Markov Chain Artificial Neural Networks Applied to Music Generation

Vinicius Lourenço [1] , Ercan Kuruoglu [2]

Tsinghua-Berkeley Shenzhen Institute

ABSTRACT

We bring a **Markov Chain Neural Network (MCNN) application in music generation**. The experiment used 371 chorales from J. Sebastian Bach as dataset. The music notes distribution are used to build a Markov Chain Transition Matrix, used later as base to build the training data. After training the MCNN, we input a first note and the network acts as a Markov Chain predicting the next state given a random variable plus the input note. As a result, we produced a random walk and a random music. [1][2][5]

INTRODUCTION

Markov Models represent a randomly changing system where the next state of the process is only dependent on the last state (Markov Property). Here we use a matrix representation for the transition probabilities from each state. [3]

$$G = (V, E, T)$$
$$E \subseteq VxV$$
$$T_{i,j} = p\ (i|j) for\ i, j \in E$$

Equation 1: *Graph and probability base representation*

The states a_{ij} from the matrix represent the note state i with the conditional probability P(X_1= j| X_0=i) of moving into state j given our actual state is i.[1][3]

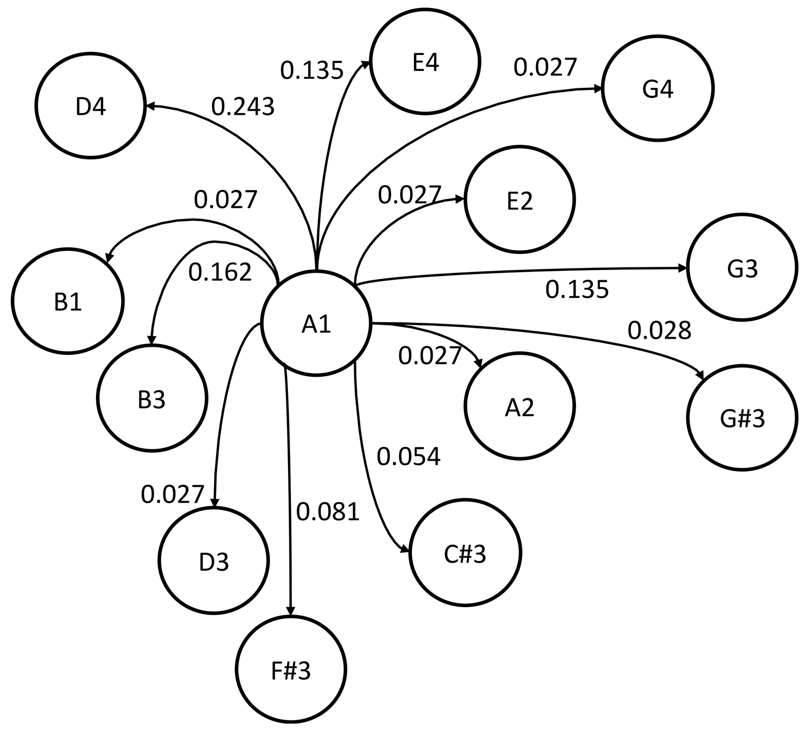

Figure 1: *Sample Graph from Transition Matrix for A1*

METHOD

The experiment followed the steps bellow:

1. Capture the 371 music data.
2. Extract the note's sequences from each music.
3. Pre-process to generate a unified data source.
4. From the data source, a counting step takes place and the Transition Matrix is built
5. The Transition Matrix is used to build a training dataset
6. Trained the MCNN
7. Run the MCNN in loop using the output of one step as the input of the next plus a random variable
8. Use the resulting array as base for the music generation

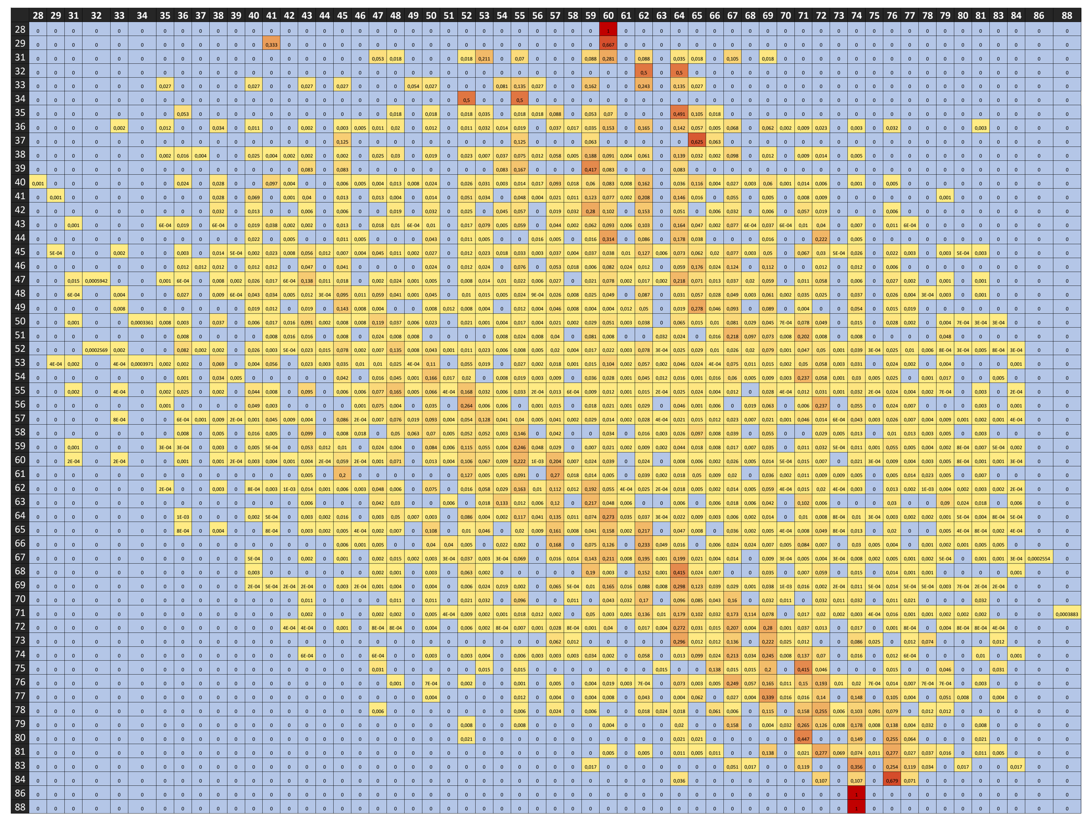

Figure 2: *Transition Matrix Heat Map*

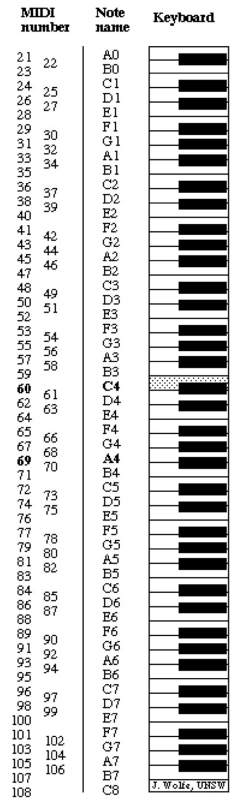

Figure 3:*MIDI translations*

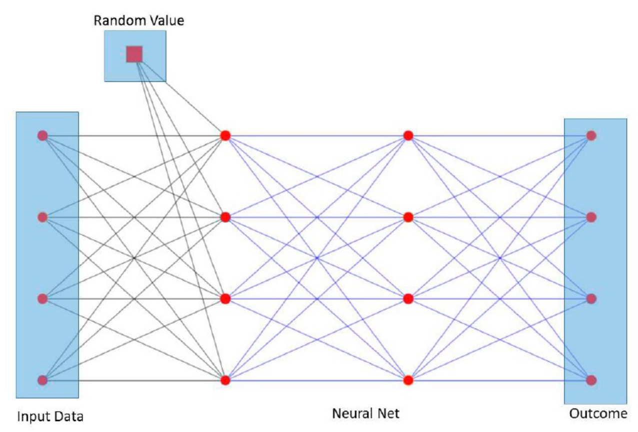

Figure 4: *Markov Chain Neural Network(MCNN) Structure[1]*

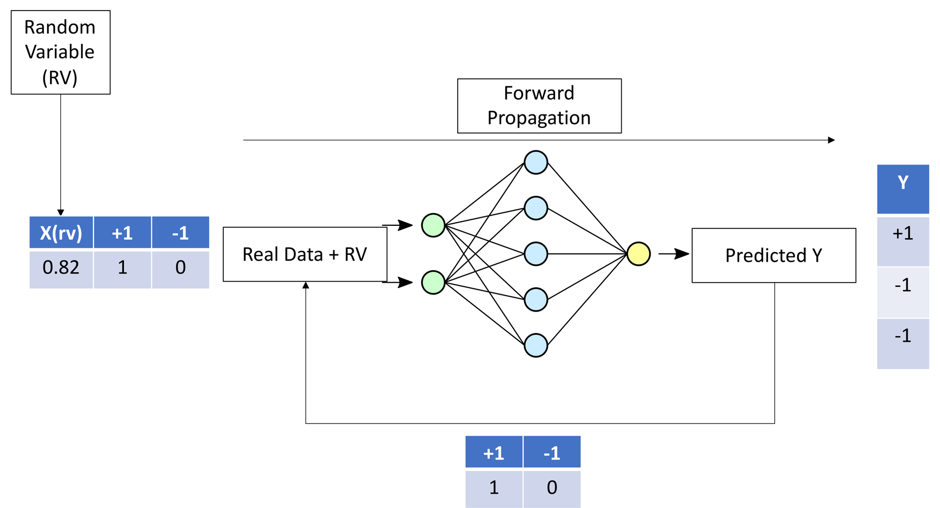

Figure 5: *Forward Propagating on MCNN with Loop for Random Walk*

RESULTS

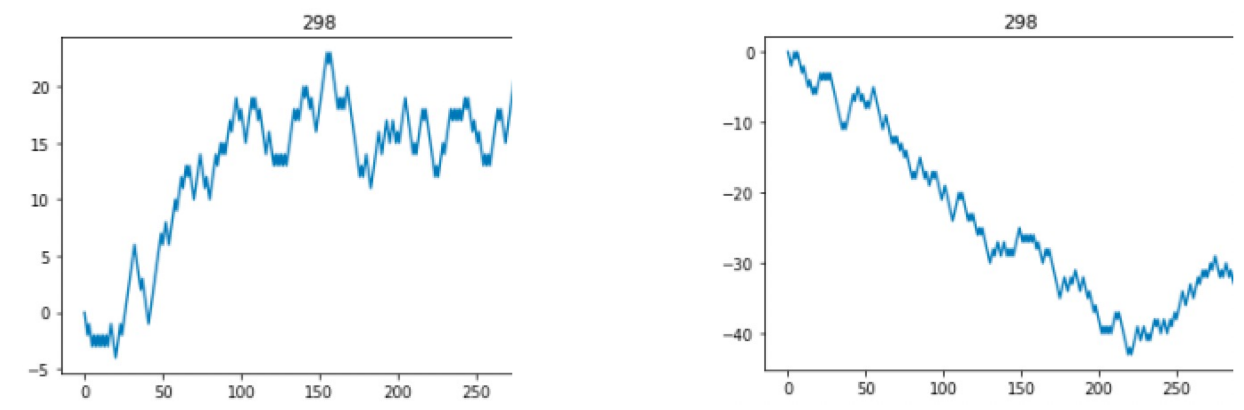

Figure 6: *Random Walk generated from the MCNN*

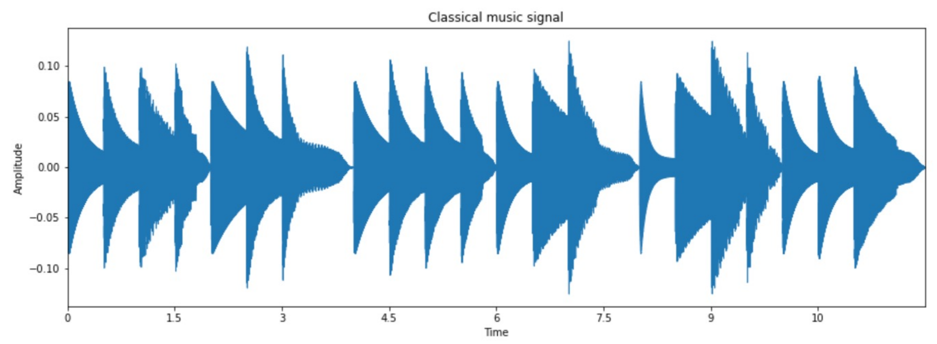

Figure 7: *Generated music waves in wav file*

CONCLUSION

The experiment was able to generate a random walk and a ANN that simulates a Markov Chain producing random musical notes transition following the transition matrix built on Johann Sebastian Bach 371 Chorales.

An important learning from this work was the fact that using one hot encoding was critical for the network learning process.

REFERENCES

[1] M. Awiszus, B. Rosenhahn, Markov chain neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 2180–2187.

[2] C. S. Sapp, Online database of scores in the humdrum file format., in: ISMIR, 2005, pp. 664–665.

[3] J. G. Kemeny, J. L. Snell, Markov chains, Vol. 6, Springer-Verlag, New York, 1976.

[4] I. A. Basheer, M. Hajmeer, Artificial neural networks: fundamentals, computing, design, and application, Journal of microbiological methods 43 (1) (2000) 3–31.

[5] D. Caldwell, Standard midi file format (1995).

[6] C. A. Mack, Fifty years of moore's law, IEEE Transactions on semiconductor manufacturing 24 (2) (2011) 202–207.

[7] M. S. Cuthbert, C. Ariza, music21: A toolkit for computer-aided musicology and symbolic music data (2010).