Music Generation

Shaotian Chen
Gehang Li

Music Generation

- What is Music Generation?
- How can we achieve it?
- Comparison of different methods

What is Music Generation?

What is Music Generation?

- Of course, compose music
- With a sequence of rules
- Achieve in making music that sounds like produced by human
- http://computoser.com/

How can we achieve it?

- By mathematical models
- By neural network

Mathematical Model

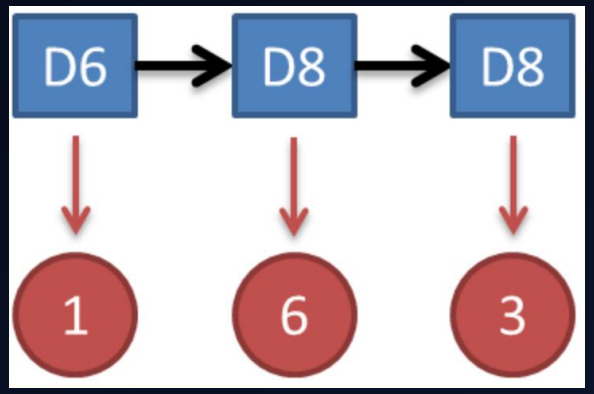
Hidden Markov Chain

- Likelihood problem
- Learning problem
- Decoding problem

Mathematical Model

Hidden Markov Chain

- The visible notes can be considered as the emission of the hidden rules
- Use samples to learn the harmony
- Generate music from the trained HMM model
- Use cross entropy to evaluate result



Mathematical Model

Hidden Markov Chain

- Lack of global structure and melodic progression
- Result may not be very good

Mathematical Model

Hidden Markov Chain

- Use different structure of HMM, like higher order HMM.

Cross-entropy (major)	Cross-entropy (minor)
2.76	2.58

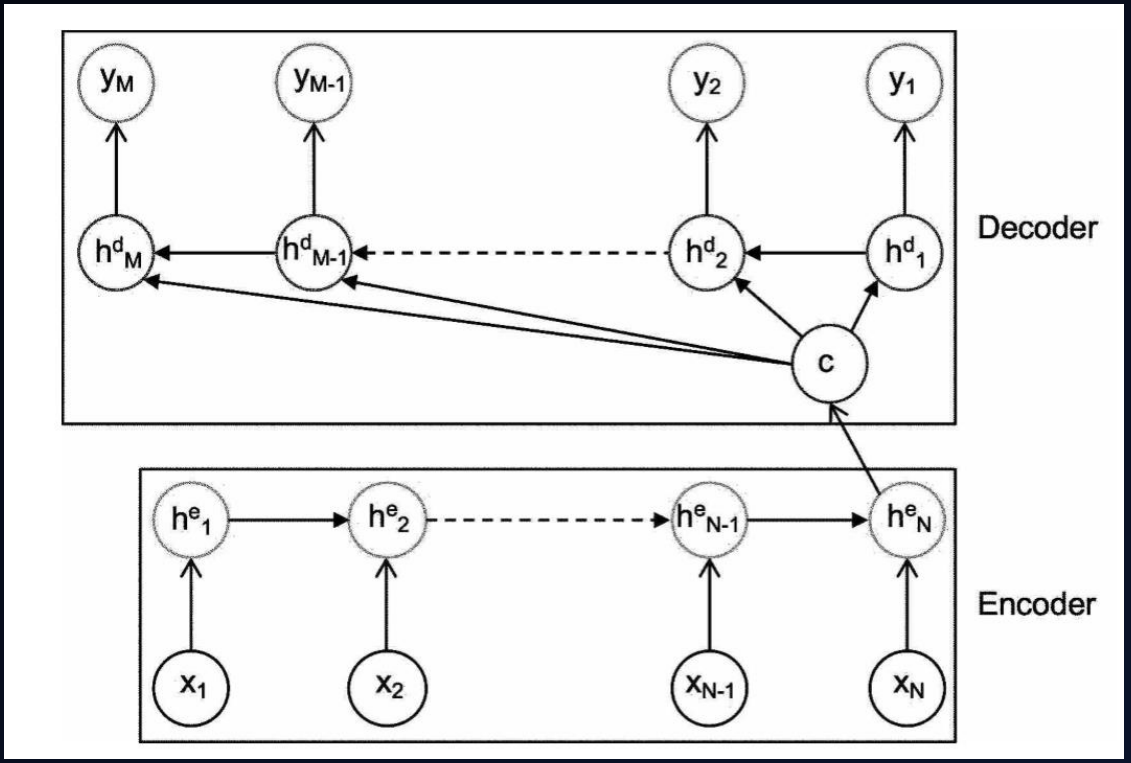


Method	Cross-entropy (major)	Cross-entropy (minor)
Local maxima	1.20	1.41
Global maximum	0.84	0.87

Neural Network

RNN+RBM

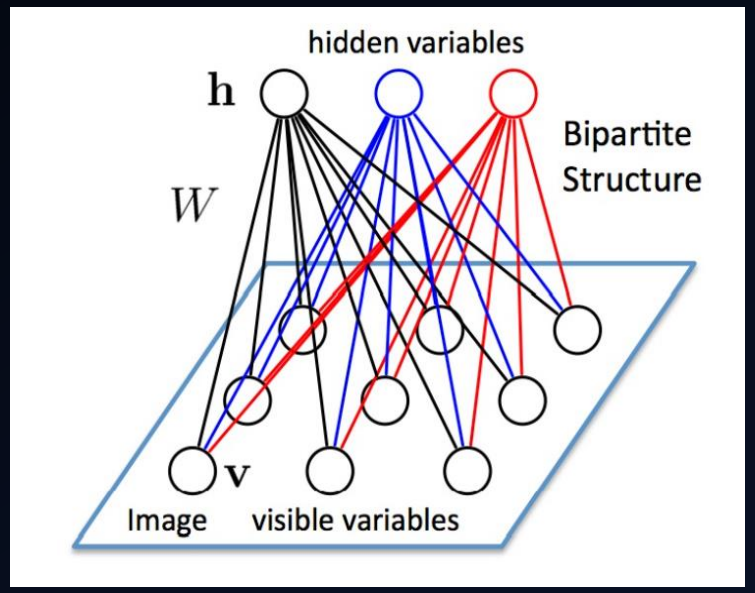
- Use RNN to capture local structure



Neural Network

RNN+RBM

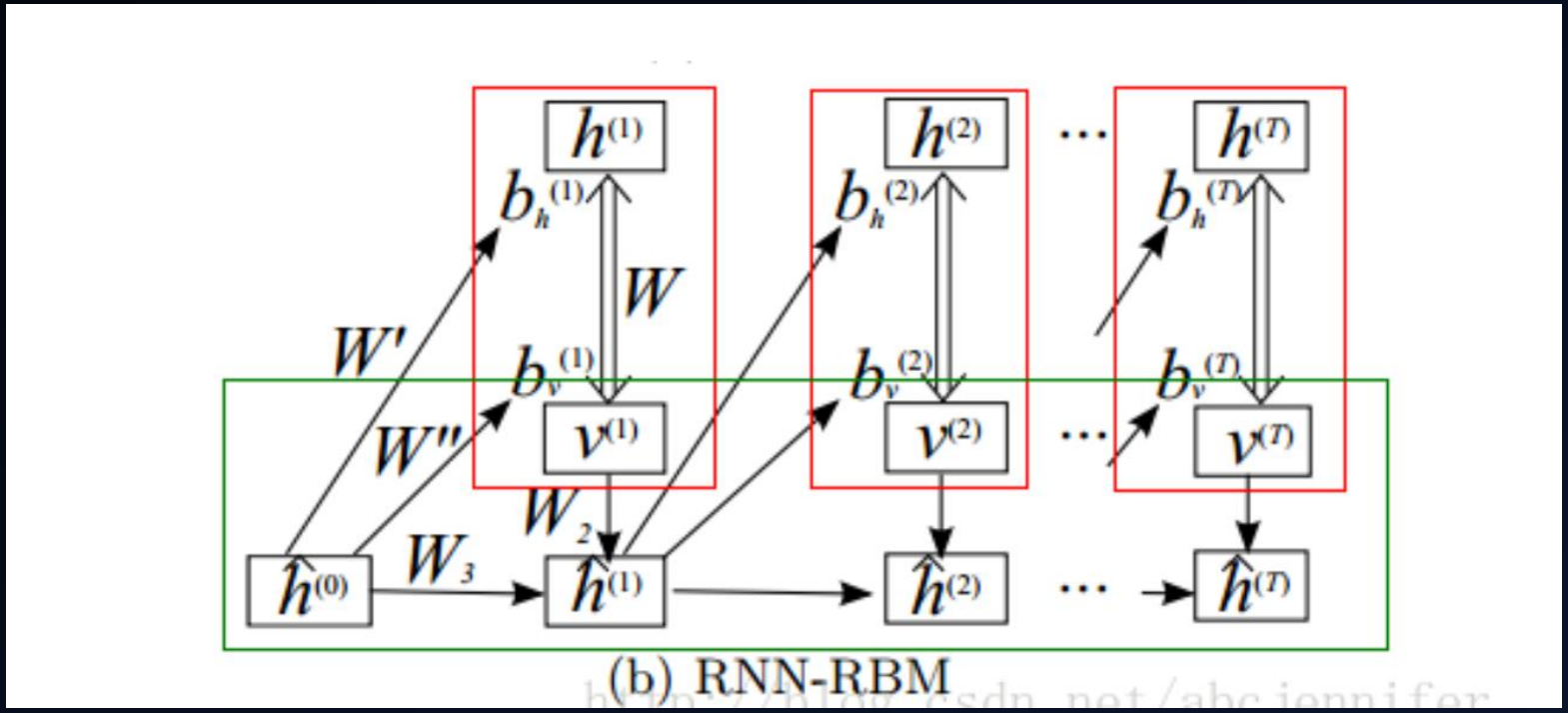
- Use Restricted Boltzmann Machine to capture global structure



Neural Network

RNN+RBM

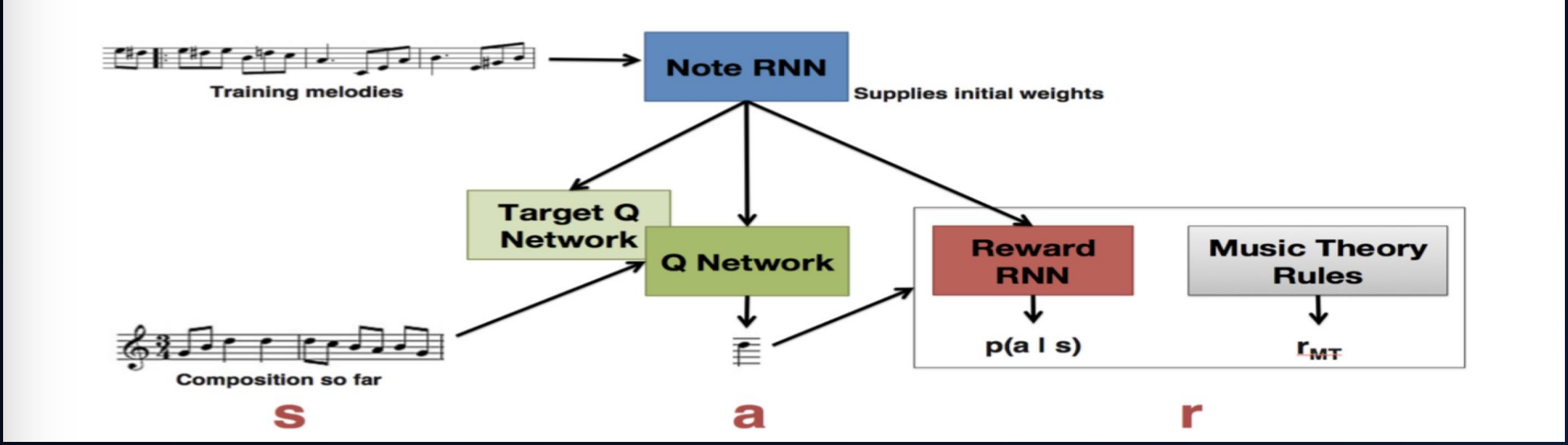
- Combine them together



(b) RNN-RBM

ML combine with Reinforcement Learning(step1)

- Given a trained RNN such as an LSTM network to supply the initial weights for the Q-network, Target-Q-network, Reward RNN



ML combine with Reinforcement Learning(step2)

- Deep Q-learning use a DQN neural network to learn the optimal Q function which follows Bellman Equation

$$Q(s_t, a_t; \pi^*) = r(s_t, a_t) + \gamma \mathbb{E}_{p(s_{t+1}|s_t, a_t)}[\max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \pi^*)]$$

- The total reward combine probabilities learned from the training data with knowledge of music theory

$$r(s, a) = \log p(a|s) + \frac{1}{c} r_{MT}(a, s)$$

- Then we get a new loss function and modified reward function

ML combine with Reinforcement Learning(step2)

$$L(\theta) = \mathbb{E}_{\beta}[(\log p(a|s) + \frac{1}{c} r_{MT}(a,s) + \gamma \max_{a'} Q(s', a'; \theta^{-}) - Q(s,a;\theta))^2]$$

$$\pi_{\theta}(a|s) = \delta(a = \arg\max_{a} Q(s,a;\theta))$$

where β is exploration policy, and θ− is the parameter of the *Target Q-network* and held fixed during the gradient computation. The moving average of θ is used as θ− . The ε-greedy method is used for exploration

ML combine with Reinforcement Learning(step3)

- Using two different off-policy :Ψ-learning and G-learning to solve the KL-regularized RL problem

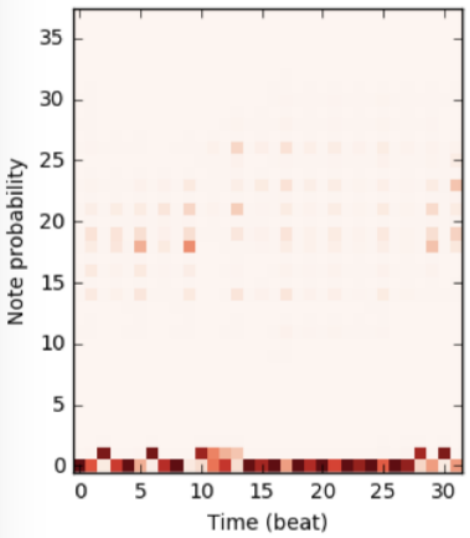
$$L(\theta) = \mathbb{E}_\beta[(\log p(a|s) + \frac{1}{c} r_{MT}(s,a) + \gamma \log \sum_{a'} e^{\Psi(s',a';\theta^-)} - \Psi(s,a;\theta))^2]$$

$$\pi_\theta(a|s) \propto e^{\Psi(s,a;\theta)}$$

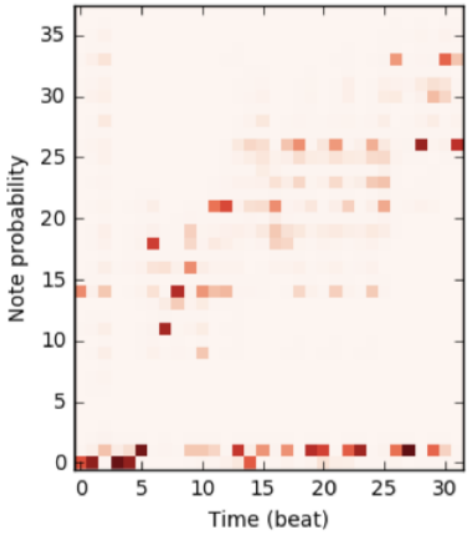
$$L(\theta) = \mathbb{E}_\beta[(\frac{1}{c} r_{MT}(s,a) + \gamma \log \sum_{a'} e^{\log p(a'|s') + G(s',a';\theta^-)} - G(s,a;\theta))^2]$$

$$\pi_\theta(a|s) \propto p(a|s) e^{G(s,a;\theta)}$$

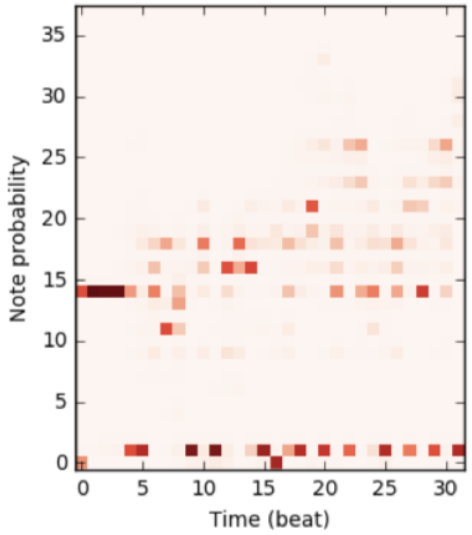
ML combine with Reinforcement Learning(results)



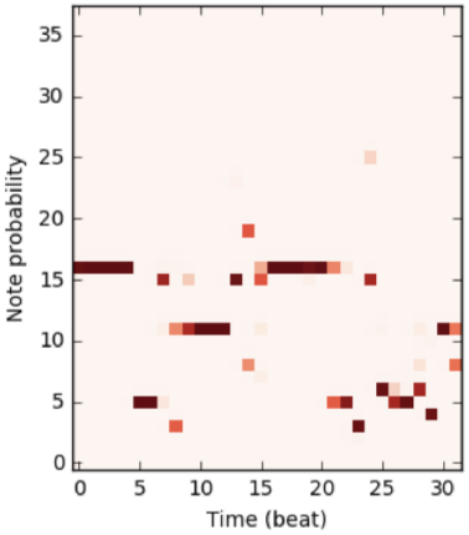
(a) *Note RNN*



(b) Q



(c) Ψ



(d) G

Comparison

HMM

- Simple to implement and train
- Low order HMM may not be able to obtain global structure

Neural Network

- Architecture of network may be very complex
- Needs a lot of examples
- Could not be directly controlled
- Avoid generating repeat sequence

Challenge of this topic

- Hard to evaluate the output
- Since the program is written by human, how to give “creativity” to the program

THANKS!

