Modeling Two-Part Counterpoint with Neural Turing Machines

Eric Nichols[1], Anna Aljanaki[2], Stefano Kalonaris[3], Gianluca Micchi, and Sebastian Stober[4] ⋆

[1] Microsoft
[2] Mooncascade
[3] RIKEN
[4] University of Potsdam
bachbimbobot@boom.cc

Abstract. Counterpoint includes musical processes of imitation, canon and antiphony, amongst other. We focus on the latter in the context of two-part polyphony. Antiphony can be assimilated it to the linguistic mechanism of question & answer, whereby a musical proposal is followed by a coherent response. While generative models for music creation 'in the style of' are now ubiquitous, little research has been carried out with respect to the notion of antiphony and how this can be modelled with memory networks. In particular, to our knowledge, no experiment has been made using Neural Turing Machines (NTM). In this paper we focus on J.S. Bach's Two-Part Inventions and we propose a NTM-based model for the generation of feasible musical responses to a given musical question. The model architecture is described and propose to evaluate the results with a validation set obtained from domain expert manual annotation, in addition to one obtained from the train-test split of the data.

Keywords: Counterpoint, Two-Part Inventions, Memory Networks, Neural Turing Machines

1 Introduction

While many attempts have been made to model counterpoint, e.g. first species counterpoint (Herremans, 2011), or to formalise its rules via computational methods (Song & Huang, 2015), little attention has been given to the process of antiphony in this context. Nevertheless, while antiphony is common in a number of other musical expressions (e.g., Afro-American, West-African music, etc.) it can also be found in counterpuntal works of Renaissance composers. We examine the Two-Part Inventions by J. S. Bach and we propose that these can be (partially) understood and/or modelled as *call & response* processes (see Fig. 1)

Bach's two-part inventions are an ideal database because they are a homogeneous body of work which offers some variance while keeping well-defined internal

⋆

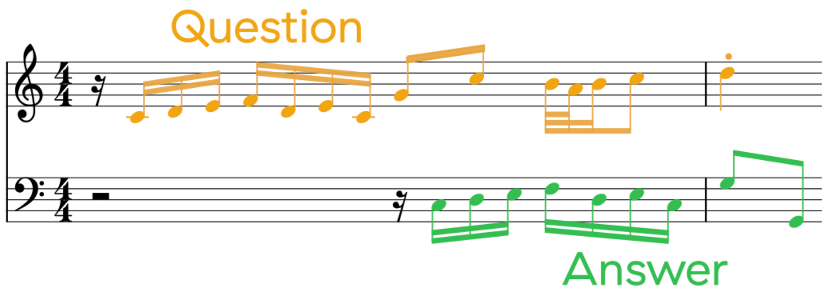

Fig. 1. The first bar of the first invention interpreted as call & response

laws. Furthermore, the two-part inventions are very well-known and regarded as a textbook example of contrapuntal music. They are relatively simple and could provide an excellent training set for a small model. We arbitrarily consider one of the parts as the question and the other as the answer. In order to represent tonal qualities and hierarchical music structures we transpose all the songs to the key of C major or A minor, accordingly. Moreover, we also arbitrarily segment each of these into four-bar chunks. These are gross simplifications, and we are aware that this representations misses out much contextual and musical value. Furthermore, we consider question and answer as asynchronous whereas, in fact, in the music pieces (the two-part inventions) these are instead synchronous and difficult to unbind. Nevertheless, we feel that this approximation is not entirely inappropriate since a composer has perfect information with respect to the composed material (since he/she is carefully writing it!). We use a Neural Turing Machine (NTM) (Graves, Wayne, & Danihelka, 2014), which is a type of memory network (Weston, Chopra, & Bordes, 2014; Ma & Príncipe, 2018) which function much the same way as an ordinary Turing Machine, where a controller decides what to read from the memory and where to move its head. The difference is that a controller is a differentiable function (a neural network). We use an LSTM with the same architecture as the sequence-to-sequence memoryless model as a controller for our NTM.

Since the chunking was made arbitrarily every four measures, it's likely that the learnt representations and the generated responses are of limited musical value. It would be advisable to use a domain-expert to manually segment the two-part variations into more semantically valid chunks. We started doing this for the first two inventions. These could be used as a validation set or they could be completed to include the whole corpus in order to train the model accordingly.

References

Graves, A., Wayne, G., & Danihelka, I. (2014). Neural turing machines. *CoRR*, *abs/1410.5401*.

Herremans, D. (2011). A variable neighbourhood search algorithm to generate first species counterpoint musical scores..

Ma, Y., & Príncipe, J. C. (2018). A taxonomy for neural memory networks. *CoRR*, *abs/1805.00327*.

Song, X.-Y., & Huang, D.-R. (2015, 08). A Study on Digital Analysis of Bach's Two-Part Inventions. , *2015*, 1-6.

Weston, J., Chopra, S., & Bordes, A. (2014). Memory networks. *CoRR*, *abs/1410.3916*.