Hindawi
Mobile Information Systems
Volume 2022, Article ID 6975824, 12 pages
https://doi.org/10.1155/2022/6975824

Research Article

Educational Evaluation of Piano Performance by the Deep Learning Neural Network Model

Yuanyuan Liao

School of Music and Dance, Nanning Normal University, Nanning, Guangxi, China

Correspondence should be addressed to Yuanyuan Liao; liaoyuanyuan@nnnu.edu.cn

Received 9 June 2022; Revised 4 July 2022; Accepted 14 July 2022; Published 29 August 2022

Academic Editor: Muhammad Zakarya

In recent years, the piano education industry has occupied a huge market. However, the automatic evaluation function of piano performance has shortcomings in existing piano education products. Deep learning (DL) algorithm and the recurrent neural network (RNN) structure can help in automatics evaluation function of the piano performance. This paper proposes a Musical Instrument Digital Interface (MIDI) piano evaluation scheme based on the RNN structure and the Spark computing engine using the Deeplearning4J DL framework. The Deeplearning4J framework can run on the Java Virtual Machine; therefore, the entire system does not require cross-platform development. The Spark distributed computing engine realizes parallelization in music data preprocessing, feature extraction, and model training. Combined with the training user interface (UI) provided by the Deeplearning4J, it can improve developmental efficiency. Additionally, the RNN parameters are analyzed. The results demonstrate that the error value of the three-layer RNN structure is smaller than other closest rivals' techniques. In particular, few piano training institutions and MIDI website datasets are used as the basis, and the experimental samples are collected. The neural network is trained, and the performance of the evaluation model is tested. The results show that the evaluation outcomes of the designed performance evaluation model for the piano are fundamentally consistent with the real levels of the players with assured feasibility; after 3k times of the training periods, the error of the RNN model is close to 0.01 and the network converges.

1. Introduction

In the 10th year of reform and opening, the piano examination system was restored, and China ushered in the stage of the explosive development of piano education for the first time [1]. In recent years, with the improvement of material living standards, the awareness of spiritual needs has deepened in people's lives, and piano education has experienced a second explosive development. Whether entering an art school for collective study or asking a pianist to teach individual study, these teaching methods have high professional requirements for piano educators. There is a serious shortage of traditional Chinese piano educators, so the contradiction between supply and demand is difficult to solve in the short term [2]. Additionally, the commission of traditional educators is generally high, and the price of wooden pianos is high, which is unaffordable for ordinary families. High cost has become an important obstacle in the development of piano education. Therefore, reducing the cost of learning has become an inevitable trend in popularizing piano education. At present, piano teachers mainly rely on their own understanding to monitor, assess, and precise students' performances in the class. Teachers and students have different perspectives on what constitutes play and music. Not only the proper or wrong note but also significant aspects like rhythm and expressiveness have an impact on the performance [3]. As a result, the conventional approach to teaching piano music contains flaws like strong subjectivity, insufficient judgement, and high unpredictability [4].

The New Generation Artificial Intelligence Development Plan, which the State Council of the People's Republic of China formally released in July 2017, emphasised the use of intelligent technology to accelerate the reform of talent training models and teaching methods and to build a new education system that includes intelligent and interactive learning [5]. The Artificial Intelligence Innovation Action Plan for Colleges and Universities, published in April 2018 by the Ministry of Education, promoted the advancement of intelligent education; explored a new teaching model based on artificial intelligence (AI); rebuilt the teaching process; and utilised AI to monitor the teaching process, analyse learning situations, and diagnose academic level [6]. In recent years, deep learning (DL) neural network, as one of the more important branches in the field of AI technology, has been fully developed, especially in computer vision, machine learning, and other directions, and education is increasingly integrated. The application of DL neural network structure in the field of education shows a trend of rapid growth [7]. Since 2015, various educational applications of AI have emerged, and a number of companies dedicated to empowering education with AI have also emerged. Under the background of the dual promotion of national policies and industry, a number of key technologies of DL neural networks are playing an increasingly important role in the field of education and are gradually being widely used [8].

Under the requirements of the times, the intelligent teaching evaluation mode has become a popular teaching evaluation method that the education community pays attention to. The DL neural network structure is integrated into the evaluation of piano performance education, which can help students obtain a more comprehensive and complete teaching evaluation [9]. This study mainly proposes a convolutional neural network (CNN)-based Musical Instrument Digital Interface (MIDI) piano performance education evaluation method based on the DL neural network structure. The evaluation grades are divided into five grades: excellent, good, medium, poor, and poor to help music teachers understand students' mastery. This study aims to provide important technical support for alleviating the inadequacy of piano coaches and reducing the work amount of piano instructors. Moreover, it can help in realizing automatic error amendment and evaluation of playing objectives and refining the productivity of music teaching for piano. The goal of this study is to offer crucial technical assistance for addressing the shortage of piano teachers, lowering the workload for piano teachers, achieving the objective assessment of piano performance education, and enhancing the effectiveness of piano music instruction. Following is a summary of our work's main contributions:

(I) The deep learning algorithm (DL) and the recurrent neural network (RNN) structure are analyzed;

(II) This study proposes a Musical Instrument Digital Interface (MIDI) piano evaluation scheme that is based on the RNN structure and the Spark computing engine and using the Deeplearning4J DL framework; and

(III) The Spark distributed computing engine realizes parallelization in music data preprocessing, feature extraction, and model training.

The remaining article is organized in the following manner: in section 2, we discuss various methods such as deep learning and neural networks. Besides, forward propagation and back propagation (BP) networks are also discussed. A deep network model construction of Piano performance evaluation based on RNN method is discussed and proposed in subsequent section 3. Discussion over the dataset is also included. Experimental outcomes and discussion are illustrated in section 4. Finally, we summarize the findings of our study in section 5 and offer ways for further research.

2. Methods

2.1. The DL Neural Network Analysis. Geoffrey Hinton first proposed DL in Science magazine, and it was studied by later generations and gradually emerged. The DL is an extension of the original basis of machine learning [10]. Compared with the machine learning network, the DL network optimizes the hierarchical data of the network structure; therefore, making the overall structure more complex. Furthermore, the internal operation algorithm has also undergone greater progress [11]. The most common algorithms are classified according to common machine learning algorithms and DL algorithms. The classification results are shown in Figure 1(a).

In Figure 1(b), DL learns the inherent laws and representation levels of sample data by analyzing the underlying laws and data structure levels within the sample data and uses the data obtained in the learning process to provide reference explanations for data in other fields [12]. The original research goal of the DL algorithm is to apply it in the field of AI, to help AI so that it can analyze and learn like humans, and to recognize various forms of data. The DL has achieved results in several fields. This should be noted that improving the analytical learning ability of AI through DL helps humans solve many complex data problems [13].

DL is a general term for data research patterns and methods. It is classified according to the specific research content. The classification results show that DL neural networks include (i) CNN is a neural network system based on convolution operation; (ii) deep belief network (DBN) performs pretraining in the form of a multilayer self-encoding neural network and further optimizes the neural network's weights by combining the discriminant information; and (iii) recurrent neural network (RNN) is a self-encoding neural network based on multilayer neurons, including autoencoder and sparse coding [14].

2.2. The RNN Structure Analysis. Currently, RNN is one of three types of neural networks that are most commonly used in the field of artificial intelligence. The characteristic of RNN is that the neurons in the hidden layer can communicate with each other. When the next input information is processed, the previous output information also affects it [15]. This memory capacity is beneficial for time series analysis. Therefore, RNN approach is extensively implemented for natural language processing (NLP), speech synthesis, speech recognition, and other fields of optimization. The structure of the basic RNN is shown in Figure 2.

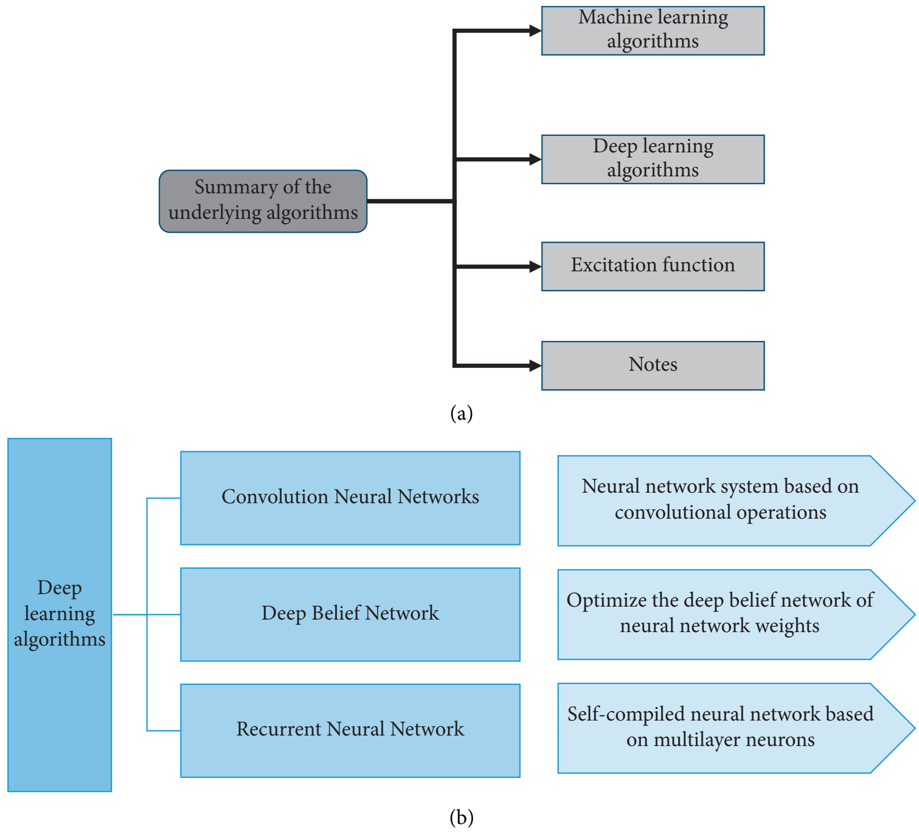


Figure 1: Algorithm classification results. (a) Algorithm basic classification. (b) Research content classification.

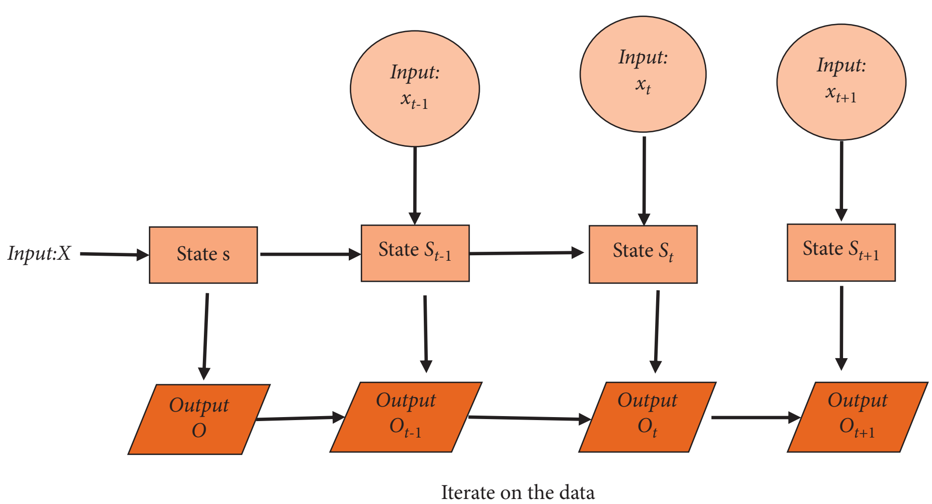


Figure 2: The structure of an RNN model.

In Figure 2, the RNN has an input x_t at every moment. Then, according to the state h_{t-1} of the RNN at the previous moment, the new state is denoted as h_t, and the output is denoted as O_t. The current state h_t of the RNN is jointly determined according to the state h_{t-1} at the previous moment and the current input x_t. At time t, the state h_{t-1} condenses the information of the previous sequence as a reference for the output. Since the length of the sequence can be extended indefinitely, however, the h state with a restricted dimension cannot store all statistics of the sequence. For that reason, the model needs to essentially learn and to retain only the utmost significant facts and statistics relevant to the subsequent jobs [16].

At time t, the expression of the output value O_t is shown as follows:

$$O_t = g(Vs_t). \quad (1)$$

In the above equation, g characterises the activation function, V represents the weight, and s_t symbolises the sum of the weights at time t.

The weight and s_t expression at time t is shown as follows:

$$s_t = f\left(Ux_t + Ws_{t-1}\right). \tag{2}$$

In the above equation, f symbolises the activation function, U represents the weight, W characterises the state transition weight matrix from the before the next moment, x_t represents the input at time t, s_{t-1} represents the weighted sum at time $t-1$. (2) is substituted into (1) to obtain the O_t expansion. RNN has a strong memory function for sequence information. The expansion is shownas follows:

$$O_t = Vf\left(Ux_t + Wf\left(Ux_{t-1} + Wf\left(Ux_{t-2} + Wf\left(Ux_{t-3} + \ldots\right)\right)\right)\right). \tag{3}$$

2.3. Forward Propagation Algorithm. The forward propagation algorithm, that is, the algorithm that realizes the function of data propagation along the forward direction, calculates the parameters of the forward propagation algorithm [17]. At time t, the hidden state s_t is shownas follows:

$$s_t = \sigma\left(Ux_t + Ws_{t-1} + b\right), \tag{4}$$

where σ represents the activation function, generally marked as tanh; and b represents the bias.

At time t, the output O_t is shown as follows:

$$O_t = Vs_t + c, \tag{5}$$

where c stands for bias.

At time t, the predicted output $\overline{y_t}$ is given by:

$$\overline{y_t} = \sigma\left(o_t\right). \tag{6}$$

At time t, the hidden state h_t is shownas follows:

$$h_t = f\left(s^{t-1}, x_t, \theta\right), \tag{7}$$

where s is the internal state, f is the excitation function, and θ is the weight coefficient inside the recurrent unit.

2.4. Backpropagation Algorithm. The RNN forward propagation algorithm is used as the basis, and the RNN backpropagation algorithm is calculated. The process of the RNN backpropagation algorithm is to calculate the gradient of each parameter of the model, that is, the gradients of U, W, V, b, and c through the transferred property of the gradient descent error [18]. The loss function is set to the cross-entropy loss function L, the output activation function is the softmax function, and the activation in the hidden layer is the tanh function.

The total loss function is shown as follows:

$$L = \sum_{t=1}^{T} L_t. \tag{8}$$

The gradients of V and c are given by the following equations, respectively:

$$\frac{\partial L}{\partial c} = \sum_{t=1}^{T} \frac{\partial L_t}{\partial L} = \sum_{t=1}^{T} \overline{y_t} - y_t, \tag{9}$$

$$\frac{\partial L}{\partial V} = \sum_{t=1}^{T} \frac{\partial L_t}{\partial V} = \sum_{t=1}^{T} \overline{y_t} - y_t\left(s_t\right)^T. \tag{10}$$

2.5. The Basic Framework of Spark. Spark is a fast, general-purpose, and scalable big data analysis computing engine based on memory. The following are the major features of the Spark framework.

(1) Rapidity: Spark supports memory-based computing, which can effectively save Input/Output (IO) resources compared to MapReduce's disk-based computing engine. In iterative operations, it runs nearly 100 times faster than MapReduce. Even, Spark is still nearly ten times faster than MapReduce on disk-based computing due to the superiority of Scala code for the functional language.

(2) Versatility: Spark provides standardized solutions to reduce enterprise development costs.

(3) Scalability: In addition to its own resource scheduler, Spark can replace Hadoop's Yarn or Mesos resource scheduler and can process all data supported by Hadoop [19]. The internal module structure of Spark is shown in Figure 3.

Figure 3: The Spark module is based on Spark Core. The Spark Structured Query Language (SQL) is built. Moreover, Spark Streaming, Mlib, Spark Gragh, and Spark SQL are mainly used for the interactive query. The Spark Streaming is mainly used for stream computing. The Mlib is used for machine learning and the Spark Gragh is a graph computing library.

2.5.1. Deeplearning4J Framework. The Deeplearning4J is a DL framework, open-sourced, and maintained by the Skymind. The Deeplearning4j is based on features developed by the Java Virtual Machine (JVM). Deeplearning4J natively supports distributed model training and making distributed modelling possible based on the theory of data parallelism. At present, the solutions for distributed modelling of deep neural networks are mainly divided into (i) model and (ii) data parallelization. The model parallelization can perform hierarchical training on multilayer neural networks; that is, the parameters of some network layers are concentrated on one node of the cluster for training. The update of parameters between each node is scheduled through the scheduler. The data parallelization saves a copy of the network model on each compute node and trains its own batch of data separately. Then, the global network parameters are updated according to a synchronous or asynchronous mechanism [20]. Deeplearning4J mainly uses data parallelization solutions. Currently, Deeplearning4J

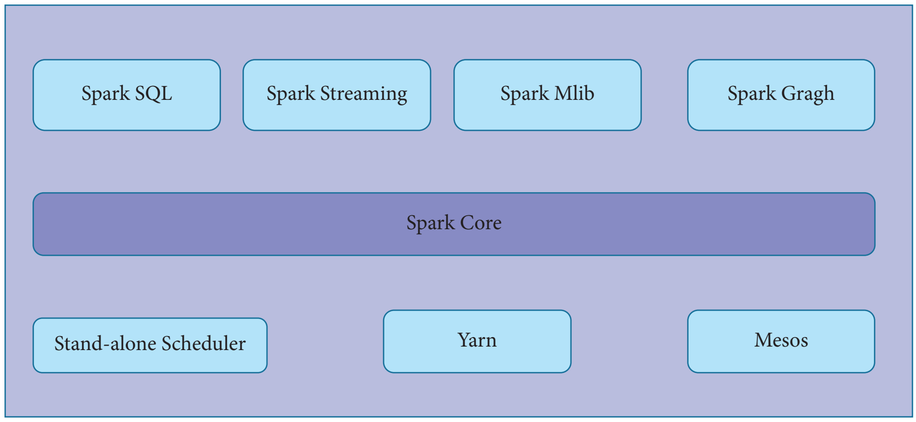


Figure 3: Structure of Spark's built-in modules.

supports two data parallelization strategies: a parameter synchronous averaging and a decentralized gradient sharing scheme [21]. The principle of data parallelization is shown in Figure 4.

In Figure 4, the principle of parameter synchronization and averaging is to define a parameter server, for instance, Parameter Server in the Driver. This example uses a weighted average algorithm to calculate the updated model parameters after collecting training parameters from multiple nodes. Then, the updated parameters are made into broadcast parameters and passed to each node [22]. The principle of decentralization is that multiple nodes are connected in pairs and can update parameters with each other. In order to solve the bottleneck problem in network transmission, the decentralized gradient sharing scheme defines a threshold for each gradient change. The gradient parameters can only be updated when a certain gradient change is greater than the threshold.

3. Model Construction of Piano Performance Evaluation Based on RNN

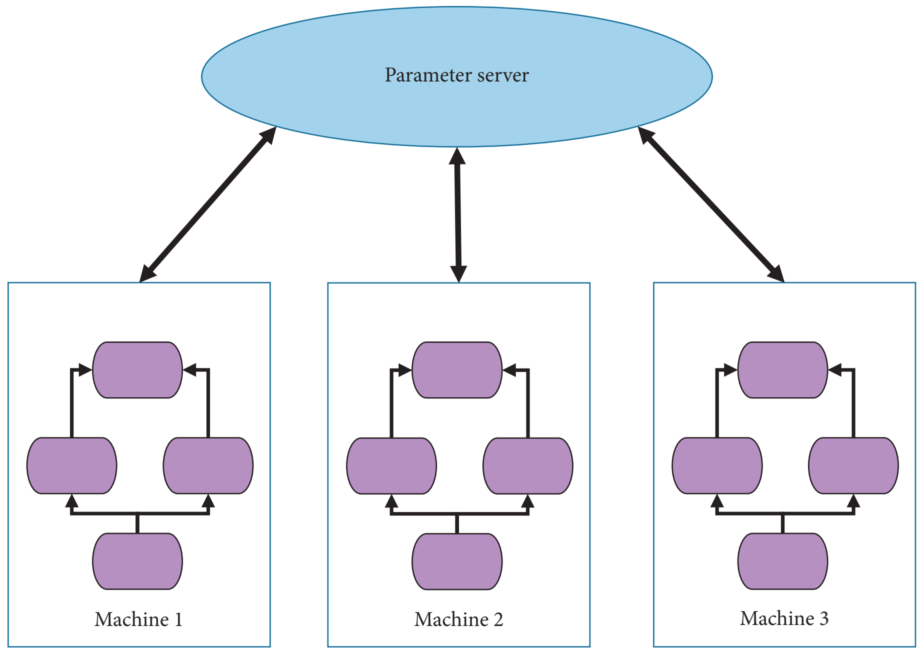
This study adopts the Deeplearning4J DL framework that trains and runs on the Spark-Yarn run mode cluster and Deeplearning4J cluster. The User Interface (UI) provided by the Deeplearning4J is used to monitor the training effect in real time and adjust the model parameters at any time. Additionally, the RNN model with attention mechanism can achieve efficient and accurate Musical Instrument Digital Interface (MIDI) piano performance evaluation. The framework of the evaluation model is shown in Figure 5.

In Figure 5, in the data acquisition module, the Sqoop tool is used to migrate data to the Hadoop Distributed File System (HDFS). In the data preprocessing module, raw data that are not suitable for training are filtered. The raw data are transformed into an input matrix form that is suitable for the RNN model training. The dataset is divided into training, validation, and test datasets. In the music evaluation classification module, the Spark-Yarn cluster is built. On the distributed framework, the RNN model is built. The preprocessed data are fed into the model training. The model parameters are adjusted in real time through the UI provided by Deeplearning4J, and the model parameters with better evaluation effects are obtained [23].

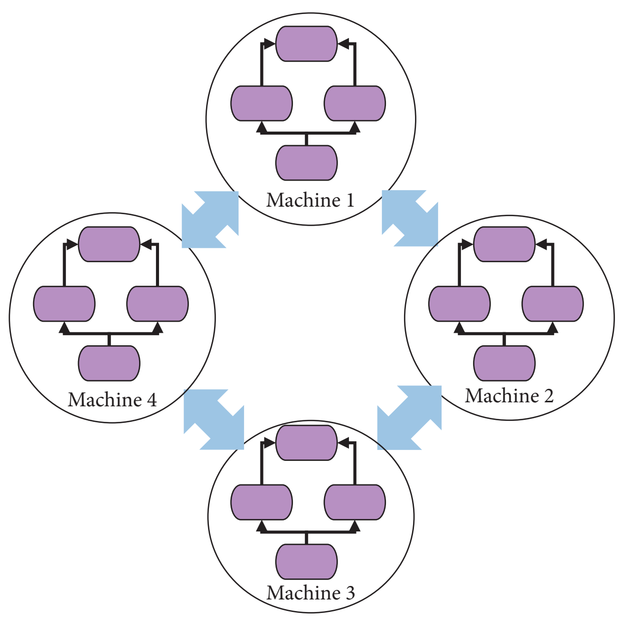
3.1. Data Preprocessing.

3.1. Data Preprocessing. Before training the model, the MIDI music data are predesigned. Timestamps in track chunks are unified to 1/16th notes, efficiently handling tedious preprocessing. The preprocessing content includes filtering music data synthesized by multitrack and MIDI software, extracting feature vectors, and designing a more reasonable model input format. There are two main types of characteristics of music. The first category is physical characteristics, including pitch and timbre. The second category is time-domain features, including short-term energy, short-term average zero-crossing rate, and short-term average amplitude. Since the MIDI music is digital music, the MIDI format can completely record the required physical characteristics, which is difficult to obtain compared to the time-domain characteristics [24]. Therefore, physical features are selected as the feature research direction. Because the timbre is only related to playing the instrument, the model only evaluates the piano music, so the timbre can directly get the default value. The ordinate of the input feature matrix is set as the time series; the abscissa is set as the key information; the matrix elements are set as the pitch information. Since the piano keyboard has 88 keys, the dimension of the abscissa is 88. The ordinate dimension is determined by the playing time of the music.

3.2. MIDI-based Piano Evaluation Neural Network Model. After obtaining a piece of piano playing audio, firstly, a data preprocessing algorithm is used to obtain the start and endpoints of each note. At this time, the time value information of the musical note is also determined. The functional requirement of this model module is to evaluate multiple pieces of MIDI music with high accuracy. The evaluation results are divided into five grades: excellent, good, medium, poor, and poor. This study uses the RNN structure and attention layer in DL for classification by *softmax* function. The complete model structure design is shown in Figure 6:



(a)



(b)

Figure 4: Principle of data parallelization; (a) principle of parameter synchronous averaging scheme; (b) principle of decentralized gradient sharing scheme.

In Figure 6, multiple subnet models are designed to realize the function of evaluating multiple pieces of piano music. Each subnet model needs to be trained separately. Multiple subnetwork models are evaluated separately for specific piano music. The evaluation subnet model obtains a feature matrix by training sample repertoires with differences in level. Then, a classification algorithm is used for evaluation. The main structures of the evaluation subnet model are input, bidirectional LSTM, attention mechanism, and output layer [25]. The input layer receives the difference sample repertoire, obtains the input feature matrix through data preprocessing, and then inputs it into the RNN structure. After the sample passes through the attention mechanism layer, the evaluation is obtained through the *softmax* function of the output layer.

3.3. Analytical Methods of Experiments

3.3.1. Data Acquisition. The data of this experiment mainly come from two parts. In fact, part of it comes from the database provided by five piano training courses in Guangxi, and the other part comes from the MIDI Show web page. The

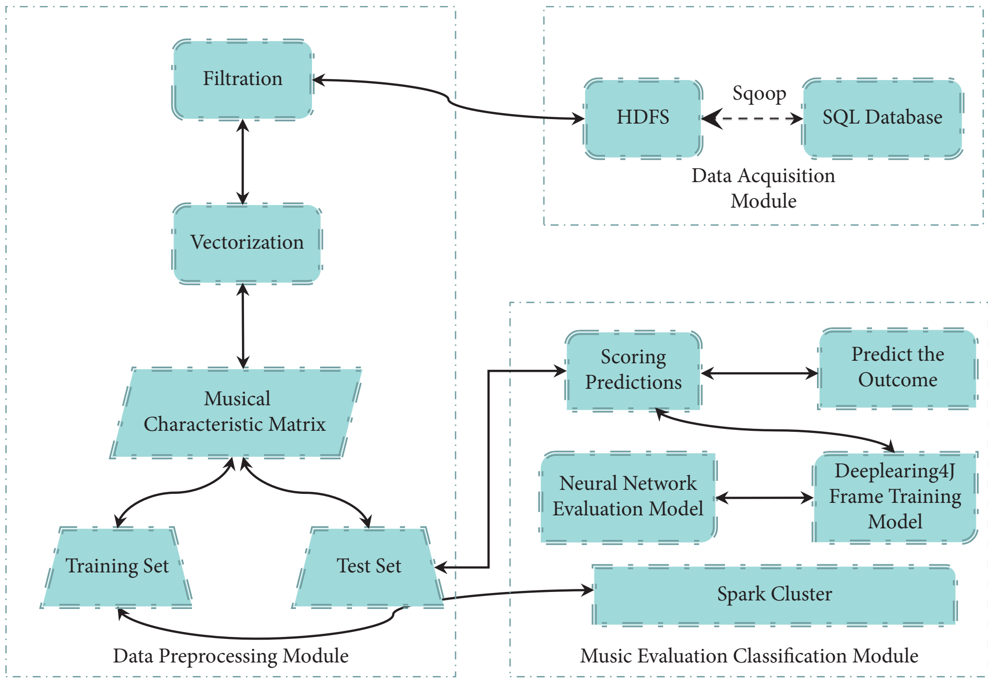


Figure 5: Framework of the piano evaluation model.

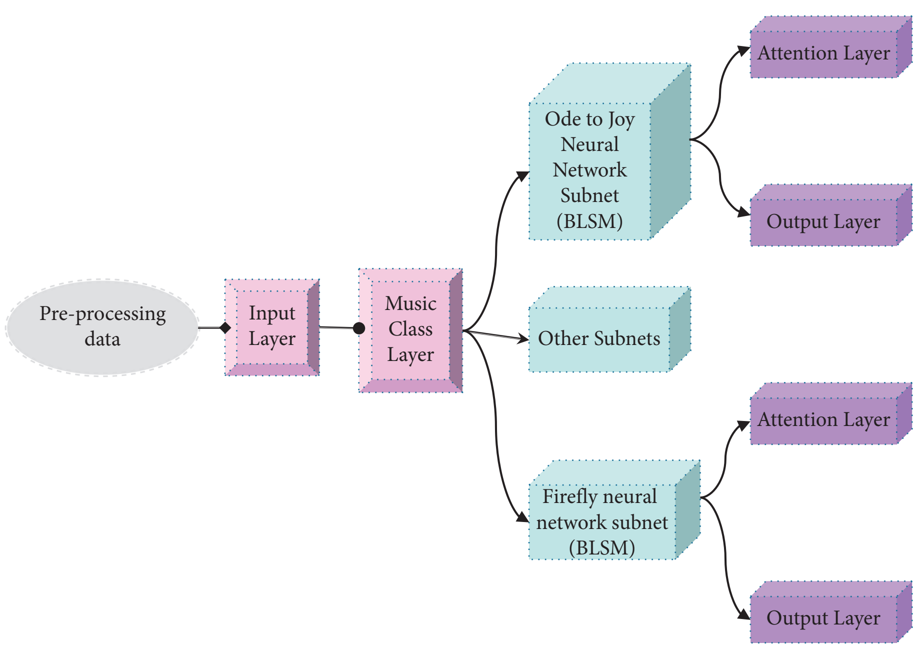


Figure 6: The structure of the piano neural network model.

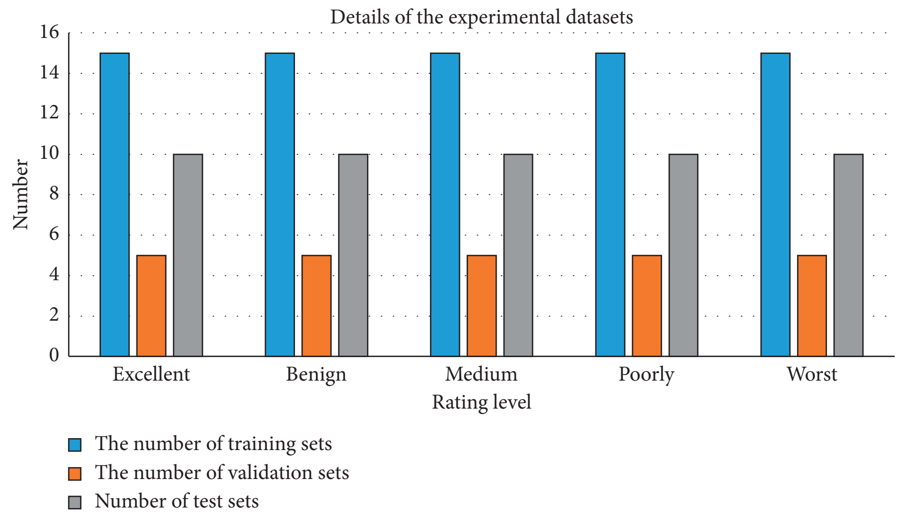


Figure 7: Details of the experimental datasets.

MIDI enthusiasts upload their works on this platform and score them. The scores are divided into five intervals, which correspond to the five evaluation levels of the system. In order to facilitate the implementation and analysis of the evaluation model, this study only selects piano music in 4/4 time [26]. The details of the experimental dataset are shown in Figure 7.

In Figure 7, 30 piano pieces are selected for each evaluation level, for a total of 150 pieces. For each evaluation level, 15 songs are selected as the training set, 5 as the validation set, and 10 as the test set.

In databases, the quality of music data is uneven. Neural network models cannot recognize raw data. Data are preprocessed. Preprocessing is divided into two Figure 8 steps:

(1) Music files synthesized by MIDI software are filtered. The filtering process is shown in Figure 8; this should be kept in mind that when the number of pitch values in the experimental dataset is less than 40 or the number of pitch values is greater 80 while is less than 20, then in that cases the MIDI music is filtered out.

(2) The filtered data are vectorized; that is, the data in the form of MIDI are converted into an input matrix that the RNN structure can read.

3.3.2. Analysis of RNN Parameters. The choice of the number of layers of the RNN structure and the corresponding number of nodes will directly affect the reliability and accuracy of the model. A reasonable number of layers of the RNN structure and the corresponding number of nodes are given so that the model is optimal, and the prediction error is minimized [27]. The value of the loss function L for different layers and different nodes of the RNN is analyzed separately, as shown in Figure 9.

In Figure 9, when the RNN structure is a single layer, the more node values, the smaller the L value. After the node value is greater than 352, the rate of decrease of the value slows down. The RNN structure is a double layer; the more node values, the smaller the L value. After the node value is greater than 176, the rate of decrease of the value slows down. When the RNN structure is three layers, the changing trend of node value and L value is not obvious. Therefore, a three-layer RNN structure is chosen [28]. Additionally, the number of nodes per layer is set to 352, 176, and 88.

3.3.3. The RNN Training. The mean square error (MSE) is used to assess the performance of the proposed RNN model. After the input of the parameters affecting the piano performance, education evaluation and the parameters and structure of the RNN are determined. Subsequently, the neural network is trained to attain the anticipated needs for precisions and accuracy. First, the standard information of each characteristic is obtained by playing and MIDI file by the piano teacher. Then, through different levels of piano performance, input features are extracted. The training method of RNN is mainly to obtain training samples by playing the *Minuet* piano piece several times by two piano teachers and three students. The input data to the RNN are derived for each play [29, 30]. The overall rhythmic feeling and level of playing expression are then assessed separately. The supplied data's range is 0–1. For this training, 10 samples were gathered. The parameters are as follows: learning rate of 0.76, momentum factor of 0.4, and error of 0.1.

4. Simulation Results

4.1. Analysis of the RNN Training Results. The RNN structure model is trained as shown in Figure 10.

In Figure 10, with the increase of training times of RNN, the resulting error value gradually decreases. After training 3000 times, the error value of the RNN structure is basically fitted with the standard error value, reaching the network convergence accuracy. Additionally, the correlation coefficient between the output of the RNN structure network and the target

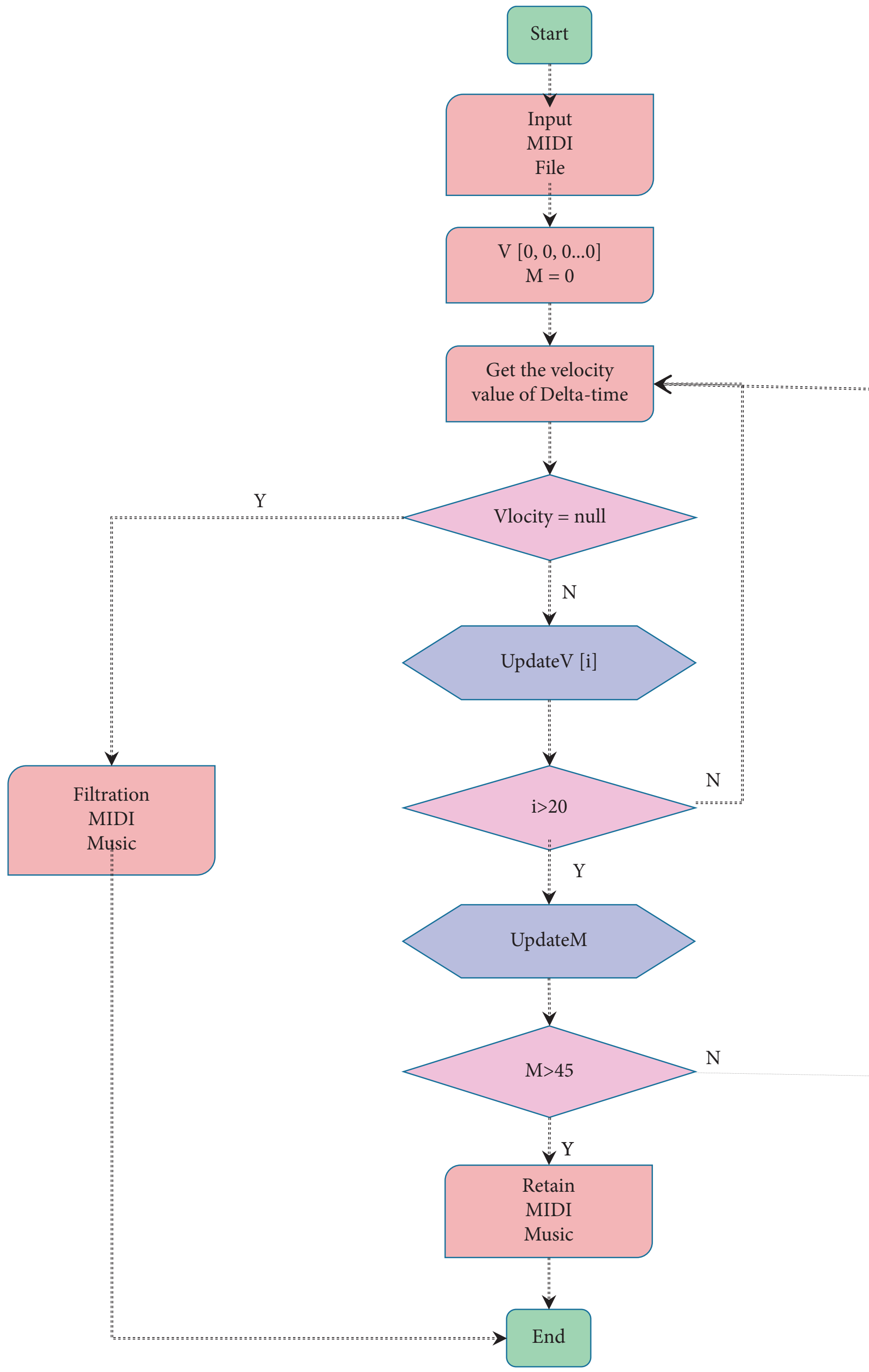


Figure 8: The data filtering process.

is as high as 0.99947, showing a high degree of fit. The designed RNN performance can meet the actual requirements.

The designed teaching evaluation model is used to evaluate the piano teacher, student A (piano level 6 level) and B (piano level 5 level), and the evaluation results are shown in Figure 11.

In Figure 11, the average value of the overall evaluation of the piano teachers is 0.92, the average value of the

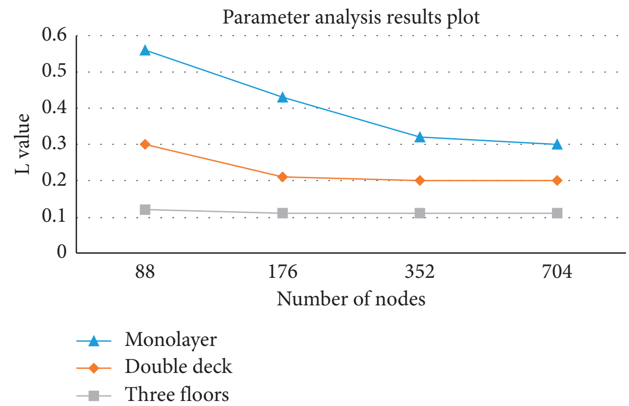
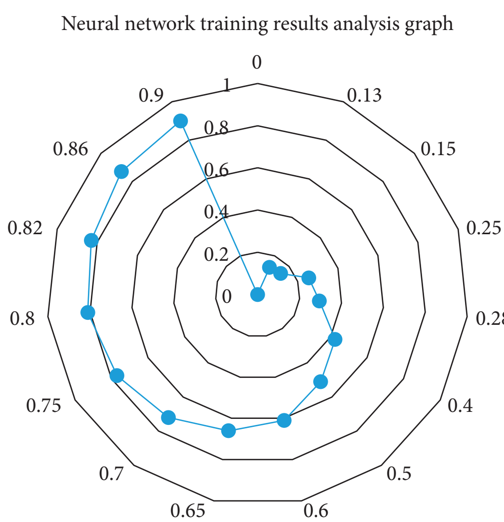
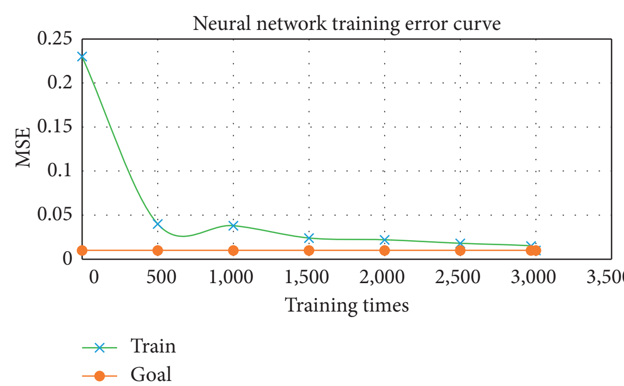


Figure 9: Results of parametric analysis.

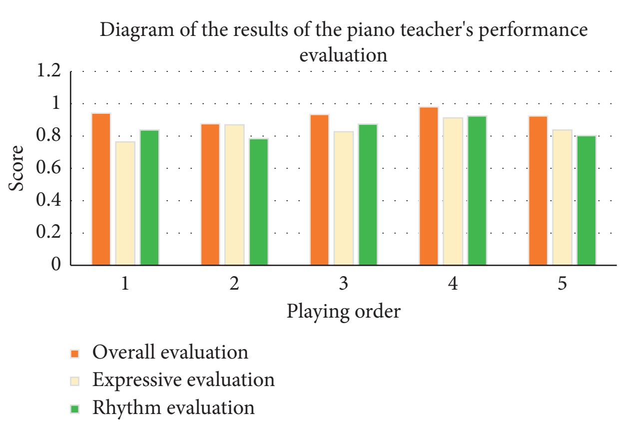


(a)

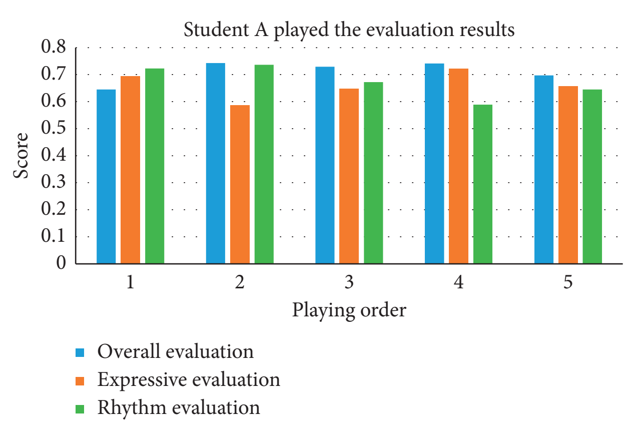


(b)

Figure 10: RNN training results; (a) analysis of MSE training results; (b) the error curve of neural network training.

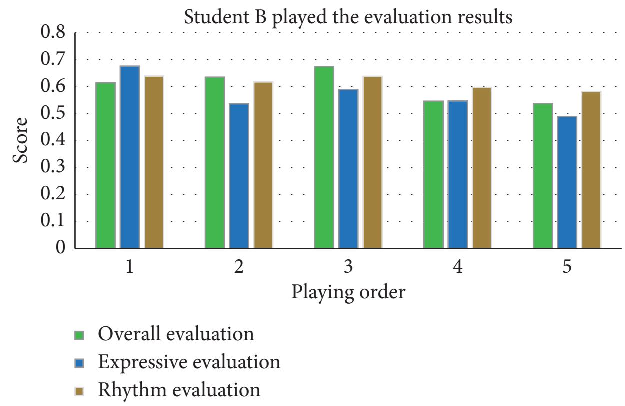


(a)



(b)

Figure 11: Continued.



(c)

Figure 11: Model evaluation results; (a) results of piano teacher performance evaluation; (b) student A performance evaluation results; (c) student B performance evaluation results.

performance evaluation is 0.83, and the average value of the rhythm evaluation is 0.82. Student A's evaluation averages are 0.71, 0.65, and 0.67. Student B's averages are 0.6, 0.59, and 0.57, respectively. The piano instructor, student A, and student B are the evaluation values provided by the model, which are generally consistent with the actual level of the performers. The data evidence that the proposed model's results can encounter the necessities and can be applied to the evaluation of piano performance education.

5. Conclusions and Future Work

The proposed RNN-based MIDI piano performance education evaluation method makes up for the shortcomings of the rule-based evaluation method, which cannot consider the coherence and expressiveness of music. First, raw data are selected from the MIDI database and the educational data of a local piano training institution. Then, the raw data are preprocessed. Next, the three-layer bidirectional LSTM neural network and the attention mechanism make it easier for the model to capture useful information. Additionally, the Spark cluster is built using the Deeplearning4J DL framework to train the model. The work efficiency is improved by adjusting model parameters through UI dependencies provided by Deeplearning4J. Additionally, the RNN parameters are analyzed. The results show that the error value of the three-layer RNN structure is smaller. Local piano training institutions and MIDI website data are used as a basis. Experimental samples are collected. These samples are used to train the neural network and to test the performance of the evaluation model. The findings demonstrate that (1) the evaluation outcomes of the developed piano performance evaluation model are largely consistent with the actual skill level of the players and have some degree of viability; and (2) after 3000 training cycles, the RNN error is close to 0.01, and the network converges. The disadvantage is that the research stays at the theoretical level.

In order to test a large amount of data, the DL network is developmental, and it is necessary to update the evaluation model according to the latest development. The purpose is to provide important technical support to improve the efficiency of piano music teaching. Similarly, deep neural networks and the impacts of the model activation functions should be analyzed in subsequent research efforts. The Spark framework runs the deep learning algorithms in parallel; however, the computational time is still limited on a single online system. Therefore, big data technologies such as cloud and edge computing should be used to improve the training efficiency in terms of computational times.

Data Availability

The experimental data used to support the findings of this study are available from the author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest regarding this work.

References

[1] Y. Izzet, E. A. Çigdem, and K. Tugçe, "Evaluation of asynchronous piano education and training in the Covid-19 era," *Educational Research and Reviews*, vol. 16, no. 4, pp. 109–117, 2021.

[2] S. Pamies, "Deconstructing modal jazz piano techniques: the relation between Debussy's piano works and the innovations of post-bop pianists," *Jazz Education in Research and Practice*, vol. 2, no. 1, pp. 76–105, 2021.

[3] Y. S. Kaleli, "The effect of individualized online instruction on TPACK skills and achievement in piano lessons," *International Journal of Technology in Education*, vol. 4, no. 3, pp. 399–412, 2021.

[4] J. Wang, "Innovative research on the teaching mode of piano group lessons under the background of big data," *Journal of Physics: Conference Series*, vol. 1744, no. 3, Article ID 032031, 2021.

[5] X. Huang, "Aims for cultivating students' key competencies based on artificial intelligence education in China," *Education and Information Technologies*, vol. 26, no. 5, pp. 5127–5147, 2021.

[6] W. Yang, "Artificial intelligence education for young children: why, what, and how in curriculum design and implementation," *Computers & Education: Artificial Intelligence*, vol. 3, Article ID 100061, 2022.

[7] S. Seo and C. Kim, "Analysis of understanding of prospective teachers' computational thinking on artificial intelligence education," *Journal of The Korean Association of Information Education*, vol. 25, no. 1, pp. 123–134, 2021.

[8] S. Liu and J. Wang, "Ice and snow talent training based on construction and analysis of artificial intelligence education informatization teaching model," *Journal of Intelligent and Fuzzy Systems*, vol. 40, no. 2, pp. 3421–3431, 2021.

[9] K. Kaur, A. Garg, X. Cui, S. Singh, and B. K. Panigrahi, "Deep learning networks for capacity estimation for monitoring SOH of Li-ion batteries for electric vehicles," *International Journal of Energy Research*, vol. 45, no. 2, pp. 3113–3128, 2021.

[10] Y. Fu, T. Wang, Y. Lei et al., "Deformable MR-CBCT prostate registration using biomechanically constrained deep learning networks," *Medical Physics*, vol. 48, no. 1, pp. 253–263, 2021.

[11] J. Lansky, S. Ali, M. Mohammadi et al., "Deep learning-based intrusion detection systems: a systematic review," *IEEE Access*, vol. 9, pp. 101574–101599, 2021.

[12] Y. Shi, Y. Zheng, K. Guo, and X. Ren, "Stock movement prediction with sentiment analysis based on deep learning networks," *Concurrency and Computation: Practice and Experience*, vol. 33, no. 6, Article ID e6076, 2021.

[13] A. Ns, D. Susan, and R. Kumar, "Naive Bayesian fusion based deep learning networks for multisegmented classification of fishes in aquaculture industries," *Ecological Informatics*, vol. 61, Article ID 101248, 2021.

[14] H. Li, G. Wang, Z. Dong et al., "Identifying cotton fields from remote sensing images using multiple deep learning networks," *Agronomy*, vol. 11, no. 1, p. 174, 2021.

[15] A. Cossu, A. Carta, V. Lomonaco, and D. Bacciu, "Continual learning for recurrent neural networks: an empirical evaluation," *Neural Networks*, vol. 143, pp. 607–627, 2021.

[16] P. B. Weerakody, K. W. Wong, G. Wang, and W. Ela, "A review of irregular time series data handling with gated recurrent neural networks," *Neurocomputing*, vol. 441, pp. 161–178, 2021.

[17] G. J. Wang, C. L. Zhu, and B. P. CVaR, "BP-CVaR: a novel model of estimating CVaR with back propagation algorithm," *Economics Letters*, vol. 209, Article ID 110125, 2021.

[18] B. Yang, C. Zeng, L. Wang et al., "Parameter identification of proton exchange membrane fuel cell via Levenberg-Marquardt backpropagation algorithm," *International Journal of Hydrogen Energy*, vol. 46, no. 44, pp. 22998–23012, 2021.

[19] Y. Ma, P. A. Zhong, B. Xu, F. Zhu, Q. Lu, and H. Wang, "Spark-based parallel dynamic programming and particle swarm optimization via cloud computing for a large-scale reservoir system," *Journal of Hydrology*, vol. 598, Article ID 126444, 2021.

[20] L. Csépányi-Fürjes and L. Kovács, "Efficiency analysis of Deeplearning4J neural network classifiers in development of transition based dependency parsers," *Acta Marisiensis. Seria Technologica*, vol. 18, no. 1, pp. 33–39, 2021.

[21] H. Li, "Piano education of children using musical instrument recognition and deep learning technologies under the educational psychology," *Frontiers in Psychology*, vol. 12, Article ID 705116, 2021.

[22] X. Chang and L. Peng, "Evaluation strategy of the piano performance by the deep learning long short-term memory network," *Wireless Communications and Mobile Computing*, vol. 2022, pp. 1–10, Article ID 6727429, 2022.

[23] N. Huang and X. Ding, "The construction of online course learning model of piano education from the perspective of deep learning," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–10, Article ID 4378883, 2022.

[24] Y. Ghatas, M. Fayek, and M. Hadhoud, "A hybrid deep learning approach for musical difficulty estimation of piano symbolic music," *Alexandria Engineering Journal*, vol. 61, pp. 10183–10196, 2022.

[25] V. Phanichraksaphong and W. H Tsai, "Automatic evaluation of piano performances for STEAM education," *Applied Sciences*, vol. 11, Article ID 11783, 2021.

[26] A. Wang and H. Mukaidani, "An efficient piano performance evaluation model using DTW based on deep learning," in *Proceedings of the 2021 IEEE 30th International Symposium on Industrial Electronics (ISIE)*, June 2021.

[27] X. Wang and Y. Chen, "Music teaching platform based on FPGA and neural network," *Microprocessors and Microsystems*, vol. 80, Article ID 103337, 2021.

[28] P. Ramoneda, N. C. Tamer, V. Eremenko, X. Serra, and M. Miron, "Score difficulty analysis for piano performance education based on fingering," in *Proceedings of the ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2022.

[29] Y. Yang, "Piano online teaching based on neural network technology," in *Proceedings of the 2021 3rd International Conference on Artificial Intelligence and Advanced Manufacture*, Manchester, USA, October 2021.

[30] W. Wang, J. Pan, H. Yi, Z. Song, and M. Li, "Audio-based piano performance evaluation for beginners with convolutional neural network and attention mechanism," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 1119–1133, 2021.