BACHDUET: A HUMAN-MACHINE DUET IMPROVISATION SYSTEM

Christodoulos Benetatos **Zhiyao Duan**
c.benetatos@rochester.edu zhiyao.duan@rochester.edu
Department of Electrical and Computer Engineering, University of Rochester

EXTENDED ABSTRACT

Summary. Back in the days of what today we refer to as *Baroque period*, improvisation was a key element of music performance and education. Great musicians such as J. S. Bach, were better known as improvisers than composers. Today, however, there is a lack of improvisation culture in classical music performance and education; classical musicians either do not know how to improvise, or cannot find other people to jam with. Motivated by this observation, we developed *BachDuet*, a system that enables real-time counterpoint improvisation between a human and a machine. This system uses a recurrent neural network to process the human musician's monophonic performance on a MIDI keyboard and generates the machine's monophonic performance in real time. We hope that it will serve as both an entertainment and practice tool for classical musicians to develop their improvisation skills.

Proposed System. We design this system to support two kinds of human-machine music improvisation: a) free counterpoint improvisation in a duet setting and b) conditional counterpoint improvisation in a duet setting over a given bass part. In both settings, the system predicts the next note of the machine's voice based on the past notes of both the machine's and the human's voices, as well as the pre-defined bass part in the second setting (Figure 1). This causal way of music generation allows us to implement this system with a real-time GUI, where the user interacts with the system by playing on a MIDI keyboard. The system operates with a preset steady tempo to ensure human-machine synchronization. A demo video can be found here [1].

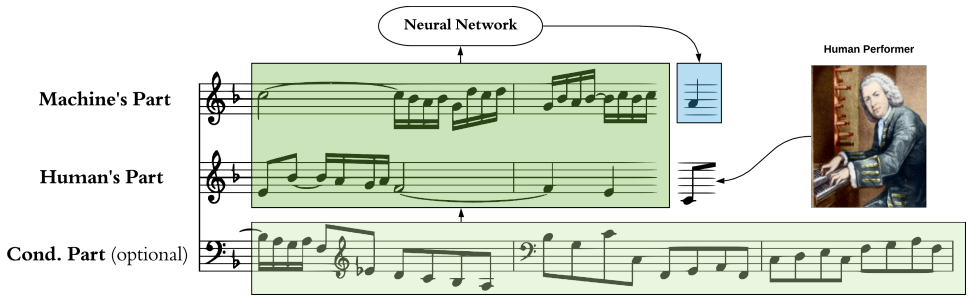

Figure 1. System overview. The system predicts the next note of the machine's voice based on the past notes of both the machine's and the human's voices, and a pre-defined bass part, when available.

The core algorithm of *BachDuet* consists of an LSTM network augmented with an stack structure serving as extra memory. We divide the music time into sixteenth note grid and assume that a note always starts and ends on grid positions (time steps). At every grid position, the network predicts the note of the machine's voice in the next time step as well as the musical key of the current time step. To encode each note, we use a combination of the MIDI number and the onset indicator. For example, a C4 quarter note spans four time steps and is encoded with four tokens: $[60\_1, 60\_0, 60\_0, 60\_0]$, where the first token denotes the onset

[1] https://www.youtube.com/watch?v=eZb0wA7k0NU.

and the following three tokens denote its continuation. To encode the metrical structure, we use the rhythm encoding scheme from [5].

We used the Bach 4-part Chorale dataset to train this network. For each chorale, we extracted all its 12 duets with one part being assigned to human and the other to machine. We further transposed each duet to all 12 keys to make the system key invariant. To address the problem of human performance errors in real-time operation, during training, we randomly injected three kinds of performance mistakes (insertion, deletion and time shifting of notes) in the human part of the duets. The amount of mistakes gradually increased as training progressed.

We designed a Graphical User Interface (GUI) (Figure 2) for this system using the PyQt library to display the performance in real time, in both the MIDI pianoroll and music notation formats. The GUI also provides several controls such as tempo preset, network memory reset, and music content initialization.

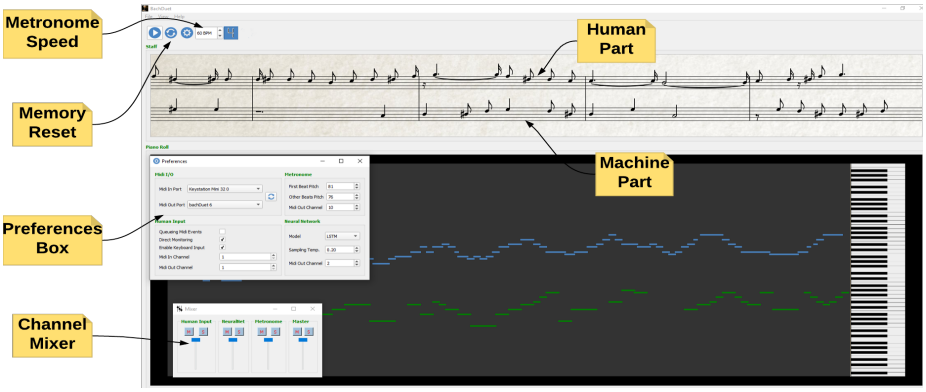

Figure 2. The system's GUI showing the generated duet in both the pianoroll and music notation formats.

Subjective Evaluation. Preliminary subjective tests show promising music improvisation outcome when the human musician has some musical background and does not deviate too much from the style of the training corpus. The next step is to conduct more in-depth subjective evaluation with a large group of musicians and audience. We plan to evaluate the usability, the quality of improvisation, and feasibility of the system as an pedagogical tool for improvisation.

Related Work. There exist a number of interactive systems that support collaborative music performance between humans and machines. However, most systems (e.g., *GenJam* [1], *Continuator* [4], and *AI Duet* [3]) only do so in a "call and response" configuration, disallowing human-machine simultaneous performance. *Voyager* [2] is an exception, but it is designed for contemporary music improvisation.

ACKNOWLEDGMENTS

This work is funded by the National Science Foundation grant No. 501700. We also thank Prof. Matthew Brown from the Eastman School of Music for the fruitful discussions.

REFERENCES

[1] John A Biles et al. Genjam: A genetic algorithm for generating jazz solos. In *Proc. International Computer Music Conference*, volume 94, pages 131–137, 1994.

[2] George E Lewis. Too many notes: Computers, complexity and culture in voyager. *Leonardo Music Journal*, pages 33–39, 2000.

[3] Yotam Mann. AI duet. https://experiments.withgoogle.com/ai/ai-duet, 2016.

[4] Francois Pachet. The continuator: Musical interaction with style. *Journal of New Music Research*, 32(3):333–341, 2003.

[5] Yujia Yan, Ethan Lustig, Joseph VanderStel, and Zhiyao Duan. Part-invariant model for music generation and harmonization. In *Proc. International Society for Music Information Retrieval*, pages 204–210, 2018.