Journal of Theoretical and Applied Information Technology
31st December 2021. Vol.99. No 24

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

# RECURRENT NEURAL NETWORK FOR THE CLASSIFICATION OF AUDIO SIGNAL COMING FROM OUD INSTRUMENT

**[1]MEHDI ZHAR, [2]OMAR BOUATTANE, [3]LHOUSSAIN BAHATTI**

Ecole Normale Supérieur de l'Enseignement Technique, SSDIA Laboratory, Mohammedia, University

Hassan II, Morocco

E-mail: [1]mehdi.zhar@gmail.com

**ABSTRACT**

In this paper, we propose a model for recognizing the playing style of the oud player according to the grandmasters, and we identify the influence rates of each oud player concerning a way to play, this solution is based on the extraction of technical characteristics of the audio signals through the signal processing mechanism. Standardization, data reduction via mathematical tools as well as a selection procedure of the optimal characteristics are also started in this work in order to allow the proposed classification model to generate a better result. After having modelled and tested several classification models of deep learning, we evoke in these papers the most adequate model answering perfectly this problem of classification of an audio signal coming from the musical instrument Oud.

Practical cases have also been developed in this work to test the relevance and efficiency of our model.

**Keywords:** *Artificial Intelligence, Classification, RNN, Music, Quarter tone, Deep Learning, Machine learning, Filtering, MFCC, Python.*

## 1. INTRODUCTION

In recent centuries, computer science has evolved greatly. It has so affected people's lives that much of our day-to-day work uses informatics in one way or the other. We have succeeded in creating numerous technical wonders through computer science. The increase in the management of large companies, social networking sites, and much more was made possible. How would life be without it is difficult to imagine? However, the study of computer science must not be stopped there and the opportunities continue to be explored. Multiple things including music have included computer science. Music is an art form that forms a major piece of our cultural work. With time it is only logical that we discover computer science's musical possibilities. That's why we are going to evoke in this work, a study based on an instrument of Arabic music which is called Oud.

The Oud is the guitar ancestor. It is an instrument of strings used in much oriental traditional music. The Oud and its origins are an exotic story. Ouds are around three thousand years old and the oldest known. Ziryab is thought to have added the fifth string to the Oud. The Oud player today plays usually the same tool as Ziryab, with just one slow change: a sixth string. In the 20th century, Munir Bashir launched the sixth string.

In oriental music for Oud, we have the Iraqi school, the oriental school, and modern school, three main schools are available. Every school represents a different and unique playstyle. How the musician puts his fingers in the handle of the Oud, the manner he holds the feather, the power of the strike on the strings, the motion of the feather, the series of notes and the speed of the game, and everything - the mixing, and more - of techniques, all of these represent the style and way of playing each oud player, regardless of the genre of the oud.

The Grandmaster of each Oud School is:

- Iraqi school: Munir BASHIR.
- Oriental school: Farid EL ATTRACHE.
- Modern school: Naseer SHAMMA.

In Artificial Intelligence, the application of classifying algorithms based on deep learning produces good recognition results, which is why we have chosen a classification model based on a recurrent neural network RNN classification algorithm. Our approach aims to propose an audio system that can generate pertinent features of the three Oud masters, and then detect the different playing styles by highlighting the gaps and differences that exist, and then predict the impact of these masters' playing styles on the Oud play.

The rest of this paper has the following structure: Section 2 describes previous work as the latest in the collection and selection of features for automated audio classifications. In Section 3 we will discuss the conceptual solution consisting of audio segmentation, characteristics of extraction, standardization of data, and detail the strategy of dimensional reductions and the conception of the classifier model. Section 4 provides details of the realistic application. Section 5 is reserved for the analysis of the results. Finally, we conclude the paper in Section 6 and discuss future relevant research.

## 2. RELATED WORK

Many works in literature explain how to identify a singer without discriminating between instrumental and singing sounds. [1] We consider an investigation into how artificial neural networks can be trained on a large corpus of melodies and converted into automatic music composers capable of providing new phrases that are consistent with the style on which they were trained, and [2] we consider an examination into how artificial neural networks could be trained on a large corpus of phrases and transformed into audio data composers able to produce new compositions that are compatible with the genre on which they were trained. On [3], a series of sinusoidal descriptors are described in order to characterize a musical signal and recognize the maximum information included in that signal. Various learning techniques were developed and tested in [4, 5, 6, 7, 8] to accomplish audio identification work. Digital music consumers typically maintain their song library on their hard drives. The duplication of music files may be an issue for them. This isn't optimal because they take up unnecessary disk space. There are already apps detecting duplicate music files, but something remains to be desired. It provides the approach of cognitive and building learning (CCL). The Digital Archiving Model has been developed (DMAM). They compare it to existing applications that detect duplicate music files for testing the efficiency of their model. The analysis shows that DMAM is better than the rest [9]. In the learning process [10, 11, 12, 13, 14] the neural networks are going to be very helpful. The algorithmic composition model [15] provides a deep (multilayer) method of monophonic melodies based on neural RNN networks with gated recurrent units (GRUs). However, the RFE-SVM is egoism, which only seeks to determine the optimal combination of classification [16]. There appears to have been a consensus among many techniques, based on its versatility, computational effectiveness, the capability of handling high-density data and the revenue for selection of characteristics, using Superstar Vectors (SVM) [17, 18]. The [19] model is an efficient audio classification system based on SVM to recognize the composer. In [20] authors present an algorithm allow to artificially compose oriental music based on calculated features. In [21], we find a new mechanism to classify the influence of oud master on the way of play of oud player, this model is based on KNN algorithm. In [22], we find an artificial algorithm of oriental music composition based on oriental gamma. Three experiments were carried out in [23], The first consists of checking the learning architecture. The second checks the system's features. The latter tests usability and listening. The results show that the melodies generated by the system are good and follow a unique style. In future, they would like the system to take emotions into account as another input for music generation. They would also like to introduce a way to include musical styles which do not rely on tagged MIDI files. In addition, they would like to enhance user interaction by incorporating other devices like mouse or joystick. The authors of [24] consider their model to be a great alternative to RNNs. They are planning to expand MidiNet into multiple tracks, speeds, and breaks. They also hope that the model can implement music theory and receive data from the recuperation models for music information. A system is presented that learns the architecture and synthetizes musical variations of the voice recording in an unattended way of a rhythmic percussion fragment is illustrated in [25]

## 3. PROPOSED MODEL

Our proposal focuses on six key components, namely audio segmentation, mathematical attributes analysis extraction, standardization and data normalization, attributes selection, and the use of the greatest precise algorithm in depth classification method RNN and finally the predict operation. The diagram illustrating our plan appears in Figure 1.

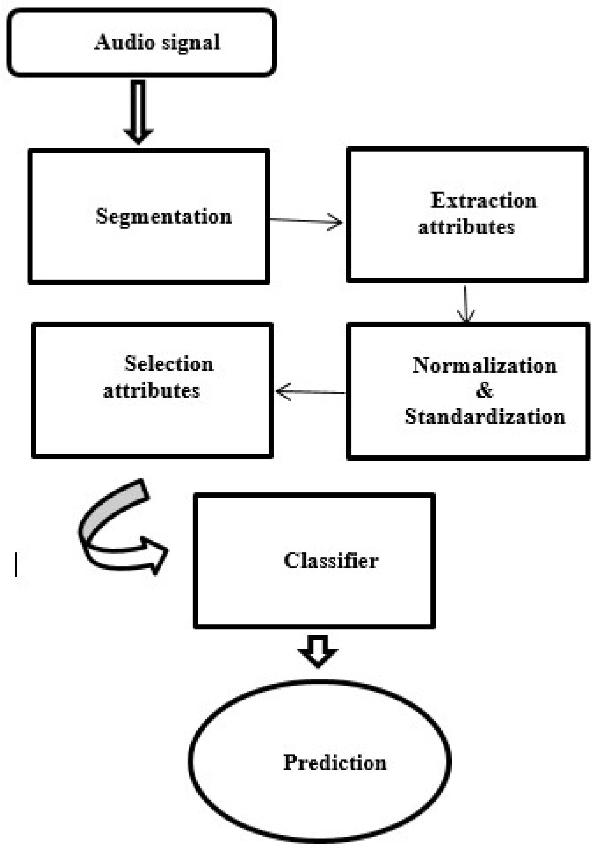


*Figure 1: Block Diagram Of Our Classification Process*

### 3.1 Audio Segmentation

The fragments are divided into different periods between 5 seconds and 100 seconds then the whole of the proposed model classification algorithm has been completed in a number of tests, in order to build the maximum precision time frames. Duration with the perfect time is 5 seconds.

### 3.2 Extraction Features

Our method consists in exploring the parameters of an audio signal via mathematical equations of signal processing. A lot of information has been extracted with the help of signal processing elements such as Zero Crossing Rate, Energy, Entropy of Energy, Spectral Centroid, Spectral Spread, Spectral Entropy, Spectral Flux, Spectral Rollof, MFCCs, Chroma Vector, Chroma Deviation.

*Table 1: Extracted Features*

| Features ID | Features Name | Description |
|---|---|---|
| 1 | Zero-Crossing Rate | The rate of signal transition over the length of a given frame. |
| 2 | Energy | The sum of the signal values squares, multiplied by the length of the corresponding frame. |
| 3 | Entropy of Energy | The entropy of normalized energies of the sub frames. It can be construed as a measure of sudden changes. |
| 4 | Spectral Centroid | The optical center of gravity. |
| 5 | Spectral Spread | The spectrum's second Key Moment. |
| 6 | Spectral Entropy | The entropy of the spectral energies is uniform for a sub frame package. |
| 7 | Spectral Flux | The square difference of the two successive frames between the normalized magnitudes |

| | | |
|---|---|---|
| | | of the spectra. |
| 8 | Spectral Rollof | The frequency at this is centered 90 percent of the spectrum's magnitude range. |
| 9 –21 | MFCCs | Mel Frequency Cepstral Coefficients form a cepstral representation where the frequency bands are measured according to the mel-scale rather than linear ones. |
| 22-33 | Chroma Vector | A 12-element representation of the spectral energy where the bins reflect the 12 groups of Western-type music (semitone spacing) in equal-tempered pitch. |
| 34 | Chroma Deviation | The standard deviation of coefficients of 12 chromiums. |

In this work, we use 26 features:

Chroma_stft, rmse, spectral_centroid, spectral_bandwidth, rolloff, zero_crossing_rate, mfcc from 1 to 20. This decision is made after testing several combinations and also for the aim to choose only features characterizing the form of the signal.

Data from signals of oriental music created by the instrument of the lute were extracted in the work. These signals are relatively short, non-stationary, with near percussive sonic characteristics. The extraction elements are the following: Based on short-term Fourier transformations, cepstral characteristics are computed according to the spectral properties:

$$S(t,F_k)=\int_{-\infty}^{+\infty} S(\tau-t)w(\tau)\exp(-j2\pi.F_k.\tau)d\tau \qquad (1)$$

All figures produced by the extraction features were calculated on average to standardize the obtained results, optimize processing operation, and increase model efficiency:

$$Average\ (s(t)) = \frac{sum\ (s(t))}{number\ (s(t))} \qquad (2)$$

### 3.3 Normalization & Standardization

Auto-learning algorithms will not function correctly without normalization. The range of all entities must therefore be normalized to ensure that each entity leads to the final interval approximately and proportionately.

We converted then the data to a level [0, 1] using the formula in order to make better use of the information's generated and start reducing the range of values:

Xsc is the normalized value, where X is an original value.

$$X_{sc} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \qquad (3)$$

### 3.4 Selected features

In this section, we will study two types of feature selection mechanisms. the first is simple filtering using the Univariate Feature Selection model, and the second is Wrapper using the Forward Selection model.

#### 3.4.1 The univariate feature selection

In this model, the Univariate Feature Selection uses certain statistical tests, such as chi-square, F-test, Mutual information to determine the force of the connection among each individual factor and the target variable. The characteristics are classified following their strength with the results. All functions are deleted from the current function space, other than a predisposed number of markers. The other characteristics are then used for the training, testing, and validation of machine models.

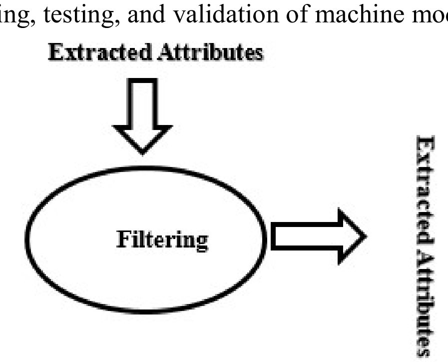


*Figure 2: The Features Selection Approach.*

Univariate selection of features aims to form the strength of the relationships among each characteristic and the objective variable, using some statistical analyses, such as chi-square, F-test, Mutual information, etc. Characteristics are classified according to their strength with the result. Information from the current area are taken away from all features other than a pre-determined number of markers. The remainder is used in the training, testing, and validation of models of the machine. Univariate Feature Selection is used as a pre-processor before the data set is covered by an

Journal of Theoretical and Applied Information Technology
31st December 2021. Vol.99. No 24

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

estimator model. The two statistical tests used in our research are F-test and Mutual information.

F-test: hypothesis testing is performed to determine if the feature and the target variable differ significantly. The correlation between the function and the goal is computed and transformed to the p-value of the F-score. For modelling, characteristics with a high F score are chosen.

$$F = \frac{\text{VBT}}{\text{VWT}} \quad (4)$$

VBT: the variance between treatments.
VWT: the variance within treatments.

$$F = \frac{\text{MS}_{\text{Treatments}}}{\text{MS}_{\text{Error}}} = \frac{SS_{Treatments}/(I-1)}{SS_{Error}/(n\tau - I)} \quad (5)$$

The variance between treatments is:

$$\sum_{i=1}^{k} n_i(\overline{Y}_i - \overline{Y})^2/(k-1) \quad (6)$$

Where Yi is the group average, neither is the number of observations in the groups, Y is the overall average of the data and K is the group number.

Treatment variance is:

$$\sum_{i=1}^{k}\sum_{j=1}^{ni}(Y_{ij} - \overline{Y_i})^2/(N-K) \quad (7)$$

In the jth in K groups and N, the sample is the total dimension, where Yij is the jth observation.

In order to generate the highest precision, the best compromise is 26 features, a test was carried out on the number of features from 1 to 26

#### *3.4.2 The univariate feature selection*

We start with the null model with the forward selection, and then we move the model one by one with each feature and select the function with a minimum p-value. Now fit a model with two characteristics by trying to combine the previous function with all the other functions. Select the minimum p-value feature again. Again. Now a model with three functions is adapted by attempting to combine two features previously selected with other features. Repeat it until we have a set of features with a p-value of features below the level of significance.

In short, the following steps are taken for the forward method:

1- Select the level of meaning, e.g. SL = 0.05, 95 percent trust.
2- Place all possible simple regression models with one feature at a time. There can be total n models. Choose the lowest p-value feature.
3- Added to the previously selected feature all models with one extra feature (s).
4- Select the minimum p-value feature again. Again. Then step 3 will go if p-value < meaning level, the process will end otherwise.

### 3.5 Classification Algorithm

In this part, we chose to design a deep learning-based classifier to produce performance with greater accuracy after passing all segmentation, extraction, normalization, and parameter selection steps.

after modeling and testing several models of deep learning algorithms, we finally obtained good results by designing a recurrent neural network (RNN) model based on 4 hidden layers and a single output layer. the hidden layers contain 100 nodes each and the output layer contains 3 nodes.

in input and for each training pulse, a vector of 26 elements is propagated in the neural network composed of 33 303 components, the table below shows in detail the composition of this proposed neural network.

*Table 2: Parameters of RNN Model*

| Layer | Output Shape | Params |
|---|---|---|
| Fc1 (Dense) | (None, 100) | 2 700 |
| Fc2 (Dense) | (None, 100) | 10 100 |
| Fc3 (Dense) | (None, 100) | 10 100 |
| Fc4 (Dense) | (None, 100) | 10 100 |
| Output (Dense) | (None, 3) | 303 |

Total parameters: 33 303, Trainable parameters: 33 303, Non trainable parameters: 0.

We specified a Sequential nature of our architecture so that layers of neurons will be added sequentially and we specified the Dense operation so that all neurons in the previous layer will be connected to all neurons in the next layer.

A mathematical equation used on a signal is an activation function. If the stimulation threshold is reached, information is transmitted or not. Specifically, the role is to decide if the neuron's

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

response is activated or not. The following function is used by a neuron only:

X = ∑ (input * weight) + biais

For hidden layers, we used the Relu activation operation:

F(x)=max(0,x)

And for the output layer we used Softmax activation operation:

$$\sigma(z) = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}} \quad for\, i = 1,\ldots, K$$

$$and \qquad (8)$$

$$z = (z_1,\ldots, z_k) \in R^k$$

and for the optimization, we opted for the use of Adam's algorithm with a step of 0.01.

the figure below illustrates the proposed RNN Model.

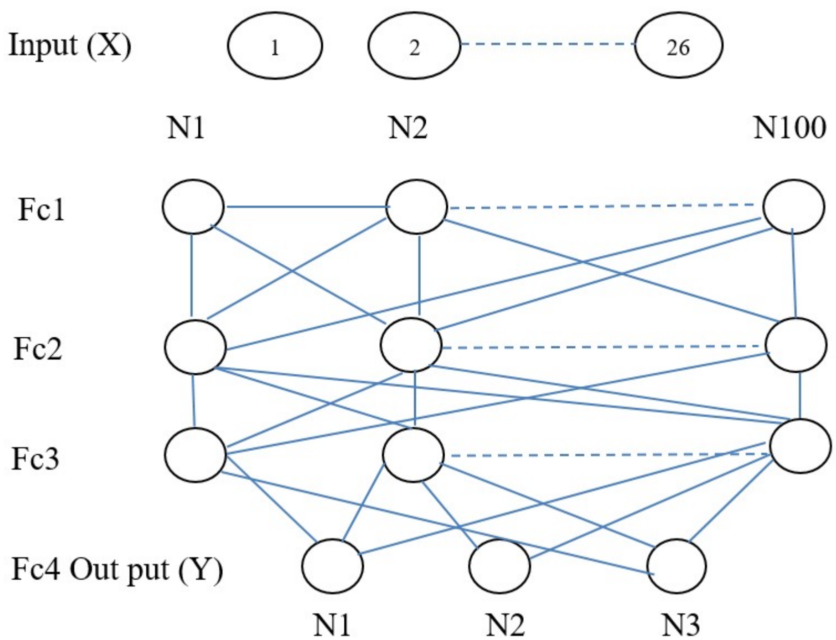


*Figure 3: Proposed RNN model.*

For training and testing, the data was split into 80% and 20%. After testing several training opportunities between 10% and 90%, this division is the best solution.

## 4. IMPLEMENTATION AND RESULTS

This article uses data from the monophonic songs of the instrument played by three world-famous masters of oud:

- FARID EL ATTRACHE, a purely eastern, traditional style defined by a sound depth, strength, and note series.

- MUNIR BACHIR - which is an Iraqi play school inspired by its master HAIDAR's Turkish play style - is defined by sharp notes, economic plum, and inverted feather.

- NASSER SHAMMA: Inspired by Western guitar playing and Iraq's school, this performance represents the modern oud school and is defined by the sharpness of the notes, the use of the chords, and arpeggios in the play.

For this method, all the musical and monophonic works of these 3 lute stars were needed. In order to achieve this model.

The following table shows the pieces used:

*Table 3. Audio Signal Fragments*

| Oud master | Duration | Input |
|---|---|---|
| Farid EL ATTRACHE | 02:15:34 | 1 629 |
| Mounir BACHIR | 05:07:57 | 3 700 |
| Nasseer Shamma | 05 : 03 : 09 | 3 645 |

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

After testing a series of differentiation times between 2 and 20, we chose a 5 second for every piece to discover the perfect choice.

The table below illustrates the accuracy according to the number of features.

Table of accuracy according to selected features (100 epochs):

*Table 4: Filter Selected Method*

| Number of features | Accuracy | Loss |
|---|---|---|
| 1 | 40% | 83% |
| 2 | 55% | 50% |
| 3 | 56% | 45% |
| 4 | 65% | 44% |
| 5 | 77% | 34% |
| 6 | 90% | 24% |
| 7 | 90% | 24% |
| 8 | 90% | 21% |
| 9 | 91% | 19% |
| 10 | 92% | 18% |
| 11 | 93% | 15% |
| 12 | 95% | 14% |
| 13 | 95% | 12% |
| 14 | 96% | 10% |
| 15 | 97% | 7% |
| 16 | 98% | 5% |
| 17 | 98% | 4% |
| 18 | 98% | 5% |
| 19 | 98% | 5% |
| 20 | 98% | 5% |
| 21 | 98% | 4% |
| 22 | 98% | 4% |
| 23 | 98% | 4% |
| 24 | 98% | 4% |
| 25 | 98% | 4% |
| 26 | 99% | 3% |

The table below illustrates the accuracy generated by using a combination of all extracted features following a wrapper method.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

*Table 5: Results of Wrapper Selected Method*

| N | Accuracy | feature names |
|---|---|---|
| 1 | [0.61145166] | ('13',) |
| 2 | [0.71008638] | ('8', '13') |
| 3 | [0.79548621] | ('2', '8', '13') |
| 4 | [0.84995821] | ('2', '8', '13', '15') |
| 5 | [0.87322374] | ('2', '8', '11', '13', '15') |
| 6 | [0.88604068] | ('2', '8', '11', '13', '15', '20') |
| 7 | [0.89063806] | ('2', '6', '8', '11', '13', '15', '20') |
| 8 | [0.9005294] | ('2', '6', '8', '11', '13', '15', '20', '21') |
| 9 | [0.90902758] | ('2', '6', '8', '11', '13', '15', '17', '20', '21') |
| 10 | [0.9148788] | ('2', '6', '8', '11', '13', '15', '17', '20', '21', '22') |
| 11 | [0.9215659] | ('2', '6', '8', '11', '13', '15', '16', '17', '20', '21', '22') |
| 12 | [0.92198384] | ('0', '2', '6', '8', '11', '13', '15', '16', '17', '20', '21', '22') |
| 13 | [0.93535804] | ('0', '2', '6', '8', '9', '11', '13', '15', '16', '17', '20', '21', '22') |
| 14 | [0.93507941] | ('0', '2', '6', '8', '9', '11', '13', '15', '16', '17', '18', '20', '21', '22') |
| 15 | [0.94524937] | ('0', '2', '6', '8', '9', '11', '13', '15', '16', '17', '18', '20', '21', '22', '24') |
| 16 | [0.94399554] | ('0', '2', '3', '6', '8', '9', '11', '13', '15', '16', '17', '18', '20', '21', '22', '23') |
| 17 | [0.93229312] | ('0', '2', '3', '6', '8', '9', '11', '12', '13', '15', '16', '17', '18', '20', '21', '22', '23') |
| 18 | [0.94775704] | ('0', '2', '3', '6', '8', '9', '10', '11', '12', '13', '15', '16', '17', '18', '20', '21', '22', '23') |
| 19 | [0.94719978] | ('0', '2', '3', '6', '7', '8', '9', '10', '11', '12', '13', '15', '16', '17', '18', '20', '21', '22', '23') |
| 20 | [0.95110059] | ('2', '3', '6', '7', '8', '9', '10', '11', '12', '13', '15', '16', '17', '18', '19', '20', '21', '22', '23', '24') |
| 21 | [0.95082196] | ('0', '2', '3', '6', '7', '8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '20', '21', '22', '23', '24') |
| 22 | [0.94970744] | ('0', '2', '3', '5', '6', '7', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20', '22', '23', '24', '25') |
| 23 | [0.94747841] | ('0', '1', '2', '3', '5', '6', '7', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20', '22', '23', '24', '25') |
| 24 | [0.95207579] | ('0', '1', '2', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13', '14', '15', '17', '18', '19', '20', '21', '22', '23', '24', '25') |
| 25 | [0.95430482] | ('0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20', '21', '23', '24', '25') |
| 26 | [0.94246308] | ('0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20', '21', '22', '23', '24', '25') |

the best result obtained is 95% accuracy by using just 2 epochs for 25 features in the input.

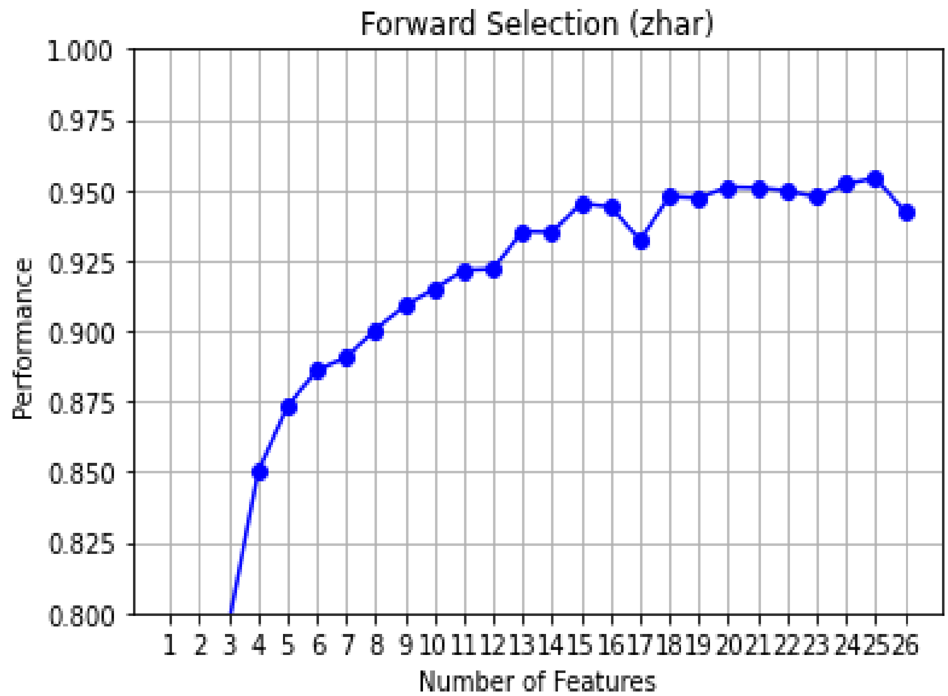


*Figure 4: Result of Forwarding Selection Method*

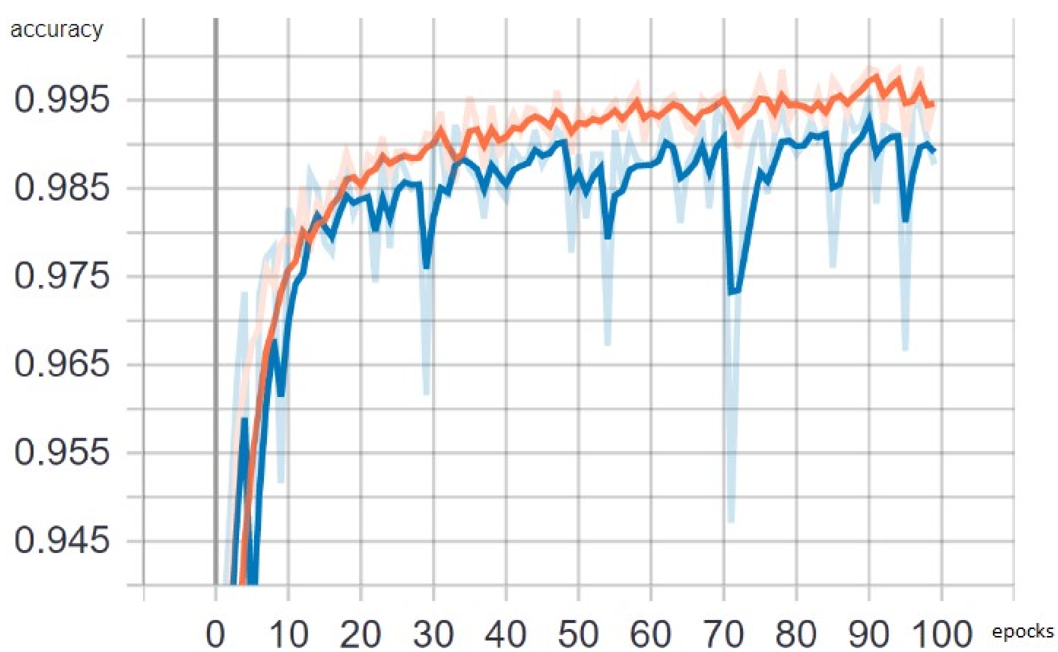


*Figure 5: Accuracy by Using Wrapper Method for 100 Epochs.*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

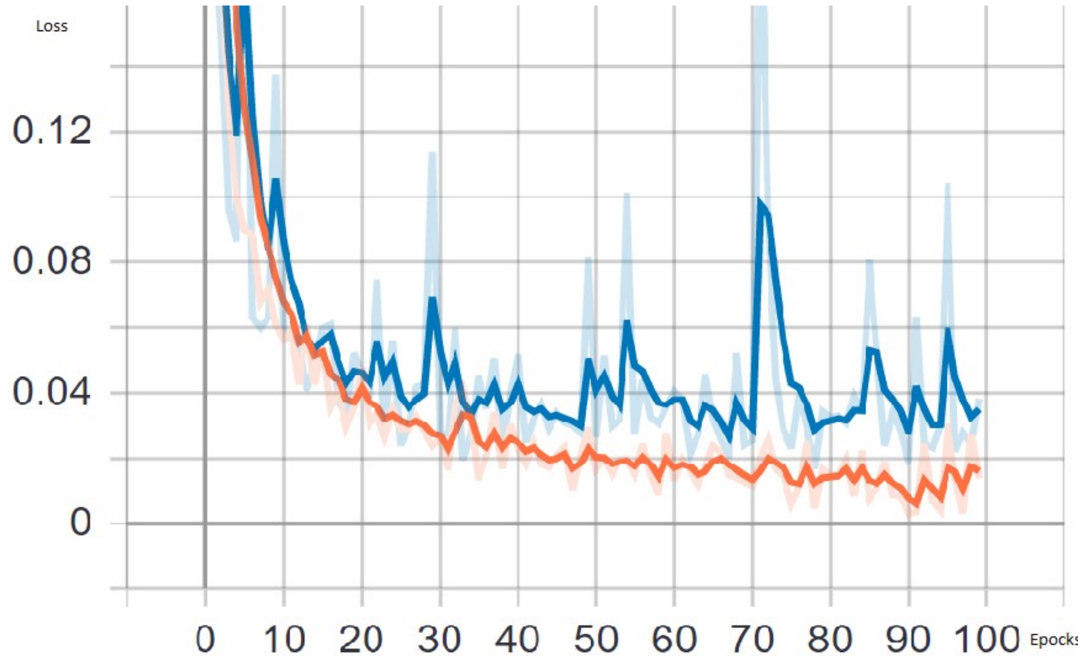


*Figure 6: Loss by Using Wrapper Method for 100 Epochs.*

The graphs above show that the best results obtained are established using the wrapper method. the best result obtained is 99% accuracy and 0.01% loss for 25 features for the input and 100 epochs.

Test of Overfitting:

To ensure the efficiency of our model, it was essential to carry out over-fitting tests, and this can only be done by injecting new data from the same oud masters to pass through the prediction mill. These data were excluded during the training and testing phases.

We injected several pieces which we divided into periods of 5 seconds. and then after generating the prediction, we average the results of the small pieces to efficiently classify them. The table below is an example of the overfitting test. For 3 pieces, the first of the class Farid Attrache, the second of the class Mounir Bashir and the third class of Nasser Shamma and after subsequently return the input values on a scale between 0 and 1, the prediction results obtained show the efficiency of our model and we were able to correctly classify the pieces.

*Table 6: Results of Over Fitting Test*

| Chroma stft | Other Features | mfcc20 | Real Result | Output | Estimated Result | Average |
|---|---|---|---|---|---|---|
| 0.36109071031120654 | - - - | -6.60006457518616 | 1 | KO | 3 | 80% |
| 0.4131565986507775 | - - - | -6.606676027484935 | 1 | OK | 1 | |
| 0.3485774676838411 | - - - | -0.8315178298003787 | 1 | OK | 1 | |
| 0.37546553777755026 | - - - | -3.647695604942445 | 1 | OK | 1 | |
| 0.3913145356911742 | - - - | -2.560305582048746 | 1 | OK | 1 | |
| 0.26173345226130634 | - - - | 3.2824009359810966 | 2 | KO | 3 | 80% |
| 0.23724080509222148 | - - - | -0.5246794680157273 | 2 | OK | 2 | |
| 0.2078389875095794 | - - - | 2.311503433246989 | 2 | OK | 2 | |
| 0.28487614560848507 | - - - | -2.1498272157872775 | 2 | OK | 2 | |
| 0.2254081164219299 | - - - | 5.0687324208512035 | 2 | OK | 2 | |
| 0.27694082085467187 | - - - | -16.496840992020545 | 3 | OK | 3 | 100% |
| 0.27694082085467187 | - - - | -16.496840992020545 | 3 | OK | 3 | |
| 0.27694082085467187 | - - - | -16.496840992020545 | 3 | OK | 3 | |
| 0.27694082085467187 | - - - | -16.496840992020545 | 3 | OK | 3 | |
| 0.27694082085467187 | - - - | -16.496840992020545 | 3 | OK | 3 | |

In this section, we have evaluated our model's relevance by using pieces from others oud players to classify the rate of influence in the way of playing of these oud Players. The effects were computed based on the sum of the predictive results of all elements and were performed by four songs of 100 songs each for five seconds. Some of the results achieved are described in the table below.

*Table 7. The Rate of the Influence in Way to Play of Oud Players*

| Musicians | Farid EL ATTRACHE | Mounir BACHIR | Naseer SHAMMA |
|---|---|---|---|
| Artist 1 | 70% | 19% | 11% |
| Artist 2 | 56% | 34% | 10% |
| Artist 3 | 3% | 67% | 30% |
| Artist 4 | 22% | 30% | 48% |

## 5. RESULTS ANALYSIS

This work, which is based on an RNN classification model, is an evolution of the classification work developed by [21], using machine learning algorithms, namely: KNN, Decision Tree, SVM. the results obtained are much more improved, especially with the modeling of a new data filtering mechanism. the table below describes the improvements in terms of accuracy rate.

*Table 8. Results Analysis*

| | SVM (with simple Filter) | Decision Tree (with simple Filter) | KNN (with simple Filter) | RNN (with Wrapper Filter) |
|---|---|---|---|---|
| Accuracy | 74.85 | 81.15% | 95.22% | 99% |

on the other hand, this field of research remains very limited in terms of publications, especially in the section which analyzes Arabic music. the modes of Arab music are numerous and much richer in terms of expression than Western music, given the presence of a quarter of the tone, which requires particular theoretical modeling.

## 6. CONCLUSION

In this paper, we have proposed a solution to the problematic of classifying the audio signals coming from the Oud instrument, we presented a methodology to recognize the influence of grandmasters on the playing style of oud players using state-of-the-art techniques for artificial intelligence, signal processing, and mathematical tools. For the aim to select only relevant data that could bring good results, we performed data extraction, normalization, and filtering methods. The information was processed by the deep RNN algorithm based on learning, compared to previous states as well as in terms of eliminated attributes and irrelevant values, RNN is much better than machine learning algorithms (KNN, SVM, Decision tree). Higher accuracy was achieved by using a multi-step training mechanism and using a set of test sets based on the percentage of data used for training and thus filtering the columns in the dataset by using the wrapper method. The results obtained were also based on test simulations by oud players by predicting the results and the influence rate. The case studies showed the effectiveness and relevance of our approach. The rate of influence of oud masters on oud players was measured using several played pieces. Each Oud player is influenced by one or more schools of playing and this will also help the Oud players to know their weaknesses concerning what they wish to improve. We intend to investigate the use of other audio features in the future, and explore other feature selection algorithms, as well as generalize this model to the study of other issues.

**ACKNOWLEDGMENT**

The H2020 Project SybSPEED, N 777720, supports this work.

**REFRENCES:**

[1] T. Ratanpara and N. Patel, “Singer identification using perceptual features and cepstral coefficients of an audio signal from Indian video songs,” EURASIP J. Audio Speech Music Process., vol. 2015, no. 1, p. 16, Jun. 2015, doi: 10.1186/s13636-015-0062-9.

[2] F. Colombo, S. Muscinelli, A. Seeholzer, J. Brea, and W. Gerstner, “Algorithmic Composition of Melodies with Deep Recurrent Neural Networks,” Jun. 2016. doi: 10.13140/RG.2.1.2436.5683.

[3] “(PDF) Sinusoidal features extraction: Application to the analysis and synthesis of a musical signal.” https://www.researchgate.net/publication/261120523_Sinusoidal_features_extraction_Application_to_the_analysis_and_synthesis_of_a_musical_signal (accessed Aug. 05, 2021).

[4] P. Herrera Boyer, G. Peeters, and S. Dubnov, “Automatic classification of musical instrument sounds,” 2003, doi: 10.1076/jnmr.32.1.3.16798.

[5] I. Kaminsky and A. Materka, “Automatic source identification of monophonic musical instrument sounds,” Dec. 1995, pp. 189–194 vol.1. doi: 10.1109/ICNN.1995.488091.

[6] G. Peeters, “Automatic Classification of Large Musical Instrument Databases Using Hierachical Classifiers with Inertia Ratio Maximization,” Oct. 2003.

[7] P. Dhanalakshmi, S. Palanivel, and V. Ramalingam, “Classification of audio signals using SVM and RBFNN,” Expert Syst. Appl., vol. 36, no. 3, Part 2, pp. 6069–6075, Apr. 2009, doi: 10.1016/j.eswa.2008.06.126.

[8] G. Guo and S. Z. Li, “Content-based audio classification and retrieval by support vector machines,” IEEE Trans. Neural Netw., vol. 14, no. 1, pp. 209–215, Jan. 2003, doi: 10.1109/TNN.2002.806626.

[9] W. Dai and K. Yu, "Contestability in the Digital Music Player Market," J. Ind. Compet. Trade, vol. 19, Jun. 2019, doi: 10.1007/s10842-018-0284-5.

[10] F. Gers, J. Schmidhuber, and F. Cummins, "Learning to Forget: Continual Prediction with LSTM," Neural Comput., vol. 12, pp. 2451–71, Oct. 2000, doi: 10.1162/089976600300015015.

[11] A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, May 2013, pp. 6645–6649. doi: 10.1109/ICASSP.2013.6638947.

[12] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," ArXiv14123555 Cs, Dec. 2014, Accessed: Aug. 05, 2021. [Online]. Available: http://arxiv.org/abs/1412.3555

[13] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," in Advances in Neural Information Processing Systems, 2014, vol. 27. Accessed: Aug. 05, 2021. [Online]. Available: https://papers.nips.cc/paper/2014/hash/a14ac55a4f27472c5d894ec1c3c743d2-Abstract.html

[14] A. Graves, "Generating Sequences with Recurrent Neural Networks," ArXiv13080850 Cs, Jun. 2014, Accessed: Aug. 05, 2021. [Online]. Available: http://arxiv.org/abs/1308.0850

[15] "The Unreasonable Effectiveness of Recurrent Neural Networks." http://karpathy.github.io/2015/05/21/rnn-effectiveness/ (accessed Aug. 05, 2021).

[16] M. L. Samb, S. Ndiaye, Y. Slimani, and M. A. Esseghir, "A Novel RFE-SVM-based Feature Selection Approach for Classification," Int. J. Adv. Sci. Technol., vol. 43, p. 10, 2012.

[17] Z. Qi, Y. Tian, and Y. Shi, "Robust twin support vector machine for pattern classification," Pattern Recognit., vol. 46, no. 1, pp. 305–316, Jan. 2013, doi: 10.1016/j.patcog.2012.06.019.

[18] R. Moraes, J. F. Valiati, and W. P. Gavião Neto, "Document-level sentiment classification: An empirical comparison between SVM and ANN," Expert Syst. Appl., vol. 40, no. 2, pp. 621–633, Feb. 2013, doi: 10.1016/j.eswa.2012.07.059.

[19] L. Bahatti, O. Bouattane, E. Ech-Chhibat, and M. Zaggaf, "An Efficient Audio Classification Approach Based on Support Vector Machines," Int. J. Adv. Comput. Sci. Appl., vol. 7, pp. 205–211, Jun. 2016, doi: 10.14569/IJACSA.2016.070530.

[20] "New Algorithm for the Development of a Musical Words Descriptor for the Artificial Composition of Oriental Music," Journal. https://astesj.com/v05/i05/p54/ (accessed Aug. 05, 2021).

[21] M. ZHAR, "Classification algorithms to recognize the influence of the playing style of the Oud masters on the musicians of the Oud instrument," Int. J. Adv. Trends Comput. Sci. Eng., vol. 9, pp. 111–118, Sep. 2020, doi: 10.30534/ijatcse/2020/1791.52020.

[22] M. ZHAR, O. BOUATTANE, and L. BAHATTI, "New Algorithm for The Development of A Musical Words Descriptor For The Artificial Synthesis Of Oriental Music," in 2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), Apr. 2020, pp. 1–8. doi: 10.1109/IRASET48871.2020.9092157.

[23] M. Navarro, W. Hashimoto, S. Rodríguez, B. Lancho, and J. Corchado Rodríguez, "Sensoring a Generative System to Create User-Controlled Melodies," Sensors, vol. 18, p. 3201, Sep. 2018, doi: 10.3390/s18103201.

[24] L.-C. Yang, S.-Y. Chou, and Yi-hsuan Yang, "MidiNet: A Convolutional Generative Adversarial Network for Symbolic-domain Music Generation using 1D and 2D Conditions," Mar. 2017.

[25] Marchini M, Purwins H. Unsupervised Analysis and Generation of Audio Percussion Sequences. In: Ystad S, Aramaki M, Kronland-Martinet R, Jensen K, editors. Exploring Music Contents. Berlin, Heidelberg: Springer; 2011. p. 205–18.