

CNN RNN architecture

Thursday, May 3, 2018 18:09

recurrence formula

$$h_t = f_w(h_{t-1}, x_t)$$

Vanilla RNN:

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$

Language model.

softmax will make output diversity than argmax.

back propagation through time.

$$\begin{aligned} h_t &= \tanh(W_{hh}h_{t-1} + W_{xh}x_t) \\ &= \tanh\left(W \cdot \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}\right) \end{aligned}$$

LSTM:

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

$$C_t = f \odot C_{t-1} + i \odot g$$

$$h_t = o \odot \tanh(C_t)$$

GRU: ~~is~~ LSTM.

max pooling \rightarrow up sampling.

convolution & transpose convolution.

$$\vec{x} * \vec{a} = X\vec{a}$$

$$\vec{x} * \vec{a}^T = X^T \vec{a}$$

t-SNE: powerful than PCA.