An End-to-End Neural Network for Polyphonic Piano Music Transcription

Samuel GUILLUY

Signal Analysis, Indexing and Transformation

March, 17 2020

Introduction

We will study a model composed of :

- **An acoustic model** which is a neural network used for estimating the probabilities of pitches in a frame of audio.
- **A language model** which is a recurrent neural network that models the correlations between pitch combinations over time.
- Finally, the acoustic and language model predictions are combined using **a probabilistic graphical model**. Inference over the output variables is performed using the beam search algorithm.

Plan of the presentation

Probabilistic definition of the model

Definition of the optimization problem

The objective of our model is to determine the mode of the conditional output distribution.

$$y^* = argmax_y P(y|x) \tag{1}$$

- y output of our model : the transcription as a high-dimensional binary vector
- x input of our model : sequence of array sound
- y^* optimal transcription

Probabilistic view of the model

Bayes formula and assumptions

First, we will factorize the join distribution of x and y :

$$P(y_0^t|x_0^t)\alpha P(y_0^{t-1}|x_0^{t-1})P(y_t|y_0^{t-1})P(y_t|x_t) \quad (2)$$

This formula can be written under the assumptions of :

$$\begin{gathered} P(y_t|y_0^{t-1}, x_0^{t-1}) = P(y_t|y_0^{t-1}) \\ P(x_t|y_0^t, x_0^{t-1}) = P(x_t|y_t) \end{gathered} \quad (3)$$

Probabilistic view of the model

Separation of the acoustic and language model

We can decompose this last formula into three parts :

- The current results from the probabilistic graphical model $P(y_0^{t-1}|x_0^{t-1})$
- The acoustic model estimates the probabilities of pitches in a frame of audio. The input is x and the output is $P(y_t|x_t)$
- The language model provides a prior probability of the current word given the previous words in a sentence. The input is y_0 and the output is $\forall t \ P(y_t|y_0^{t-1})$.

Acoustic Model

Convnets model for acoustic modelling - $P(y_t|x_t)$

Advantages of Convolutional Neural Networks :

- **Preserve the spatial structure** of the inputs
- The use of convolution products allows to produce a feature map which acts like filters
- **Use of the context** : better prediction accuracies can be achieved by incorporating information over several frames of inputs

Acoustic Model

Others model for acoustic modelling - $P(y_t|x_t)$

1. **Use of a pre processing :**
Mel-Frequency Cepstral Coefficients is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency.
2. **Other type of networks**
   - **Recurrent Neural Network** : recursive connections between the hidden layer activations at sometime t and the hidden layer activations at $t-1$.
   - **Attention network** as Neural machine translation by jointly learning to align and predict

Music Language Model (MLM)

Neural Autogressive Distribution Estimator (NADE) - $P(y_t|y_0^{t-1})$

A NADE estimates the joint distribution over high dimensional binary variables. It is similar to a fully sigmoid network.

$$\begin{aligned} h_i &= \sigma(W_{:,<i} y_0^{i-1} + b_h) \\ P(y_i|y_0^{i-1}) &= \sigma(V_i h_i + b_v^i) \end{aligned} \tag{4}$$

Music Language Model (MLM)

RNN - Neural Autogressive Distribution Estimator - $P(y_t|y_0^{t-1})$

Combine the NADE model with a RNN to learn high dimensional temporal distributions for the MLM.

$$\begin{aligned} b_v^t &= b_v + W_1 h_t \\ b_h^t &= b_h + W_2 h_t \end{aligned} \tag{5}$$

W_1 and W_2 are weight matrices from the RNN hidden state to the visible and hidden biases.

Combination of the two models

Result: beam.pop()
for *t=1 to T* **do**
    for *l,m_l in beam* **do**
        for *k=1 to K* **do**
            $y^{'} = m_a.next\_most\_probable()$
            l' $= \log P_l(y'|s)P_a(y'|x_t)$
            $m_l' \longleftarrow m_l \ with \ y_t := y'$
            new_beam.insert(l+l',m_l)
        end
    end
    beam $\longleftarrow$ *new_beam*
end

Algorithm 1: Beam Search Algorithm

Combination of the two models

Adaptation to the chord case

How to extract the most probable y' from the acoustic model ?

- set M the number of element from the chord
- set P the number of possible element from the chord
- take randomly M element among the P most possible value

My Implementation of the model

The dataset

MusicNet is a collection of classical music recordings, together with over 1 million annotated labels indicating the precise time of each note in every recording, the instrument that plays each note, and the note's position in the metrical structure of the composition.

The input of our model is thus an array of the sound.

The output of the model is a binary matrix representation of the chords played during the sound.

My Implementation of the model

The architecture of my model

Acoustic Model : CNN + cos/sin filters ; CNN + constant Q transform filters
Language Model : NADE
Probabilistic Graphical Model : beam search algorithms adapt for chords

Limitation of the model

About the assumption of the equation 3

The assumption about the independence of the pitches is not verified as the pitches are highly correlated (harmonies, chords) in polyphonic music.
The assumption about the predictions at time t are only a function of the input at t and is independent of all other inputs and outputs.

Conclusion

Improvement of our Implementation

- For the Acoustic model : The frames at the beginning and end of the audio are zero padded so that a context window can be applied to each frame.
- For the Language Model : add RNN to the NADE model.
- For the Probabilistic Graphical Model : use the hash table beam object. In order to prune better similar solution.