Hindawi
Scientific Programming
Volume 2021, Article ID 7196197, 9 pages
https://doi.org/10.1155/2021/7196197

Research Article

Construction of Music Teaching Evaluation Model Based on Weighted Naïve Bayes

Xiongjun Xia and Jin Yan

Hunan Normal University, Changsha, Hunan, China

Correspondence should be addressed to Jin Yan; 202010160242@hunnu.edu.cn

Received 19 July 2021; Revised 7 August 2021; Accepted 20 August 2021; Published 17 September 2021

Academic Editor: Mian Ahmad Jan

Evaluation of music teaching is a highly subjective task often depending upon experts to assess both the technical and artistic characteristics of performance from the audio signal. This article explores the task of building computational models for evaluating music teaching using machine learning algorithms. As one of the widely used methods to build classifiers, the Naïve Bayes algorithm has become one of the most popular music teaching evaluation methods because of its strong prior knowledge, learning features, and high classification performance. In this article, we propose a music teaching evaluation model based on the weighted Naïve Bayes algorithm. Moreover, a weighted Bayesian classification incremental learning approach is employed to improve the efficiency of the music teaching evaluation system. Experimental results show that the algorithm proposed in this paper is superior to other algorithms in the context of music teaching evaluation.

1. Introduction

Recently, music has become an important part of education. With the reform and development of music education and the introduction of foreign advanced teaching methods, music education as an important part of quality education has been paid more and more attention [1]. Music teaching has shifted from focusing on skills to cultivating students' emotions, attitudes, and values. Music teaching evaluation has shifted from summative evaluation to formative evaluation, from partial evaluation to comprehensive evaluation, and from focusing on results to focusing on the process [2].

The task of developing and implementing an effective evaluation process for music education seems to be more abstract than producing evaluation methods in other areas of teaching. Traditional methods to the evaluation of music to date failed to prepare evaluators with comprehensive information required to make educational decisions about music teacher performance. For educational leaders to make valid decisions about music education in our schools, more precise and accurate evaluation models will be required about the effectiveness and nature of music teacher evaluation. Music performance evaluation (MPE) is the process of identification, assessment, and modeling the impact of music on the human listener [3]. The majority of the initial researches in MPE investigated symbolic data collected from musical instrument digital interface (MIDI) devices. Recently, attention has moved to the investigation of audio signals.

The concept of applying technology to evaluate music teaching education is old. An initial attempt to point out the importance of systematic assessment of music acts for enhancing learning was carried out by Seashore in the 1930s [4]. One of the initial works discovering the benefits of computer-aided evaluation techniques for music was accomplished by Allvin [5]. He described the application of pitch detection to analyze errors in musical presentation and provide a positive response to the learners. However, currently, the music signal is different from the original sound wave. The musical signal is sampled, multiple times per second, and converted by an analog-to-digital converter into a series of digital values. These values characterize the digital audio signal and are used to regenerate the music. This musical signal comprises numerous acoustic information

and features. Usually, these audio features are extracted from the small segments of the audio signal and then combined into a more abstract feature vector [6].

Music teaching evaluation systems usually depend upon extracting prominent and standard audio features from voice signal and then applying machine learning algorithms to describe the value of the performance. Kosina [7], for example, extracted audio features from the 3 second segments of musical sounds and classified the features with a neural network classifier. Knight et al. [8] execrated a group of audio features and employed a support vector machine (SVM) classifier to classify the tonal characteristics of trumpet performances into "good" or "bad" labels. An evaluation system based on the identification of pitch interval for the assessment of singing voice was presented in [9]. Abeßer et al. used the features of sound pitch and intonation a music sound and instrumental performances to determine the rhythmic accuracy [10]. Luo et al. extracted the music spectral, timbral, and pitch-based features to predict mistakes of the violin players [11].

Features based on associated pitch contours were extracted by Bozkurt et al. to investigate vocal conservatory exam sounds and categorize them as "pass "or "fail" [12]. Han and Lee [13] used the mel-frequency cepstrum coefficients (MFCCs) as features to detect common musical mistakes. Wu and Lerch [14] utilized sparse coding to extract features for developing regression models for the evaluation of music performances. The models exhibited enhancement in performance compared to other models.

Despite the different feature sets and machine learning algorithms used by the above works, the basic idea that links them together is that they all are based on features and classification algorithms tuned for a specific task [15–18]. Motivated by the triumph of these music evaluation systems, the inspiration of this study is to discover the potential of machine learning in the form of Naïve Bayes classifier for the valuation of music performance. In this study, we employ an improved Naïve Bayes classifier (NBC) for the music teaching evaluation. The Naïve Bayes method has become greatly popular for building classifiers. This is because of the ability of NBC to acquire prior knowledge, the distinctive knowledge representation of NBC, and the accuracy of NBC. Although NBC is capable of extraordinary classification performance [19, 20], questions remain about their performance in multilabel classification.

The remaining sections of the paper are organized as follows. In Section 2, we provide an outline of the commonly known machine learning techniques. In Section 3, we present the proposed music teaching evaluation model. The results are given in Section 4, and finally, the proposed work is summarized in Section 5.

2. Overview of Classification Algorithms

Classification is an important problem in supervised learning. Its purpose is to summarize the unique audio features and find a model or accurate description for each class. The trained classification model is used to classify the features set in the dataset, and the class of the new data with unknown labels can be recognized through the trained model. The classification problem is mainly divided into two phases: learning and classification. In the learning phase, according to the known training dataset, we use the appropriate learning method to train a classifier. In the process of classification, new input instances are classified by the learned classifier.

In machine learning, the common classification algorithms are NB, SVM, KNN, decision tree (DT), and artificial neural network (ANN). Among them, the NB algorithm is a classification method built on the Bayesian theorem, which introduces the theory of feature condition independence, and the classification model is easy to understand. SVM is a linear classification model with the largest hyperplane defined in the feature space. KNN assumes that given a training dataset, the class of the instance has been determined. When classifying new test cases, KNN first judges the class label of the k-nearest neighbor training instance and then uses the majority voting method to predict the class. The DT model is a tree structure, which denotes the process of predicting data item examples based on features. The DT learning process usually comprises three steps: feature selection, tree generation, and decision tree pruning. ANN can effectively solve the problem of nonlinear comprehensive evaluation and reduce the influence of human factors on decision-making results. It can fully model any complex nonlinear relationship and model the nonlinear process without knowing the cause of the data.

Different machine learning algorithms exhibit different properties. The effect of classification is mostly related to the characteristics of data and application background. Table 1 provides a comparative analysis of several common classification algorithms.

Based on a given problem of music teaching evaluation, the different machine learning algorithms can be selected for music teaching evaluation. Taking a sequence of evaluation attributes as input data and an evaluation grade as a class label, a classification algorithm can give the most likely class label for the new evaluation attribute, namely, the evaluation result. To make sure the reliability of the evaluation results, it is indispensable to use the appropriate evaluation index to construct the classifier. Accuracy is an important index to evaluate the performance of a classifier, which is the ratio of correctly classified data samples to the total number of applied samples in a given dataset. Accuracy can be computed as follows:

$$\text{Acc} = \frac{N_c}{N}, \tag{1}$$

where Acc is the accuracy rate, N_c is the number of correctly classified samples, and N shows the total number of total samples.

K-fold cross-validation is a key statistical assessment technique, mostly used to assess the performance of machine learning algorithms. In this study, 10-fold cross-validation was applied in addition to accuracy to gauge the performance of the proposed technique.

Table 1: Comparison and analysis of classification algorithms.

Algorithm	Advantage	Shortcoming
NB	Classification efficiency is stable and suitable for multiclassification tasks and incremental training and has the best performance when the correlation of feature attributes is small.	It cannot make a very accurate estimation of the class probability, and the time efficiency is low when there are many attributes or the correlation between attributes is large.
SVM	It uses less training set and can deal with high-dimensional sparse text data, which has a certain robustness.	It depends too much on the position of positive and negative examples around the classification surface, and the selection of kernel function lacks guidance. When there are many samples, the training speed is slow.
KNN	Without parameter estimation, it is easy to deal with the case of a large number of categories, and the method is simple and stable.	The sample size is large; the space complexity is high; the memory overhead is large; and the selection of K value also directly affects the performance of classification.
DT	The model has a tree structure, readability, and high classification speed.	It is easy to overfit and ignore the correlation between attributes.
BP network	It has the ability of self-adaptive, strong nonlinear mapping, and fault tolerance.	It has slow convergence, is easy to fall into local minimum, and has strong sample dependence.

3. Music Teaching Evaluation Recognition Model

3.1. Principle of Naïve Bayes Algorithm. In the classification process of NB, the Bayesian theorem is applied. During classification, the prior probability of each category is obtained by learning a large training set, and then compute the posterior probability of an item x. Finally, the item x is predicted as one of the categories with higher posterior probability. Suppose D is the training dataset, $T = \{T_1, T_2 \ldots, T_n\}$ is the set of attribute variables, and n is the total attributes. $C = \{C_1, C_2, \ldots, C_m\}$ is the set of class variables; K is the number of classes; then a training sample can be represented as $\{\{x_1, x_2, \ldots, x_n, C, \}; j \in m, C_j$. It means that the class label of the training sample is given, and a test sample x can be represented as the probability of $\{x_1, x_2, \ldots, x_n\}$ judging that the test sample is related to a certain class. This can be computed using the following equation:

$$p(C_j \mid X) = \arg\max_{C_j} \frac{p(X \mid C_j) p(C_j)}{p(X)}. \tag{2}$$

NB is an effective classification algorithm. The classification model has the advantages of simple explanation, high computational efficiency, and good stability, and its performance is superior to DT, SVM, and other classifiers in some cases. The Naïve Bayesian model adopts the simplest network structure, as shown in Figure 1.

The root node C is the class variable, and the leaf nodes $A_1, A_2, \ldots, A_n$ are the attribute variables. The NB classification depends upon the ordinary Bayesian theory, which avoids the condition that the attributes are independent. If $p(X)$ is a constant, then the NB model can be represented using the following equation:

$$p(C_j \mid X) = \arg\max_{C_j} p(X \mid C_j) p(C_j), \tag{3}$$

where $p(C_j)$ is a kind of prior probability that can be learned from training data as follows:

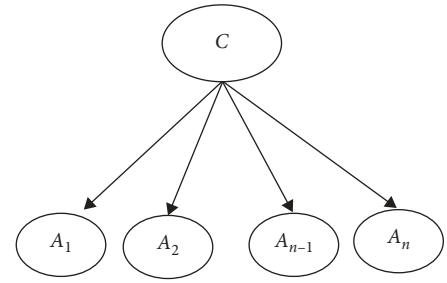

Figure 1: NB classification model.

$$p(C_j) = \frac{s_j}{s}, \tag{4}$$

where s_j is class in training samples C_j and s represents training samples.

NB algorithm assumes that all attribute variables are conditionally independent, and there is no relationship between attributes, so these attribute variables are also independent of the class attribute. The computation cost of $p(X \mid C_j)$ is very large, and the condition of independence can reduce the computation cost, but it will also lose some computational accuracy. Based on the conditional independence hypothesis, $p(X \mid C_j)$ the equation can be simplified as follows:

$$p(X \mid C_j) = \prod_{i=1}^{n} p(X \mid C_j), \tag{5}$$

where $p(x_1 \mid C_j), p(x_2 \mid C_j), \ldots, p(x_n \mid C_j)$ can be obtained from the training set. Combined with the above three equations, the class of test samples can be determined.

3.2. Determination of Weight Attributes Based on Weighted Naïve Bayes. To reduce the computational cost, in the NB algorithm, it is assumed that the condition attributes are independent of each other, and their importance in decision-making is the same for all attributes, that is, the weight is equal to 1. However, in practical application, the significance

of each conditional attribute varies; therefore, when the default ownership weight is 1, the accuracy of classification is reduced. In this study, the weighted Naïve Bayes (WNB) algorithm is used to assign an appropriate weight to the attributes based on their contribution to classification. It retains the computation speed and also decreases the influence of the attribute conditional independence statement on the classifier performance. The weighted NB is computed as follows:

$$p(C_j \mid X) = \arg\max_{C_j} p(C_j) \prod_{i=1}^{n} p(A_i \mid C_j)^{wi}, \quad (6)$$

where w_i denotes the weight of attribute A_i. The weight governs the significance of various classification attributes. The higher the value of w_i is, the greater the corresponding attribute A will be. In many applications, the prediction of a precise weight for each attribute is the key problem of the WNB model.

Based on the correlation between the evaluation attributes of music teaching evaluation data and the comprehensive evaluation value, it can be observed that the value of each index has a distinct effect on the results of the evaluation. In this study, we propose a new method to define the weight of each evaluation attribute by calculating the correlation probability of attributes in class. Each attribute A_i may have n different values, where $n \in N$. Suppose a concrete instance z, when the value of z is a_k for a given class C_j, attribute a A_i about C_j the correlation probability and uncorrelated probability are calculated using Equations (7) and (8), respectively:

$$p(A_i \mid \text{rel}) = \frac{\text{count}(A_i = a_k \wedge C_j)}{\text{count}(A_i = a_k)}, \quad (7)$$

$$p(A_i \mid \text{norel}) = 1 - p(A_i \mid \text{rel}), \quad (8)$$

where count is the statistical number; when the value of A_i is a_k and belongs to C_j for class, the attribute weight can be computed using the following equation:

$$w(A_i, a_k, j) = \frac{p(A_i \mid \text{rel})}{p(A_i \mid \text{norel})}. \quad (9)$$

Hence, the equation of the WNB algorithm is as follows:

$$p(C_j \mid X) = \arg\max_{C_j} p(C_j) \prod_{i=1}^{n} p(A_i \mid C_j)^{w(A_i, a_k, j)}. \quad (10)$$

In a dataset, suppose there are l class labels, n attributes, and K attributes for each attribute, then the total number of weights for all attributes is $l * n * k$. Different values of the same attribute lead to different weights. When the value of the same attribute is the same, the weight is different under different categories. Finally, corresponding to each attribute value, the weight of the correlation probability with the present class label is used for computation, and the output of each class is compared. The class with the maximum value is the classification result. The specific steps of the WNB algorithm are given in Algorithm 1.

3.3. Incremental Learning of WNB Algorithm. In the WNB algorithm, the idea of incremental learning can reduce the performance requirements of the algorithm. The Bayesian classifier has a unique characteristic of incremental learning. A large part of the calculation process can be carried out in the way of incremental learning, which can reduce the time consumption of the algorithm. At the same time, the prediction performance of the classification algorithm is not affected by the completeness of training samples. Generally, the more complete the training samples are, the stronger their prediction ability and generalization ability are. In practice, the training samples of the classifier need to be gradually completed, and it is difficult to complete all the samples at one time. The incremental learning process of Bayes is to update the original class prior probability according to the new sample data $p(C_j)$ and attribute conditional probability $p(x_1 \mid C_j)$. The incremental learning of the algorithm is Algorithm 1:

The classifier does not need to retrain the classification model but only needs to import the newly added data into the classification model and modify the relevant parameters of the classifier. The modified equation of prior probability of Bayesian incremental algorithm is as follows:

$$P(C_j) = \begin{cases} \frac{N}{N+1} P(C_j) + \frac{N}{N+1}, & \text{when } c_t = c_j, \\ \frac{N}{N+1} P(C_j), & \text{when } c_t \neq c_j. \end{cases} \quad (11)$$

Likewise, the modified equation of the conditional probability of the Bayesian incremental algorithm is as follows:

$$P(C_j) = \begin{cases} \frac{N}{N+1} P(x_j \mid C_j) + \frac{N}{NC_j + 1}, & \text{when } c_t = c_j \text{ and } x_i = a_k, \\ P(x_j \mid C_j), & \text{when } c_t \neq c_j, \\ \frac{NC_j}{NC_j + 1} P(x_j \mid C_j), & \text{when } c_t = c_j \text{ and } x_i \neq a_k, \end{cases} \quad (12)$$

where $p(C_j)$ and $p(x_1 \mid C_j)$ are the class prior probability and attribute conditional probability updated after adding new samples, n is the total number of original data records, Nc_j is the total number of original data records means belonging to category C_j, and a_k is the value of a feature. In addition, when a new sample is added, it is necessary to count the number of samples of each category corresponding to the attribute value of the new sample set again. Combined with the statistical value of the old sample data, the correlation probability and noncorrelation probability value of the attribute are updated, to update the weight of each attribute. Using equations (9) and (10), and weighted

Input: test case to be classified
Step 1: scan all the training sample data and count the class label C. At the same time, the statistical attribute a, the number of samples whose value is AE, and the number of samples whose value is not a, are recorded in the count table.
Step 2: based on the information in the count table, for all attributes values, the correlation probability, and noncorrelation probability are calculated using equations (7) and (8), and the results are saved in the RP table.
Step 3: acquire the weight parameter. Using the information in the RP table, the weights of all attributes for various class labels are calculated using formula (9), and the weights are saved in the weight table.
Step 4: learn a priori probability. Using the number of classes in the count table, equation (4) is used to calculate the a priori probability of all class labels and save it to the correlation probability (CP) table. Meanwhile, equation (5) is used in the calculation of conditional probability for all attributes, and the results are saved in the conditional probability table (CPT) table.
Step 5: implement classification. When predicting the category of a test case, the CP table and the CPT table are called first, and then the corresponding value in the weight table is called using the specific value of each attribute. Finally, the posterior probability of the case belonging to each category is calculated using equation (10), and the maximum posterior probability is found out, and the class label is assigned.
Output class labels

Algorithm 1: Step of weighted Naïve Bayes classification.

Bayes equation (12), the category C_j of each data record x can be calculated.

4. Results

4.1. Comparative Analysis of Classification Accuracy of Traditional Machine Learning Algorithms and NB Algorithm. In this section, the five well-known machine learning classification algorithms are used to evaluate the existing teaching evaluation dataset, and the feasibility of the algorithm is judged by its accuracy. For each classification algorithm, we used the algorithm function provided by Sklearn, a machine learning library of the Python language. The default parameters were used to compare the experimental results. The dataset was divided into 220 training sets and 70 test sets. We applied 10-fold cross-validation in which the dataset was categorized into 10 equal parts. At each iteration, 1 part was used for testing, and the remaining 9 parts were used for training the classifier. The final results were averages as shown in Table 2.

The average classification results of these classification algorithms are shown in Figure 2.

For the same experimental dataset, the average computation time of each algorithm is shown in Table 3.

NB algorithm has high classification accuracy and the minimum running time on the dataset. The highest average computation time of 34.9 s was taken by the BP algorithm, whereas the NB algorithm took only 0.06 s to complete the classification task. Therefore, this validates the selection of the NB algorithm as the best candidate machine learning algorithm for the construction of the music teaching evaluation model.

4.2. Comparative Analysis of Classification Accuracy of NB and WNB Algorithms. We compared the performance of standard NB and WNB classifiers based on classification accuracy and competition time. All the 290 music samples were divided into 10 equal parts with 220 samples used for training and 70 samples for testing the classification performance. Using 10-fold cross-validation, the classification performance of the NB and WNB algorithms was measured. The results are given in Table 4.

The experimental results show that the average accuracy of the NB algorithm is 0.71%, whereas the average accuracy of the WNB is 0.75%. It can be observed that the classification accuracy of the WNB algorithm is better than that of the traditional NB algorithm in the music teaching evaluation dataset.

Since neural networks have rapidly gained attraction within the music fields [21] Among the different classes of the neural network, the backpropagation (BP) neural network is probably the most widely used. When the BP neural network algorithm processes the training dataset, it unifies the original score value (percentage system) into a decimal of the [0, 1] interval through data normalization and sets an error threshold to form a model to identify the evaluation level of the new sample data. We experimented with the BP algorithm, where all the 220 music samples were arbitrarily chosen as the training set and 70 data samples as the test set. According to the number of attributes, the input layer node was set to 10; the hidden layer node was 5; the output layer node was 1, the activation function was tanh(); the learning rate was 0.01; and the number of cycles was 10,000. After the neural network algorithm training, the test results are shown in Table 5.

Based on the above analysis, since in the actual evaluation process, the percentage system evaluation value given is generally very high, which leads to model overfitting, resulting in a high prediction level. Therefore, after preprocessing, the percentage score value was discretized into the evaluation value of the five grade system, and the data of different grades were randomly selected and mixed into the training dataset, in which the training set contained 220 and the test set contained 70 data samples. The results of the BP network and WNB algorithm were computed and are shown in Table 6 and Figure 3.

According to Table 6, the comparison chart of classification accuracy between the NB and WNB algorithms is shown in Figure 3.

Table 2: Record of classification accuracy.

	10-fold cross-validation										
Algorithm	1	2	3	4	5	6	7	8	9	10	Average
NB	0.8	0.757	0.8	0.743	0.757	0.8	0.757	0.729	0.729	0.771	0.764
SVM	0.671	0.586	0.671	0.757	0.757	0.7	0.7	0.657	0.743	0.714	0.695
KNN	0.686	0.757	0.671	0.743	0.671	0.686	0.686	0.671	0.743	0.686	0.7
DT	0.65	0.628	0.686	0.7	0.642	0.717	0.7	0.657	0.7	0.657	0.674
BP	0.771	0.728	0.714	0.814	0.7	0.714	0.7	0.628	0.729	0.729	0.723

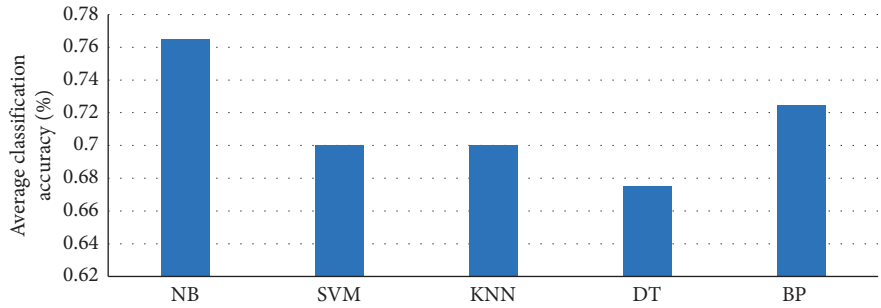

Figure 2: Classification accuracy of common algorithms.

Table 3: Average computation time (s).

Algorithm	NB	SVM	KNN	DT	BP
Average time consumption	0.068	28.961	3.353	0.131	34.908

Table 4: Classification accuracy of NB algorithm and WNB algorithm.

	10-fold cross-validation										
Algorithm	1	2	3	4	5	6	7	8	9	10	Average
NB	0.729	0.700	0.714	0.671	0.700	0.714	0.729	0.757	0.729	0.686	0.713
WNB	0.700	0.757	0.771	0.729	0.786	0.729	0.843	0.743	0.700	0.729	0.750

In the actual teaching evaluation dataset, because there are more excellent grades in the evaluation grade and fewer other grades when using hierarchical data to train the classification model when the extracted training datasets are different, the experimental results will be affected to a great extent. In this experiment, the average accuracy of the WNB algorithm is 0.767, and that of the BP algorithm is 0.683. The experimental results show that the performance of WNB is better than that of the BP algorithm. Likewise, the average computation time of WNB is 0.025 s, while the average computation time of the BP algorithm is about 34 s. Compared with the BP algorithm, the time consumption of the WNB algorithm is less, and the recognition accuracy is higher. Hence, the WNB algorithm has greater advantages in music teaching evaluation.

4.3. Experimental Analysis of WNB with Incremental Learning-Based WNB. We completed the construction of an incremental classification model based on WNB with 220 training samples and 70 test samples and gradually increased the training sample sets. The specific calculation results of test samples in each phase of random sampling increment are shown in Table 7.

Using incremental learning strategy, the calculation results are more inclined to the correct category, that is, the probability value of the correct category increases, while the probability value of other categories decreases. With incremental strategy, the changes of average classification accuracy are shown in Table 8, which shows a very small degradation in performance while adding more samples.

WNB algorithm with incremental learning referred to as adding WNB algorithm; using the same experimental dataset, the consumption time of WNB with incremental strategy is shown in Figure 4.

The introduction of the incremental model not only helps improve the classification model. At the same time, because the incremental model does not need to train and calculate the previously trained dataset again, it only needs to classify and calculate the newly added data, directly merge with the previous training values, and update the relevant parameters of the model, which saves time and increase the

Table 5: Some experimental results of the BP algorithm.

True evaluation value	Rank value	Predicted calculated value	Error range	Forecast level
0.98863	Excellent	0.927	0.060	Excellent
0.94318	Excellent	0.908	0.034	Excellent
1	Excellent	0.928	0.071	Excellent
0.97727	Excellent	0.946	0.031	Excellent
0.98863	Excellent	0.953	0.034	Excellent
0.96591	Excellent	0.924	0.041	Excellent
1	Excellent	0.918	0.081	Excellent
0.88636	Good	0.927	−0.041	Excellent
0.54545	Unqualified	0.819	−0.274	Good
0	Unqualified	0.291	−0.291	Unqualified

Table 6: Classification accuracy of BP and WNB algorithms.

Frequency	1	2	3	4	5	6	7	8	9	10	Average
BP	0.743	0.657	0.671	0.629	0.771	0.657	0.7	0.729	0.643	0.629	0.683
WNB	0.729	0.786	0.814	0.729	0.757	0.771	0.729	0.843	0.729	0.786	0.767

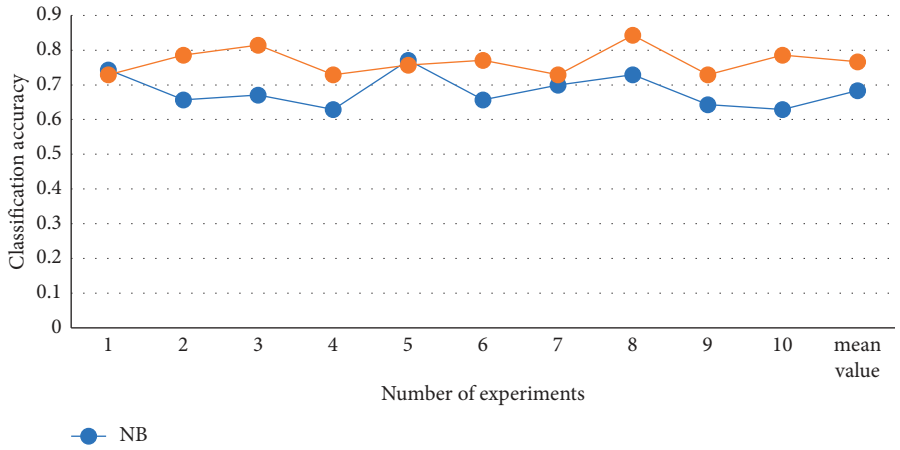

Figure 3: Comparison of classification accuracy of BP and WNB algorithms.

Table 7: Calculation results of test data.

Number of increments	50 more		Add 100		150 more		Add 180	
	Before increment	After increment	Before increment	After increment	Before increment	After increment	Before increment	After increment
Grade	Excellent		Excellent		Good		Good	
Excellent	$1.1e-09$	$2.5e-09$	$3.9e-08$	$2.1e-08$	$3.5e-06$	$2.9e-06$	$1.3e-05$	$1.2e-05$
Good	$1.5e-12$	$5.6e-15$	$7.2e-14$	$5.1e-13$	$2.9e-06$	$7.4e-06$	$1.9e-08$	$4.1-08$
Secondary	0.0	0.0	0.0	0.0	$3.0e-05$	$7.3e-06$	0.0	0.0
Qualified	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Unqualified	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Judgment result	Excellent	Excellent	Excellent	Excellent	Secondary	Good	Excellent	Excellent

Table 8: Classification accuracy before and after the increment.

Increment value	Before increment	Add 50	Add 100	Add 150	Add 180
Average accuracy	0.721	0.732	0.724	0.731	0.751

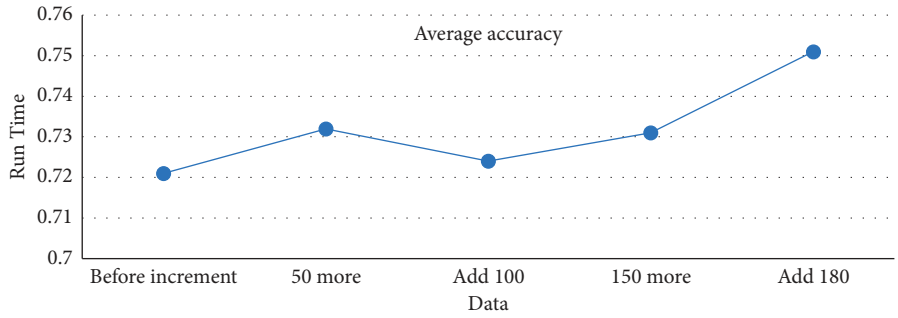

Figure 4: Algorithm running time comparison.

effectiveness of the classification model. Although the results of the proposed model are encouraging, more work is required to test the performance on other musical datasets and advanced machine learning algorithms.

5. Conclusion

This article explores the construction of a music teaching evaluation model using machine learning techniques. By comparing the potential of traditional classification algorithms, the weighted Naïve Bayes algorithm is presented as the baseline algorithm for the music teaching evaluation process to evaluate the music teaching effect. In addition, the concept of incremental learning is introduced to improve the classification performance of the weighted Naïve Bayes classifier. The experimental results show that the introduction of the incremental learning approach not only optimizes the performance of the classifier but also reduces the computation time of the algorithm. Results validate that the algorithm proposed in this paper is superior to other algorithms in the performance of music teaching evaluation.

Data Availability

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

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] F. Pachet and A. Zils, "Automatic extraction of music descriptors from acoustic signals," in *Proceedings of the International Symposium on Music Information Retrieval (ISMIR'04)*, Barcelona, Spain, October 2004.

[2] E. Pampalk, "A MATLAB toolbox to compute music similarity from audio," in *Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR'04)*, Barcelona, Spain, October 2004.

[3] M. Schedl, "An explorative, hierarchical user interface to structured music repositories," Master's Thesis, Vienna University of Technology, Vienna, Austria, 2003.

[4] C. E. Seashore, "The psychology of music," *Music Educators Journal*, vol. 23, no. 2, pp. 20–22, 1936.

[5] R. L. Allvin, "Computer-assisted music instruction: a look at the potential," *Journal of Research in Music Education*, vol. 19, pp. 131–143, 1971.

[6] C. N. Silla, C. A. A. Kaestner, and A. L. Koerich, "The latin music database: a database for the automatic classification of music genres (in Portuguese)," in *Proceedings of 11th Brazilian Symposium on Computer Music*, pp. 167–174, So Paulo, Brazil, September 2007.

[7] K. Kosina, "Music genre recognition," MSc. Dissertation, Fachschule Hagenberg, Hagenberg, Austria, 2002.

[8] T. Knight, F. Upham, and I. Fujinaga, "The potential for automatic assessment of trumpet tone quality," in *Proceedings of the International Society of Music Information Retrieval Conference (ISMIR)*, pp. 573–578, Miami, FL, USA, October 2011.

[9] C. Dittmar, E. Cano, J. Abeßer, and S. Grollmisch, "Music information retrieval meets music education," in *Multimodal Music Processing*, M. Müller, M. Goto, and M. Schedl, Eds., Vol. 3, Springer, Berlin, Germany, 2012.

[10] J. Abeßer, J. Hasselhorn, C. Dittmar, A. Lehmann, and S. Grollmisch, "Automatic quality assessment of vocal and instrumental performances of ninth-grade and tenth-grade pupils," in *Proceedings of the International Symposium on Computer Music Multidisciplinary Research (CMMR)*, Marseille, France, October 2013.

[11] Y. J. Luo, L. Su, Y. H. Yang, and T. S. Chi, "Detection of common mistakes in novice violin playing,," in *Proceedings of the International Society of Music Information Retrieval Conference (ISMIR)*, pp. 316–322, Taipei, Taiwan, October 2015.

[12] B. Bozkurt, O. Baysal, and D. Yuret, "A dataset and baseline system for singing voice assessment," in *Proceedings of the International Symposium on Computer Music Multidisciplinary Research (CMMR)*, pp. 430–438, Matosinhos, Portugal, September 2017.

[13] Y. Han and L. Lee, "Hierarchical approach to detect common mistakes of beginner flute players," in *Proceedings of the International Society of Music Information Retrieval Conference (ISMIR)*, pp. 77–82, Curitiba, Brazil, November 2014.

[14] C. W. Wu and A. Lerch, ""Learned features for the assessment of percussive music performances," in *Proceedings of the International Conference on Semantic Computing (ICSC)*, Laguna Hills, CA, USA, February 2018.

[15] X. Yu, J. Yang, and Z. Xie, "Training SVMs on a bound vectors set based on fisher projection," *Frontiers of Computer Science*, vol. 8, no. 5, pp. 793–806, 2014.

[16] X. Yu, D. Zhan, L. Liu, H. Lv, L. Xu, and J. Du, "A privacy-preserving cross-domain healthcare wearable recommendation algorithm based on domain-dependent and domain-independent

feature fusion," *IEEE Journal of Biomedical and Health Informatics*, 2021.

[17] B. Whitman, D. Roy, and B. Vercoe, "Learning word meanings and descriptive parameter spaces from music," in *Proceedings Of the HLT-NAACL03 Workshop on Learning Word Meaning From Non-linguistic Data*, Edmonton, Canada, March 2003.

[18] P. Knees, E. Pampalk, and G. Widmer, "Artist classification with web-based data," in *Proceedings Of the International Symposium On Music Information Retrieval (ISMIR'04)*, Barcelona, Spain, October 2004.

[19] D. Li, I. K. Sethi, N. Dimitrova, and T. McGee, "Classification of general audio data for content-based retrieval," *Pattern Recognition Letters*, vol. 22, no. 5, pp. 533–544, 2001.

[20] B. Whitman and P. Smaragdis, "Combining musical and cultural features for intelligent style detection," in *Proceedings of the 3rd International Conference on Music Information Retrieval ISMIR*, Paris, France, October 2002.

[21] M. Yu, T. Quan, Q. Peng, X. Yu, and L. Liu, "A model-based collaborate filtering algorithm based on stacked AutoEncoder," *Neural Computing and Applications*, vol. 10, pp. 1–9, 2021.