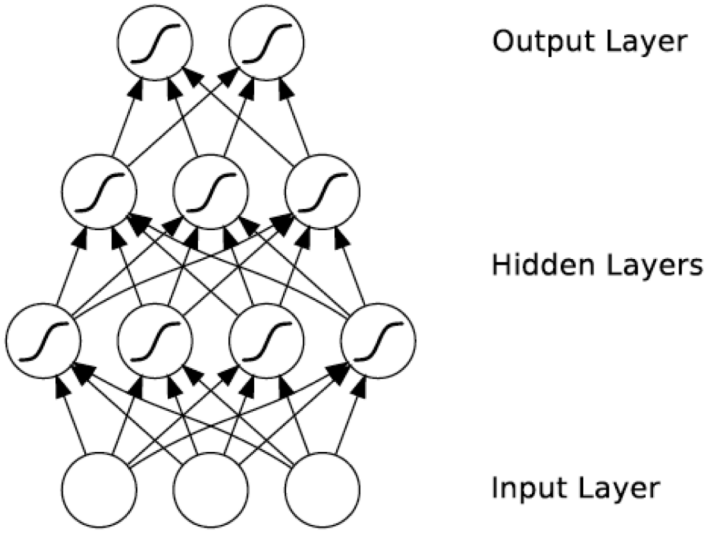
Recurrent Neural Networks PART I

Minlie Huang
aihuang@tsinghua.edu.cn
Dept. of Computer Science and Technology
Tsinghua University
http://coai.cs.tsinghua.edu.cn/hml/

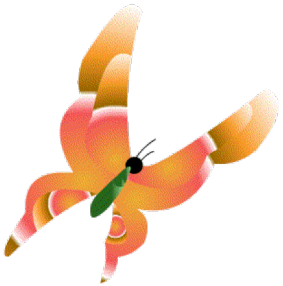
What's wrong with MLPs?

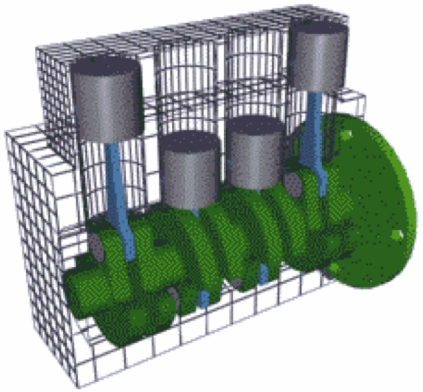
- Problem 1: Can't model sequences
  - Fixed-sized inputs & outputs
  - No temporal structure, thus no dynamics

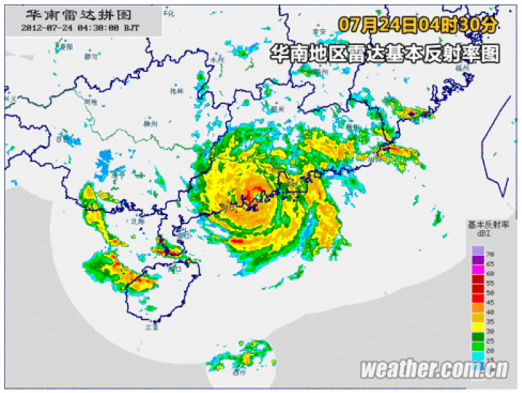
- Problem 2: Pure feed-forward processing
  - No "memory", no feedback

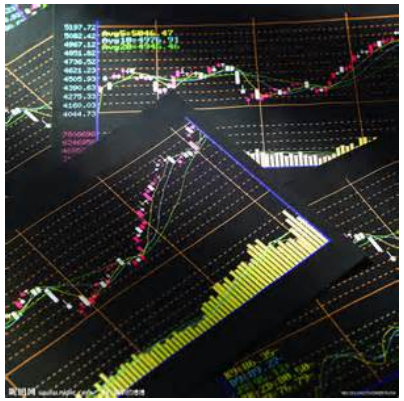


Dynamic Systems





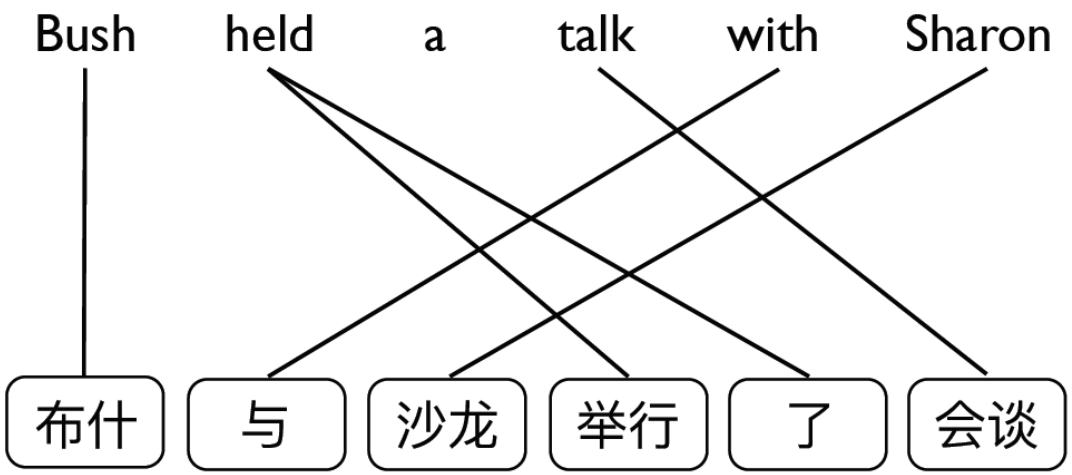




Sequences are everywhere

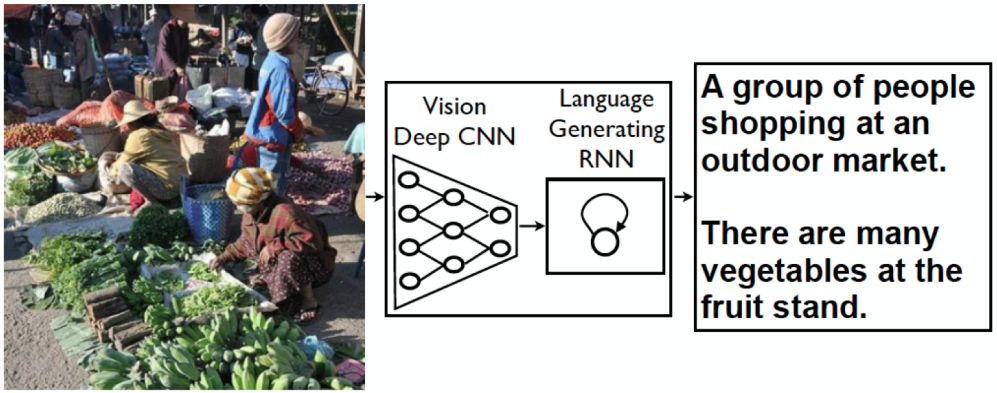
- Signal to natural language

- Translation from English to Chinese



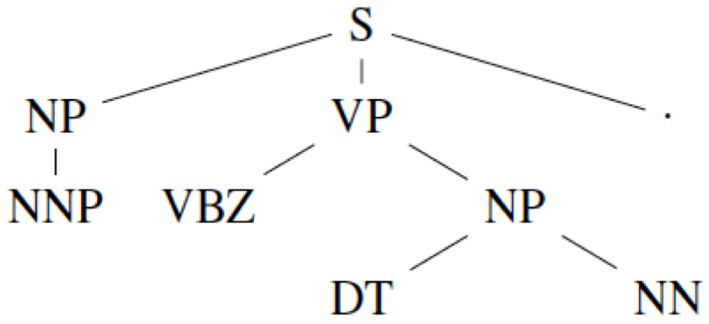
Sequences are everywhere

- Image to description



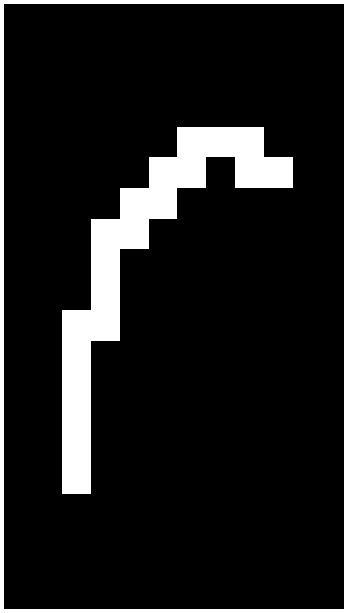
- Natural language to parsing tree

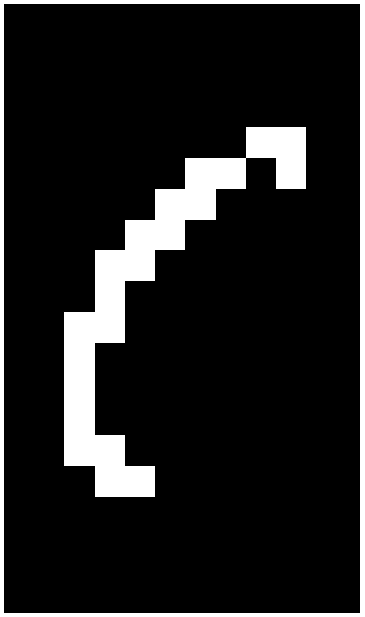
John has a dog . →



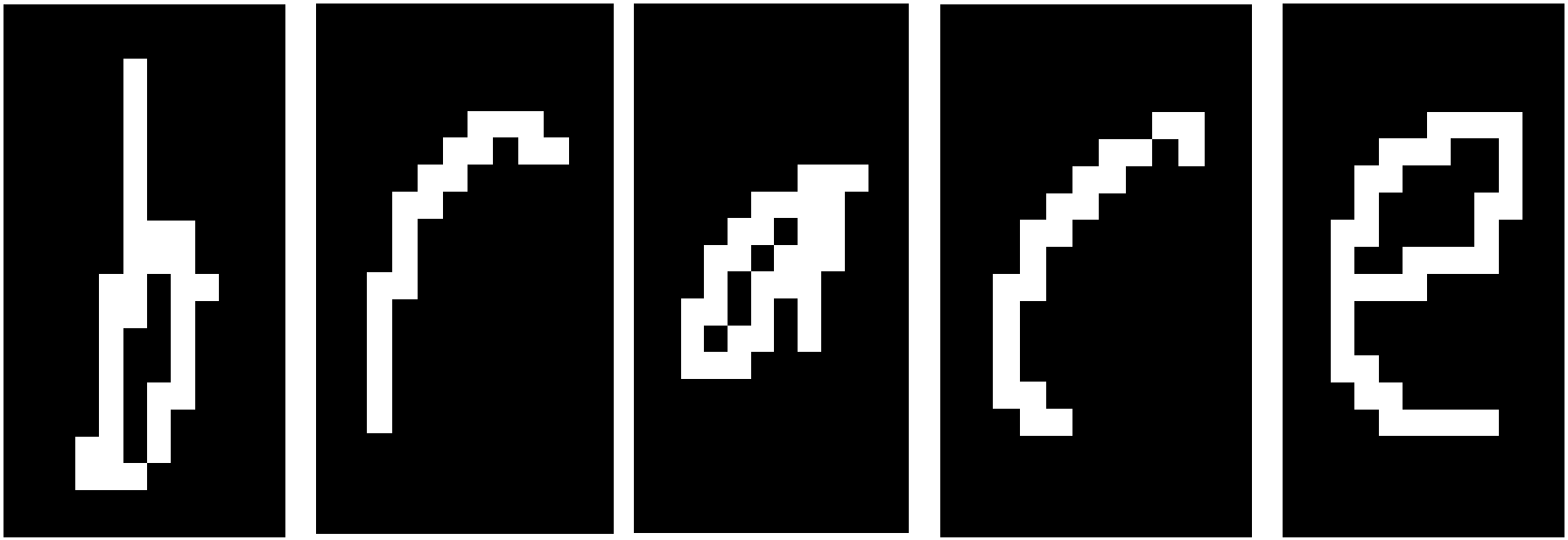
John has a dog . → (S (NP NNP)$_{NP}$ (VP VBZ (NP DT NN)$_{NP}$)$_{VP}$.)$_{S}$

Why model sequences?

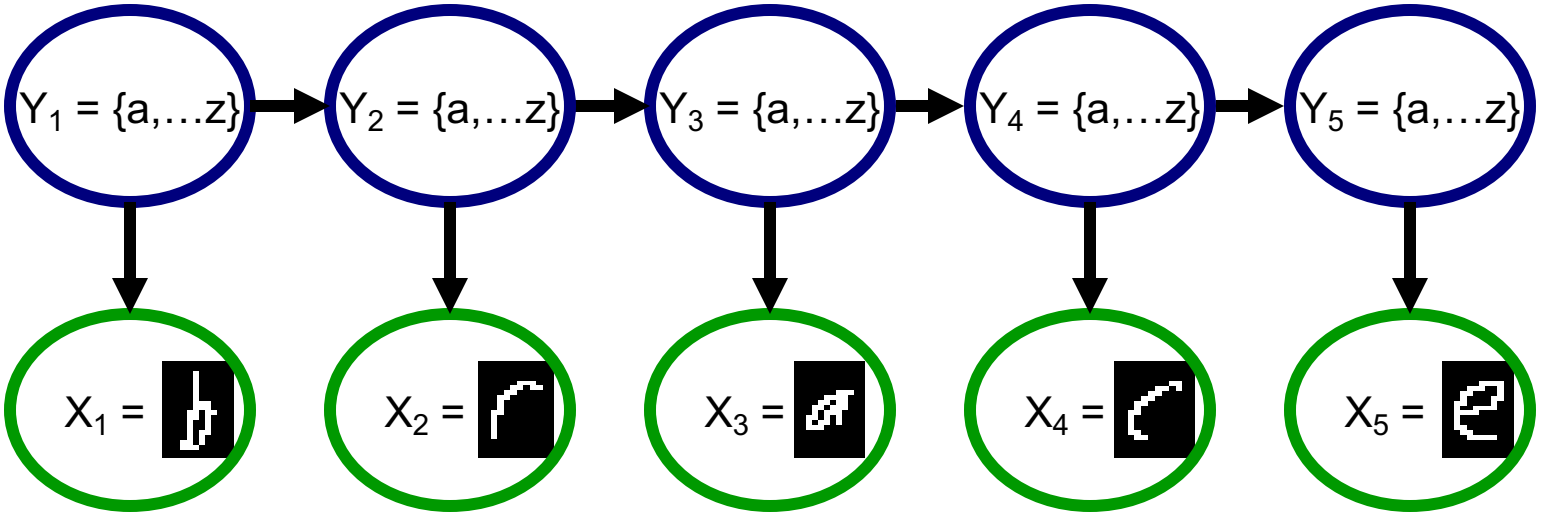




Why model sequences?



Why model sequences?

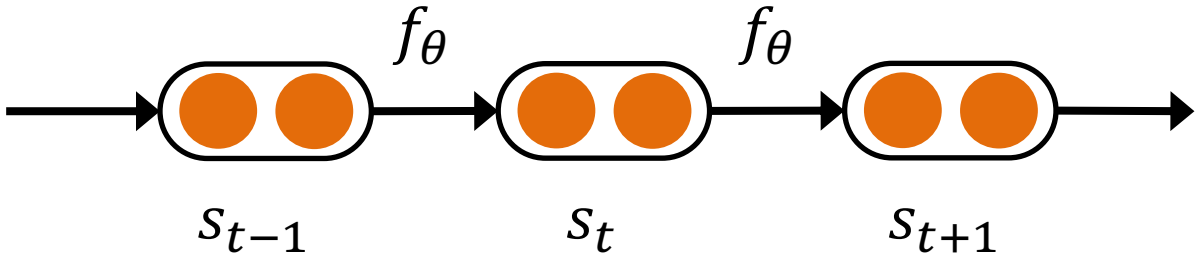


Hidden Markov Model (HMM):

$$P(X_{1\ldots n}, Y_{1\ldots n}) = P(Y_1)P(X_1|Y_1) \cdot \prod_{i=2}^{n} P(Y_i|Y_{i-1})P(X_i|Y_i)$$

How to model sequences?

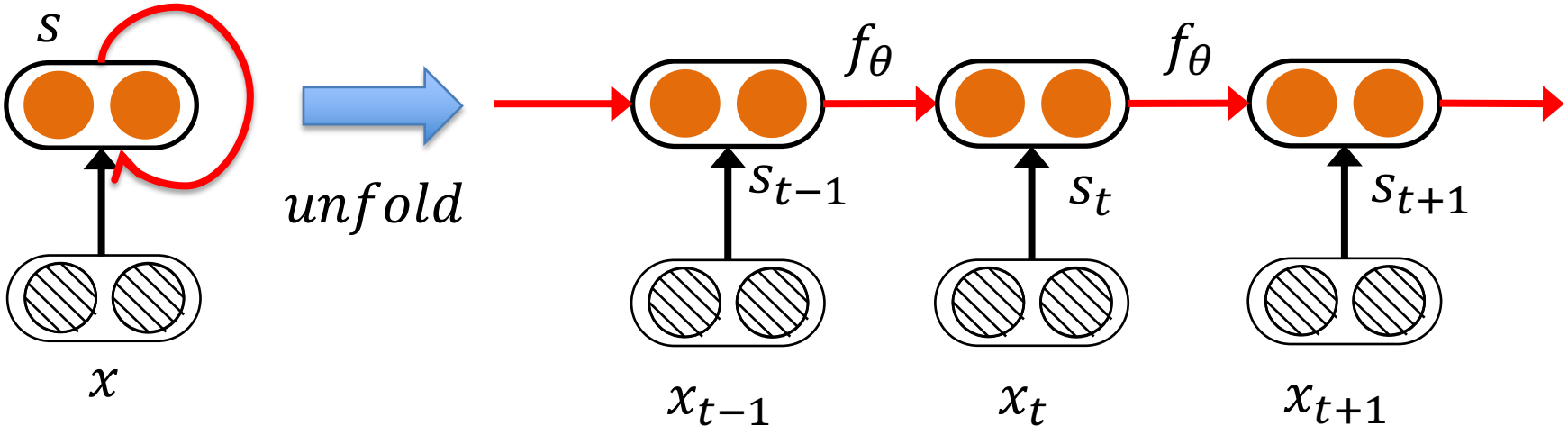
- No input



$$s_t = f_\theta(s_{t-1})$$

How to model sequences?

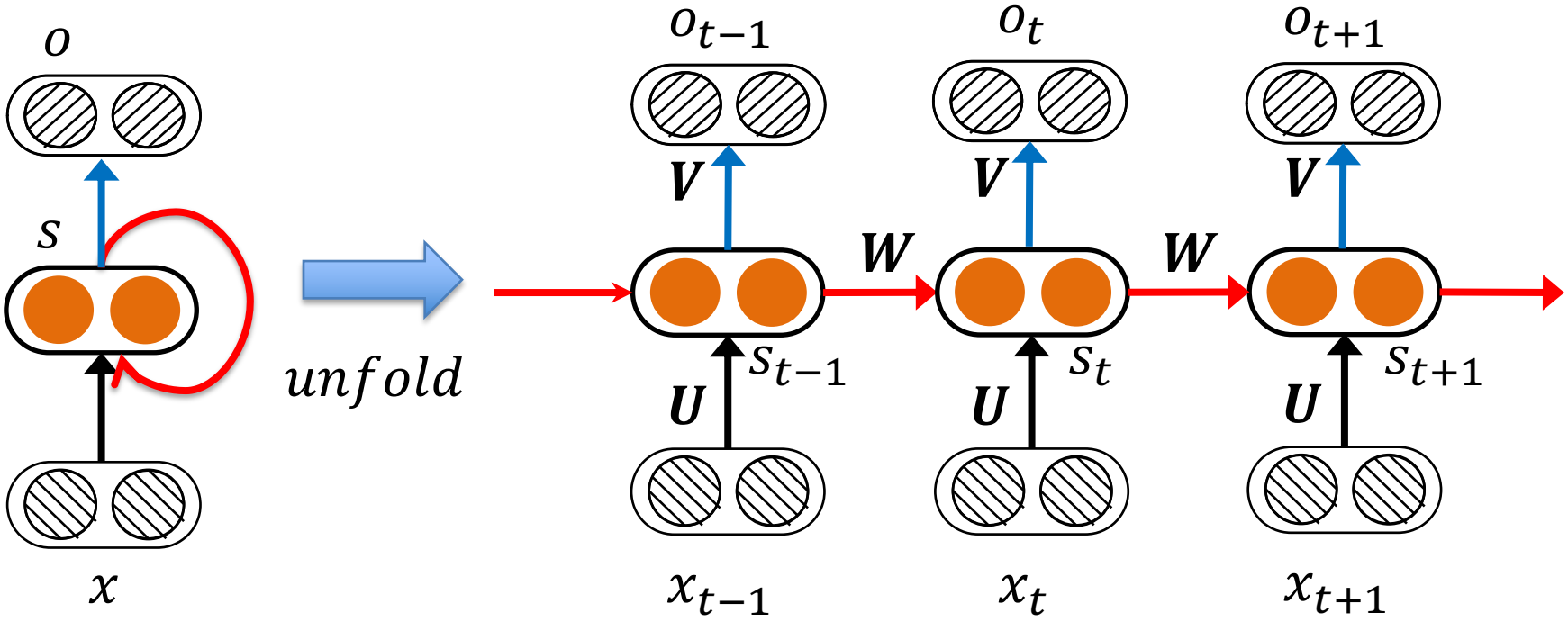
- With inputs



$$s_t = f_\theta(s_{t-1}, x_t)$$

How to model sequences?

- With inputs and outputs

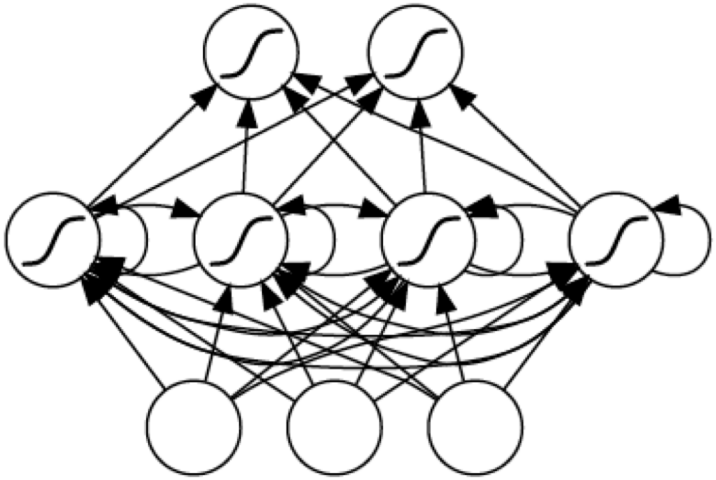


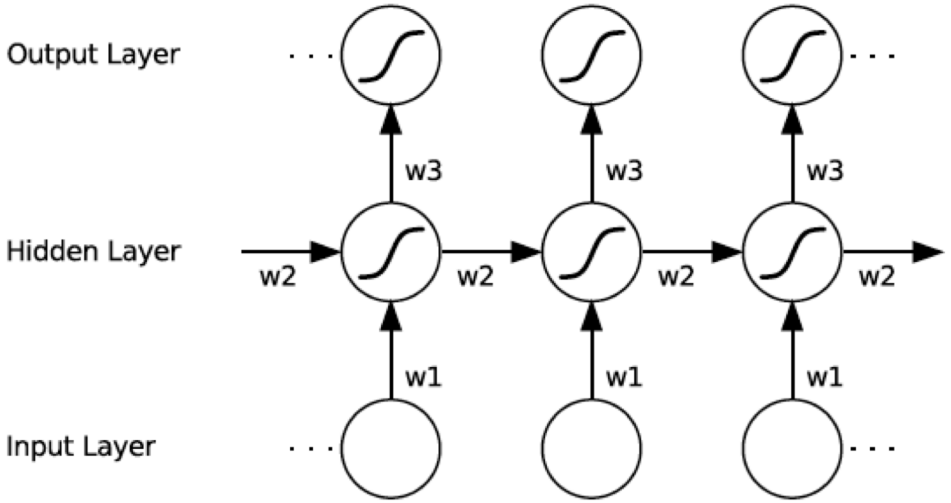
$$s_t = f(Ws_{t-1} + Ux_t)$$

$$o_t = g(Vs_t)$$

How to model sequences?

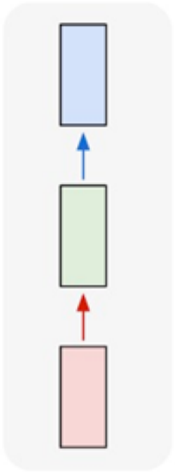
- With neural nets





How to model sequences?

one to one

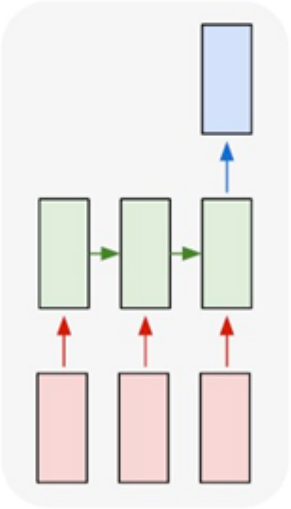


Input: No sequence

Output: No sequence

Example: “standard” classification / regression

many to one

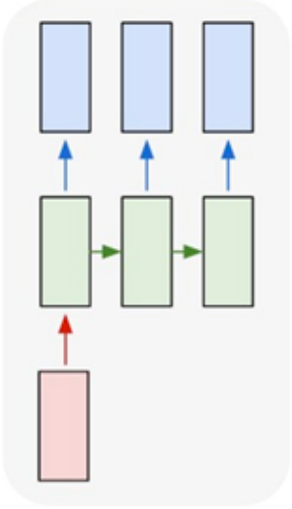


Input: No sequence

Output: Sequence

Example: Image2Caption

one to many

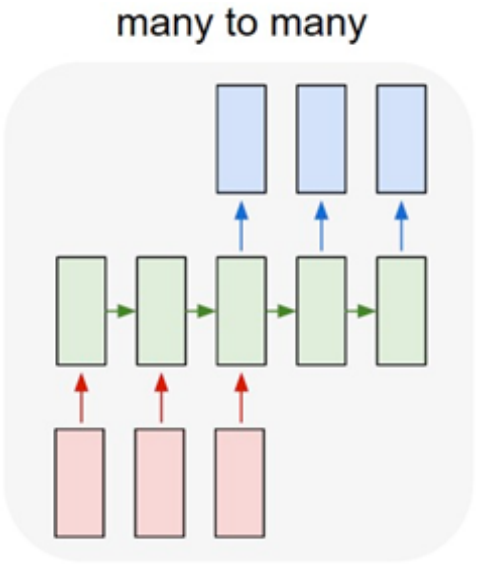


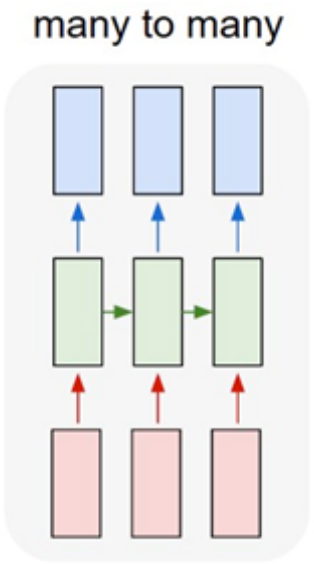
Input: Sequence

Output: No sequence

Example: sentence classification, multiple-choice question answering

How to model sequences?

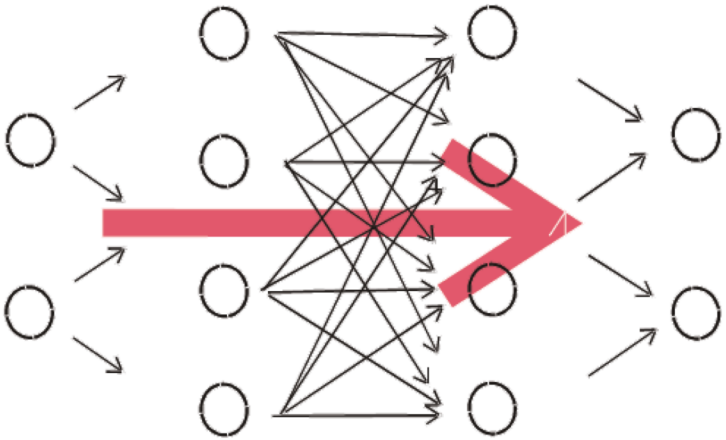




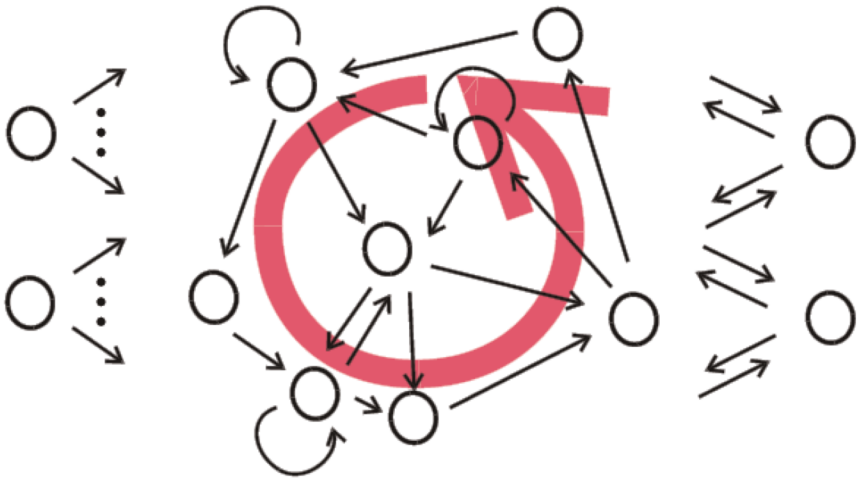
Input: Sequence, **Output**: Sequence

Example: machine translation, video captioning, open-ended question answering, video question answering

Architecture



Feed-forward



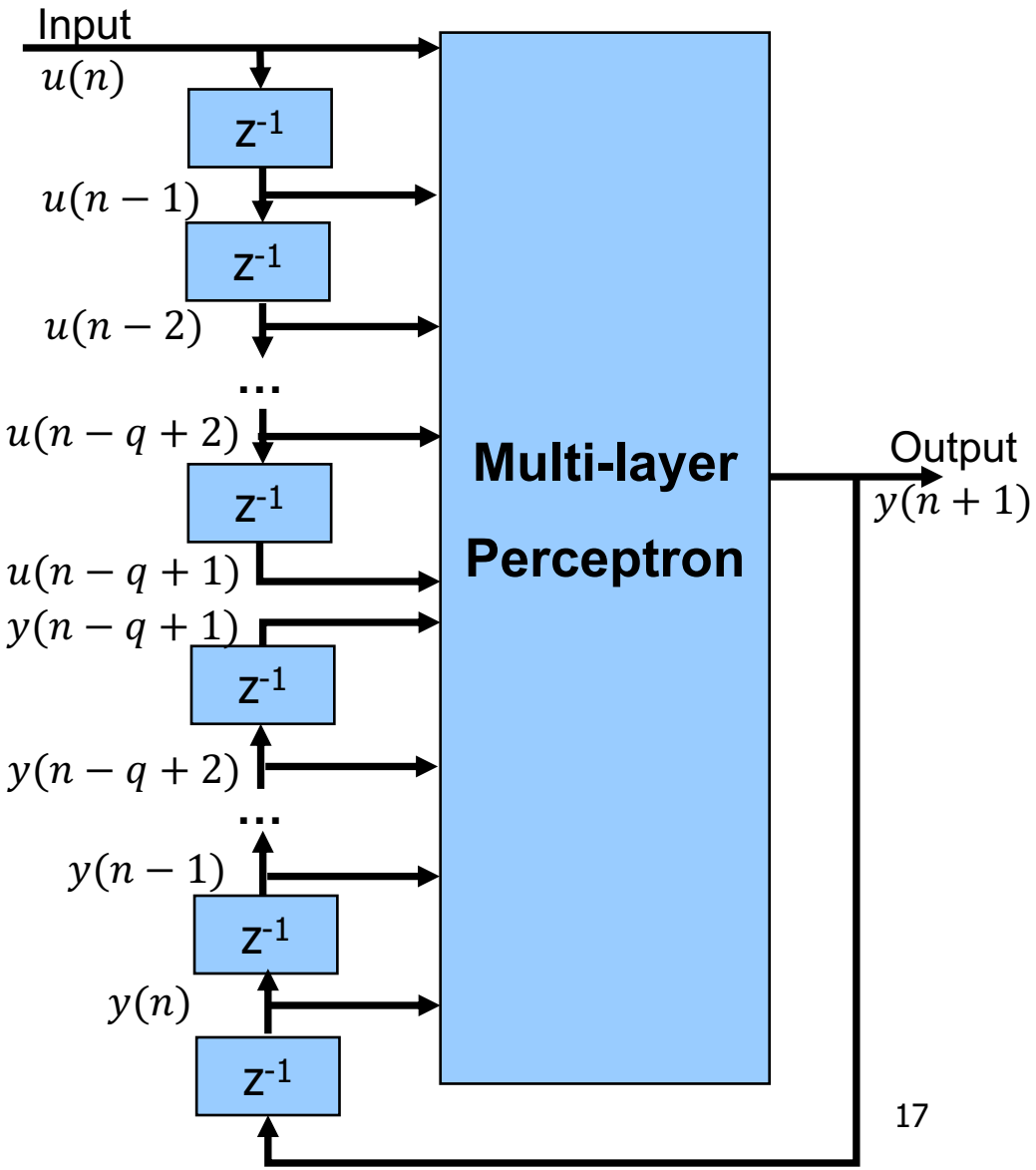
Recurrent

Architecture

- Three types of typical architectures
  - Input-output recurrent model
  - State-space model
  - Recurrent multilayer perceptron
- Common features
  - Incorporate a static multilayer perceptron or parts thereof
  - Exploit the nonlinear mapping capability of the multilayer perceptron

Architecture #1

- **Input-output recurrent model** (nonlinear auto-regressive with exogenous input model) with the feed-back of output



Input-output recurrent model

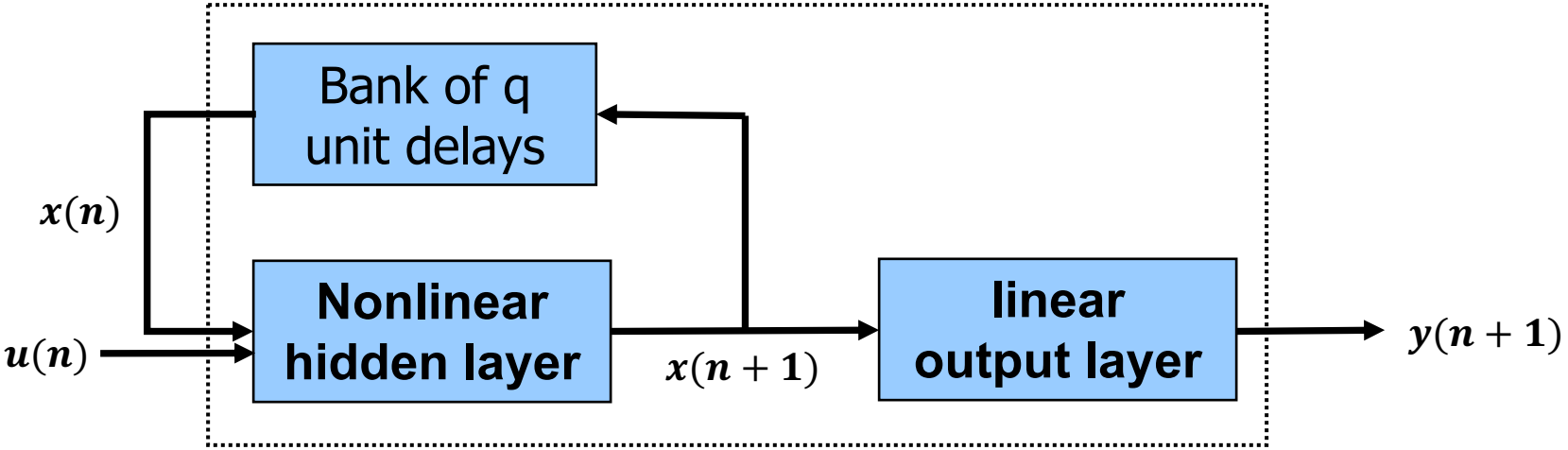
- Input $u(n)$ → Output $y(n+1)$
- Present and past input values : $u(n), u(n-1), u(n-2), \ldots, u(n-q+1)$
- Output $y(n+1)$ is regressed by

$$y(n+1) = \mathrm{F}(y(n), \ldots, y(n-q+1), u(n), \ldots u(n-q+1))$$

where F is a nonlinear function of its arguments

Architecture #2

- State-space model



Recurrence is in the state variables

State-space model

- The hidden neurons define the state of the network.
- Input $u(n)$, hidden $x(n)$, output $y(n)$
- The hidden layer is nonlinear, but the output layer is linear

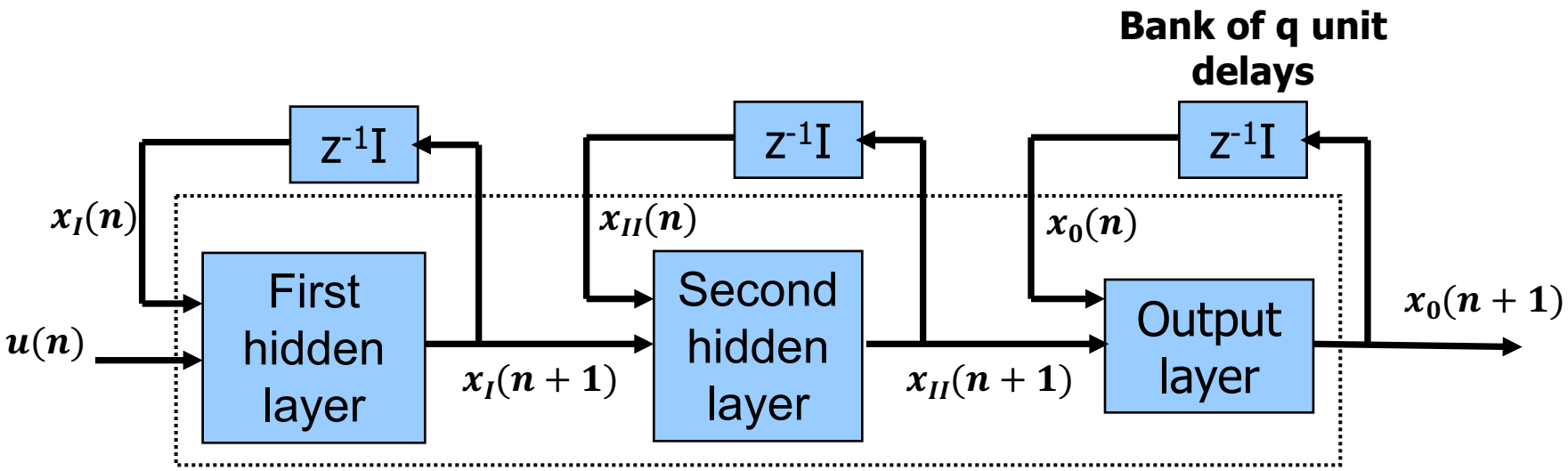
$$x(n+1) = f\big(x(n), u(n)\big)$$

$$y(n) = C \cdot x(n)$$

where f is a nonlinear function characterizing the hidden layer, and C is the matrix of synaptic weights characterizing the output layer

Architecture #3

- Recurrent multilayer perceptron (RMLP)



Stacking multiple Architecture #2

RMLP

- Output of the 1st, 2nd …hidden layers are $x_I(n)$, $x_{II}(n)$ … respectively
- Input is $u(n)$ in the first layer

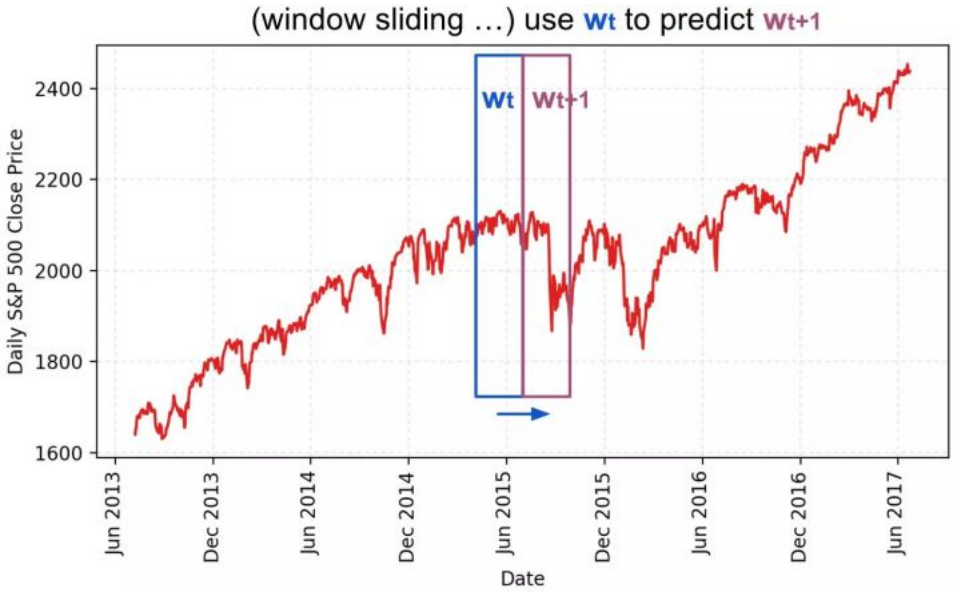
$$x_I(n+1) = \varphi_I(x_I(n), u(n))$$
$$x_{II}(n+1) = \varphi_{II}(x_{II}(n), x_I(n+1))$$
$$\ldots$$
$$x_o(n+1) = \varphi_o(x_o(n), x_K(n+1))$$

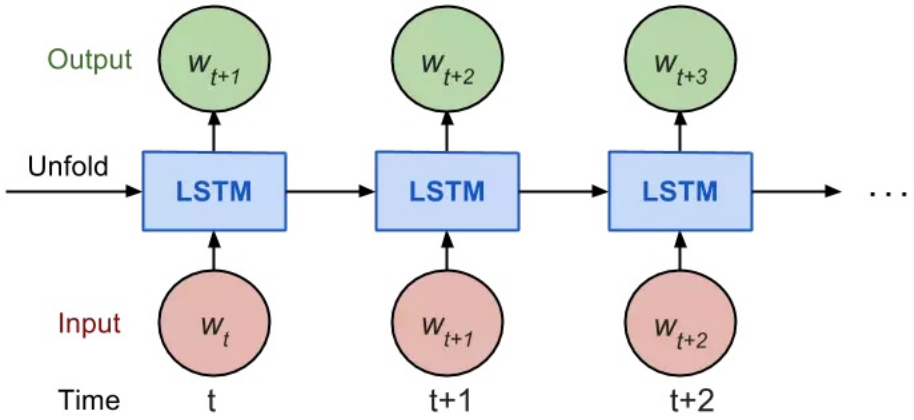
where φ denotes the activation function characterizing corresponding layer of the RMLP, and K denotes the number of hidden layers in the network

Application

- Unstructured Sequential Prediction
  - Stock, air quality, etc.
- Structured Prediction (mostly in NLP)
  - Name entity recognition
  - Part of speech tagging
  - Machine translation
  - Speech recognition
  - Question answering
  - Image caption
- Other applications
  - Melody generation

Application: Stock Prediction

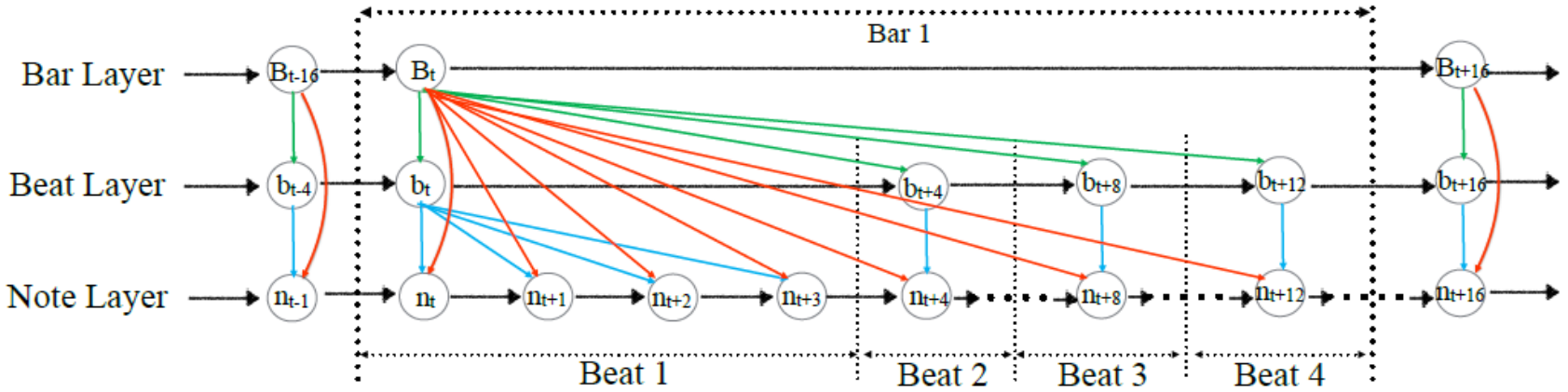


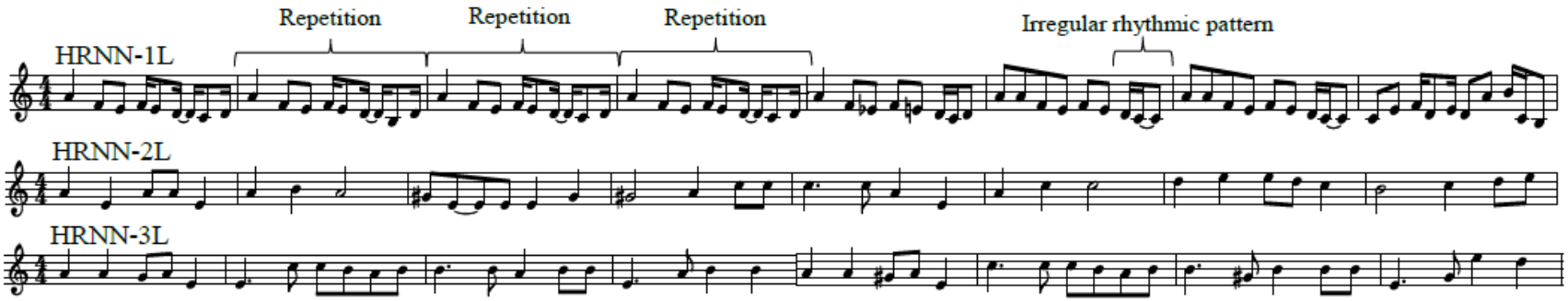


$$W_0 = (p_0, p_1, \ldots, p_{w-1})$$
$$W_1 = (p_w, p_{w+1}, \ldots, p_{2w-1})$$
$$\ldots$$
$$W_t = (p_{tw}, p_{tw+1}, \ldots, p_{(t+1)w-1})$$

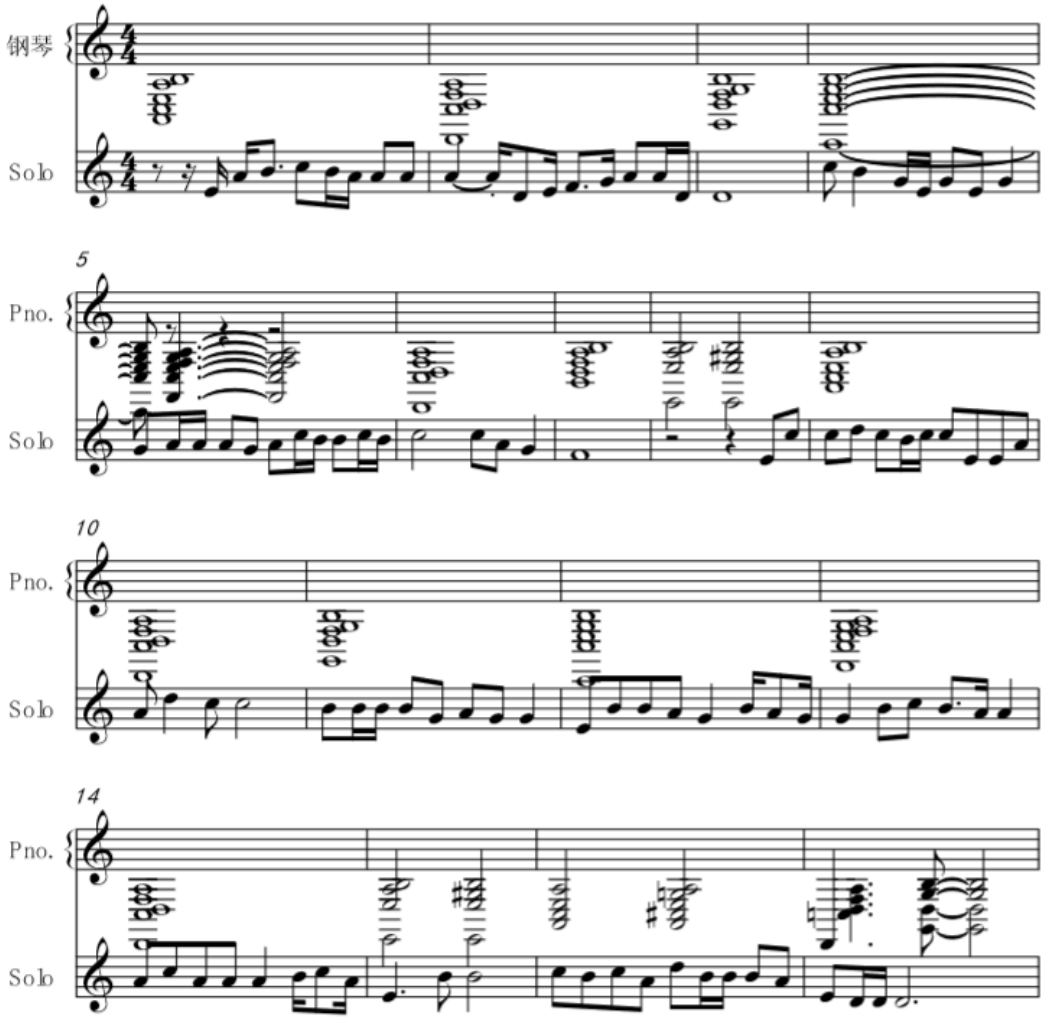
$$f(W_0, W_1, \ldots, W_t) \approx W_{t+1}$$

Application: Melody Generation



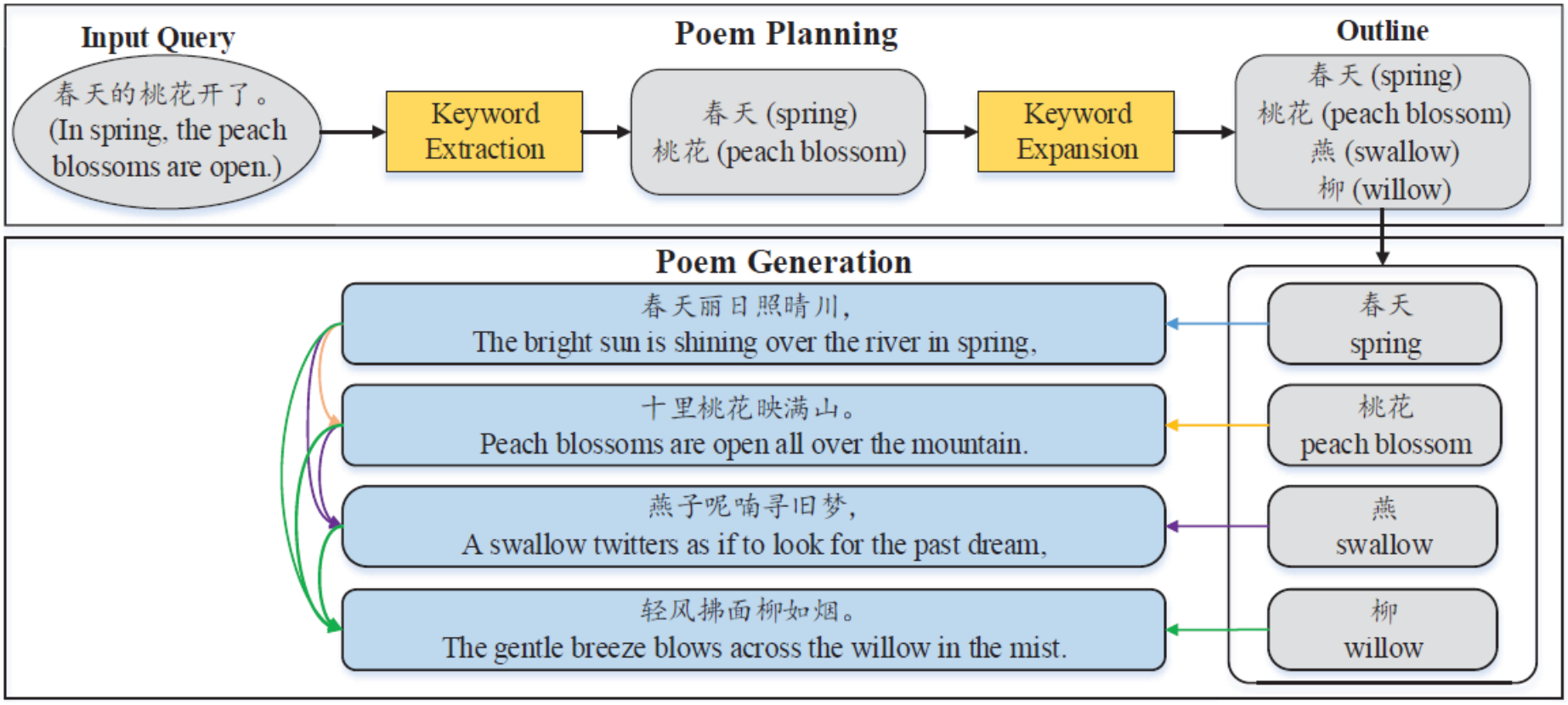


Application: Melody Generation



Encode chord based on domain knowledge with hierarchical recurrent neural network

Application: Poem Generation



Zhe Wang, Wei He, Hua Wu, Haiyang Wu, Wei Li, Haifeng Wang, Enhong Chen. Chinese Poetry Generation with Planning based Neural Network. COLING 2016.

End