Recurrent Neural Network-based Music Language Models for Improving Automatic Music Transcription

Abstract

In this paper, we propose a system for automatic music transcription (AMT) which incorporates prior information from a music language model (MLM) based on recurrent neural networks. AMT is the process of converting an acoustic music signal into a symbolic notation, and is considered to be a fundamental problem in music signal processing. The MLMs are trained on sequences of symbolic polyphonic music. We train Recurrent Neural Network (RNN)-based models, as they are capable of capturing complex temporal structures present in symbolic music data. Similar to the function of language models in automatic speech recognition, we use the MLMs to generate a prior probability for the occurrence of a sequence. The acoustic AMT model is based on probabilistic latent component analysis, and prior information from the MLM is incorporated into the transcription framework using Dirichlet priors. We test our hybrid models on a dataset of multiple-instrument polyphonic music and report a significant improvement in terms of F-measure, when compared to using an acoustic-only model.

1. Introduction

Automatic Music Transcription (AMT) involves automatically generating a symbolic representation of an acoustic musical signal (Benetos et al., 2013a). AMT is considered to be a fundamental topic in the field of music information retrieval (MIR) and has numerous applications in related fields in music technology, such as interactive music applications and computational musicology. Typically, the output of an AMT system is a *piano-roll* representation, which is a two-dimensional matrix representation of a musical piece where the X-axis represents time quantized into regular intervals, and the Y-axis represents the keys of a piano in increasing pitch. A cell in this matrix is 1 if the key represented by its Y-coordinate is sounding at the time instant represented by its Y-coordinate.

The majority of recent transcription papers utilise and expand *spectrogram factorisation* techniques, such as non-negative matrix factorisation (NMF) (Li & Seung, 1999) and its probabilistic counterpart, probabilistic latent component analysis (PLCA) (Smaragdis et al., 2006). Spectrogram factorisation techniques decompose a two-dimensional spectrogram of the input audio signal into a product of spectral templates (that typically correspond to musical notes) and component activations (that indicate whether each note is active at a given time frame). Spectrogram factorisation-based AMT systems include the work by Bertin et al. (Bertin et al., 2010), who proposed a Bayesian framework for NMF, which considers each pitch as a model of Gaussian components in harmonic positions. Benetos and Dixon (Benetos & Dixon, 2012) proposed a convolutive model based on PLCA, which supports the transcription of multiple-instrument music and supports tuning changes and frequency modulations (modelled as shifts across log-frequency).

In terms of connectionist approaches for AMT, Nam et al. (Nam et al., 2011) proposed a method where features suitable for transcribing music are learned using a deep belief network consisting of stacked restricted Boltzmann machines (RBMs). The model performed classification using support vector machines and was applied to piano music. Böck and Schedl used recurrent neural networks (RNNs) with Long Short-Term Memory units for performing polyphonic piano transcription (Bock & Schedl, 2012), with the system being particularly good at recognising note onsets.

There is no doubt that a reliable acoustic model is important for generating accurate symbolic transcriptions of a given music signal. However, since music exhibits a fair amount of structural regularity much like language, it is natural for one to think of the possibility of improving transcription accuracy using a *music language model* (MLM) in a manner akin to the use of a language model to improve the performance of a speech recognizer (Rabiner & Juang, 1993). In (Boulanger-Lewandowski et al., 2012), the predictions of a polyphonic MLM were used to this end, which was further developed in (Boulanger-Lewandowski et al., 2013), where an input/output extension of the RNN-RBM

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

was proposed that learned to map input sequences to output sequences in the context of AMT. Another example of symbolic information which can improve the performance of acoustic models are *score informed* approaches, which have been applied in music research tasks such as source separation (Ewert & Müller, 2012), voice separation (Ewert & Müller, 2011) and tonic identification (Sentürk et al., 2013).

In the present work, we make use of the predictions made by a Recurrent Neural Network (RNN) and a Recurrent Neural Network-Neural Autoregressive Distribution Estimator (RNN-NADE) based polyphonic MLM proposed in (Boulanger-Lewandowski et al., 2012) to refine the transcriptions of a PLCA-based AMT system (Benetos & Dixon, 2012; Benetos et al., 2013b). Information from the MLM is incorporated into the PLCA-based acoustic model as a prior for pitch activations during the parameter estimation stage. Contrary to the system of (Boulanger-Lewandowski et al., 2012) where the MLM model improved the transcription output in a post-processing step, the proposed system jointly incorporates the MLM information in the transcription system during the parameter estimation stage. In addition, experiments are performed on real-world music recordings (compared to synthesized MIDI files) of multiple-instrument polyphonic music (compared to piano-only music). It was observed that combining the two models in this way boosts transcription accuracy by +3% on the Bach10 dataset of multiple-instrument polyphonic music, compared to using the acoustic AMT system only.

The outline of this paper is as follows. The PLCA-based transcription system is presented in Section 2. The RNN-based polyphonic music prediction system that is used as a music language model is described in Section 3. The combination of the two aforementioned systems is presented in Section 4. The employed dataset, evaluation metrics, and experimental results are shown in Section 5; finally, conclusions are drawn and future directions are indicated in Section 6.

2. Automatic Music Transcription System

For combining acoustic and music language information in an automatic transcription context, we employ the transcription model of (Benetos & Dixon, 2012), which supports the transcription of multiple-instrument polyphonic music and also supports pitch deviations and frequency modulations. The model of (Benetos & Dixon, 2012) is based on probabilistic latent component analysis (PLCA), which is a latent variable analysis method which has been used for decomposing spectrograms (Shashanka et al., 2008) and can be viewed as a probabilistic version of non-negative matrix factorization (Li & Seung, 1999). For computational efficiency purposes, we employ the fast implementation from (Benetos et al., 2013b), which utilized pre-extracted note templates that are also pre-shifted across log-frequency, in order to account for frequency modulations or tuning changes. In addition, as was shown in (Smaragdis & Mysore, 2009), PLCA-based models can utilise priors for estimating unknown model parameters, which will be useful in this paper for informing the acoustic transcription system with symbolic information.

The transcription model takes as input a normalised log-frequency spectrogram $V_{\omega,t}$ (ω is the log-frequency index and t is the time index) and approximates it as a bivariate probability distribution $P(\omega, t)$. $P(\omega, t)$ is decomposed into a series of log-frequency spectral templates per pitch, instrument, and log-frequency shifting (which indicates deviation with respect to the ideal tuning), as well as probability distributions for pitch, instrument, and tuning.

The model is formulated as:

$$P(\omega, t) = P(t) \sum_{p,f,s} P(\omega|s, p, f) P_t(f|p) P_t(s|p) P_t(p) \quad (1)$$

where p denotes pitch, s denotes the musical instrument source, and f denotes log-frequency shifting. $P(t)$ is the energy of the log-spectrogram, which is a known quantity. $P(\omega|s, p, f)$ denotes pre-extracted log-spectral templates per pitch p and instrument s, which are also pre-shifted across log-frequency. The pre-shifting operation is made in order to account for pitch deviations, without needing to formulate a convolutive model across log-frequency by f. $P_t(f|p)$ is the time-varying log-frequency shifting distribution per pitch, $P_t(s|p)$ is the time-varying source contribution per pitch, and finally, $P_t(p)$ is the pitch activation, which essentially is the resulting music transcription. As a time-frequency representation in the log-frequency domain we use the constant-Q transform (CQT) with a log-spectral resolution of 60 bins/octave (Schörkhuber & Klapuri, 2010).

The unknown model parameters ($P_t(f|p)$, $P_t(s|p)$, $P_t(p)$) can be iteratively estimated using the expectation-maximisation (EM) algorithm (Dempster et al., 1977). For the *Expectation* step, the following posterior is computed:

$$P_t(p, f, s|\omega) = \frac{P(\omega|s, p, f) P_t(f|p) P_t(s|p) P_t(p)}{\sum_{p,f,s} P(\omega|s, p, f) P_t(f|p) P_t(s|p) P_t(p)} \quad (2)$$

For the *Maximization* step (without using any priors) unknown model parameters are updated using the posterior computed from the Expectation step:

$$P_t(f|p) = \frac{\sum_{\omega,s} P_t(p, f, s|\omega) V_{\omega,t}}{\sum_{f,\omega,s} P_t(p, f, s|\omega) V_{\omega,t}} \quad (3)$$

$$P_t(s|p) = \frac{\sum_{\omega,f} P_t(p,f,s|\omega)V_{\omega,t}}{\sum_{s,\omega,f} P_t(p,f,s|\omega)V_{\omega,t}} \tag{4}$$

$$P_t(p) = \frac{\sum_{\omega,f,s} P_t(p,f,s|\omega)V_{\omega,t}}{\sum_{p,\omega,f,s} P_t(p,f,s|\omega)V_{\omega,t}} \tag{5}$$

We consider the sound state templates to be fixed, so no update rule for $P(\omega|s,p,f)$ is applied. Using fixed templates, 20-30 iterations using the update rules presented in the present section are sufficient for convergence. The output of the system is a pitch activation which is scaled by the energy of the log-spectrogram:

$$P(p,t) = P(t)P_t(p) \tag{6}$$

After performing 5-sample median filtering for note smoothing, thresholding is performed on $P(p,t)$ followed by note pruning with a lower duration threshold of 40ms (corresponding to the length of one time frame) in order to convert $P(p,t)$ into a binary piano-roll representation, which is the output of the transcription system, and is also used for evaluation purposes.

3. Polyphonic Music Prediction System

Taking inspiration from speech recognition, it has been known that a good statistical model of symbolic music can help the transcription process (Cemgil, 2004). However there are 2 main reasons for the use of MLMs in AMT not being more common.

1. Training models that capture the temporal structure and complexity in symbolic polyphonic music is not an easy task. In speech recognition, often simple language models like n-grams work extremely well. However, music has a more complex structure and simple statistical models like n-grams and HMMs fail to model these characteristics accurately (Boulanger-Lewandowski et al., 2012).

2. There is no consensus on how to incorporate this prior information within the transcription system. However, recently there have been some successful attempts at using this prior information to improve the accuracy on AMT tasks (Boulanger-Lewandowski et al., 2012; 2013).

In this section we discuss the details of the music prediction system and the models used. In the next section we discuss how we incorporate the predictions from these models in a PLCA-based music transcription system.

3.1. Recurrent Neural Network

A recurrent neural network (RNN) is a powerful model for time-series data which is known to account for long-term temporal dependencies, over multiple time-scales when trained effectively. Given a sequence of inputs $v_1, v_2, \ldots, v_T$ each in $\mathbb{R}^n$, the network computes a sequence of hidden states $\hat{h}_1, \hat{h}_2, \ldots, \hat{h}_T$ each in $\mathbb{R}^m$, and a sequence of predictions $\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_T$ each in $\mathbb{R}^k$ by iterating the equations

$$\begin{aligned} \hat{h}_t &= e(W_{\hat{h}x}v_t + W_{\hat{h}\hat{h}}\hat{h}_{t-1} + b_{\hat{h}}) && (7)\\ \hat{y}_t &= g(W_{y\hat{h}}) && (8) \end{aligned}$$

where $W_{y\hat{h}}$, $W_{\hat{h}x}$, $W_{\hat{h}\hat{h}}$ are the weight matrices and $b_{\hat{h}}$, b_y are the biases and e and g are activation functions which are typically non-linear and applied element-wise.

In theory, a recurrent neural network can be trained using the gradient-based Back-Propagation Through Time algorithm (Werbos, 1990) using the exactly computable error gradients in the network. However, 1^{st} order gradient methods fail to correctly train RNNs for many real-world problems. This difficulty has been associated with what is known as the *vanishing/exploding gradients* phenomenon (Bengio et al., 1994), where the errors exhibit exponential decay/growth as they are back-propagated through time. RNNs have been proposed over the years (Hochreiter & Schmidhuber, 1997; Jaeger, 2002; Martens & Sutskever, 2011).

However, recent in work in the field of neural networks and deep learning has led to several improvements in gradient based optimization methods that make training of RNNs on real-world data possible. Most notably, the Hessian Free (HF) optimization algorithm has been used to train RNNs successfully on several real world datasets, including symbolic polyphonic music data (Martens & Sutskever, 2011). Apart from second order methods like HF, several modifications to first-order gradient based methods exist that currently form the state of the art in training RNNs (Bengio et al., 2012).

3.2. Recurrent Neural Network-based models

One of the drawbacks of using RNNs to predict polyphonic symbolic music is that the outputs of the network, $\hat{y}_T$ at any time step T, are conditionally independent given the sequence of input vectors $v_1, v_2, \ldots, v_T$. This is a severe constraint when used for modelling polyphonic music, where notes often appear in very correlated patterns within a frame. In order to overcome this limitation, models derived from RNNs have been proposed which are better at modelling high-dimensional sequences (Sutskever et al., 2008; Boulanger-Lewandowski et al., 2012).

The first RNN-based model that tried to model high-dimensional sequences is the Recurrent Temporal Restricted Boltzmann Machine (RTRBM) (Sutskever et al., 2008). The RTRBM is a conditional RBM, where the RBM parameters at time t are a function of the visible and hidden hidden units until time $\tau < t$. The hidden states of the RTRBM are updated as if they belonged to a regular RNN, but they are binary during inference. A major constraint with the RTRBM is that the hidden states are responsible for conveying both temporal information and modelling the conditional distribution at each time-step. The RTRBM model was later extended to the more general RNN-RBM model (Boulanger-Lewandowski et al., 2012). In this model, there is a separate hidden state which conveys the temporal information. The hidden state of the RBM is then free to model only the distribution at that time step. It was found that the RNN-RBM performs better than the RTRBM at modelling symbolic polyphonic music (Boulanger-Lewandowski et al., 2012).

For our prediction system, we make use of a variant of the RNN-RBM, called the RNN-NADE. The only difference is that the conditional distributions at each step are modelled by a Neural Autoregressive Distribution Estimator (NADE) (Larochelle & Murray, 2011) as opposed to an RBM. As discussed in the next section, to combine the predictions with the transcription system, we need individual pitch activation probabilities at each time-step. Obtaining these probabilities from an RBM is intractable as it requires summing over all possible hidden states. However the NADE is a tractable distribution estimator and we can easily obtain these probabilities from the NADE. The NADE models the probability of occurrence of a vector p as:

$$P(p) = \prod_{i=1}^{D} P(p_i|\mathbf{p_{<i}}) \tag{9}$$

where $p \in \mathbb{R}^D$,p_i is the pitch activation and $\mathbf{p_{<i}}$ is the vector containing all the pitch activations p_j such that $j < i$.

In our system we utilise each of the conditional probabilities $p(p_i|\mathbf{p_{<i}})$ as probabilities of the pitch activations. Although the pitch activation probabilities are only conditioned on $\mathbf{p_{<i}}$, we hypothesize that this will be a better model than the RNN, where the pitch activation probabilities are completely independent. Another motivation for using the NADE is that the gradients can be computed exactly, and therefore we can employ HF optimization for training the RNN-NADE.

4. Combining Transcription and Prediction

In this section, we describe the process for combining the acoustic model with the music language model for deriving an improved transcription. Firstly, the input music signal is transcribed using the process described in Section 2. The resulting piano-roll representation of the transcription system is considered to be a sequence $p_1, p_2, \ldots, p_T$ that is placed as input to the MLM presented in Section 3. For the RNN-NADE, we compute the probability $P(p_i|\mathbf{p_{<i}})$ for all time frames, and use that as prior information for the combined model, with the prior information denoted as $P_{MLM}(p,t)$, where $P_{MLM}(p = i,t) = P(p_i|\mathbf{p_{<i}})$. For the RNN, the prediction output is directly denoted as $P_{MLM}(p,t)$, since pitch probabilities are independent.

As shown in (Smaragdis & Mysore, 2009), PLCA-based models use multinomial distributions; since the Dirichlet distribution is conjugate to the multinomial, a Dirichlet prior can be used to enforce structure on the pitch activation distribution $P_t(p)$. Following the procedure of (Smaragdis & Mysore, 2009), we define the Dirichlet hyperparameter for the pitch activation as:

$$\alpha_t(p) = \frac{P_t(p)P_{MLM}(p,t)}{\sum_p P_t(p)P_{MLM}(p,t)} \tag{10}$$

where $\alpha_t(p)$ essentially is a pitch activation probability which is filtered through a pitch indicator function computed from the symbolic prediction step (the denominator is simply for normalisation purposes).

The recording is then re-transcribed, using as additional information the prior computed from the transcription step. The modified update for the pitch activation which replaces (5) is given by:

$$P_t(p) = \frac{\sum_{\omega,f,s} P_t(p,f,s|\omega)V_{\omega,t} + \kappa\alpha_t(p)}{\sum_{p,\omega,f,s} P_t(p,f,s|\omega)V_{\omega,t} + \kappa\alpha_t(p)} \tag{11}$$

where κ is a weight parameter expressing how much the prior should be imposed; as in (Smaragdis & Mysore, 2009), the weight decreases from 1 to 0 throughout the iterations. To summarize, the transcription creates a symbolic prediction, which in turn improves the subsequent re-transcription of the music signal. An overview of the complete transcription-prediction system architecture can be seen in Fig. 4.

5. Evaluation

5.1. Dataset

For testing the transcription system, we employ the Bach10 dataset (Duan et al., 2010), which is a freely available multi-track collection of multiple-instrument polyphonic music, suitable for multi-pitch detection experiments. It consists of ten recordings of J.S. Bach chorales, performed by violin, clarinet, saxophone, and bassoon. Pitch ground truth for each instrument is also provided. Due to the tonal

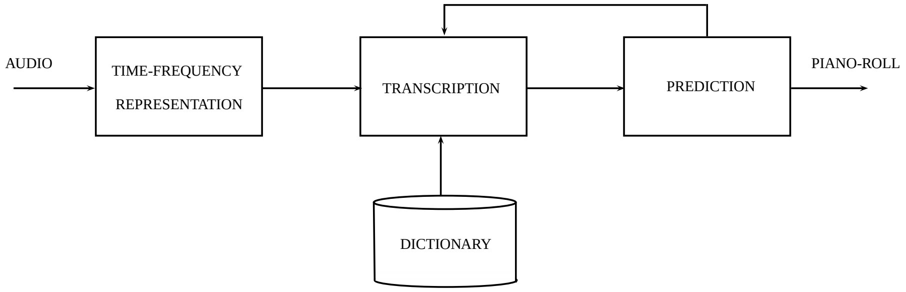

Figure 1. Proposed system diagram.

and homogeneous content of the dataset, it is suitable for testing the incorporation of music language models in a transcription system. For training the transcription system, pre-extracted and pre-shifted spectral templates are extracted for the instruments present in the dataset, using isolated note samples from the RWC database (Goto et al., 2003).

For training the MLMs we use the Nottingham dataset[1], a collection of 1200 music pieces in symbolic ABC format, which contain simple chord combinations and tunes. We trained the RNN and the RNN-NADE models using both Stochastic Gradient Descent (SGD) and HF to compare performance. The inputs to both the models are sequences of length 200 where each frame of the sequence is a binary vector of length 88 which covers the full range of the piano from A0 to C8. We train both the RNN and the RNN-NADE to predict the next vector given a sequence of input vectors. We train the models by minimizing the negative log-likelihood of the sequences using the cross-entropy objective $\sum_i t_i \log p_i + (1 - t_i)\log(1 - p_i)$ where i sums over all the dimensions of the binary vector and t_i is the pitch target.

5.2. Metrics

For evaluating the performance of the proposed system for multi-pitch detection, we employ the precision, recall, and F-measure metrics, which are commonly used in transcription evaluations (MIREX):

$$Pre = \frac{N_{tp}}{N_{sys}}, \quad Rec = \frac{N_{tp}}{N_{ref}}, \quad F = \frac{2 \cdot Rec \cdot Pre}{Rec + Pre} \qquad (12)$$

where N_{tp} is the number of correctly detected pitches, N_{sys} is the number of detected pitches, and N_{ref} is the number of ground-truth pitches. As in the public evaluations on multi-pitch detection carried out through the MIREX framework (MIREX), a detected note is considered correct is if its pitch is the same as the ground truth pitch and its onset is within a 50ms tolerance interval of the ground-truth onset.

Model	Pre
RNN (SGD)	67.89%
RNN (HF)	69.61%
RNN-NADE (SGD)	68.89%
RNN-NADE (HF)	**70.61%**

Table 1. Validation results for MLMs

5.3. Results

To validate the performance of the MLMs, we calculate the precision as defined in (12) of the prediction on unseen sequences of music from the Nottingham dataset. The Nottingham dataset contains 1200 folk melodies out of which we utilise 694 tracks for training, 173 tracks for validation and 170 for testing [2]. For both the RNN and RNN-NADE models we sample 10 vectors from the conditional distribution at each time-step and calculate the expected precision against the ground truth. The reported precision is found by finding the mean over the predictions of every frame. Table 1 shows the results of the validation experiments. These results are of the same order as the prediction accuracies reported in (Boulanger-Lewandowski et al., 2012). We found that for both the models, HF optimization gave better precision than SGD. Training with HF was also easier as there were less hyper parameters to be tuned when compared to SGD where learning rate needs to be updated to make sure training is effective.

Multi-pitch detection experiments are performed using the proposed system, with various configurations. A first configuration only considers the transcription system from Section 2. A second configuration takes the output of the transcription system and gives it as input to the prediction system of Section 3, where the final piano-roll is the output of the prediction step. A third configuration is the one presented in Section 4, where the recording is re-transcribed,

[1] ifdo.ca/∼seymour/nottingham/nottingham.html

[2] http://www-etud.iro.umontreal.ca/ boulanni/icml2012

Configuration	F	Pre	Rec
Configuration 1	62.02%	58.51%	66.12%
Configuration 2 - RNN-SGD	62.29%	59.08%	65.98%
Configuration 3 - RNN-SGD	63.85%	61.14%	66.90%
Configuration 2 - RNN-HF	62.44%	59.28%	66.07%
Configuration 3 - RNN-HF	62.87%	60.03%	66.11%
Configuration 2 - NADE-SGD	62.62%	59.70%	65.92%
Configuration 3 - NADE-SGD	64.08%	61.96%	66.44%
Configuration 2 - NADE-HF	62.20%	59.14%	65.68%
Configuration 3 - NADE-HF	**65.16**%	**62.80**%	**67.78**%

Table 2. Transcription results using various system configurations.

having the prediction as a prior information for estimating the pitch activations. For the prediction system, experiments were made using both the RNN-NADE and the RNN.

Results using the various system configurations are displayed in Table 2. It can be seen that the best performance is achieved by the 3rd configuration when using the NADE-HF model for prediction, which surpasses the acoustic-only transcription system by more than 3%. In general, it can be seen that by using the prediction system as a post-processing step (2nd configuration) always leads to an improvement over the acoustic-only model (1st configuration). A similar trend can be observed when integrating the prediction information as a prior in the transcription system (configuration 3) compared to just using th prediction system as post-processing (configuration 2); a improvement is always reported. Another observation ca be made when comparing the RNN-NADE with the RNN with the former providing a clear improvement.

As an example of the proposed system's performance, th spectrogram and raw output of the transcription-predictio system using the 3rd configuration is displayed for recording from the Bach10 dataset in Fig. 2, whereas th post-processed transcription output along with the groun truth for the same recording is shown in Fig. 3. It ca be seen that the system detects correctly most of the tim pitches and their respective durations, althougn there is als a large number of false alarms.

For comparison with the method of (Duan et al., 2010) (where the Bach10 dataset was first introduced), the proposed method using the frame-based accuracy metric which is defined in the same paper by Duan et al., reaches 74.3% for the NADE-HF using the 3rd configuration, whereas the method of (Duan et al., 2010) reaches 69.7%.

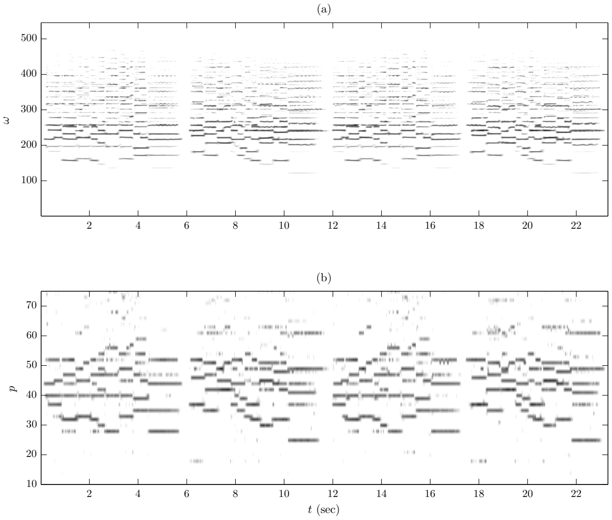

Figure 2. (a) The spectrogram $V_{\omega,t}$ for recording "Ach Lieben Christen" from the Bach10 dataset. (b) The pitch activation $P(p,t)$ using the transcription-predicton system using the 3rd configuration, with the NADE-HF.

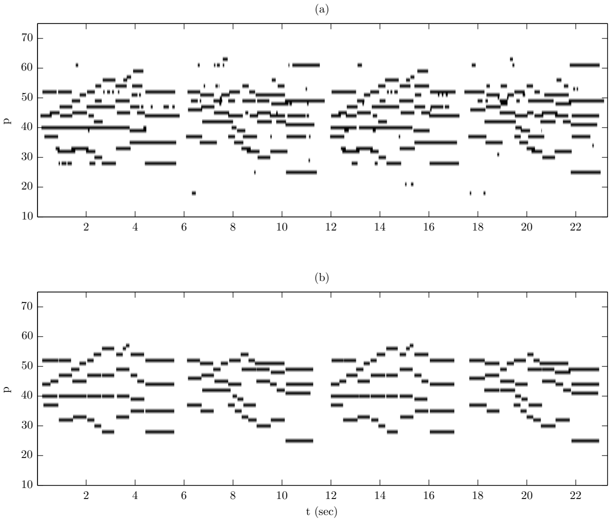

Figure 3. Transcription example for recording "Ach Lieben Christen" from the Bach10 dataset. (a) The post-processed output of the transcription-predicton system using the 3rd configuration, with the NADE-HF. (b) The pitch ground truth of the recording.

6. Conclusions

In this paper, we proposed a system for automatic music transcription which incorporated prior information from a polyphonic music prediction model based on recurrent neural networks. The acoustic transcription model was based on probabilistic latent component analysis, and information from the prediction system was incorporated using Dirichlet priors. Experimental results using the Bach10 dataset of multiple-instrument recordings showed that there is a significant improvement of 3% in terms of F-measure, with statistical significance supported by (Benetos, 2012), when combining a music language model with an acoustic transcription model instead of using an acoustic-only transcription system.

In the future, we would like to evaluate the proposed system using language models trained from different sources to see if this helps the MLMs generalize better. In addition, we aim to evaluate the proposed model on additional datasets of multiple-instrument polyphonic music. We will also investigate different system configurations, by bootstrapping the system for demonstrating that an improved transcription can lead to an improved prediction, and so on. We will also investigate the effect of using different RNN architectures like Long Short Term Memory (LSTM) and bi-directional RNNs and LSTMs. Finally, we would like to extend and improve the current models for high-dimensional sequences to better fit the requirements for music language modelling.

References

Benetos, E. *Automatic transcription of polyphonic music exploiting temporal evolution*. PhD thesis, Queen Mary University of London, December 2012.

Benetos, E., Dixon, S., Giannoulis, D., Kirchhoff, H., and Klapuri, A. Automatic music transcription: challenges and future directions. *Journal of Intelligent Information Systems*, 41(3):407–434, December 2013a. doi: 10.1007/s10844-013-0258-3.

Benetos, Emmanouil and Dixon, Simon. A shift-invariant latent variable model for automatic music transcription. *Computer Music Journal*, 36(4):81–94, 2012.

Benetos, Emmanouil, Cherla, Srikanth, and Weyde, Tillman. An effcient shiftinvariant model for polyphonic music transcription. In *6th International Workshop on Machine Learning and Music*, 2013b.

Bengio, Yoshua, Simard, Patrice, and Frasconi, Paolo. Learning long-term dependencies with gradient descent is difficult. *Neural Networks, IEEE Transactions on*, 5 (2):157–166, 1994.

Bengio, Yoshua, Boulanger-Lewandowski, Nicolas, and Pascanu, Razvan. Advances in optimizing recurrent networks. 2012.

Bertin, N., Badeau, R., and Vincent, E. Enforcing harmonicity and smoothness in bayesian non-negative matrix factorization applied to polyphonic music transcription. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3):538–549, March 2010.

Bock, S. and Schedl, M. Polyphonic piano note transcription with recurrent neural networks. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 121–124, March 2012.

Boulanger-Lewandowski, N., Bengio, Y., and Vincent, P. Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. In *29th International Conference on Machine Learning*, Edinburgh, Scotland, UK, 2012.

Boulanger-Lewandowski, N., Bengio, Y., and Vincent, P. High-dimensional sequence transduction. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3178–3182, May 2013.

Cemgil, A. T. *Bayesian Music Transcription*. PhD thesis, Radboud University of Nijmegen, 2004.

Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1): 1–38, 1977.

Duan, Z., Pardo, B., and Zhang, C. Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(8):2121–2133, November 2010.

Ewert, Sebastian and Müller, Meinard. Score-informed voice separation for piano recordings. In *ISMIR*, pp. 245–250, 2011.

Ewert, Sebastian and Müller, Meinard. Score-informed source separation for music signals. In *Multimodal Music Processing*, pp. 73–94, 2012.

Goto, M., Hashiguchi, H., Nishimura, T., and Oka, R. RWC music database: music genre database and musical instrument sound database. In *International Conference on Music Information Retrieval*, Baltimore, USA, October 2003.

Hochreiter, Sepp and Schmidhuber, Jürgen. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Jaeger, Herbert. Adaptive nonlinear system identification with echo state networks. In *Advances in neural information processing systems*, pp. 593–600, 2002.

Larochelle, Hugo and Murray, Iain. The neural autoregressive distribution estimator. *Journal of Machine Learning Research*, 15:29–37, 2011.

Li, D. D. and Seung, H. S. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, October 1999.

Martens, James and Sutskever, Ilya. Learning recurrent neural networks with hessian-free optimization. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 1033–1040, 2011.

MIREX. Music Information Retrieval Evaluation eXchange (MIREX). http://music-ir.org/mirexwiki/.

Nam, J., , Ngiam, J., Lee, H., and Slaney, M. A classification-based polyphonic piano transcription approach using learned feature representations. In *12th International Society for Music Information Retrieval Conference*, pp. 175–180, Miami, Florida, USA, October 2011.

Rabiner, Lawrence and Juang, Biing-Hwang. Fundamentals of speech recognition. 1993.

Schörkhuber, C. and Klapuri, A. Constant-Q transform toolbox for music processing. In *7th Sound and Music Computing Conf.*, Barcelona, Spain, July 2010.

Sentürk, Sertan, Gulati, Sankalp, and Serra, Xavier. Score informed tonic identification for makam music of turkey. In *14th Int. Soc. for Music Info. Retrieval Conf. Curitiba, Brazil.(to appear)*, volume 195, pp. 206, 2013.

Shashanka, M., Raj, B., and Smaragdis, P. Probabilistic latent variable models as nonnegative factorizations. *Computational Intelligence and Neuroscience*, 2008. Article ID 947438.

Smaragdis, P. and Mysore, G. Separation by “humming”: user-guided sound extraction from monophonic mixtures. pp. 69–72, October 2009.

Smaragdis, P., Raj, B., and Shashanka, Ma. A probabilistic latent variable model for acoustic modeling. In *Neural Information Processing Systems Workshop*, Whistler, Canada, December 2006.

Sutskever, Ilya, Hinton, Geoffrey E, and Taylor, Graham W. The recurrent temporal restricted boltzmann machine. In *Advances in Neural Information Processing Systems*, pp. 1601–1608, 2008.

Werbos, Paul J. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10): 1550–1560, 1990.