City Research Online

City, University of London Institutional Repository

Citation: Lambert, A., Weyde, T. & Armstrong, N. (2015). Perceiving and predicting expressive rhythm with recurrent neural networks. In: Proceedings of the 12th International Conference in Sound and Music Computing. . Maynooth, Ireland: SMC15. ISBN 9780992746629

This is the published version of the paper.

This version of the publication may differ from the final published version.

Permanent repository link: https://openaccess.city.ac.uk/id/eprint/16489/

Link to published version:

Copyright: City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

Reuse: Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

City Research Online: http://openaccess.city.ac.uk/ publications@city.ac.uk

PERCEIVING AND PREDICTING EXPRESSIVE RHYTHM WITH RECURRENT NEURAL NETWORKS

Andrew J. Lambert
City University London
andrew.lambert.1@city.ac.uk

Tillman Weyde
City University London
t.e.weyde@city.ac.uk

Newton Armstrong
City University London
newton.armstrong.1@city.ac.uk

ABSTRACT

Automatically following rhythms by beat tracking is by no means a solved problem, especially when dealing with varying tempo and expressive timing.

This paper presents a connectionist machine learning approach to expressive rhythm prediction, based on cognitive and neurological models. We detail a multi-layered recurrent neural network combining two complementary network models as hidden layers within one system.

The first layer is a Gradient Frequency Neural Network (GFNN), a network of nonlinear oscillators which acts as an entraining and learning resonant filter to an audio signal. The GFNN resonances are used as inputs to a second layer, a Long Short-term Memory Recurrent Neural Network (LSTM). The LSTM learns the long-term temporal structures present in the GFNN's output, the metrical structure implicit within it. From these inferences, the LSTM predicts when the next rhythmic event is likely to occur.

We train the system on a dataset selected for its expressive timing qualities and evaluate the system on its ability to predict rhythmic events. We show that our GFNN-LSTM model performs as well as state-of-the art beat trackers and has the potential to be used in real-time interactive systems, following and generating expressive rhythmic structures.

1. INTRODUCTION

> "Composition is not a matter of filling or dividing time, but rather of generating time." [1]

The examination of the expressive qualities of music has been ongoing since the Ancient Greeks [2]. For instance, performers have been shown to express the higher metrical structures within a piece of music by tending to slow down at the end of certain phrases [3].

What Roads is alluding to in the above quote is that it is the perception of rhythmic events that provides a subjective experience of time to the listener. As the performer expressively varies the temporal dynamics, metrical dissonances and consonances are formed, affecting our perception of musical time and our expectation of rhythmical events. Our research concerns this interplay of metric *perception*, expectational *prediction* with respect to expressive variations on musical timing.

In order to achieve rhythmic prediction, we need to first overcome the current problem with perceiving expressive timing. Automatically processing an audio signal to determine pulse event onset times (beat tracking) is a mature field, but it is by no means a solved problem. Analysis of beat tracking failures has shown that a big problem for beat trackers is varying tempo and expressive timing [4, 5].

We take a cognitive approach, utilising a neurologically inspired model of rhythm perception known as a Gradient Frequency Neural Network (GFNN) [6]. In a GFNN a network of oscillators are distributed across a frequency spectrum. Internal connections between oscillators in the network can be learned via Hebbian learning. When stimulated by a signal, the GFNN resonates nonlinearly, producing larger amplitude responses at related frequencies along the spectrum. When the frequencies in a GFNN are distributed within a rhythmic range, resonances can occur at integer ratios to the pulse. These resonances can be interpreted as the perception of a hierarchical metrical structure.

GFNNs have shown promise even when dealing with more complex input, such as syncopated rhythms [7] and polyrhythms [8]. The oscillators' entrainment properties make them good candidates for solving the expressive timing problem and so the GFNN forms the basis of our *perception* layer.

In our system the GFNN is coupled with a Long Short-Term Memory Neural Network (LSTM) [9], which is a type of recurrent neural network able to learn long-term dependencies in a time-series. The LSTM takes the role of *prediction* in our system; it reads the GFNN's resonances to make predictions about the expected rhythmic events in the piece.

A future goal of our research is to use the GFNN-LSTM model for expressive rhythmic *production*. That is, the generation of new expressive timing structures based on its own output and/or other music agents' output. This system would be fast enough to operate in real-time.

In this paper, Section 2 details previous work in this area, Section 3 details a rhythm prediction experiment we have conducted with the GFNN-LSTM model and shares its results. Finally, Section 4 offers conclusions and points to future work.

Copyright: ©2015 Andrew J. Lambert et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 Unported License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

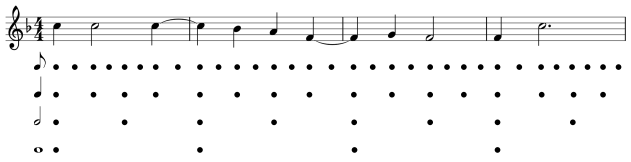

Figure 1. Metrical levels marked with Lerdahl and Jackendoff's 'dot notation'. The pulse level in this score would be at the crotchet (quarter note) level.

2. BACKGROUND

2.1 Pulse and Metre

A central idea in Lerdahl and Jackendoff's *Generative Theory of Tonal Music* (GTTM) is the notion of structures in music which are not present in the music itself, but perceived and constructed by the listener [10].

GTTM presents a detailed grammar of the inferred hierarchies a listener perceives when they listen to and understand a piece of music. Lerdahl and Jackendoff define four such hierarchies in tonal music, however in this paper we focus predominantly on *metrical structure*, considering other grammars only in relation to this.

A natural and often subconscious behaviour when we listen to music is that we tap our feet or nod our heads along to it. By doing so, we are reducing the music we hear into a series of periodic events. These events can sometimes be present in the music, but are often only implied by the rhythm of the music events and are constructed psychologically in the listener's mind. This process is known as *beat induction*; it is still an elusive psychological phenomenon that is under active research [11, 12], and has been claimed to be a fundamental musical trait [13].

When performing beat induction, one listener may tap along at twice the rate of another listener. In fact, there are several ways in which the music can be tapped along to, existing in a hierarchically layered relationship. The layers of beats are referred to in GTTM as 'metrical levels' and together they form a hierarchical metrical structure.

The beats at any given level can be perceived as 'strong' or 'weak'. If a beat on a particular level is perceived as strong, then it also appears in the next highest level, which creates the aforementioned hierarchy of beats. Theoretically, large measures, phrases, periods, and even higher order forms are possible in this hierarchy. Figure 1 illustrates a metrical analysis of a score.

Although tapping along at any metrical level is perfectly valid, humans often choose a common, comfortable period to tap to. Lerdahl and Jackendoff explain this selection process as a *preference rule* [14]. In general, this common period is referred to as the 'beat', but it is a problematic term since a beat can also refer to a singular rhythmic event or a metrically inferred event. Here we use a term that has recently grown in popularity in music theory: 'pulse' [15].

2.2 Nonlinear Resonance

GTTM is a musicological theory beginning with (but not limited to) the musical score as a source for analysing metre. What actually occurs in our brains as we listen to music and perform metre induction is another matter entirely.

Entrainment is the phenomena that occurs when two or more oscillations become synchronised in frequency and phase. It has been studied in a variety of disciplines such as mathematics and chemistry [16–18]. One can observe entrainment in action by placing several metronomes on a connected surface; over time the metronomes will synchronise [19].

Jones was among the first to propose an entrainment theory for the way we perceive, attend and memorise temporal events [20]. Jones posits that rhythmic patterns such as music potentially entrain a hierarchy of oscillations, forming an *attentional rhythm*. These attentional rhythms inform an expectation of when events are likely to occur, by extending the entrained period into the future.

Large takes this idea one step further with the notion of *nonlinear resonance* [6]. He states that musical structures occur at similar time scales to fundamental modes of brain dynamics, causing the nervous system to resonate to the rhythmic patterns. According to this theory, perceptions of pulse and metre perception arise as patterns of nervous system activity.

$$\frac{dz}{dt} = z(\alpha + i\omega + (\beta_1 + i\delta_1)|z|^2 + \frac{(\beta_2 + i\delta_2)\varepsilon|z|^4}{1-\varepsilon|z|^2}) + kP(\epsilon, x(t))A(\epsilon, \bar{z}) \quad (1)$$

Eq. (1) shows the differential equation that defines a Hopf normal form oscillator with its higher order terms fully expanded. This form is referred to as the canonical model, and was derived from a model of neural oscillation in excitatory and inhibitory neural populations [21]. z is a complex valued variable, $\bar{z}$ is its complex conjugate, and ω is the driving frequency in radians per second. α is a linear damping parameter, and β_1, β_2 are amplitude compressing parameters, which increase stability in the model. δ_1, δ_2 are frequency detuning parameters, and ε controls the amount on nonlinearity in the system. $x(t)$ is a time-varying external stimulus, which is also coupled nonlinearly and consists of passive part, $P(\varepsilon, x(t))$, and an active part, $A(\varepsilon, \bar{z})$, controlled by a coupling parameter k.

The α parameter acts as a bifurcation parameter: when $\alpha < 0$ the model behaves as a damped oscillator, and when $\alpha > 0$ the model oscillates spontaneously, obeying a limit-cycle. The gradual dampening of the amplitude allows the oscillator to maintain a long temporal memory of previous stimulation. This oscillator will resonate to an external stimulus that contains frequencies at integer ratio relationships to its natural frequency. Ratios such as 1:1, 2:1, 1:2, 3:1, 1:3, 3:2, and 2:3 are common and even higher order integer ratios are possible.

Optionally, canonical oscillators can be coupled together with a connectivity matrix as is shown in Eq. (2).

$$\frac{dz}{dt} = f(z, x(t)) + \sum_{i \neq j} c_{ij} \frac{z_j}{1-\sqrt{\epsilon} z_j} \cdot \frac{1}{1-\sqrt{\epsilon}\bar{z}_i} \quad (2)$$

Where $f(z, x(t))$ is the right hand side of Eq. (1) and c_{ji}

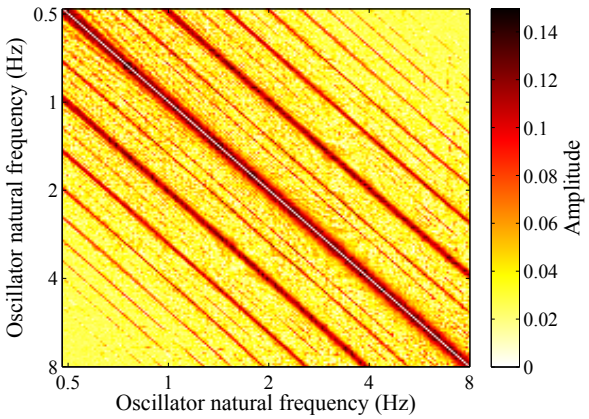

Figure 2. Amplitudes of connectivity matrix. Hebbian parameters are set to the following: $\lambda = .001, \mu_1 = -1, \mu_2 = -50, \epsilon_c = 16, \kappa = 1$, oscillator parameters are set to a limit cycle behaviour. Strong connections have formed at high-order integer ratios.

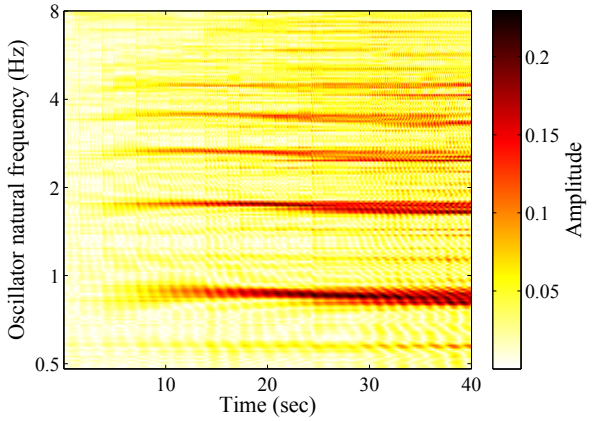

Figure 3. Amplitudes of oscillators over time.

is a complex number representing phase and magnitude of a connection between the i^{th} and j^{th} oscillator.

Hebbian learning can be incorporated on these connections, in a similar way to Hoppensteadt and Izhikevich [22]. This can allow resonance relationships between oscillators to form stronger bonds and is shown in Eq. (3).

$$\frac{dc_{ij}}{dt} = c_{ij}(\lambda + \mu_1|c_{ij}|^2 + \frac{\epsilon_c\mu_2|c_{ij}|^4}{1-\epsilon_c|c_{ij}|^2}) + \kappa\frac{z_i}{1-\sqrt{\epsilon_c}z_i}.\frac{z_j}{1-\sqrt{\epsilon_c}\bar{z_j}}.\frac{1}{1-\sqrt{\epsilon_c}z_j} \quad (3)$$

Here λ, μ_1, μ_2, ϵ_c and κ are all canonical Hebbian learning parameters.

Figure 2 shows a connectivity matrix after Hebbian learning has taken place. In this example the oscillators have learned connections to one another in the absence of any stimulus due to the oscillators operating in their limit cycle behaviour. Connections have been learned at high order integer ratios.

2.3 Gradient Frequency Neural Networks

Connecting several canonical oscillators together with a connection matrix forms a *Gradient Frequency Neural Network* (GFNN) [21]. When the frequencies in a GFNN are distributed within a rhythmic range and stimulated with music, resonances can occur at integer ratios to the pulse.

Velasco and Large connected two GFNN networks together in a pulse detection experiment for syncopated rhythms [7]. The two networks were modelling the sensory and motor cortices respectively. In the first network, the oscillators were set to a bifucation point between damped and spontaneous oscillation ($\alpha = 0, \beta_1 = -1, \beta_2 = -0.25, \delta_1 = \delta_2 = 0$ and $\varepsilon = 1$). The second network was tuned to exhibit double limit cycle bifurcation behaviour ($\alpha = 0.3, \beta_1 = 1, \beta_2 = -1, \delta_1 = \delta_2 = 0$ and $\varepsilon = 1$), allowing for greater memory and threshold properties. The first network was stimulated by a rhythmic stimulus, and the second was driven by the first. The two networks were also internally connected in integer ratio relationships such as 1:3 and 1:2. The results showed that the predictions of the model match human performance, implying that the brain may be adding frequency information to a signal to infer pulse and metre. Other rhythmic studies with GFNNs include rhythm categorisation [23] and polyrhythmic analysis [8].

Figure 3 shows the amplitude response of a GFNN to a rhythmic stimulus over time. Darker areas represent stronger resonances, indicating that that frequency is relevant to the music. A hierarchical structure can be seen to emerge from around 8 seconds, in relation to the pulse which is just below 2Hz in this example. At around 24 seconds, a tempo change occurs, which can be seen by the changing resonances in the figure. These resonances can be interpreted as a perception of the hierarchical metrical structure.

2.4 Beat Tracking

By far the most common form of automatically predicting rhythmic events is that of automatically processing an audio signal to determine pulse event onset times. In Music Information Retrieval (MIR) this is known as *beat tracking*.

Automated beat tracking has a long history of research [24]. The MIR Evaluation eXchange (MIREX) [1] project runs a beat tracking task each year, which evaluates several submitted systems against various datasets. This provides an easy way to discern what the current state-of-the-art is in terms of beat tracking, which lately has been Böck and Schedl's system [25].

State-of-the-art beat trackers do a relativity good job of finding the pulse in music with a strong beat and a steady tempo, yet we are still far from matching the human level of beat induction. Furthermore, despite a recent surge in new beat-tracking systems, there has been little improvement over Klapuri et al.'s system [26].

Grosche et al. [4] have performed an in-depth analysis

[1] http://www.music-ir.org/mirex/

of beat tracking failures on the Chopin Mazurka dataset [2] (MAZ). MAZ is a collection of audio recordings comprising on average 50 performances of each of Chopin's Mazurkas. Grosche et al. found that properties such as expressive timing and ornamental flourishes were contributing to the beat trackers' failures.

Holzapfel et al. [5] have selected 'difficult' excerpts for a new beat tracking dataset by a selective sampling approach. This is now publicly available as the SMC dataset [3] . The SMC excerpts are tagged with a selection of signal property descriptors, which allows for an overview of what contributes to an excerpt's difficulty. Most of the descriptors refer to temporal aspects of the music, such as slow or varying tempo, ornamentation, and syncopation, and over half of the dataset is tagged with the most prominent tag: expressive timing.

From this it is clear that being able to track expressive timing variations in performed music is one area in which there is much room for improvement. This has been attempted in many cases, most notably in the work of Dixon [27] and Dixon and Goebl [28]. However, these systems do not perform well on today's standard datasets, scoring poorly on the SMC dataset in 2014's MIREX results.

2.5 Neural Network Music Models

Todd [29] and Mozer [30] were among the first to utilise a connectionist machine learning approach to music generation. One of the major advantages of this approach is that it replaces rule-based systems, which can be strict, lack novelty, and not deal with unexpected inputs very well. Instead, the structure of existing musical examples are learned by the network and generalisations are made from these learned structures to compose new pieces. Both Todd and Mozer's systems are recurrent networks that are trained to predict melody. They take as input the current musical context as a pitch class and note onset marker and predict the same parameters at the next time step.

Whilst Todd and Mozer were mainly concerned with predicting pitch sequences over time, Gasser et al. [31] have taken a connectionist approach to perceive and produce rhythms that conform to particular metres. Their neural network model *SONOR* is a self-organising network of adaptive oscillators that uses Hebbian learning to prefer patterns similar to those it has been exposed to in a learning phase. A single input/output (IO) node operates in two modes, perception and production. In the perception mode, the IO node is excited by patterns of strong and weak beats, conforming to a specific metre. Hebbian learning is used to create connections and between the oscillators in the network. Once these connections have been learned, the network can be switched to production mode, reproducing patterns that match the metre of the stimuli.

Recurrent neural networks (RNNs) such as the those used in the above systems can be good at learning temporal patterns. However, as noted by Todd [29] and Mozer [30], they often lack global coherence due to the lack of long-term memory. This results in sequences with good local structures, but long-term dependencies are often lost. One way of tackling this problem is to introduce a series of time lags into the network input, so that past values of the input are presented to the network along with the present. Kalos [32] used a model of a similar type to generate music data in symbolic MIDI format. One advantage of this method is that it performs well on polyphonic music, but the time lag method still does not capture long-term structure very successfully.

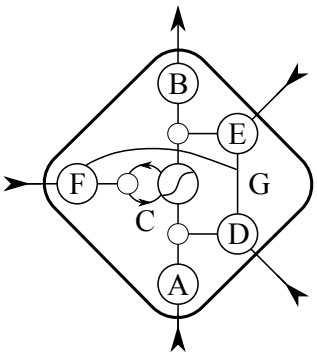

Figure 4. A single LSTM memory block showing (A) input, (B) output, (C) CEC, (D) input gate, (E) output gate, (F) forget gate and (G) peephole connections.

2.6 Long Short-Term Memory

Introduced by Hochreiter and Schmidhuber in 1997, Long Short-Term Memory Neural Networks (LSTMs) were designed to overcome the problem of modelling long term structures. Whilst RNNs can theoretically learn infinitely long patterns, in practice this is difficult due to the 'vanishing gradient problem' [9]. It can take as little as 5 time steps for this problem to occur in an RNN [33]. In an LSTM, a self-connected node known as the Constant Error Carousel (CEC) ensures constant error flow back through time, meaning that LSTMs can bridge time lags in excess of 1000 time steps [9].

A simplified diagram of an LSTM memory block can be seen in Figure 4. The input and output gates control how information flows into and out of the CEC, and the forget gate controls when the CEC is reset. The input, output and forget gates can be connected via 'peepholes'. For a full specification of the LSTM model we refer to [9] and [34].

As time-series predictors, LSTMs perform very well, as is shown by Böck and Schedl's beat tracker [25]. LSTMs have also had some success in generative systems. Eck and Schmidhuber [35] trained LSTMs which were able to improvise chord progressions in the blues and more recently Coca et al. [36] used LSTMs to generate melodies that fit within user specified parameters.

Lambert et al. have combined a GFNN with an LSTM (GFNN-LSTM) as two layers in an RNN chain and used it to predict melodies [37, 38]. Providing nonlinear resonance data from the GFNN helped to improve melody prediction with an LSTM. This is due to the LSTM being

[2] http://www.mazurka.org.uk/

[3] http://smc.inescporto.pt/research/data-2/

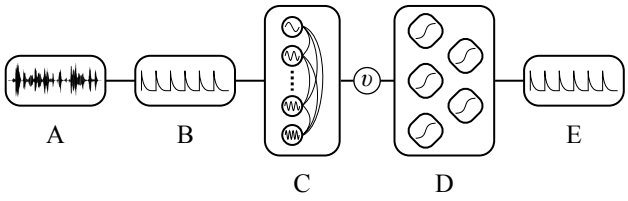

Figure 5. An overview of our GFNN-LSTM system showing (A) audio input, (B) mid-level representation, (C) GFNN, (D) LSTM, and (E) rhythm prediction output. The variable ν can be a mean field function or full connectivity.

able to make use of the relatively long temporal resonance in the GFNN output, and therefore model more coherent long-term structures. Here we take this work further by working with audio data and differing tempos.

3. EXPERIMENTS

We have performed an experiment where we have trained a GFNN-LSTM to predict expressive rhythmic events from audio data. The system takes audio data as input and outputs an event activation function. The system operates in a number of stages which are detailed below. A schematic of the system is provided in Figure 5.

The pieces in the MAZ dataset are expressively performed by various performers and vary in tempo and dynamics throughout the performance. However, the pieces are all within the same genre and are all performed on the piano, making drawing conclusions about the rhythmic aspects more valid. We have collected a subset of 50 excerpts, each 40 seconds long, by randomly choosing the full pieces and slicing 40 seconds worth of data.

When processing audio data for rhythmic events, it is common to first transform the audio signal into a more rhythmically meaningful form from which these events can be inferred. This representation could be extracted note onsets in binary form, or a continuous function that exhibits peaks at likely onset locations [39]. These functions are called *onset detection functions* and their outputs are known as *mid-level representations*.

Since we are dealing with expressively rich audio, we have chosen an onset detection function which is sensitive both to sharp and soft attack events such as those found in the MAZ piano performances. From Bello et al.'s tutorial on onset detection in music signals [40], we have selected the complex spectral difference onset detection function. This is a good general onset detector which works well with a variety of timbres. It is a continuous function that can be converted into binary onset data by using suitable threshold levels for peak picking. A sample rate of 86.025Hz was used, which was recently found to yield accurate detection results [41].

3.1 GFNN layer

The GFNN was implemented in MATLAB using the GrFNN Toolbox [42]. It consisted of 192 oscillators, logarithmically distributed with natural frequencies in a rhythmic range of 0.5Hz to 8Hz. The GFNN was stimulated by rhythmic time-series data in the form of the mid-level representation the audio data.

We have selected two parameter sets for the oscillators themselves, which affect the way the oscillators behave. The first is set to the bifurcation point between damped and spontaneous oscillation. We term this 'critical mode', as the oscillator resonates with input, but the amplitude decays over time in the absence of input: $\alpha = 0, \beta_1 = \beta_2 = -1, \delta_1 = \delta_2 = 0, \epsilon = 1$. By setting $\delta_1 = 1$, we define the second parameter set: 'detune mode'. These parameters allow the oscillator to change its natural frequency more freely, especially in response to strong stimuli. This essentially allows more entrainment to occur, so should allow for greater tracking of tempo changes. We obtained these values from the examples provided with the GrFNN Toolbox.

We have also selected three approaches to performing the Hebbian learning in the GFNN layer. The first approach simply has no connectivity between oscillators and therefore no learning activated at all (None). This is so that we can measure the effect (if any) that learning in the GFNN layer has on the overall predictions of the system.

The second approach is to activate online Hebbian learning with the following parameters: $\lambda = 0, \mu_1 = -1, \mu_2 = -50, \epsilon_c = 4$ and $\kappa = 1$ (Online). Under these parameters, the network should learn connections between related frequencies as they resonate to the stimulus.

The third approach is where generic initial connections have first been set in the network, learned by operating the oscillators in limit cycle mode (InitOnline). In this mode, the internal connections can be learned in the absence of any stimulus and results in a connectivity matrix shown in Figure 2. This provides a much more general state for the connection matrix to be in and potentially overcomes the limitations of the fixed frequency connections learned in online-only mode.

We found in some initial experimentation that during learning phase, the differential equations that drive the connectivity matrix can tend to spiral off to infinity. To ensure greater stability in the system, we have limited the connections in the connectivity matrix to have a magnitude less than $\frac{1}{\sqrt{\epsilon_c}}$ (0.5 in our experiments). We also and rescaled all stimuli to be in the range $0 <= x(t) <= 0.25$.

3.2 LSTM layer

The LSTM was implemented in Python using the PyBrain library [43]. For each variation of the GFNN, we trained two LSTM topologies. The first had 192 linear inputs, one for each oscillator in the GFNN, which took the real part of each oscillator's output. The second topology took only one linear input, which consisted of the mean field of the GFNN. The mean field reduces the dimensionality of the input whilst retaining frequency information within the signal.

All networks used the standard LSTM model with peephole connections enabled. The number of hidden LSTM blocks in the hidden layer was fixed at 10, with full recurrent connections. The number of blocks was chosen based on previous results which found it to provide reasonable

prediction accuracy, whilst minimising the computational complexity of the LSTM [38].

All networks had one single linear output, which serves as a rhythmic event predictor. The target data used was the output of the onset detection algorithm, where the samples were shifted so that the network was predicting what should happen next. The input and target data was normalised before training.

Training was done by backpropagation through time [44] using RProp$^-$ [45]. During training we used 5-fold cross-validation [46]. Training stopped when the total error had not improved for 20 epochs, or when this limit was reached, whichever came sooner.

3.3 Evaluation

The two main aims of this experiment were to firstly create a meaningful internal representation of metrical structure, and secondly to create good predictions in terms of the rhythmic structure. Therefore we are evaluating the system on its ability to predicted expressively timed rhythmic events, whilst varying the parameters of the GFNN and connectivity.

The results have been evaluated using the standard information retrieval metrics of precision, recall and F-measure. Events are predicted using a gradient threshold of the output data. The threshold looks for peaks in the signal by tracking gradient changes from positive to negative. When this gradient change occurs, an onset has taken place and is marked as such.

These events were subject to a tolerance window of $\pm$58.1ms. This means that an onset can occur within this time window and still be deemed a true positive. At the sample rate used in this experiment, this equates to 5 samples either side of an event. We also ensured that neither the target nor the output can have onsets faster than a rate of 16Hz, which is largely considered to be the limit of where rhythm starts to be perceived as pitch [6]. These are limitations to our evaluation method, but since we are mainly interested in predicted rhythmic structures and are not explicitly evaluating the production of expressive micro-timing, we believe they are acceptable concessions.

The first 5 seconds of output by the network are ignored, making the evaluation only on the final 35 seconds of predictions.

Table 1 and Table 2 display the results of the experiment, Figure 6 shows an example network output. These numerical metrics and visual figures provide some indication of how well the system is capturing the rhythmic structures. However, this information may be better understood by listening to the predicted rhythms. To this end, the reader is invited to visit this paper's accompanying website [4], where we have assembled a collection of audio examples and further output plots for each network's target and output data.

3.4 Discussion

We can see from the results that the best overall network incorporates detune oscillators, online learning with initial generic connections in the GFNN layer, and mean field connections.

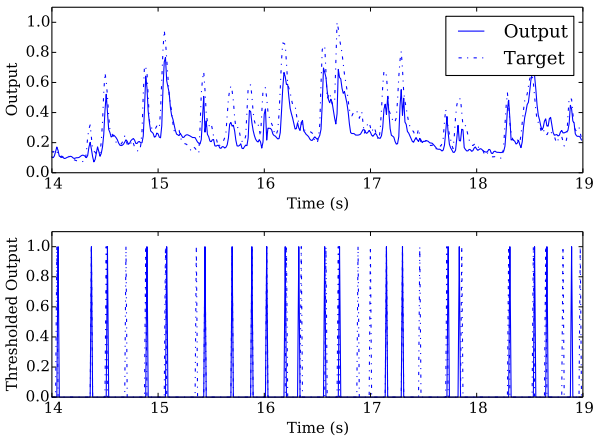

Figure 6. The output of the GFNN-LSTM. The top figure shown the predicted onset likelihood, the bottom figure displays the threshold events.

The mean field networks always outperformed the LSTMs with full connections to the GFNN. This is probably due to the mean field being able to capture the most resonant frequencies, whilst filtering out the noise of some less resonant frequencies. The resulting signal to the LSTM would therefore be more relevant for predicting rhythmic events. However, this may be due to the limited number of LSTM blocks in each network forming a bottleneck in the fully connected networks. Increasing number of hidden LSTM blocks may mitigate this limitation.

Another downside of the mean field networks is shown in the standard deviation figures. Whist performance improved in all cases using the mean field, the standard deviation also increased. This means there was a greater range of performances between the folds and could possibly indicate some networks being trained to local optima. During training we observed that the mean field networks took many more epochs for errors to converge.

The detuning oscillators outperformed the critical oscillators in all cases. This can be attributed to the greater amount of entrainment occurring in the network. Tempo changes can be tracked as an entrainment process between a local population of oscillators in the network. Where there is a local area of strong resonance the oscillators will take on very near frequencies to one another. As the stimulus frequency changes, this local area will be able to follow it, moving the local resonance area along the frequency gradient.

When compared to the results of our previous work on rhythm prediction with the GFNN-LSTM model [38], these results may at first seem a little underwhelming. The best network in our previous experiment achieved a rhythm prediction mean F-measure of 82.2%, compared with the 71.8% mean achieved here. However, this reflects the added difficulty of the task being undertaken here. Our previous work was on symbolic music at a fixed tempo and no expressive variation, whereas this study is undertaken on

[4] http://andyroid.co.uk/research/gfnn_lstm_rhythm_prediction/

Learning	LSTM	Precision	Recall	F-measure
None	Full	0.6114 (0.035)	0.6182 (0.034)	0.6059 (0.021)
None	Mean	0.6878 (0.100)	0.6883 (0.067)	0.6823 (0.081)
Online	Full	0.5637 (0.043)	0.6185 (0.076)	0.5798 (0.042)
Online	Mean	0.6862 (0.039)	0.6401 (0.050)	0.6548 (0.042)
InitOnline	Full	0.5982 (0.055)	0.6230 (0.041)	0.6000 (0.018)
InitOnline	Mean	**0.7032 (0.031)**	**0.6979 (0.041)**	**0.6958 (0.036)**

Table 1. Critical oscillation mode results. These results show the mean results calculated on the validation data. The number in brackets denotes the standard deviation.

Learning	LSTM	Precision	Recall	F-measure
None	Full	0.5972 (0.027)	0.6508 (0.036)	0.6161 (0.027)
None	Mean	0.7208 (0.058)	0.6891 (0.069)	0.6959 (0.057)
Online	Full	0.5831 (0.044)	0.6443 (0.067)	0.6020 (0.015)
Online	Mean	0.6943 (0.028)	0.6911 (0.045)	0.6866 (0.034)
InitOnline	Full	0.5666 (0.023)	0.6787 (0.033)	0.6114 (0.013)
InitOnline	Mean	**0.7239 (0.013)**	**0.7178 (0.061)**	**0.7142 (0.033)**

Table 2. Detune oscillation mode results. These results show the mean results calculated on the validation data. The number in brackets denotes the standard deviation.

audio data performed in expressive way at varying tempos. The overall best single system (Detune oscillators, InitOnline connections, and Mean input) was achieving an F-measure of 77.2%, which is extremely promising.

For comparison with other systems, the best beat tracker performance on MAZ submitted to MIREX in 2014 scored an F-measure of 71.5% (see [47]). Whilst this is not a direct comparison as we are predicting expressive rhythm, not pulse events, we believe it shows our system is at least comparable to state-of-the-art systems.

4. CONCLUSIONS

In this paper we have detailed a multi-layered recurrent neural network model for expressively timed rhythmic perception and prediction. The model consists of a perception layer, provided by a GFNN, and a prediction layer provided by an LSTM. We have evaluated the GFNN-LSTM on a dataset selected for its expressive timing qualities and found it to perform at a compatible standard to a previous experiment undertaken on symbolic data.

Our system's performance is comparable to state-of-the-art beat tracking systems. For the purposes of rhythm generation, we feel that the F-measure results reported here are already in a good range. Greater values may lead to too predictable and repetitive rhythms, lacking in the novelty expected in human expressive music. On the other hand, lower values may make the generated rhythms too random and irregular, so that they may even not be perceived as rhythmic at all. To make any firm conclusions on this, we would need to conduct formal listening tests based on the rhythms we have generated with our system. This is left for future work.

By using an oscillator network to track the metrical structure of expressively timed audio data, we have moved towards real-time processing of audio signals. We intend to extend this initial system for complete use as a MuMe system. Firstly, we will incorporate polyphonic rhythms into the system, instead of outputting a single rhythm output. Secondly, incorporating some melody model as in our previous work would be of use for complete autonomy of the system as a musical agent. Finally, we will close the feedback loop by connecting the system's output to its input. This would allow indefinite generation of new rhythmic structures which can be evaluated for their novelty. In doing so we will have created an expressive, generative, real-time agent.

Acknowledgments

Andrew J. Lambert is supported by a PhD studentship from City University London.

5. REFERENCES

[1] C. Roads, "Rhythmic Processes in Electronic Music," in *Joint 40th International Computer Music Conference and 11th Sound & Music Computing conference*, Athens, Greece, 2014.

[2] A. Gabrielsson and E. Lindström, "The role of structure in the musical expression of emotions," *Handbook of music and emotion: Theory, research, applications*, pp. 367–400, 2010.

[3] E. F. Clarke, "Generative principles in music performance." 1988.

[4] P. Grosche, M. Müller, and C. S. Sapp, "What Makes Beat Tracking Difficult? A Case Study on Chopin Mazurkas," in *in Proceedings of the 11th International Society for Music Information Retrieval Conference, 2010*, 2010, pp. 649–654.

[5] A. Holzapfel, M. E. P. Davies, J. Zapata, J. Oliveira, and F. Gouyon, "Selective Sampling for Beat Tracking Evaluation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 9, pp. 2539–2548, Nov. 2012.

[6] E. W. Large, "Neurodynamics of Music," in *Music Perception*, ser. Springer Handbook of Auditory Research, M. R. Jones, R. R. Fay, and A. N. Popper, Eds. Springer New York, Jan. 2010, no. 36, pp. 201–231. [Online]. Available: http://0-link.springer.com.wam.city.ac.uk/chapter/10.1007/978-1-4419-6114-3_7

[7] M. J. Velasco and E. W. Large, "Pulse Detection in Syncopated Rhythms using Neural Oscillators," in *12th International Society for Music Information Retrieval Conference*, Miami, FL, 2011, pp. 185–190.

[8] V. Angelis, S. Holland, P. J. Upton, and M. Clayton, "Testing a Computational Model of Rhythm Perception Using Polyrhythmic Stimuli," *Journal of New Music Research*, vol. 42, no. 1, pp. 47–60, 2013. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/09298215.2012.718791

[9] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[10] F. Lerdahl and R. Jackendoff, *A generative theory of tonal music*. Cambridge, Mass.: MIT press, 1983.

[11] G. Madison, "An Auditory Illusion of Infinite Tempo Change Based on Multiple Temporal Levels," *PLoS ONE*, vol. 4, no. 12, p. e8151, Dec. 2009. [Online]. Available: http://dx.doi.org/10.1371/journal.pone.0008151

[12] J. London, *Hearing in Time: Psychological Aspects of Musical Meter*. Oxford University Press, May 2012. [Online]. Available: http://0-www.oxfordscholarship.com.wam.city.ac.uk/view/10.1093/acprof:oso/9780199744374.001.0001/acprof-9780199744374

[13] H. Honing, "Without it no music: beat induction as a fundamental musical trait," *Annals of the New York Academy of Sciences*, vol. 1252, no. 1, pp. 85–91, 2012.

[14] F. Lerdahl and R. Jackendoff, "An Overview of Hierarchical Structure in Music," *Music Perception: An Interdisciplinary Journal*, vol. 1, no. 2, pp. 229–252, Dec. 1983. [Online]. Available: http://www.jstor.org/stable/40285257

[15] S. Grondin, *Psychology of Time*. Emerald Group Publishing, 2008.

[16] C. Huygens, *Horologium oscillatorium, sive de motu Pendulorum ad Horologia aptato demonstrationes geometricae*. Muguet, 1673.

[17] Y. Kuramoto, *Chemical oscillations, waves and turbulence*. Springer, Berlin, 1984.

[18] S. H. Strogatz, *Nonlinear dynamics and chaos: with applications to physics, biology and chemistry*. Perseus publishing, 2001.

[19] J. Pantaleone, "Synchronization of metronomes," *American Journal of Physics*, vol. 70, no. 10, pp. 992–1000, Oct. 2002. [Online]. Available: http://scitation.aip.org/content/aapt/journal/ajp/70/10/10.1119/1.1501118

[20] M. R. Jones, "Time, our lost dimension: Toward a new theory of perception, attention, and memory," *Psychological Review*, vol. 83, no. 5, pp. 323–355, Sep. 1976. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=pdh&AN=1977-07367-001&site=ehost-live

[21] E. W. Large, F. V. Almonte, and M. J. Velasco, "A canonical model for gradient frequency neural networks," *Physica D: Nonlinear Phenomena*, vol. 239, no. 12, pp. 905–911, Jun. 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167278910000187

[22] F. C. Hoppensteadt and E. M. Izhikevich, "Synaptic organizations and dynamical properties of weakly connected neural oscillators II. Learning phase information," *Biological Cybernetics*, vol. 75, no. 2, pp. 129–135, 1996.

[23] R. Bååth, E. Lagerstedt, and P. Gärdenfors, "An Oscillator Model of Categorical Rhythm Perception," in *Proceedings of the 35th Annual Conference of the Cognitive Science Society*, M. Knauff, M. Pauen, N. Sebanz, and I. Wachsmuth, Eds. Austin, TX: Cognitive Science Society, 2013, pp. 1803–1808.

[24] P. E. Allen and R. B. Dannenberg, "Tracking musical beats in real time," in *Proceedings of the 1990 International Computer Music Conference*, vol. 140143, 1990.

[25] S. Böck and M. Schedl, "Enhanced Beat Tracking with Context-Aware Neural Networks," in *Proceedings of the 14th International Conference on Digital Audio Effects (DAFx-11)*, Paris, France, 2011.

[26] A. Klapuri, A. Eronen, and J. Astola, "Analysis of the meter of acoustic musical signals," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 342–355, Jan. 2006.

[27] S. Dixon, "Automatic extraction of tempo and beat from expressive performances," *Journal of New Music Research*, vol. 30, no. 1, pp. 39–58, 2001.

[28] S. Dixon and W. Goebl, "Pinpointing the beat: Tapping to expressive performances," in *Proc. of International Conference on Music Perception and Cognition*, Sydney, Austrailia, 2002, pp. 617–620.

[29] P. M. Todd, "A Connectionist Approach to Algorithmic Composition," *Computer Music Journal*, vol. 13, no. 4, pp. 27–43, Dec. 1989. [Online]. Available: http://www.jstor.org/stable/3679551

[30] M. C. Mozer, "Neural network music composition by prediction: Exploring the benefits of psychoacoustic constraints and multi-scale processing," *Connection Science*, vol. 6, no. 2-3, pp. 247–280, 1994.

[31] M. Gasser, D. Eck, and R. Port, "Meter as Mechanism: A Neural Network Model that Learns Metrical Patterns," *Connection Science*, vol. 11, no. 2, pp. 187–216, 1999. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/095400999116331

[32] A. Kalos, "Modeling MIDI Music as Multivariate Time Series," in *IEEE Congress on Evolutionary Computation, 2006. CEC 2006*, 2006, pp. 2058–2064.

[33] F. Gers and J. Schmidhuber, "LSTM recurrent networks learn simple context-free and context-sensitive languages," *IEEE Transactions on Neural Networks*, vol. 12, no. 6, pp. 1333–1340, Nov. 2001.

[34] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to Forget: Continual Prediction with LSTM," *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, Oct. 2000. [Online]. Available: http://dx.doi.org/10.1162/089976600300015015

[35] D. Eck and J. Schmidhuber, "Finding temporal structure in music: blues improvisation with LSTM recurrent networks," in *Proceedings of the 2002 12th IEEE Workshop on Neural Networks for Signal Processing, 2002*, 2002, pp. 747–756.

[36] A. Coca, D. Correa, and L. Zhao, "Computer-aided music composition with LSTM neural network and chaotic inspiration," in *The 2013 International Joint Conference on Neural Networks (IJCNN)*, Aug. 2013, pp. 1–7.

[37] A. Lambert, T. Weyde, and N. Armstrong, "Beyond the Beat: Towards Metre, Rhythm and Melody Modelling with Hybrid Oscillator Networks," in *Joint 40th International Computer Music Conference and 11th Sound & Music Computing conference*, Athens, Greece, 2014.

[38] ——, "Studying the Effect of Metre Perception on Rhythm and Melody Modelling with LSTMs," in *Tenth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2014.

[39] E. D. Scheirer, "Tempo and beat analysis of acoustic musical signals," *The Journal of the Acoustical Society of America*, vol. 103, no. 1, pp. 588–601, Jan. 1998. [Online]. Available: http://scitation.aip.org/content/asa/journal/jasa/103/1/10.1121/1.421129

[40] J. P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler, "A tutorial on onset detection in music signals," *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 5, pp. 1035–1047, 2005.

[41] M. E. P. Davies and M. Plumbley, "Context-Dependent Beat Tracking of Musical Audio," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1009–1020, Mar. 2007.

[42] E. W. Large, J. C. Kim, K. L. Lerud, and D. Harrell, "GrFNN Toolbox 1.0: Matlab tools for simulating signal processing, plasticity and pattern formation in gradient frequency neural networks," 2014. [Online]. Available: https://github.com/GrFNN/Toolbox-1.0

[43] T. Schaul, J. Bayer, D. Wierstra, Y. Sun, M. Felder, F. Sehnke, T. Rückstieß, and J. Schmidhuber, "PyBrain," *Journal of Machine Learning Research*, vol. 11, pp. 743–746, 2010.

[44] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[45] C. Igel and M. Hüsken, "Improving the Rprop learning algorithm," in *Proceedings of the second international ICSC symposium on neural computation (NC 2000)*. Citeseer, 2000, pp. 115–121.

[46] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *IJCAI*, vol. 14, 1995, pp. 1137–1145.

[47] S. Böck, F. Krebs, and G. Widmer, "A multi-model approach to beat tracking considering heterogeneous music styles," in *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR 2014), Taipei, Taiwan*, 2014.