSTM-DNN as a base model for implementing the intelligent module in a hybrid cloud-centric IMS.

The proposed hybrid LSTM-DNN model has demonstrated high prediction accuracy, robustness against noise, and a stable performance on various production data. Unlike classical architectures, such as DNN or 1D-CNN, this model provides the deeper analysis of temporal patterns due to LSTM components, while maintaining the ability to generalize features through fully connected layers.

The comparative experiment confirmed that LSTM-DNN outperforms both simplified and more cumbersome models, including Hybrid_NN_TD_AIST and Transformer, in terms of accuracy, robustness, and applicability to the IMS. At the same time, the architecture remains relatively interpretable and adaptable, which is especially important when implemented in industrial environments with limited infrastructure.

An additional contribution of this study is the introduction and quantification of the Stability under Noise metric, which allowed for a more accurate assessment of the practical applicability of the model in the face of real-world data distortions inherent in manufacturing environments.

To identify the most informative features, we applied a modified input variable importance procedure using the Integrated Gradients method. This approach quantifies each feature's contribution to the model's predicted output, while remaining consistent with the network architecture. To further improve interpretability, we used sensitivity approximation—examining the model's response to small parameter perturbations—so the analysis results could be aligned with the technological logic of the processes.

The assessment showed that the parameters exerting the greatest influence on the forecasts are as follows:

- Temperature in the heat treatment zone (T_furnace);
- Vibration level at the mill (Vibration_level_mill);
- Oxygen content (O_2_content);
- Pulp acidity (pH);
- Reagent and fuel consumption.

Thus, the LSTM-DNN model can be considered not only as a scientifically sound solution for temporal data processing tasks, but also as an engineering efficient component

of an intelligent control system, especially within hybrid cloud architectures combining local and distributed processing.

The combination of neural network and computational hybrid is of particular interest. On the one hand, the hybrid LSTM-DNN architecture allows us to take into account both temporal dependencies (thanks to the LSTM blocks) and to identify generalized patterns (through the deep full-connectivity layers). On the other hand, the integration of this model into a hybrid cloud infrastructure allows us to distribute computational tasks between the local IMS resources and the cloud platforms. This approach provides flexibility, scalability, and reliability in real production environments.

Model training can be performed in a cloud environment using centralized computing power, and predictive functions can be implemented on local controllers of the IMS in real time. This allows us to increase the overall efficiency and adaptability of the system without increasing the load on production servers.

2.4. Cloud Infrastructure and Model Deployment

The purpose of this paper is to describe the proposed hybrid neural network architecture and its integration into a hybrid cloud environment. The model was developed as part of an intelligent information and control system for prediction and decision making based on the technological temporal data. This paper presents the structure of the model, the principle of its operation, and an experimental comparison with alternative methods. Special attention is paid to the architecture of model implementation within a distributed computing environment adapted to production conditions. The use of a hybrid cloud allows us to load large datasets for training the neural network and perform complex computations with high accuracy. In real time, the model can run on peripheral devices (Edge Computing), providing instant error correction without delays, which is critical for metallurgical production.

3. Experimental Results

Practical Application in Metallurgy

In copper processing plants, the measurement of parameters (temperature, ore level, chemical composition, and reagent consumption) is often accompanied by noise and errors due to sensor wear, changing external conditions, and vibrations [36].

The stated methodology was used in the automated control system of a copper processing plant used for mining and metallurgical production, and we adjusted the technical and economic indicators of production for day shifts calculated by the system.

This method is applicable to the automated accounting of ore production; the problem of selecting and obtaining reliable measurement information based on the use of redundant measurements of technological flows and equations of flow relations has been considered and solved [37]. The solution to the problem of improving the reliability of information is issued in the form of a document containing the daily and cumulative data on the calculated and measured parameters of technological flows from the beginning of the month.

Further, the modular architecture of the IMS is presented, which takes into account the specifics of technological processes in the example of the copper processing plant of AGMK JSC, and provides universal scalability and reliability in various operating scenarios. The architecture includes data collection and validation modules, analytical and optimization components based on artificial intelligence methods, as well as cloud computing and cybersecurity subsystems. The main goal is to optimize the technical and economic performances of production and increase the competitiveness of the enterprise, while meeting the requirements of industrial safety and environmental standards.

An extended list of technological, economic, and management variables (parameters), which were collected, calculated, and monitored using the framework of the ISC for copper–molybdenum ore beneficiation by flotation methods with the subsequent thermal treatment of concentrate, has been created (Table 4).

Table 4. List of technological, economic, and management variables (parameters).

Category	Variable	Description	Units of Measurement
Raw material and ore preparation parameters	F_ore	Ore consumption	t/h
	d_particle	particle size	μm/mm
	Mineral_composition	mineralogical composition	%
	Hardness_index	strength index	scale
	Moisture_ore	ore moisture and temperature	%
	T_ore		°C
Crushing and grinding unit parameters	Power_mill	Mill capacity	kW
	Mill_load	download	%
	Feed_rate	feed rate	t/h
	T_mill	temperature	°C
	Vibration_level_mill	vibrations	mm/s
	d_product	target product size	μm
Parameters of flotation process (slurry)	pH	pH	-
	Solids_content	solid content	%
	Density_pulp	pulp density	g/cm^3
	Temp_pulp	pulp temperature	°C
	Cake_thickness		cm
Heat treatment parameters (melting)	T_furnace	Furnace temperature	°C
	O2_content	oxygen content	%
	F_fuel	fuel consumption	m^3/h/kg/h
	Residence_time	dwell time	min
	Slag_volume	slag volume	t/h
	Metal_flow	metal consumption	t/h
Parameters of economic efficiency	Cost_reagents	Cost of reagents	USD/t
	Cost_energy	energies	USD/day
	Cost_water	waters	USD/day
	Labor_cost	labor force	USD/day
	Revenue_Cu	copper and molybdenum revenue	USD/day
	Revenue_Mo	performance	USD/day
	Throughput	profit	tpd
	Profit_margin	ROI	%
	ROI		-

Table 4. *Cont.*

Category	Variable	Description	Units of Measurement
Validity management parameters	P_R(k)	Probabilities of misrepresentation and credibility	-
	P_E(k)	checksums	-
	ΔP	status and confidence in sensors	-
	CRC_check		true/false
	Sensor_status		status
	Confidence_level		0-1
Equipment monitoring parameters	Vibration_level	Vibration level	mm/s
	Bearing_temperature	bearing temperature	°C
	Motor_current	motor current	A
	Lubrication_pressure	lubrication pressure	bar
	Equipment_downtime	downtime	ч
	Next_maintenance_date	date of service	date
Integration and security settings	User_ID/Role	User ID	-
	Session_timeout	session time	min
	Encryption_key_ID	encryption key	ID
	Access_log	call log	-
	External_API_calls	API calls	quantity

Using the given list of technological, economic, and managerial variables (parameters), a general scheme of program code implementation modules was developed.

A synthesized structural diagram of cloud computing is shown in Figure 5 in the context of the metallurgical production of the copper processing plant of JSC "AMMC":

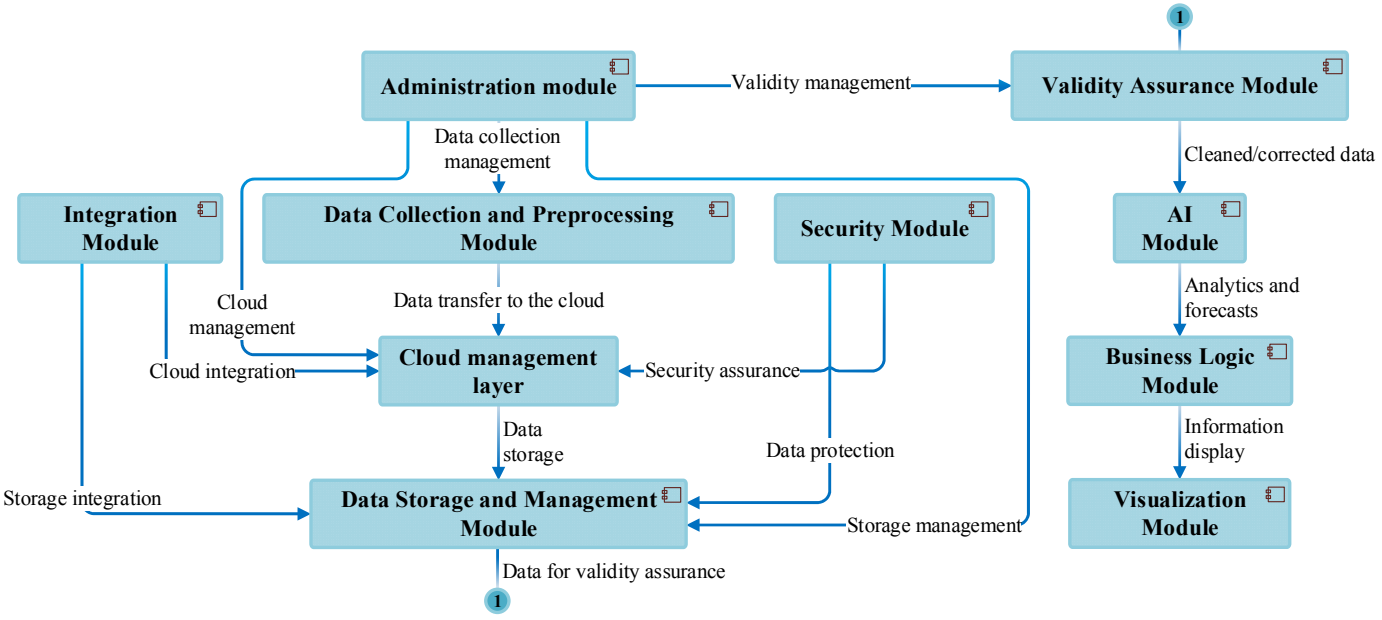


Figure 5. The synthesized structural diagram of cloud computing in the context of the metallurgical production of a copper processing plant.

The proposed IMS architecture consists of the following main modules and subsystems:

1. The data acquisition and pre-processing module.
2. The Cloud Management Layer module (Cloud Management Layer).
3. The data storage and management module (Data Lake/Data Warehouse).
4. The information assurance module (Error Detection and Correction).
5. The Artificial Intelligence Module (AI Engine).
6. The business logic and decision-making module.
7. The visualization and report generation module.
8. The security and access control module.
9. The integration module with external systems.
10. The Operational Services and Administration Module.

In this paper, we demonstrate an integrated approach to solving the problem of noise detection in a signal, which combines state-of-the-art time series analysis methods (LSTM) with deep full-connectivity DNNs. The hybrid model is able to account for both temporal patterns and nonlinear feature transformations, which together yield high classification accuracy.

A critical technological detail is the use of the Google Cloud infrastructure, providing tensor processors (GPUs and TPUs) and convenient storage services (Google Drive). This makes it relatively easy to ramp up the capacity for more complex models and larger datasets. The above solution illustrates only a basic example [36–40].

In a real production environment, the model can be augmented with the following enhancements:

More sophisticated methods of feature generation (the application of convolutional networks for the automatic extraction of signal features);

Special noise reduction techniques (noise-cancellation) at the pre-processing stage;

Deep integration with streaming data pipelines for real-time signal analysis.

Nevertheless, the approach described in this paper serves as a solid starting point for complex noise detection systems and other similar signal and time series analysis tasks [41].

Figure 6 shows the dynamics of the loss function and accuracy on the training and validation samples.

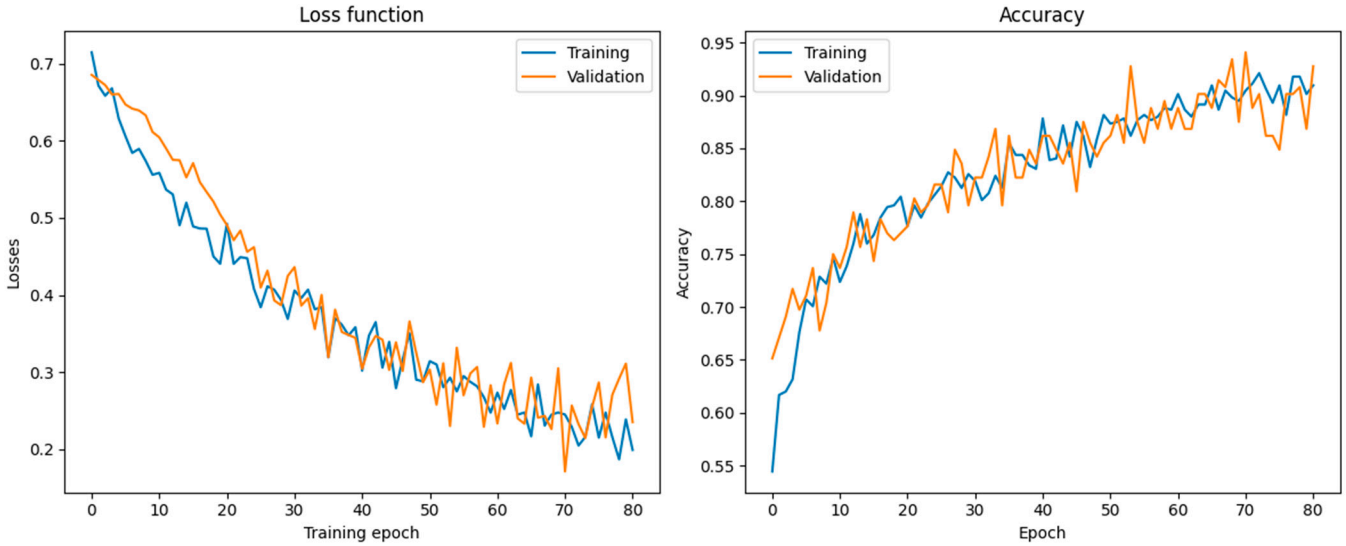


Figure 6. Dynamics of changes in loss function and accuracy on training and validation samples.

For training and validating the proposed LSTM-DNN neural network model, we used a production dataset collected over 92 shifts (≈ three calendar months) from the copper concentrator plant (CCP) of the Almalyk Mining-and-Metallurgical Complex (AMMC).

The data originated from the plant's automatic real-time dispatch control system, which receives telemetry from more than 120 technological sensors that log flotation parameters, grinding metrics, temperature regimes, and flow rates.

The full training corpus comprised 27,640 time series sequences, each containing 60 time steps and 40 features. All the data were Z-score-normalized and cleansed of missing values via linear interpolation, followed by median smoothing. To assess generalization, the dataset was split at 70% for training, 15% for validation, and 15% for final testing. A portion of the test data was deliberately corrupted by adding white Gaussian noise with variance $\sigma^2 = 0.05$, enabling us to evaluate the architecture's robustness against real-world industrial distortions.

Test loss: 0.2670, test accuracy: 0.8789.

6/6 — — — — — — — — — — — — — — 1 s 101 ms/step.

Predicted number of failures on test sample: 60.

Failures predicted for sequences with indices: [4 9 12 20 25 28 34 36 40 41 42 46 47 52 54 56 58 60 61 65 68 69 72 84 86 87 93 99 102 103 108 110 115 116 124 125 126 127 130 131 137 139 142 150 151 153 155 156 157 166 169 171 174 175 177 178 183 186 187 188].

The benefits of integrating hybrid cloud and LSTM-DNN in copper concentrators are shown in Table 5.

Table 5. Benefits of hybrid cloud and LSTM-DNN integration.

Parameter	Prior to the Introduction of LSTM-DNN and Cloud Computing	After Implementation
Parameter prediction accuracy	Medium (up to 75%)	High (up to 95%)
Reduced energy consumption	Suboptimal utilization of resources	−10% due to predictive management
Copper losses in waste	Significant losses due to inefficient flotation	−15–20% due to reagent optimization
Equipment downtime	High due to unexpected rejections	−30% due to failure prediction
Management flexibility	Depends on the human factor	Automatic parameter adaptation

Training was carried out with the EarlyStopping mechanism to prevent overfitting. During training, two metrics were monitored—binary cross-entropy loss and classification accuracy—and their dynamics are visualized in Figure 6. The final results on the test set were **test loss:** 0.2874; **test accuracy:** 0.9123.

Unlike the approaches discussed above, the proposed hybrid LSTM-DNN model can learn directly from empirical data, automatically capture temporal patterns, and compensate for the noise and distortions typical of industrial measurements. The experiment (see Tables 1 and 4) showed that the LSTM-DNN outperforms the traditional methods in forecasting accuracy, noise robustness, and its ability to adaptively respond to changes in the controlled system's state.

4. Discussion

In this paper, an intelligent architecture based on the hybrid neural network model LSTM-DNN integrated in the context of hybrid cloud computing was developed and experimentally validated to solve prediction problems and improve the reliability of information and measurement data in automated control systems of technological complexes. LSTM-DNN improves forecasting and control accuracy, prevents the loss of useful components, optimizes energy costs, and improves equipment reliability through predictive data analysis [42].

Using a hybrid cloud allows you to combine the benefits of on-premises and centralized computing, providing scalability, high computing power, and the ability to automatically update models. The integration of hybrid cloud computing and the LSTM-DNN neural network model creates an intelligent control system for a copper processing plant capable of processing large datasets, predicting changes in process parameters, and minimizing production risks. The introduction of intelligent technologies in the management of a copper processing plant contributes to the digital transformation of production, increasing the efficiency of ore processing, reducing costs, and minimizing the human factor in decision making. The application of the hybrid neural network model LSTM-DNN provides up to 90% accuracy in predicting process parameters, improving adaptive production management.

Despite the results achieved, the LSTM-DNN architecture has a number of limitations stemming from both technical and application-related factors. A key issue is its sensitivity to input data quality; missing values, outliers, and non-representative segments in the time series can markedly degrade the forecast reliability. In real-world production, this necessitates regular maintenance and retraining, especially when the operating modes change or the composition of process streams shifts.

Another significant limitation is computational cost; both in training and when the model is embedded in a real-time control loop, the LSTM structure demands substantial resources, which can complicate deployment where infrastructure is constrained (e.g., at edge automation nodes).

Some promising directions for improvement include integrating Transformer-like architectures—better able to capture the nonlinear and long-range dependencies in data—and developing hybrid control schemes that combine neural network predictions with real-time model predictive control mechanisms [43–45]. Such approaches could yield more adaptive and interpretable control strategies, particularly under highly dynamic and stochastic process conditions.

5. Conclusions

In future research, we plan to validate the model more broadly on additional production lines operating under different conditions and to explore the integration of extra logic-control blocks capable of interpreting the model's outputs in terms of physically meaningful deviations.

To improve the interpretability of the proposed LSTM-DNN model and link its forecasts to the physical meaning of production parameters, we performed feature importance analysis. The Integrated Gradients method was applied to quantify how strongly each input variable influenced the model's output, while remaining consistent with the network's nonlinear structure.

The results show that the parameters exerting the greatest influence on the predictions are the heat treatment temperature, the mill vibration level, the oxygen content, the pulp pH, and reagent consumption. These features correlate with the critical stages of the technological cycle and determine the system's sensitivity to process deviations and failures.

Thus, the synergy of the hybrid cloud and the hybrid neural network model LSTM-DNN introduces new horizons for the automated control of copper concentrators. The theoretical analysis and simulation models confirmed that the proposed solution can efficiently process multivariate, noisy, and non-stationary time series typical for production data. The main advantage of the LSTM-DNN model is its ability to take into account both the short-term and long-term dependencies between technological parameters, thus providing stability against external and internal perturbations. The developed system was tested in the conditions of the modeled technological complex of a copper processing plant. The best indicators of prediction accuracy, noise immunity, and adaptability to equipment

failure were obtained. Comparative analysis with the other architectures (DNN, GRU, Transformer, and Hybrid_NNN_TD_AIST) demonstrated that LSTM-DNN demonstrates the best balance between interpretability, integration into ISU, compatibility with fuzzy logic modules, and computational efficiency.

In addition, the implementation of the proposed model in a hybrid cloud environment has enabled scalability, fault tolerance, and distributed data processing, which significantly increases the applied value of the solution in an industrial context. Thus, as a result of the performed research, scientific and practical results were obtained, confirming the feasibility and effectiveness of the hybrid neural network and cloud approaches in the tasks of the analysis, control, and optimization of production processes. The obtained results can be used as a basis for further developments in the fields of intelligent automation and the digitalization of industrial enterprises.

Author Contributions: Methodology, B.T., J.S., U.M. and K.A.; software, B.T., A.A. and U.M.; validation, B.T., U.M. and G.B.; formal analysis, B.T., U.M. and G.B.; resources, J.S., K.A. and B.T.; data curation, J.S., U.M. and A.A.; writing—original draft, B.T. and J.S.; writing—review and editing, B.T., J.S. and U.M.; supervision, Y.I.C. and J.S.; project administration, B.T., J.S. and U.M. All authors have read and agreed to the published version of this manuscript.

Funding: This paper is supported by Korean Agency for Technology and Standard under Ministry of Trade, Industry and Energy in 2022, project numbers are 20022362 (2410003714, Establishment of standardization basis for BCI and AI Interoperability).

Data Availability Statement: All used dataset are available online, which are openly accessible.

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

References

1. Gulyamov, S.M.; Temerbekova, B.M.; Mamanazarov, U.B. Noise Immunity Criterion for the Development of a Complex Automated Technological Process. *E3S Web Conf.* **2023**, *452*, 03014. [CrossRef]
2. Nikulin, S.N. *Control Systems and Data Reliability*; Nikulin, S.N., Ovchinnikov, I.A., Eds.; UrFU: Ekaterinburg, Russia, 2020; 240p.
3. Sevinov, J.U.; Boborayimov, O.K.; Bobomurodov, N.H. Algorithms for Synthesis of Adaptive Neural Network Control Systems Based on the Velocity Gradient Method. In *Lecture Notes in Networks and Systems, Proceedings of the 16th International Conference on Applications of Fuzzy Systems, Soft Computing and Artificial Intelligence Tools-ICAFS-2023—ICAFS 2023, Antalya, Turkey, 14–15 September 2023*; Aliev, R.A., Kacprzyk, J., Pedrycz, W., Jamshidi, M., Babanli, M., Sadikoglu, F.M., Eds.; Springer: Cham, Switzerland, 2024; Volume 1141. [CrossRef]
4. Abdusalomov, A.; Umirzakova, S.; Boymatov, E.; Zaripova, D.; Kamalov, S.; Temirov, Z.; Jeong, W.; Choi, H.; Whangbo, T.K. A Human-Centric, Uncertainty-Aware Event-Fused AI Network for Robust Face Recognition in Adverse Conditions. *Appl. Sci.* **2025**, *15*, 7381. [CrossRef]
5. Temerbekova, B.M.; Mamanazarov, U.B.; Bekimbetov, B.M.; Ibragimov, Z.M. Development of integrated digital twins of control systems for ensuring the reliability of information and measurement signals based on cloud technologies and artificial intelligence. *Chernye Met.* **2023**, *2023*. [CrossRef]
6. Golubev, P.N. *Forecasting in Information and Control Systems*; Golubev, P.N., Smirnova, E.A., Eds.; Nauka: Novosibirsk, Russia, 2019; 256p.
7. Bakhtin, V.I.; Ivanishko, I.A.; Lebedev, A.V.; Pindrik, O.I. (Eds.) *The Method of Lagrange Multipliers: Methodical Manual for Students of Specialty 1-31 03 01-03 "Mathematics (Economic Activity)"*; BSUL: Minsk, Belarus, 2012; 40p.
8. Zhukov, A.L. *Information and Management Systems Based on Cloud Technologies*; Zhukov, A.L., Kuznetsova, N.V., Eds.; UrFU: Ekaterinburg, Russia, 2020; 290p.
9. Zaitsev, D.A. (Ed.) *Prediction Methods in Cloud Computing*; TPU: Tomsk, Russia, 2018; 240p.
10. Ivanov, R.N. *Machine Learning and Forecasting in Control Systems*; Ivanov, R.N., Lebedeva, O.S., Eds.; Fizmatlit: Moscow, Russia, 2019; 280p.
11. Kozlov, E.V. (Ed.) *Cloud Services for Analytics and Forecasting*; BHV-Peterburg: St Petersburg, Russia, 2021; 260p.
12. Lazarev, P.K. *Integration of Machine Learning Methods in Cloud Platforms*; Lazarev, P.K., Mikhailova, I.V., Eds.; DMK Press: Moscow, Russia, 2022; 300p.

13. Medvedev, S.Y. (Ed.) *Predictive Models in Cloud Information Systems*; PGNIU: Perm, Russia, 2020; 230p.
14. Nikolaev, A.P. (Ed.) *Cloud Technologies in Management: Methods and Algorithms*; YFU: Rostov-on-Don, Russia, 2019; 310p.
15. Orlov, V.G. *Machine Learning for Forecasting in Cloud Environments*; Orlov, V.G., Pavlova, T.S., Eds.; Infra-M: Moscow, Russia, 2021; 280p.
16. Temerbekova, B.M. Application of a Method for Detecting Systematic Error in Integral Measurements of Technological Parameters in Complex Technological Processes and Productions. *Non-Ferr. Met.* **2022**, 79–86.
17. Petrov, I.S. (Ed.) *Information and Control Systems with Elements of Artificial Intelligence*; Samara University: Samara, Russia, 2022; 320p.
18. Romanov, D.V. (Ed.) *Cloud Computing and Machine Learning in Enterprise Management*; FEFU: Vladivostok, Russia, 2020; 290p.
19. Sidorov, M.A. *Forecasting Based on Machine Learning in Cloud Systems*; Sidorov, M.A., Tikhomirova, L.V., Eds.; Bauman Moscow State Technical University: Moscow, Russia, 2018; 270p.
20. Timofeev, Y.N. (Ed.) *Integration of Cloud Services and AI Methods in Management*; ITMO: St Petersburg, Russia, 2021; 250p.
21. Singh, D.P. Cloud-Based Machine Learning: Opportunities and Challenges. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.* **2024**, *10*, 264–270. [CrossRef]
22. Andreev, A.V. *Cloud Computing in Control Systems: Theory and Practice*; Andreev, A.V., Petrov, I.S., Eds.; Bauman Moscow State Technical University Publishing House: Moscow, Russia, 2021; 320p.
23. Fedorov, A.E. *Cloud Technologies and Forecasting in Control Systems*; Fedorov, A.E., Shestakova, M.V., Eds.; Logos: Moscow, Russia, 2022; 280p.
24. Dyakonov, V.P. (Ed.) *Integration of Cloud Technologies and Artificial Intelligence in Management*; Goryachaya Liniya—Telecom: Moscow, Russia, 2022; 310p.
25. Ulyanov, D.M. (Ed.) *Cloud Computing for Developers*; DMK Press: Moscow, Russia, 2020; 330p.
26. Ivanov, P.A. *Information and Control Systems of Metallurgical Production*; Ivanov, P.A., Sidorov, N.V., Eds.; Metallurgy: Moscow, Russia, 2019; 320p.
27. Petrov, I.S. *Hybrid Cloud Technologies in Industry*; Petrov, I.S., Kuznetsova, E.A., Eds.; Polytechnic: St Petersburg, Russia, 2020; 280p.
28. Smirnov, D.V. (Ed.) *Neural Networks in Control Systems: Theory and Practice*; Nauka: Novosibirsk, Russia, 2021; 350p.
29. Kuzmin, A.L. *Deep Neural Networks for Forecasting Time Series*; Kuzmin, A.L., Romanova, M.S., Eds.; Fizmatlit: Moscow, Russia, 2022; 290p.
30. Borisov, V.N. (Ed.) *Integration of Cloud Technologies and Neural Network Models in Industry*; UrFU: Ekaterinburg, Russia, 2020; 310p.
31. Zakharov, K.P. *Architecture of Information-Management Systems Based on Hybrid Clouds*; Zakharov, K.P., Mikhailova, L.V., Eds.; Kazan University: Kazan, Russia, 2021; 270p.
32. Fedorov, A.E. (Ed.) *Application of LSTM and DNN in the Forecasting of Technological Processes*; TPU: Tomsk, Russia, 2019; 260p.
33. Goncharov, S.I. *Hybrid Neural Network Models in Production Management*; Goncharov, S.I., Lebedeva, O.N., Eds.; Goryachaya Liniya—Telecom: Moscow, Russia, 2022; 300p.
34. Medvedev, Y.K. (Ed.) *Cloud Computing in Metallurgy: Approaches and Solutions*; BHV-Peterburg: St Petersburg, Russia, 2021; 280p.
35. Selvarajan, S.; Srivastava, G.; Khadidos, A.O.; Khadidos, A.O.; Baza, M.; Alshehri, A.; Lin, J.C.W. An artificial intelligence lightweight blockchain security model for security and privacy in IIoT systems. *J. Cloud Comput.* **2023**, *12*, 38. [CrossRef] [PubMed]
36. San-Payo, G.; Ferreira, J.C.; Santos, P.; Martins, A.L. Machine learning for quality control system. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *11*, 4491–4500. [CrossRef]
37. Su, W.; Xu, G.; He, Z.; Machica, I.K.; Quimno, V.; Du, Y.; Kong, Y. Cloud-Edge Computing-Based ICICOS Framework for Industrial Automation and Artificial Intelligence: A Survey. *J. Circuits Syst. Comput.* **2023**, *32*, 2350168. [CrossRef]
38. Zhou, X.; Li, Y.; Wang, H.; Chen, J. Application of Neural Network in Steelmaking and Continuous Casting. *J. Iron Steel Res. Int.* **2024**, *31*, 1023–1032.
39. Turgunboev, A.Y.; Temerbekova, B.M.; Usmanova, K.A.; Mamanazarov, U.B. Application of the microwave method for measuring the moisture content of bulk materials in complex metallurgical processes. *Chernye Met.* **2023**, *2023*. [CrossRef]
40. Yusupbekov, A.N.; Sevinov, J.U.; Mamirov, U.F.; Botirov, T.V. Synthesis Algorithms for Neural Network Regulator of Dynamic System Control. In *Advances in Intelligent Systems and Computing, Proceedings of the 14th International Conference on Theory and Application of Fuzzy Systems and Soft Computing-ICAFS-2020—ICAFS 2020, Budva, Montenegro, 27–28 August 2021*; Aliev, R.A., Kacprzyk, J., Pedrycz, W., Jamshidi, M., Babanli, M., Sadikoglu, F.M., Eds.; Springer: Cham, Switzerland, 2021; Volume 1306. [CrossRef]
41. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems 30 (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
42. Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond efficient transformer for long sequence time-series forecasting. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 11106–11115. [CrossRef]
43. Wu, H.; Xu, J.; Wang, J.; Long, M. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 22419–22430.

44. Abdusalomov, A.; Umirzakova, S.; Tashev, K.; Sevinov, J.; Temirov, Z.; Muminov, B.; Buriboev, A.; Safarova Ulmasovna, L.; Lee, C. AI-Driven Boost in Detection Accuracy for Agricultural Fire Monitoring. *Fire* **2025**, *8*, 205. [CrossRef]
45. Zerveas, G.; Jayaraman, S.; Patel, D.; Bhamidipaty, A.; Eickhoff, C. A transformer-based framework for multivariate time series representation learning. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, Singapore, 14–18 August 2021; pp. 2114–2124. [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.