Language Understanding

Mar 21st, 2017

Intelligent Conversational Bot

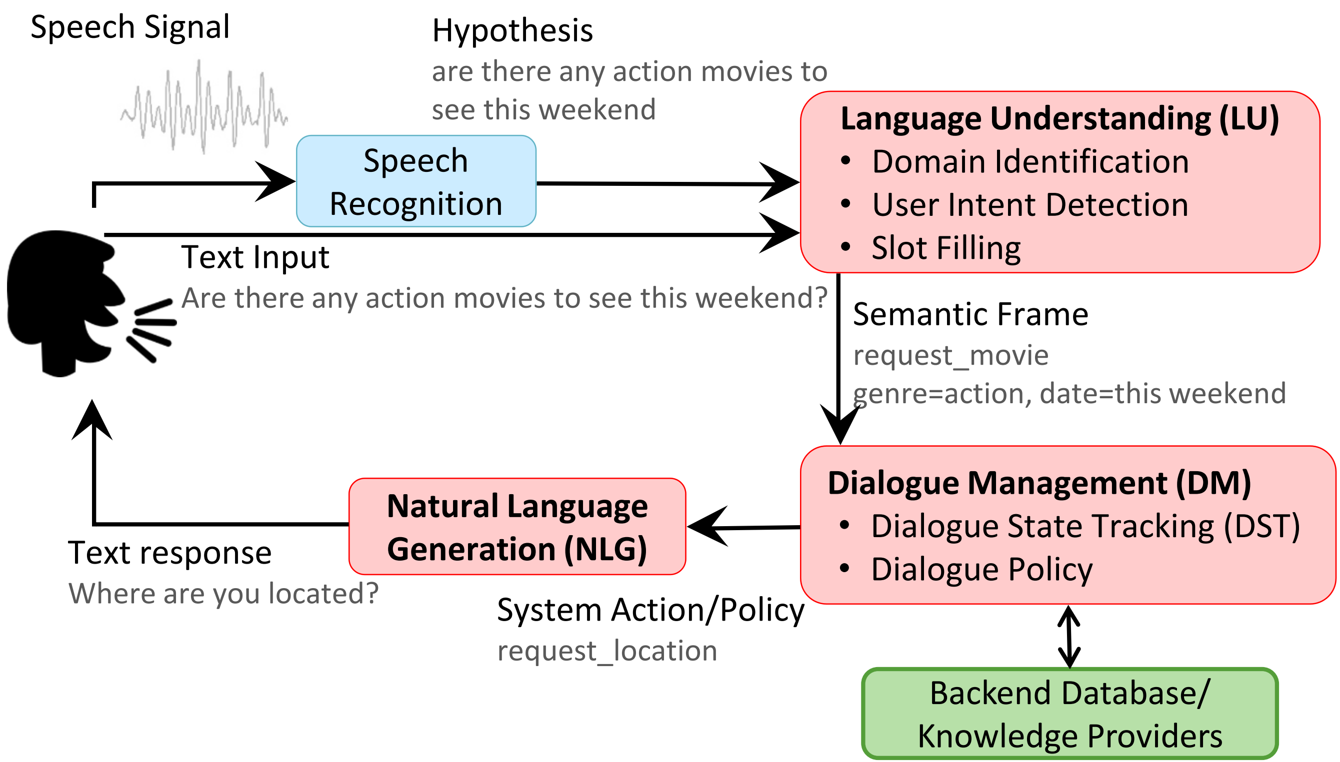
YUN-NUNG (VIVIAN) CHEN
WWW.CSIE.NTU.EDU.TW/~YVCHEN/S105-ICB

國立臺灣大學
National Taiwan University

Review

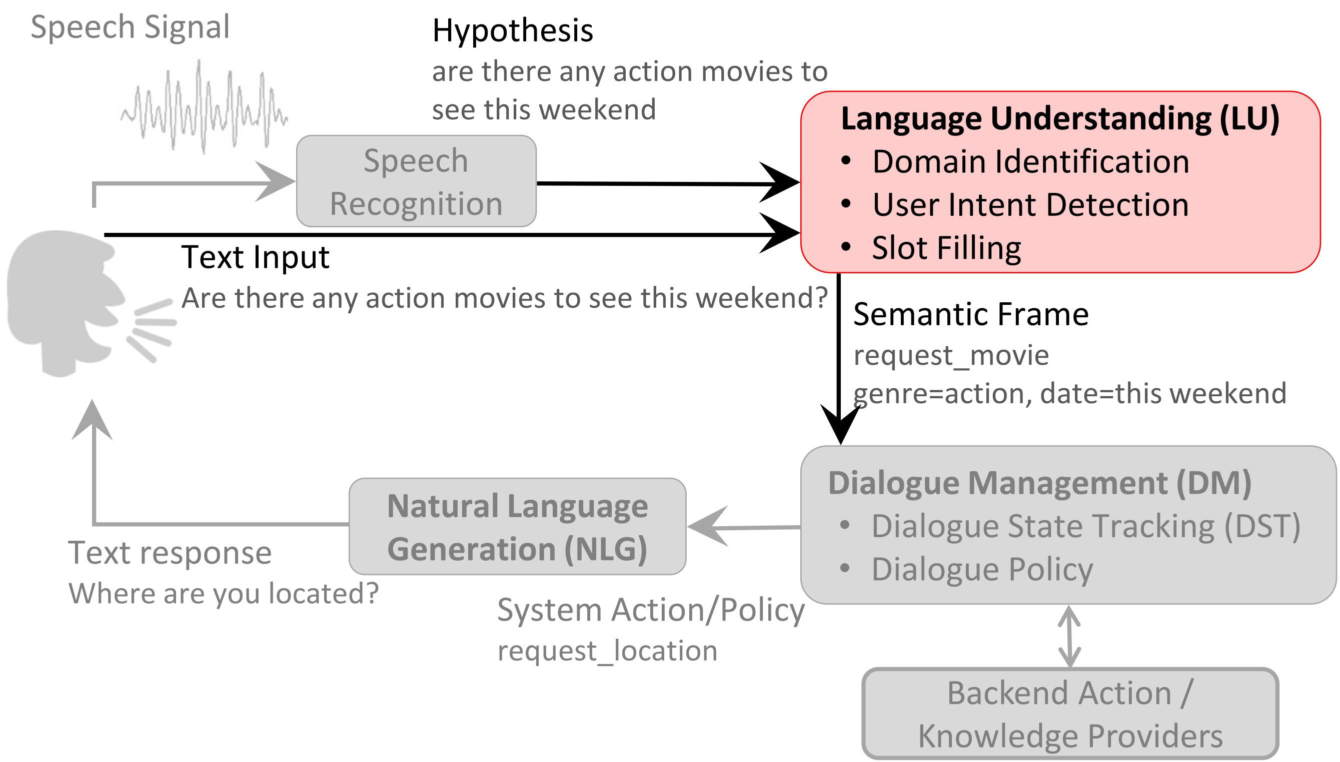
Task-Oriented Dialogue System (Young, 2000)

http://rsta.royalsocietypublishing.org/content/358/1769/1389.short



Task-Oriented Dialogue System (Young, 2000)

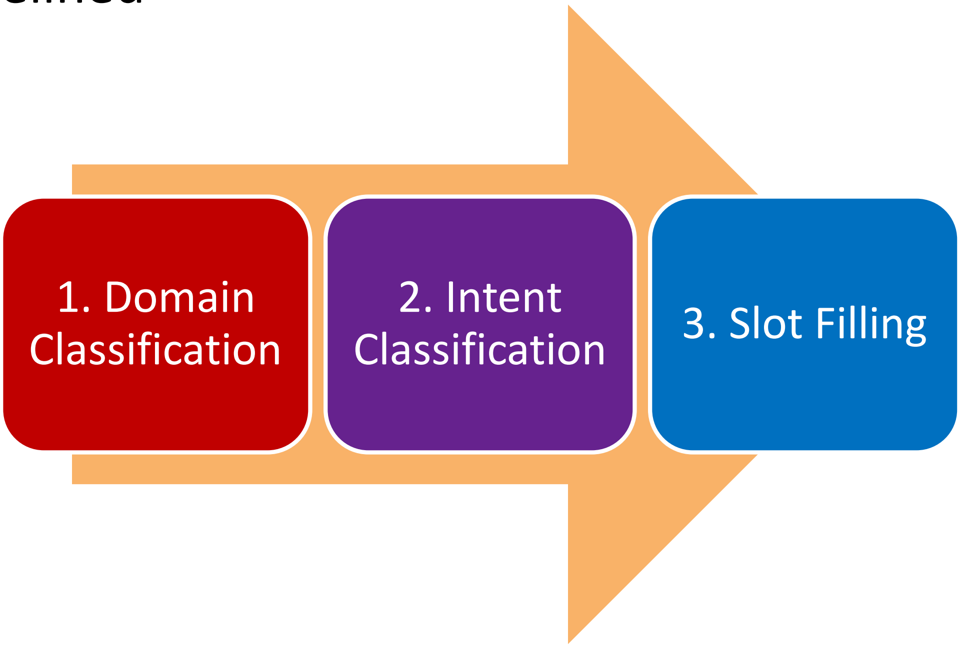
http://rsta.royalsocietypublishing.org/content/358/1769/1389.short



Conventional LU

Language Understanding (LU)

- Pipelined



LU – Domain/Intent Classification

As an **utterance classification** task

- Given a collection of utterances u_i with labels c_i, $D = \{(u_1, c_1), \ldots, (u_n, c_n)\}$ where $c_i \in C$, train a model to estimate labels for new utterances u_k.

find me a cheap taiwanese restaurant in oakland

Domain	Intent
Movies	find_movie, buy_tickets
<u>Restaurants</u>	<u>find_restaurant</u>, find_price, book_table
Music	find_lyrics, find_singer
Sports	…
…	

Conventional Approach

Data

dialogue utterances annotated with domains/intents

Model

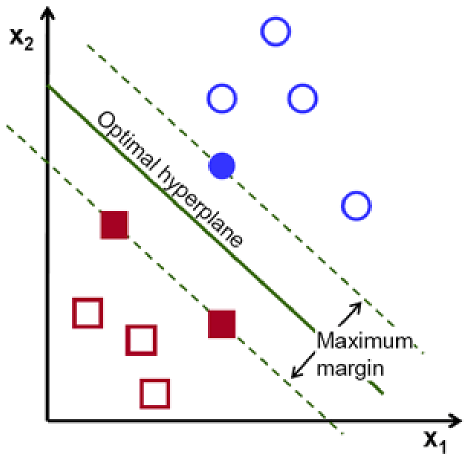
machine learning classification model
e.g. support vector machine (SVM)

Prediction

domains/intents

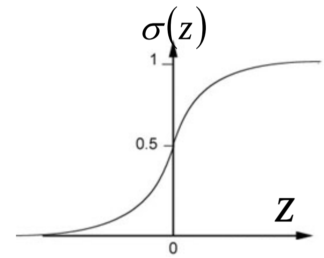
Theory: Support Vector Machine

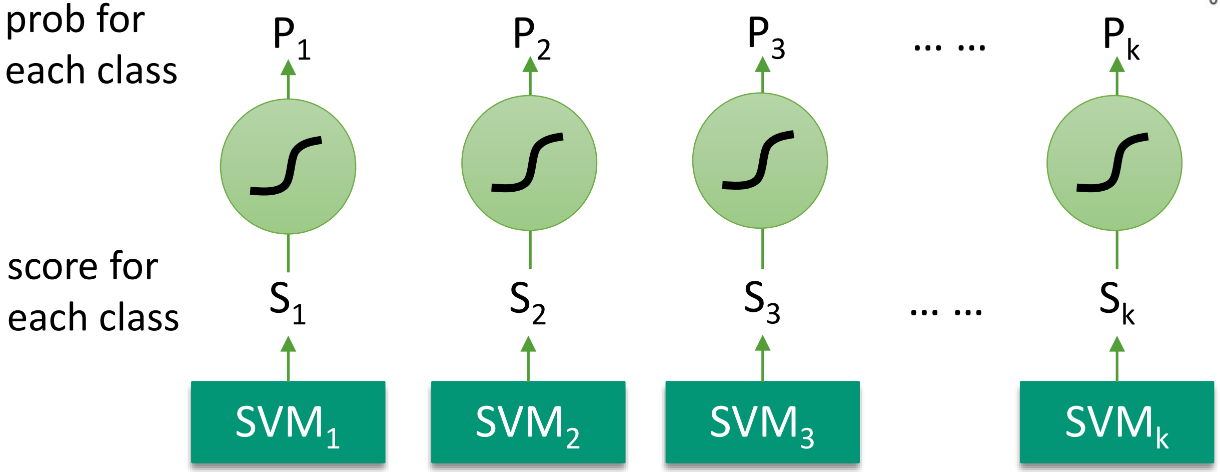
- SVM is a maximum margin classifier
  - Input data points are mapped into a high dimensional feature space where the data is linearly separable
  - Support vectors are input data points that lie on the margin



Theory: Support Vector Machine

- Multiclass SVM
  - Extended using *one-versus-rest* approach
  - Then transform into probability





Domain/intent can be decided based on the estimated scores

LU – Slot Filling

As a sequence tagging task

- Given a collection tagged word sequences, $S=\{((w_{1,1},w_{1,2},\ldots, w_{1,n1}), (t_{1,1},t_{1,2},\ldots,t_{1,n1})), ((w_{2,1},w_{2,2},\ldots,w_{2,n2}), (t_{2,1},t_{2,2},\ldots,t_{2,n2})) \ldots\}$ where $t_i \in M$, the goal is to estimate tags for a new word sequence.

flights from Boston to New York today

	flights	from	Boston	to	New	York	today
Entity Tag	O	O	B-city	O	B-city	I-city	O
Slot Tag	O	O	B-dept	O	B-arrival	I-arrival	B-date

Conventional Approach

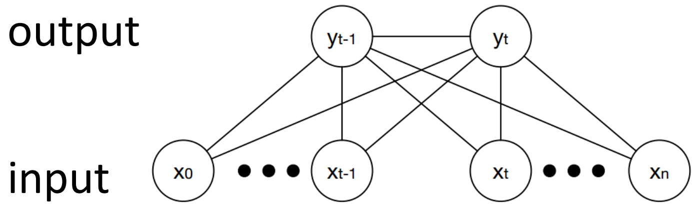
Data — dialogue utterances annotated with slots

Model — machine learning tagging model
e.g. conditional random fields (CRF)

Prediction — slots and their values

Theory: Conditional Random Fields

- CRF assumes that the label at time step t depends on the label in the previous time step t-1



- Maximize the log probability $\log p(\mathbf{y} \mid \mathbf{x})$ with respect to parameters λ

$$p(y \mid x) = \frac{1}{Z(x)} \exp(\sum_i \lambda_i f_i(x, y))$$

$$= \prod_t \frac{1}{Z(x)} \exp(\sum_i \lambda_i f_i(x, y_t, y_{t-1}))$$

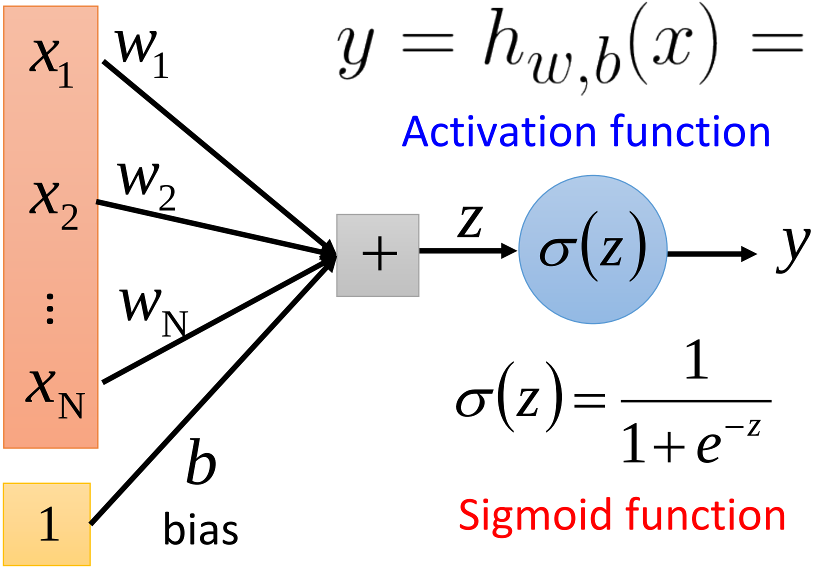
Slots can be tagged based on the y that maximizes $p(y|x)$

Neural Network Based LU

A Single Neuron

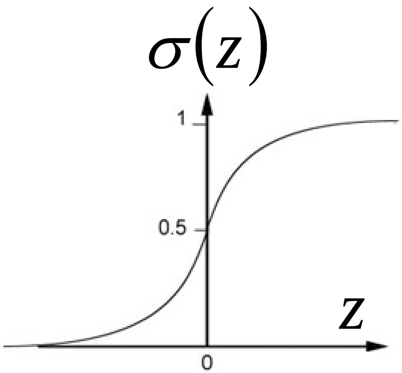
$$y = h_{w,b}(x) = \sigma(w^T x + b)$$

Activation function



$$\sigma(z) = \frac{1}{1+e^{-z}}$$

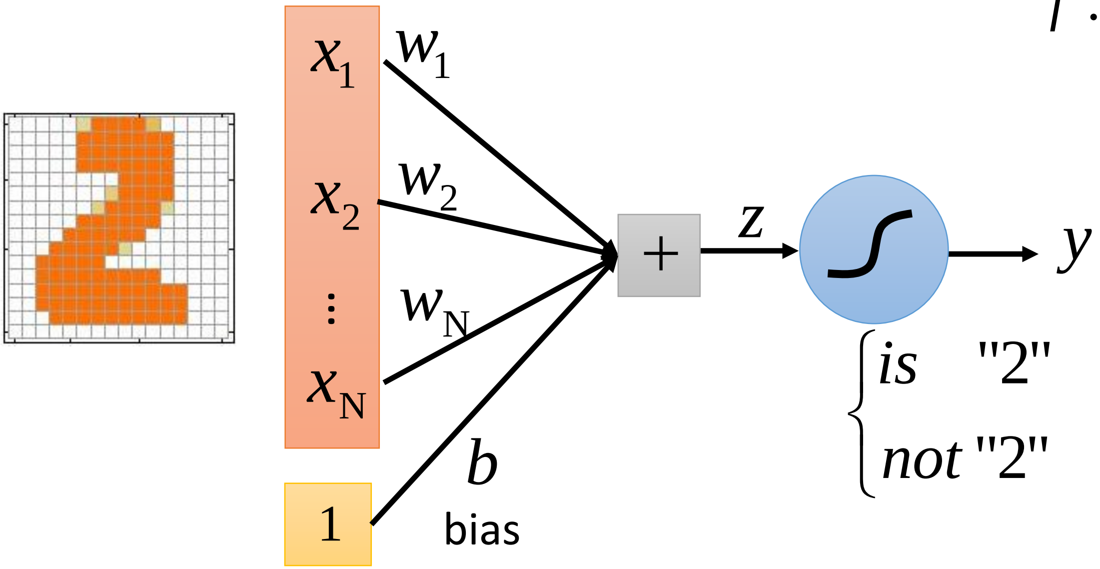
Sigmoid function



w, *b* are the parameters of this neuron

A Single Neuron

$$f : R^N \rightarrow R^M$$



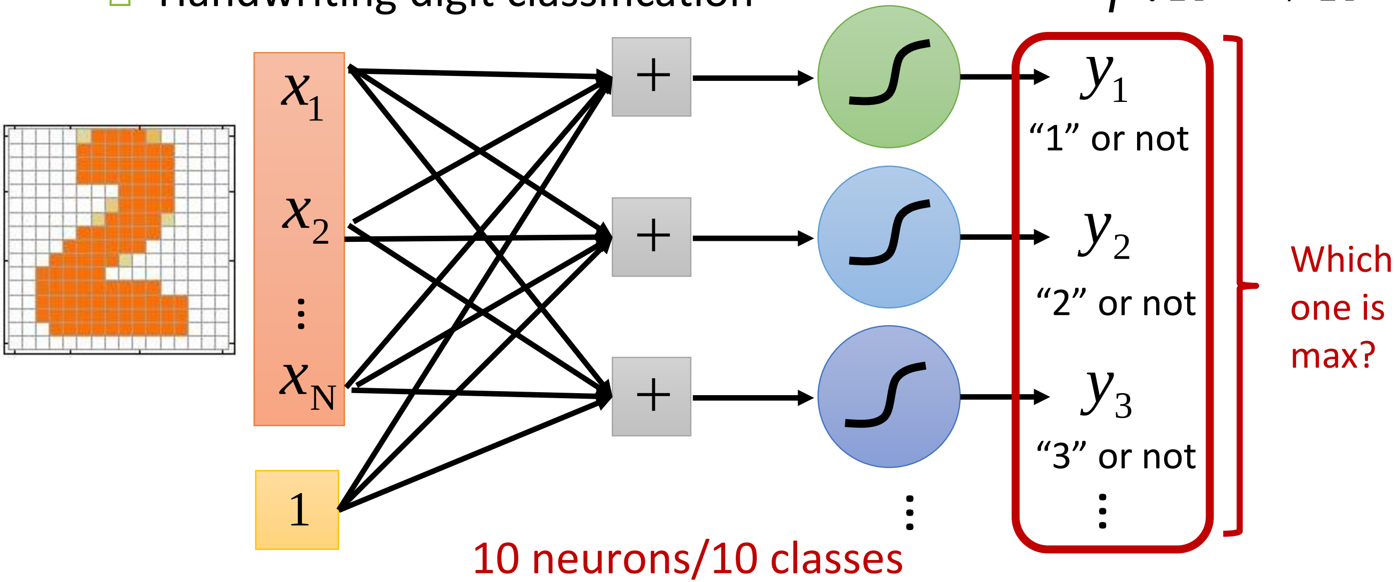
$$\begin{cases} is \quad "2" & y \geq 0.5 \\ not \; "2" & y < 0.5 \end{cases}$$

A single neuron can only handle binary classification

A Layer of Neurons

- Handwriting digit classification

$$f : R^N \rightarrow R^M$$

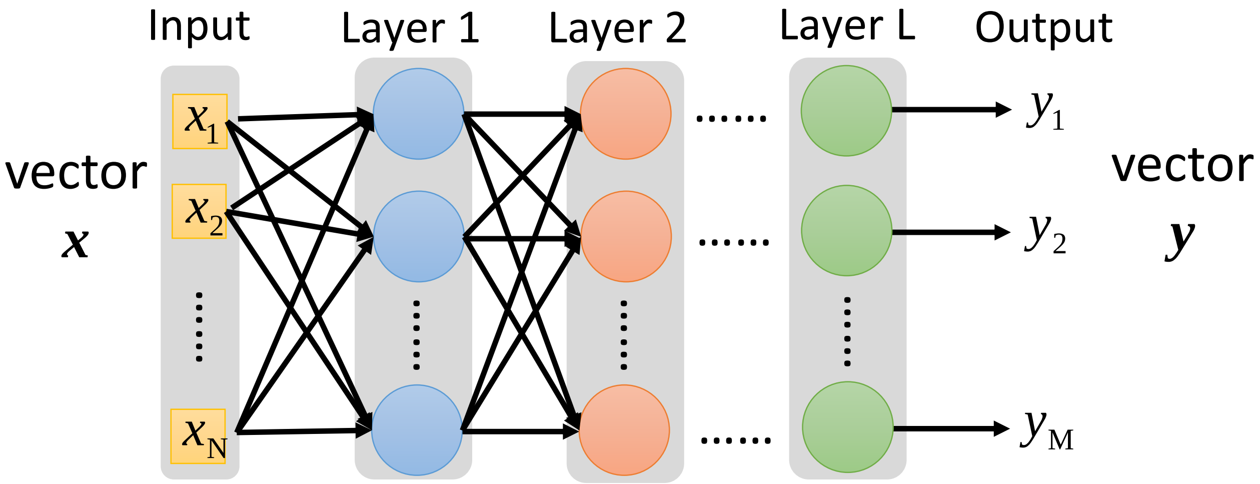


A layer of neurons can handle multiple possible output, and the result depends on the max one

Deep Neural Networks (DNN)

- Fully connected feedforward network

$$f : R^N \rightarrow R^M$$

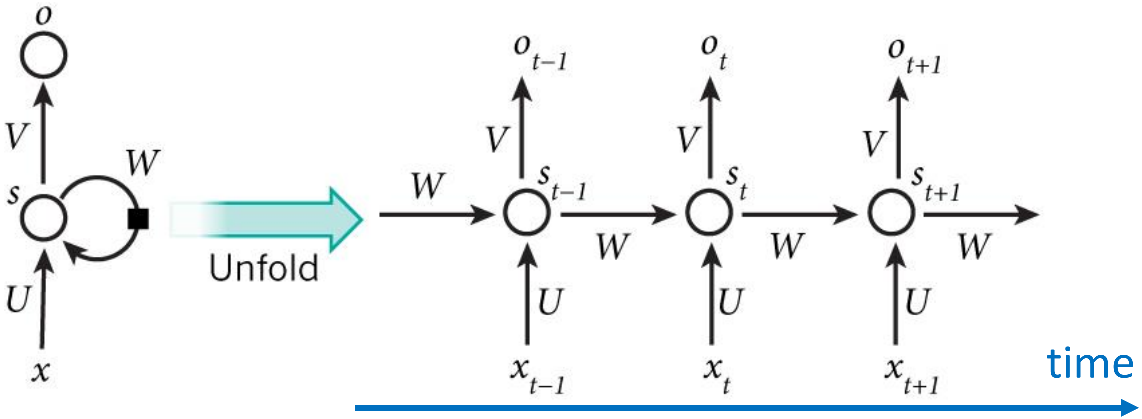


Deep NN: multiple hidden layers

Recurrent Neural Network (RNN)

$$s_t = \sigma(W s_{t-1} + U x_t) \qquad \sigma(\cdot)\text{: tanh, ReLU}$$

$$o_t = \mathrm{softmax}(V s_t)$$

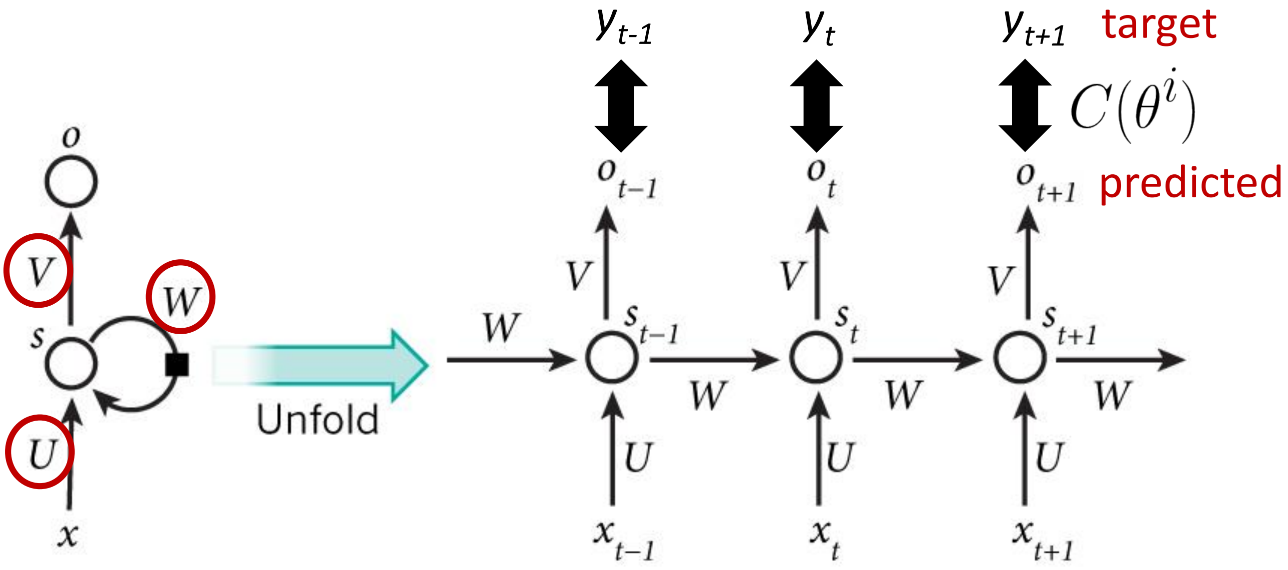


RNN can learn accumulated sequential information (time-series)

http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/

Model Training

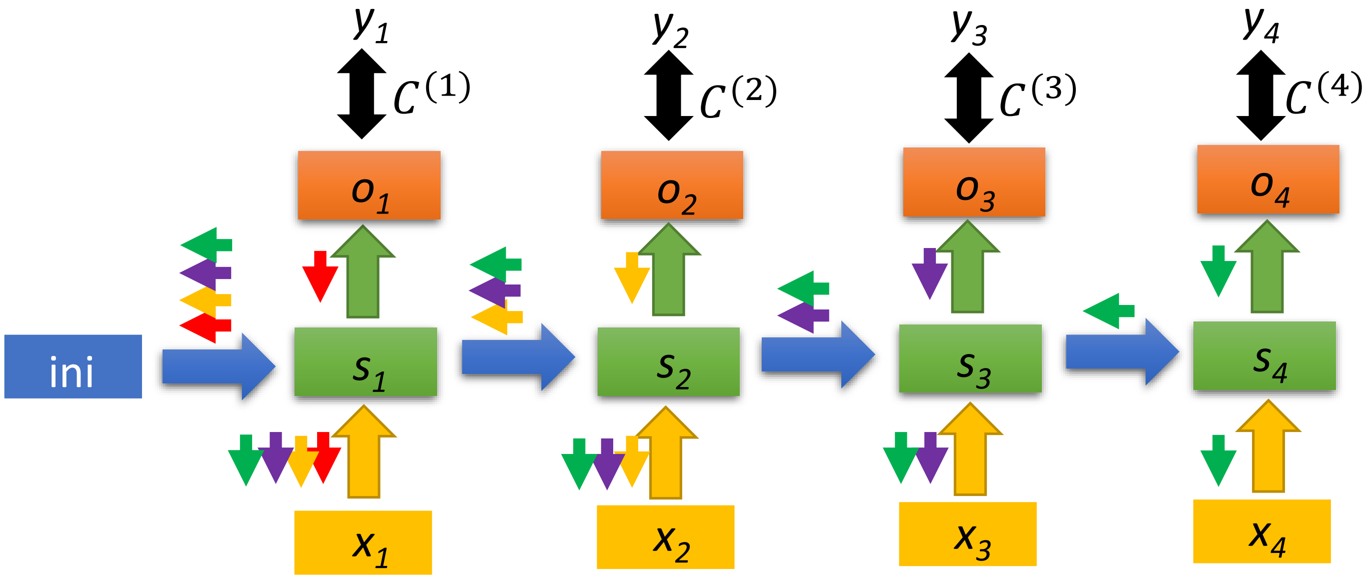
- All model parameters $\theta = \{U, V, W\}$ can be updated by SGD



BPTT

Forward Pass: Compute s_1, s_2, s_3, s_4 ……

Backward Pass: For $C^{(4)}$ For $C^{(3)}$ For $C^{(2)}$ For $C^{(1)}$



The model is trained by comparing the correct sequence tags and the predicted ones

Deep Learning Approach

Data: dialogue utterances annotated with semantic frames (user intents & slots)

Model: deep learning model (classification/tagging)
e.g. recurrent neural networks (RNN)

Prediction: user intents, slots and their values

Classification Model

As an **utterance classification** task

- Given a collection of utterances u_i with labels c_i, $D = \{(u_1, c_1), \ldots, (u_n, c_n)\}$ where $c_i \in C$, train a model to estimate labels for new utterances u_k.

- Input: each utterance u_i is represented as a feature vector f_i
- Output: a domain/intent label c_i for each input utterance

How to represent a sentence using a feature vector

Sequence Tagging Model

As a sequence tagging task

- Given a collection tagged word sequences, $S=\{((w_{1,1},w_{1,2},\ldots, w_{1,n1}), (t_{1,1},t_{1,2},\ldots,t_{1,n1})), ((w_{2,1},w_{2,2},\ldots,w_{2,n2}), (t_{2,1},t_{2,2},\ldots,t_{2,n2})) \ldots\}$ where $t_i \in M$, the goal is to estimate tags for a new word sequence.

- Input: each word $w_{i,j}$ is represented as a feature vector $f_{i,j}$
- Output: a slot label t_i for each word in the utterance

How to represent a word using a feature vector

Word Representation

- Atomic symbols: ***one-hot*** representation

car $[0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ \ldots\ 0]$

car

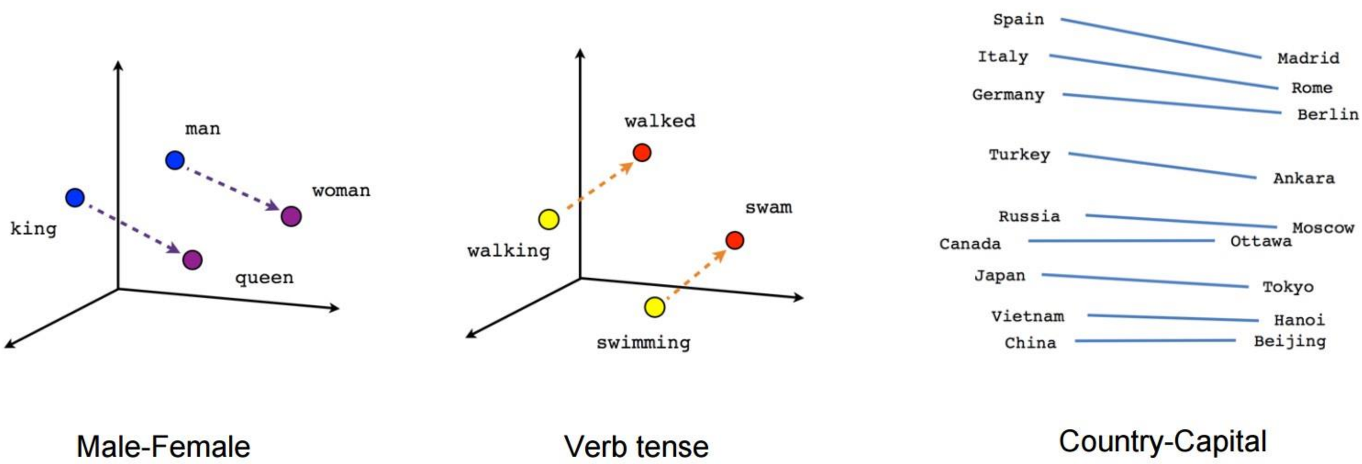
> Issues: difficult to compute the similarity (i.e. comparing “car” and “motorcycle”)

$[0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ \ldots\ 0]$ AND $[0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ \ldots\ 0] = 0$

car motorcycle

Word Representation

- Neighbor-based: low-dimensional dense word embedding



Idea: words with similar meanings often have similar neighbors

Chinese Input Unit of Representation

- Character
  - Feed each char to each time step
- Word
  - Word segmentation required

你知道美女與野獸電影的評價如何嗎?

⬇

你/知道/美女與野獸/電影/的/評價/如何/嗎

Can two types of information fuse together for better performance?

LU – Domain/Intent Classification

As an **utterance classification** task

- Given a collection of utterances u_i with labels c_i, $D = \{(u_1, c_1), \ldots, (u_n, c_n)\}$ where $c_i \in C$, train a model to estimate labels for new utterances u_k.

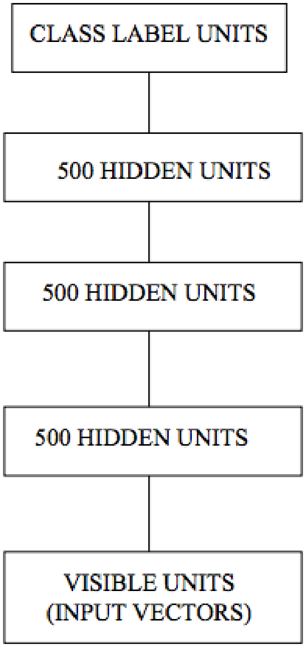
find me a cheap taiwanese restaurant in oakland

Domain	Intent
Movies	find_movie, buy_tickets
Restaurants	find_restaurant, find_price, book_table
Music	find_lyrics, find_singer
Sports	…
…	

Deep Neural Networks for Domain/Intent Classification – I (Sarikaya et al, 2011)

http://ieeexplore.ieee.org/abstract/document/5947649/

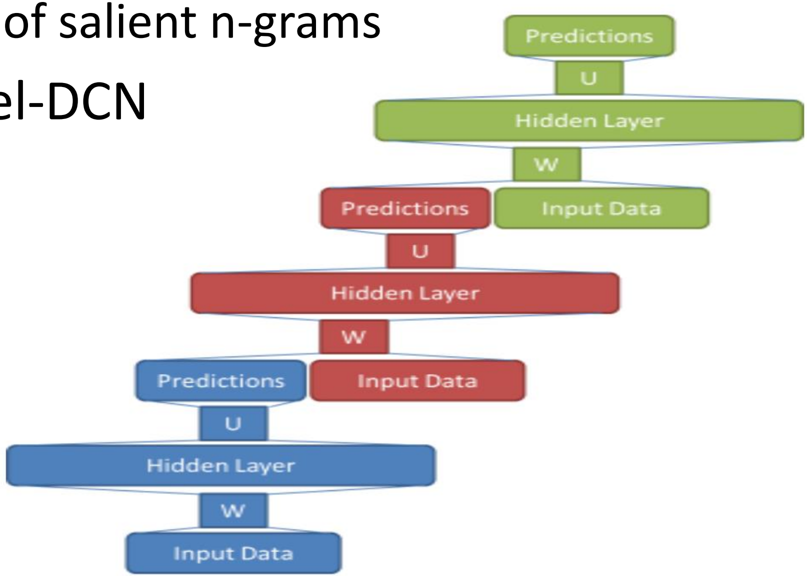
- Deep belief nets (DBN)
  - Unsupervised training of weights
  - Fine-tuning by back-propagation
  - Compared to MaxEnt, SVM, and boosting



Deep Neural Networks for Domain/Intent Classification – II (Tur et al., 2012; Deng et al., 2012)

http://ieeexplore.ieee.org/abstract/document/6289054/; http://ieeexplore.ieee.org/abstract/document/6424224/

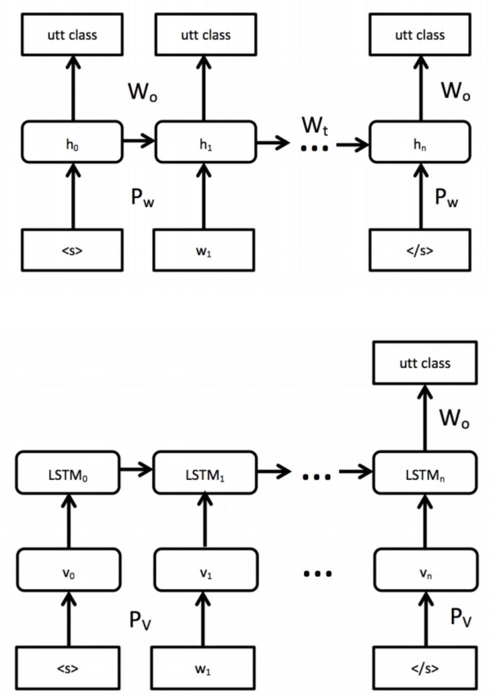
- Deep convex networks (DCN)
  - Simple classifiers are stacked to learn complex functions
  - Feature selection of salient n-grams
- Extension to kernel-DCN



Deep Neural Networks for Domain/Intent Classification – III (Ravuri and Stolcke, 2015)

https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/RNNLM_addressee.pdf

- RNN and LSTMs for utterance classification
- Word hashing to deal with large number of singletons
  - Kat: #Ka, Kat, at#
  - Each character n-gram is associated with a bit in the input encoding



LU – Slot Filling

As a **sequence tagging** task

- Given a collection tagged word sequences, $S=\{((w_{1,1},w_{1,2},\ldots, w_{1,n1}), (t_{1,1},t_{1,2},\ldots,t_{1,n1})), ((w_{2,1},w_{2,2},\ldots,w_{2,n2}), (t_{2,1},t_{2,2},\ldots,t_{2,n2})) \ldots\}$ where $t_i \in M$, the goal is to estimate tags for a new word sequence.

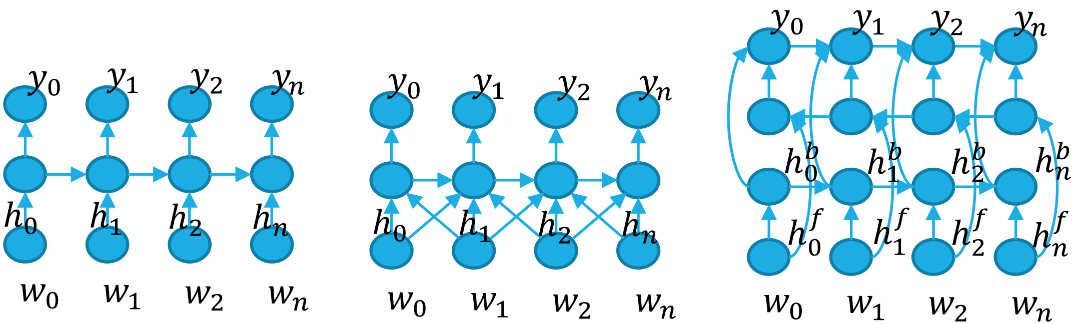
flights from Boston to New York today

	flights	from	Boston	to	New	York	today
Entity Tag	O	O	B-city	O	B-city	I-city	O
Slot Tag	O	O	B-dept	O	B-arrival	I-arrival	B-date

Recurrent Neural Nets for Slot Tagging – I
(Yao et al, 2013; Mesnil et al, 2015)

http://131.107.65.14/en-us/um/people/gzweig/Pubs/Interspeech2013RNNLU.pdf; http://dl.acm.org/citation.cfm?id=2876380

- Variations:
  a. RNNs with LSTM cells
  b. Input, sliding window of n-grams
  c. Bi-directional LSTMs



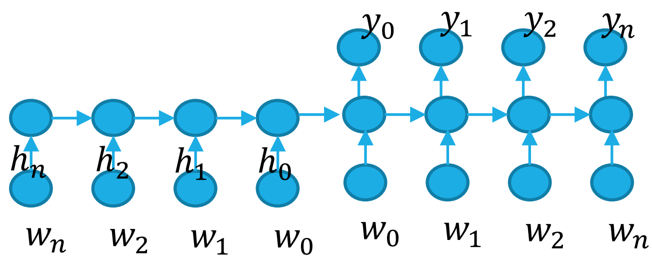
(a) LSTM (b) LSTM-LA (c) bLSTM

Recurrent Neural Nets for Slot Tagging – II

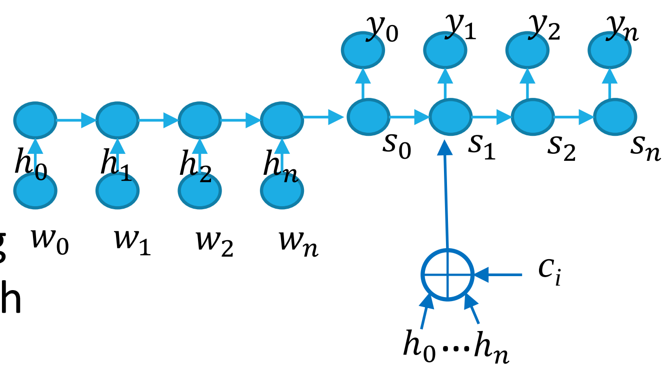
(Kurata et al., 2016; Simonnet et al., 2015)

http://www.aclweb.org/anthology/D16-1223

- Encoder-decoder networks
  - Leverages sentence level information



- Attention-based encoder-decoder
  - Use of attention (as in MT) in the encoder-decoder network
  - Attention is estimated using a feed-forward network with input: h_t and s_t at time t



Joint Semantic Frame Parsing

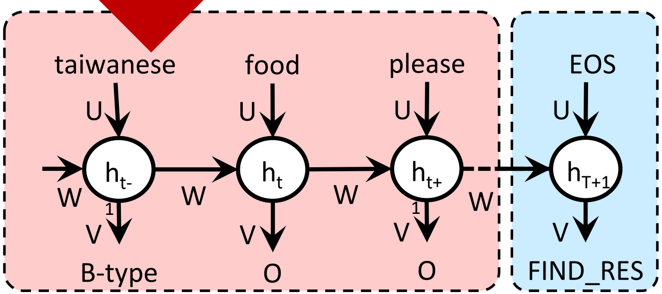
https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/IS16_MultiJoint.pdf; https://arxiv.org/abs/1609.01454

Sequence-based (Hakkani-Tur et al., 2016)

- Slot filling and intent prediction in the same output sequence

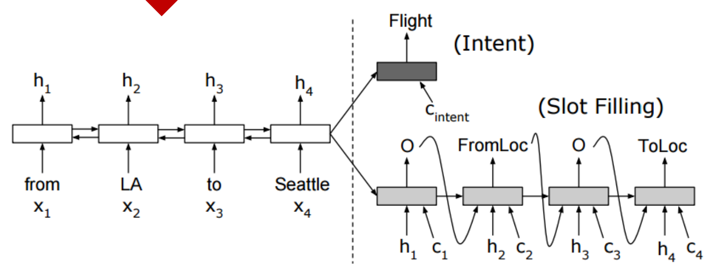
Parallel (Liu and Lane, 2016)

- Intent prediction and slot filling are performed in two branches



Slot Filling

Intent Prediction



Milestone 1 – Language Understanding

3) Collect and annotate data
4) Use machine learning method to train your system
   - Conventional
     - SVM for domain/intent classification
     - CRF for slot filling
   - Deep learning
     - LSTM for domain/intent classification and slot filling
5) Test your system performance

Concluding Remarks