Ideas

Aciditeam

September 21, 2015

While being self-sufficient, those notes focus on the recent results in optimization and don't pretend to be an exhaustive review of optimization methods.

Chapter 1

Music generation

Chapter 2

Orchestral inference

2.1 Projective symbolic orchestration

2.1.1 Factored Gated Conditional RBM

The two next steps should be easy to implement and imply important changes :

- take dynamics into consideration
- event level

Other leads should be quickly investigated, although not spending to much time on it

- Greatly increase the number of factor units. It should greatly improve the "expressiveness" of the network.

2.2 Signal/symbolic networks

The idea is to build a system where orchestration is controlled in real-time by a set of faders which in they turn control some spectral features. It would be a purely generative system that would not rely on a piano score, but instead would be more an ambient generator (I imagine the result as "sound layers").

2.2.1 Learning step

The learning step rely on an orchestral database composed of the scores (xml preferably, midi otherwise) and a recording. Each track is divided in (short) time frames (both score and recording). The two Siamese networks respectively receive in input a symbolic frame and its corresponding signal frame (or its FFT, or any other perceptually relevant transformation). The training is then divided in two steps

1. unsupervised step, in order to learn relevant features in the last layer
2. supervised step in order to force the output of the two networks to be close in a metric space.

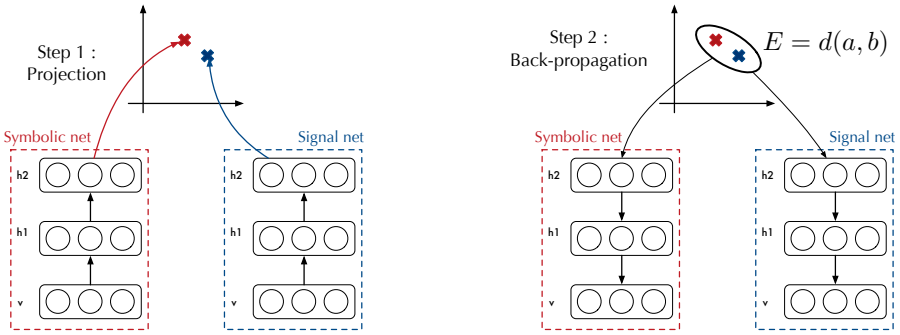

Figure 2.1: Supervised learning step in the Siamese network.

The structure of the two networks have to be determined (recurrent nets or simple deep architectures ? Stacked memory ?), but the first step would rely on state of the art methods in automatic features extraction. A first remark for the second step is that it requires the two networks to have the same number of output units $|O|$. Those two sets of output units define two points in $[0, 1]\,|O|$ (or $\mathbb{R}_+^{|O|}$ if we use a different type of unit for the output ?). The distance between those two points then define an error function that we can differentiate in order to fine-tuned the two networks with back-propagation.

2.2.2 Generation step

A fader is associated to each dimensions of our "projection" space (space defined by the output units). The position of each fader then define the coordinate of the synthesis point. We associate to each output unit the value of each coordinate of the synthesis point and back-propagate those values in the "symbolic" network to obtain the orchestration and in the "spectral" network to observe the ideal signal/spectrogram (probably not totally the same that the one corresponding to the symbolic score played).

2.2.3 Unsolved questions

- Input for the "spectral/signal" network
- Which architecture ? (recurrent nets or simple deep architectures ? Stacked memory, Deep LSTM ?)
- Which type of input **and** output units ?
- How to deal with continuity of orchestration ? A lead could be to define Siamese network for past frames and force recent past to be close to the actual frame in the projection space.

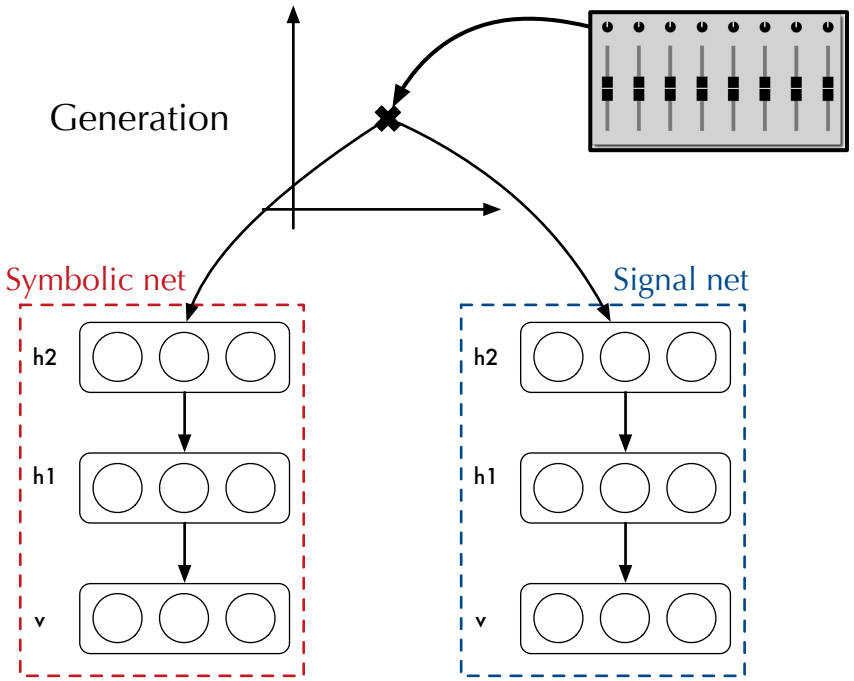

Figure 2.2: Generation step in the Siamese network.