Intelligent Conversational Bot

Language Generation

May 2nd, 2017

YUN-NUNG (VIVIAN) CHEN

WWW.CSIE.NTU.EDU.TW/~YVCHEN/S105-ICB

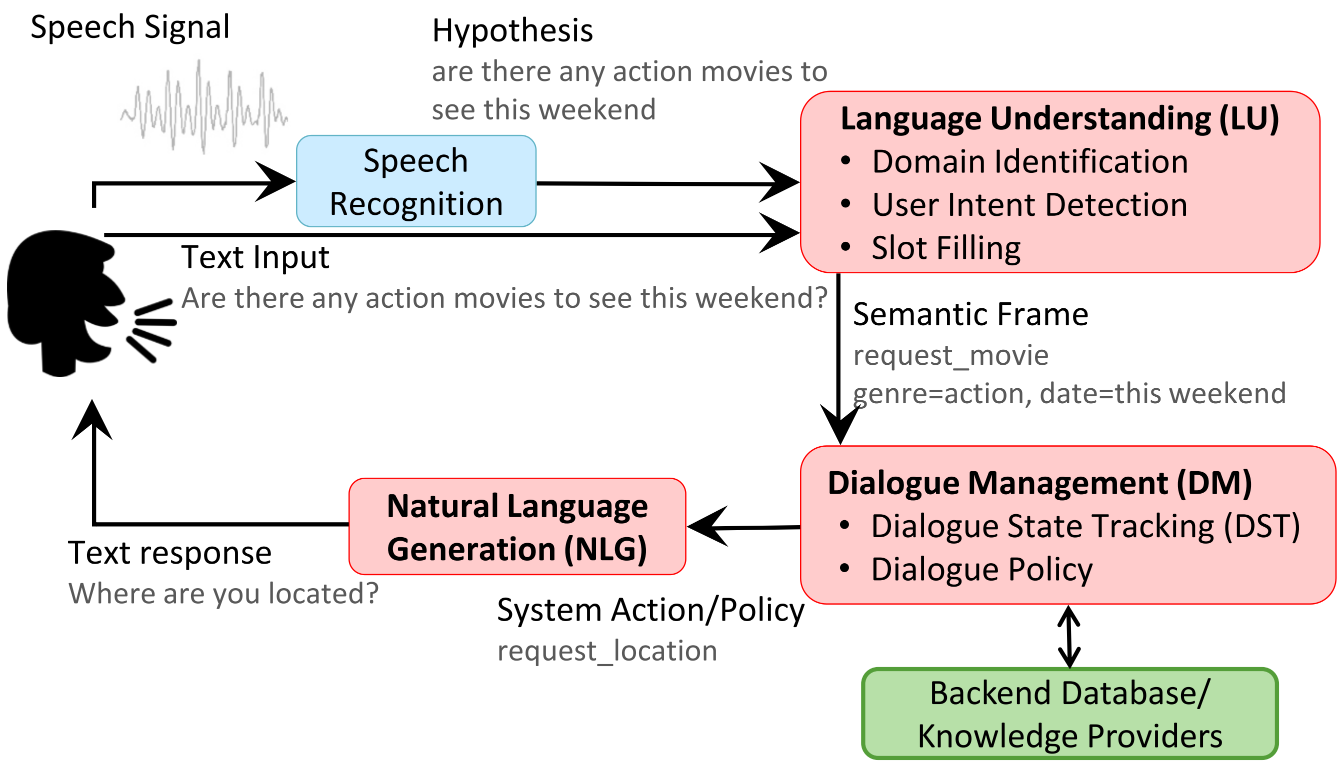
國立臺灣大學
National Taiwan University

Slides credit from Shawn

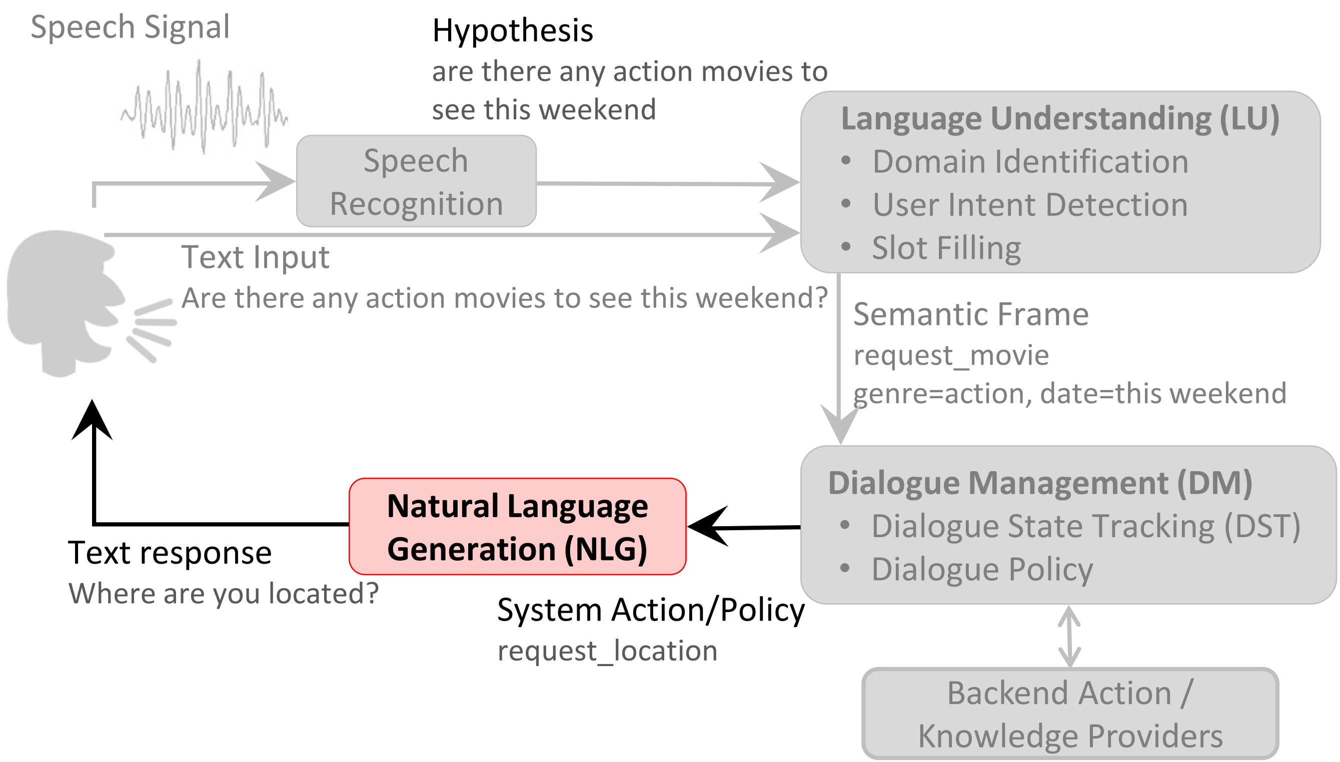
Review

Task-Oriented Dialogue System (Young, 2000)

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



Task-Oriented Dialogue System (Young, 2000)



Language Modeling

Language Modeling

- Goal: estimate the probability of a word sequence

$$P(w_1, \cdots, w_m)$$

- Example task: determinate whether a sequence is grammatical or makes more sense

recognize speech
or
wreck a nice beach

If P(recognize speech)
> P(wreck a nice beach)

Output =
"recognize speech"

N-Gram Language Modeling

- Goal: estimate the probability of a word sequence

$$P(w_1, \cdots, w_m)$$

- N-gram language model
  - Probability is conditioned on a window of $(n\text{-}1)$ previous words

$$P(w_1, \cdots, w_m) = \prod_{i=1}^{m} P(w_i \mid w_1, \cdots, w_{i-1}) \approx \prod_{i=1}^{m} P(w_i \mid w_{i-(n-1)}, \cdots, w_{i-1})$$

  - Estimate the probability based on the training data

$$P(\text{beach}|\text{nice}) = \frac{C(nice\ beach)}{C(nice)}$$

Count of "nice beach" in the training data

Count of "nice" in the training data

Issue: some sequences may not appear in the training data

N-Gram Language Modeling

- Training data:
  - The dog ran ……
  - The cat jumped ……

$$P(\text{ jumped} \mid \text{dog }) = \cancel{0}\ \ 0.0001$$

$$P(\text{ ran} \mid \text{cat }) = \cancel{0}\ \ 0.0001$$

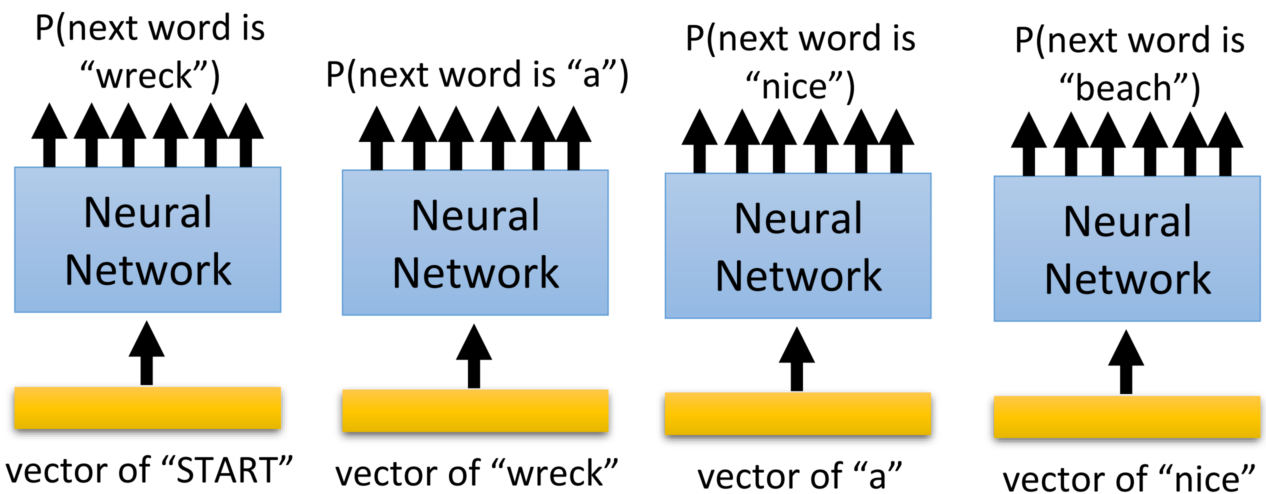
give some small probability
→ smoothing

- The probability is not accurate.
- The phenomenon happens because we cannot collect all the possible text in the world as training data.

Neural Language Modeling

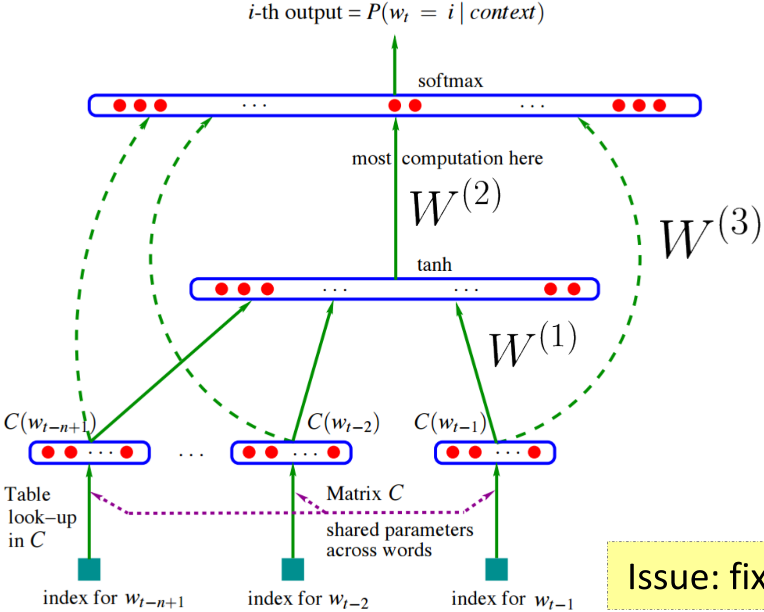
- Idea: estimate $P(w_i \mid w_{i-(n-1)}, \cdots, w_{i-1})$ not from count, but from the NN prediction

P("wreck a nice beach") = P(wreck|START)P(a|wreck)P(nice|a)P(beach|nice)

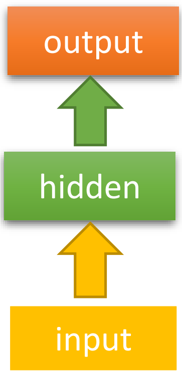


Neural Language Modeling

$$\hat{y} = \text{softmax}(W^{(2)}\sigma(W^{(1)}x + b^{(1)}) + W^{(3)}x + b^{(3)})$$



Probability distribution of the next word

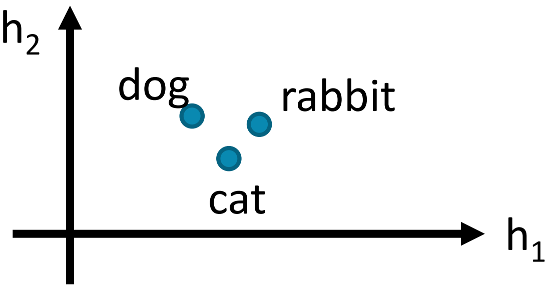


context vector

Issue: fixed context window for conditioning

Neural Language Modeling

- The input layer (or hidden layer) of the related words are close

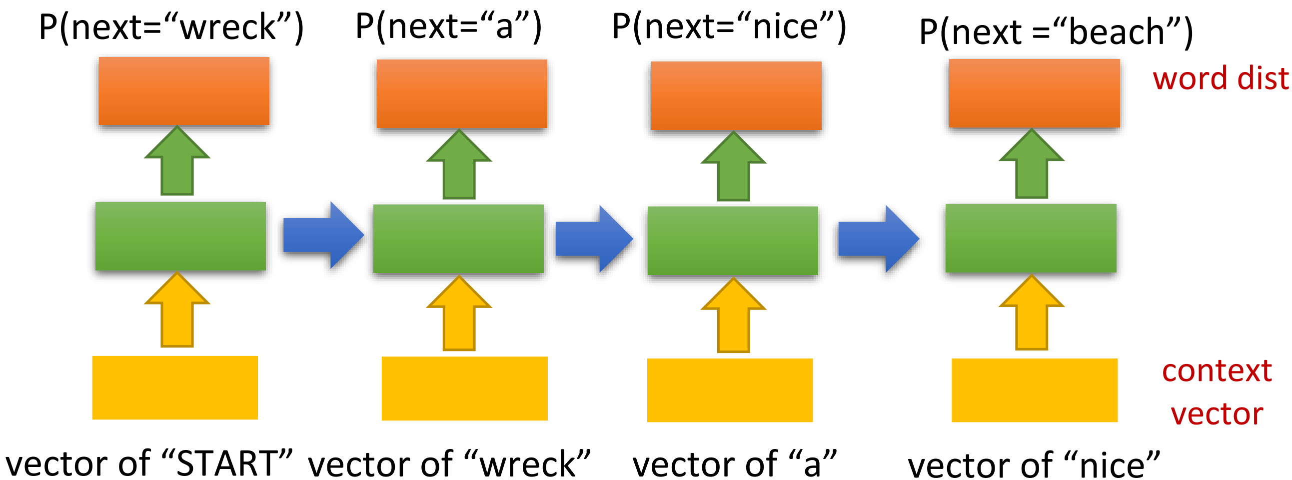


  - If P(jump|dog) is large, P(jump|cat) increase accordingly (even there is not "… cat jump …" in the data)

Smoothing is automatically done

RNNLM

- Idea: condition the neural network on all previous words and tie the weights at each time step
- Assumption: temporal information matters



Idea: pass the information from the previous hidden layer to leverage all contexts

Natural Language Generation

Traditional Approaches

Natural Language Generation (NLG)

- Mapping dialogue acts into natural language

inform(name=Seven_Days, foodtype=Chinese)

Seven Days is a nice Chinese restaurant

Template-Based NLG

- Define a set of rules to map frames to NL

Semantic Frame	Natural Language
confirm()	"Please tell me more about the product your are looking for."
confirm(area=$V)	"Do you want somewhere in the $V?"
confirm(food=$V)	"Do you want a $V restaurant?"
confirm(food=$V,area=$W)	"Do you want a $V restaurant in the $W."

Pros: simple, error-free, easy to control
Cons: time-consuming, rigid, poor scalability

Class-Based LM NLG (Oh and Rudnicky, 2000)

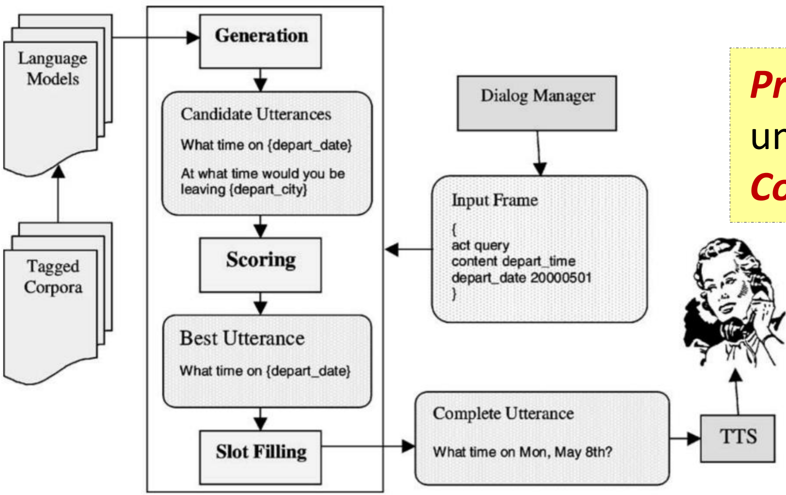
http://dl.acm.org/citation.cfm?id=1117568

- Class-based language modeling

$$P(X \mid c) = \sum_t \log p(x_t \mid x_0, x_1, \cdots, x_{t-1}, c)$$

- NLG by decoding $X^* = \arg\max_X P(X \mid c)$

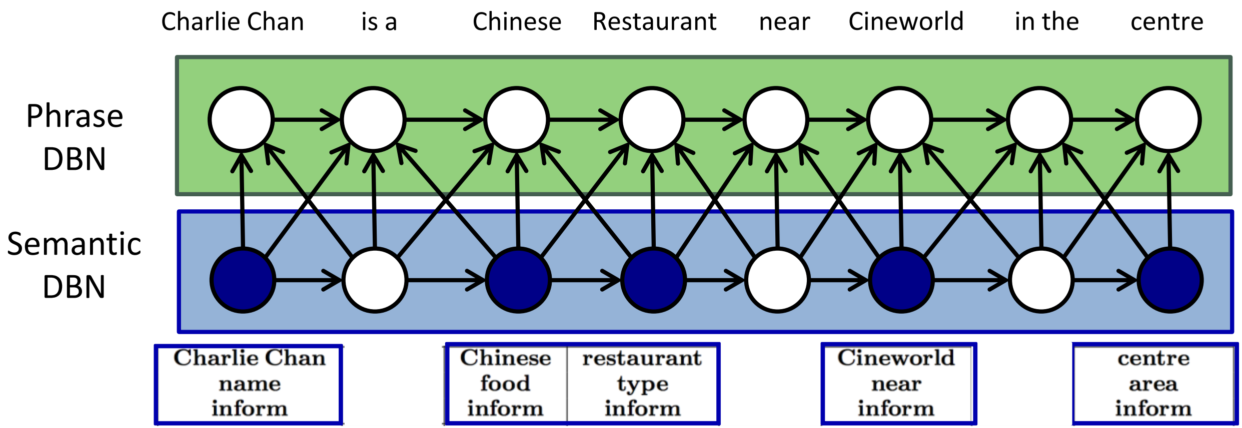
Classes:
inform_area
inform_address
…
request_area
request_postcode



Pros: easy to implement/ understand, simple rules
Cons: computationally inefficient

Phrase-Based NLG (Mairesse et al, 2010)

http://dl.acm.org/citation.cfm?id=1858838



Inform(name=Charlie Chan, food=Chinese, type= restaurant, near=Cineworld, area=centre)

realization phrase semantic stack

r_t	s_t	h_t	l_t
<s>	START	START	START
The Rice Boat	inform(name(X))	X	inform(name)
is a	inform	inform	EMPTY
restaurant	inform(type(restaurant))	restaurant	inform(type)
in the	inform(area)	area	inform
riverside	inform(area(riverside))	riverside	inform(area)
area	inform(area)	area	inform
that	inform	inform	EMPTY
serves	inform(food)	food	inform
French	inform(food(French))	French	inform(food)
food	inform(food)	food	inform
</s>	END	END	END

Pros: efficient, good performance

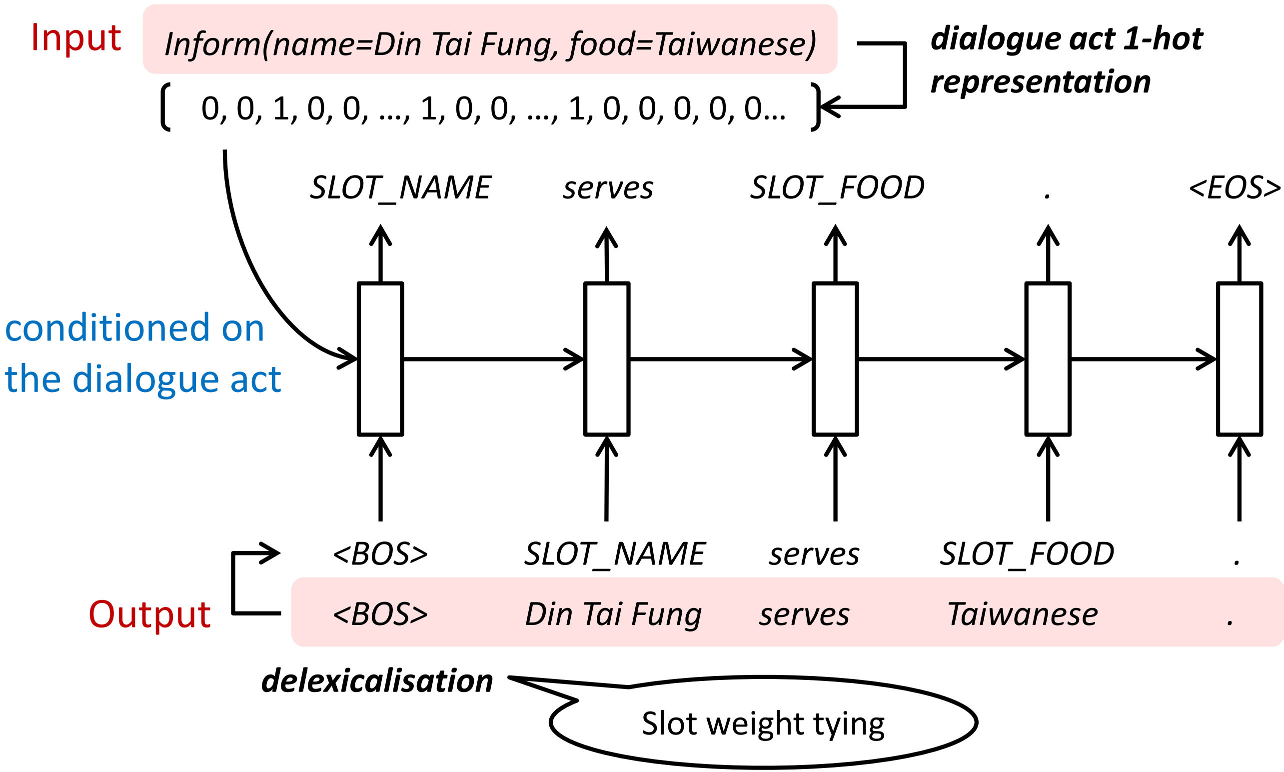
Cons: require semantic alignments

Natural Language Generation

Deep Learning Approaches

RNN-Based LM NLG (Wen et al., 2015)

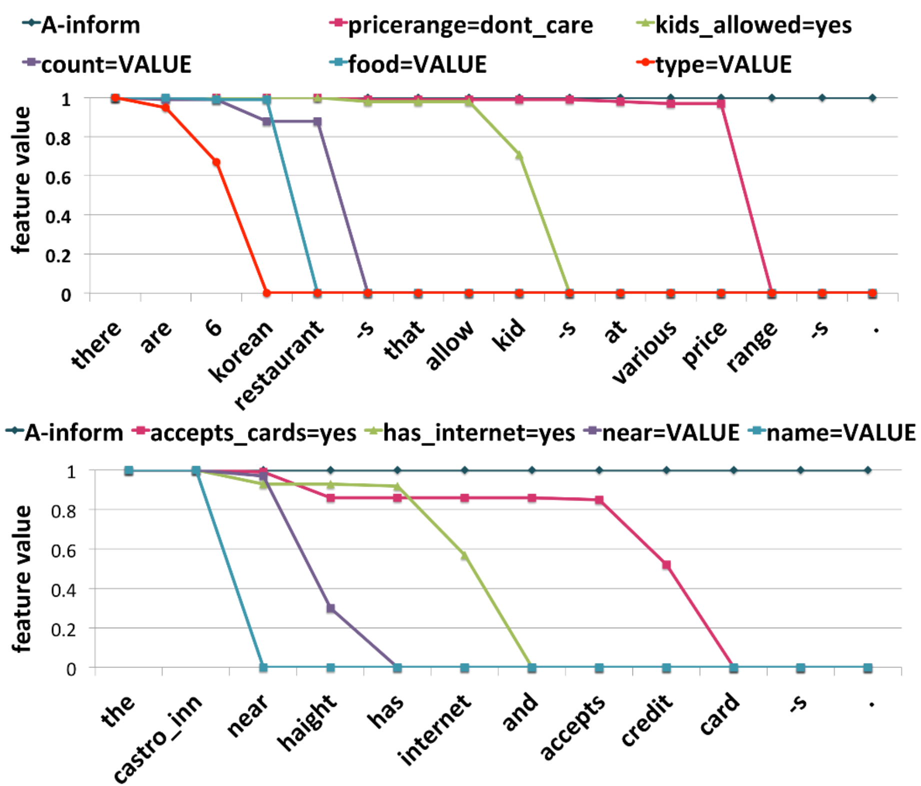
http://www.anthology.aclweb.org/W/W15/W15-46.pdf#page=295



Handling Semantic Repetition

- Issue: semantic repetition
  - Din Tai Fung is a great Taiwanese restaurant that serves Taiwanese.
  - Din Tai Fung is a child friendly restaurant, and also allows kids.
- Deficiency in either model or decoding (or both)
- Mitigation
  - Post-processing rules (Oh & Rudnicky, 2000)
  - Gating mechanism (Wen et al., 2015)
  - Attention (Mei et al., 2016; Wen et al., 2015)

Visualization



Semantic Conditioned LSTM (Wen et al., 2015)

http://www.aclweb.org/anthology/D/D15/D15-1199.pdf

- Original LSTM cell

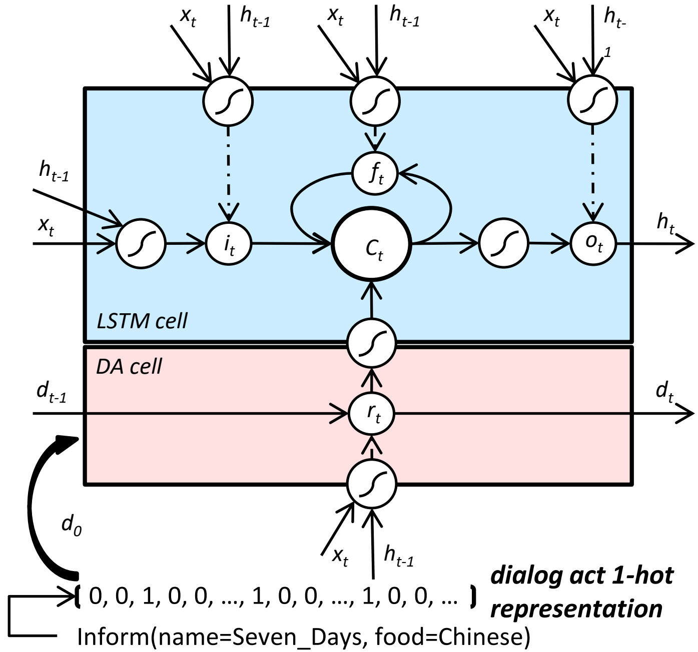
$$\mathbf{i}_t = \sigma(\mathbf{W}_{wi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1})$$
$$\mathbf{f}_t = \sigma(\mathbf{W}_{wf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1})$$
$$\mathbf{o}_t = \sigma(\mathbf{W}_{wo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1})$$
$$\hat{\mathbf{c}}_t = \tanh(\mathbf{W}_{wc}\mathbf{x}_t + \mathbf{W}_{hc}\mathbf{h}_{t-1})$$
$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \hat{\mathbf{c}}_t$$
$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t)$$

- Dialogue act (DA) cell

$$\mathbf{r}_t = \sigma(\mathbf{W}_{wr}\mathbf{x}_t + \mathbf{W}_{hr}\mathbf{h}_{t-1})$$
$$\mathbf{d}_t = \mathbf{r}_t \odot \mathbf{d}_{t-1}$$

- Modify Ct

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \hat{\mathbf{c}}_t + \tanh(\mathbf{W}_{dc}\mathbf{d}_t)$$



Idea: using gate mechanism to control the generated semantics (dialogue act/slots)

Attentive Encoder-Decoder for NLG

- Slot & value embedding

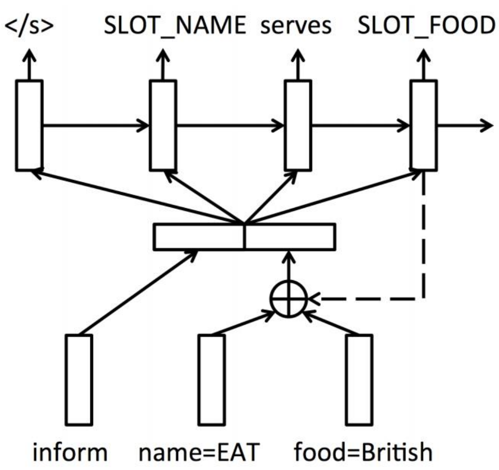
$$\mathbf{z}_i = \mathbf{s}_i + \mathbf{v}_i$$

- Attentive meaning representation

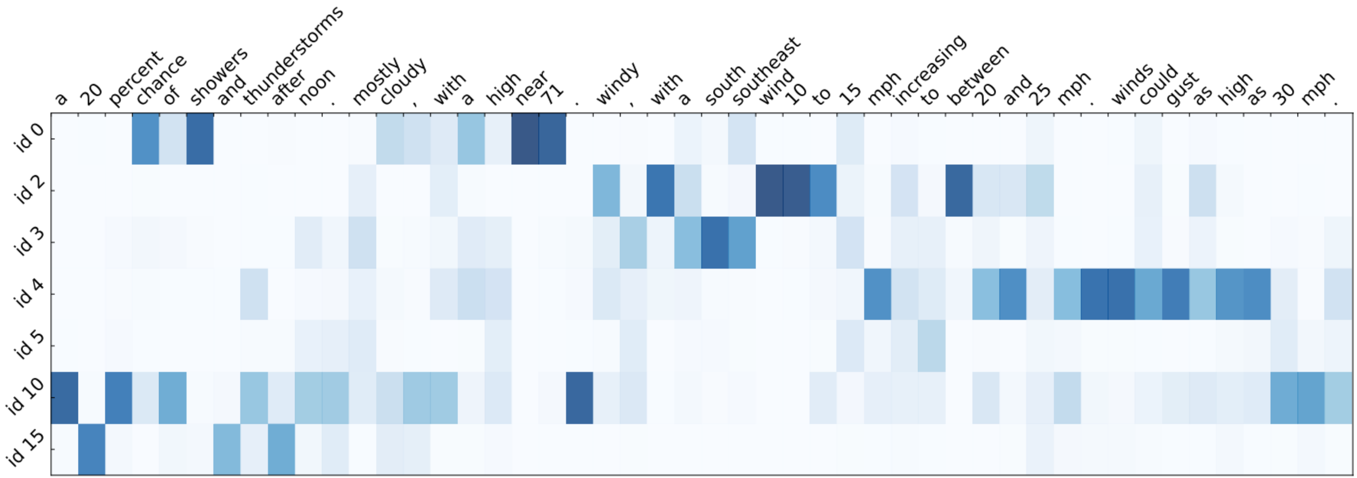
$$e_{ti} = \mathbf{v}^{\mathrm{T}} \tanh(\mathbf{W}_{hm}\mathbf{h}_{t-1} + \mathbf{W}_{zm}\mathbf{z}_i)$$

$$\alpha_{ti} = \mathrm{softmax}(e_{ti})$$

$$\mathbf{d}_t = \mathbf{a} \oplus \sum_i \boldsymbol{\alpha}_{ti}\mathbf{z}_i$$



Attention Heat Map

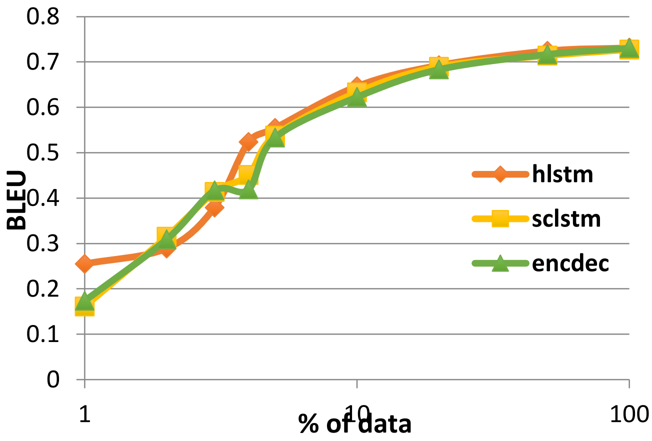


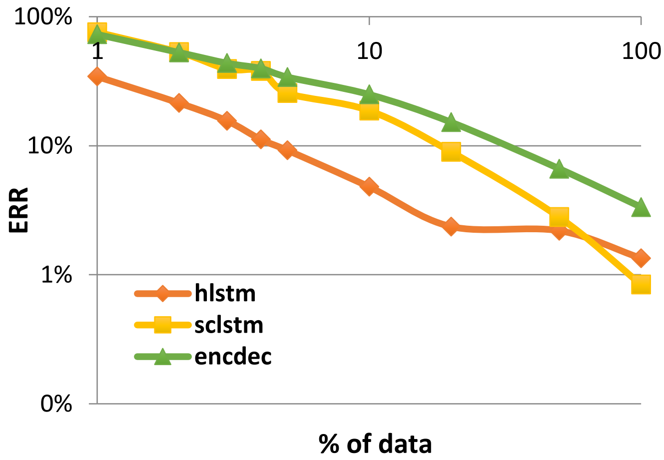
Record details:

id-0: temperature(time=06-21, min=52, mean=63, max=71); id-2: windSpeed(time=06-21, min=8, mean=17, max=23);
id-3: windDir(time=06-21, mode=SSE); id-4: gust(time=06-21, min=0, mean=10, max=30);
id-5: skyCover(time=6-21, mode=50-75); id-10: precipChance(time=06-21, min=19, mean=32, max=73);
id-15: thunderChance(time=13-21, mode=SChc)

Figure 3: An example generation for a set of records from WEATHERGOV.

Model Comparison



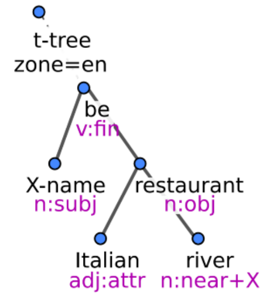


Structural NLG (Dušek and Jurčíček, 2016)

https://www.aclweb.org/anthology/P/P16/P16-2.pdf#page=79

- Goal: NLG based on the syntax tree
  - Encode trees as sequences
  - Seq2Seq model for generation

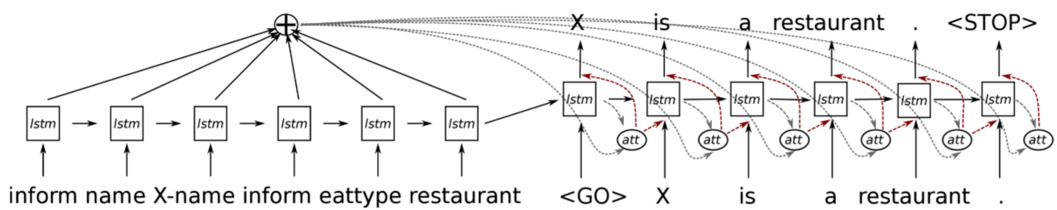
inform(name=X-name,type=placetoeat,eattype=restaurant,
area=riverside,food=Italian)



(<root> <root> ((X-name n:subj) be v:fin ((Italian adj:attr) restaurant n:obj (river n:near+X))))

X-name n:subj be v:fin Italian adj:attr restaurant n:obj river n:near+X

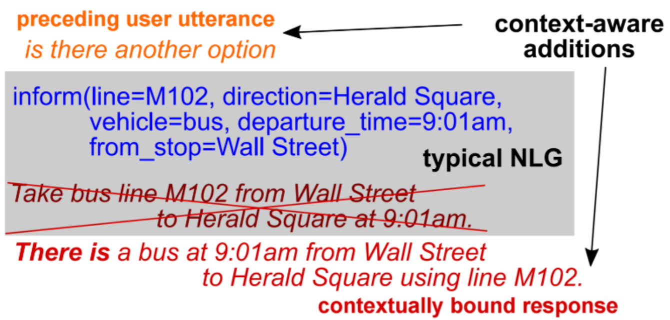
X is an Italian restaurant near the river.

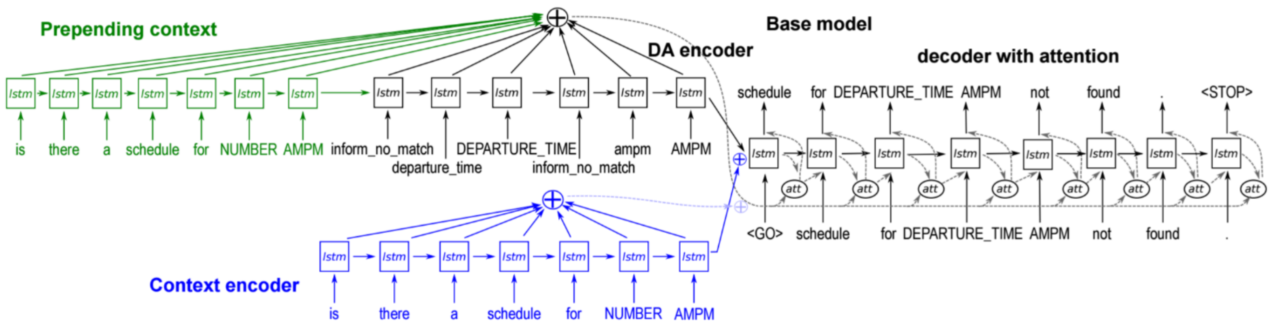


Contextual NLG (Dušek and Jurčíček, 2016)

https://www.aclweb.org/anthology/W/W16/W16-36.pdf#page=203

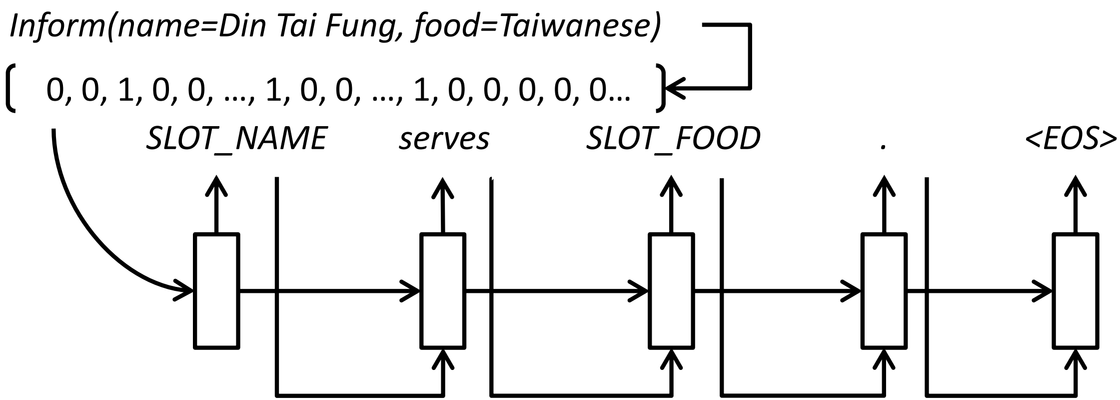
- Goal: adapting users' way of speaking, providing context-aware responses
  - Context encoder
  - Seq2Seq model





Decoder Sampling Strategy

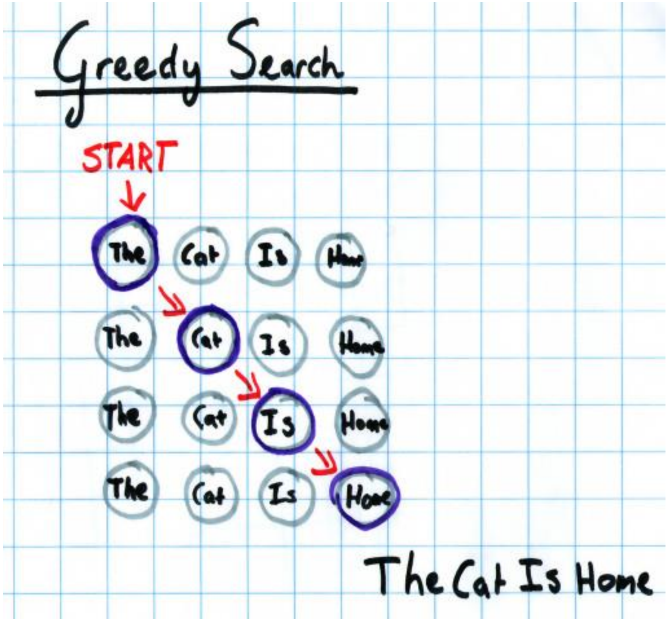
- Decoding procedure



  - Greedy search
  - Beam search
  - Random search

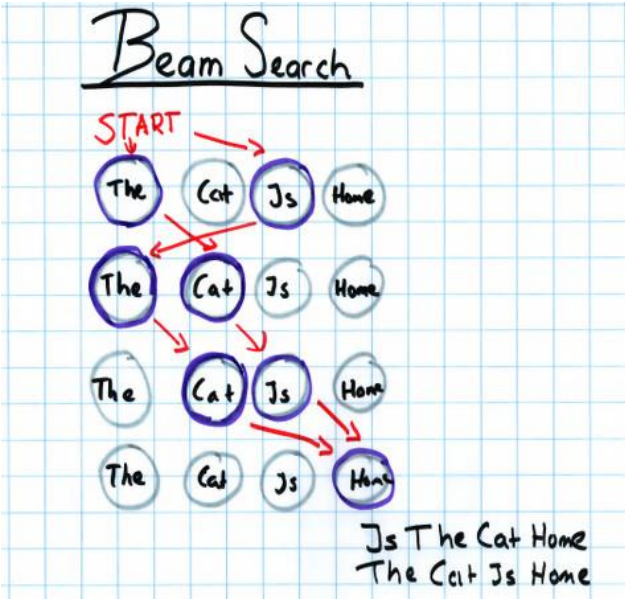
Greedy Search

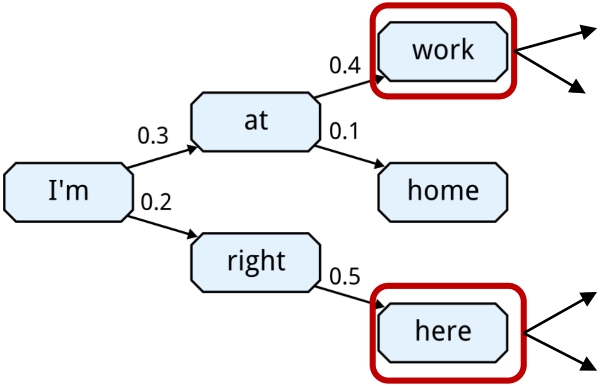
- Select the next word with the *highest* probability



Beam Search

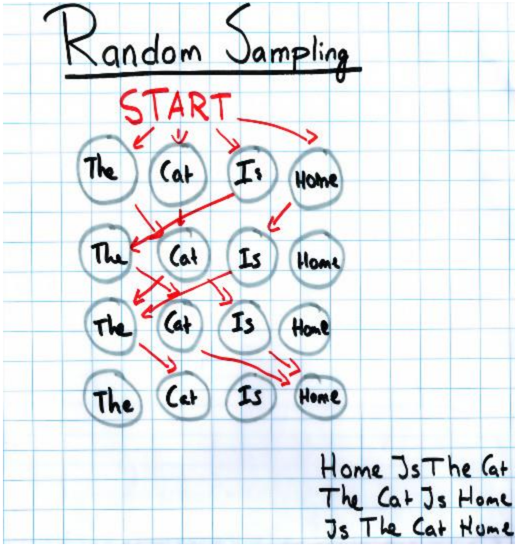
- Select the next *k-best* words and keep a beam with width=k for following decoding





Random Search

- Randomly select the next word
  - Higher diversity
  - Can follow a probability distribution



Chit-Chat Generation

Chit-Chat Bot

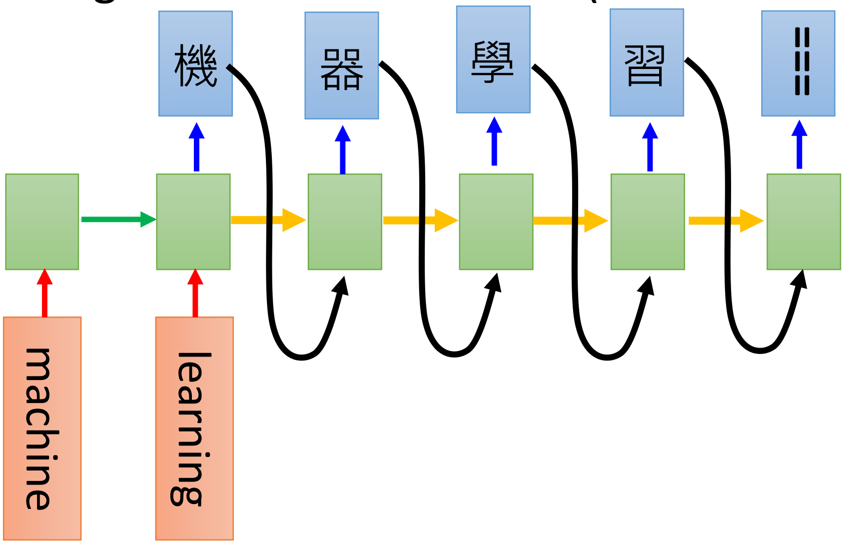
- Neural conversational model
  - Non task-oriented

Human: *what do you think about tesla ?*
Machine: *he 's a good conductor .*
Human: *what do you think about bill gates ?*
Machine: *he 's a good man .*
Human: *What do you think about messi ?*
Machine: *he 's a great player .*
Human: *what do you think about cleopatra ?*
Machine: *oh , she 's very regal .*
Human: *what do you think about england during the reign of elizabeth ?*
Machine: *it was a great place .*

Human: *what is the purpose of life ?*
Machine: *to serve the greater good .*
Human: *what is the purpose of living ?*
Machine: *to live forever .*
Human: *what is the purpose of existence ?*
Machine: *to find out what happens when we get to the planet earth .*
Human: *where are you now ?*
Machine: *i 'm in the middle of nowhere .*
Human: *what is the purpose of dying ?*
Machine: *to have a life .*
Human: *what is the purpose of being intelligent ?*
Machine: *to find out what it is .*
Human: *what is the purpose of emotions ?*
Machine: *i don 't know .*

Many-to-Many

- Both input and output are both sequences → **Sequence-to-sequence learning**
  - E.g. *Machine Translation* (machine learning→機器學習)

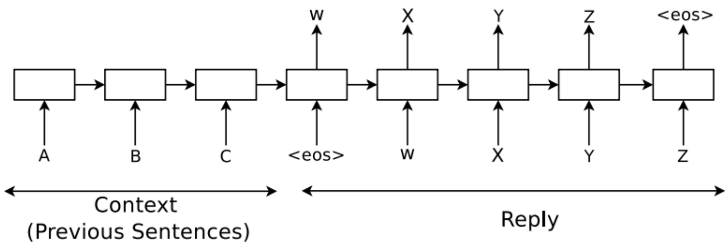


[Ilya Sutskever, NIPS'14][Dzmitry Bahdanau, arXiv'15]

A Neural Conversational Model

- Seq2Seq

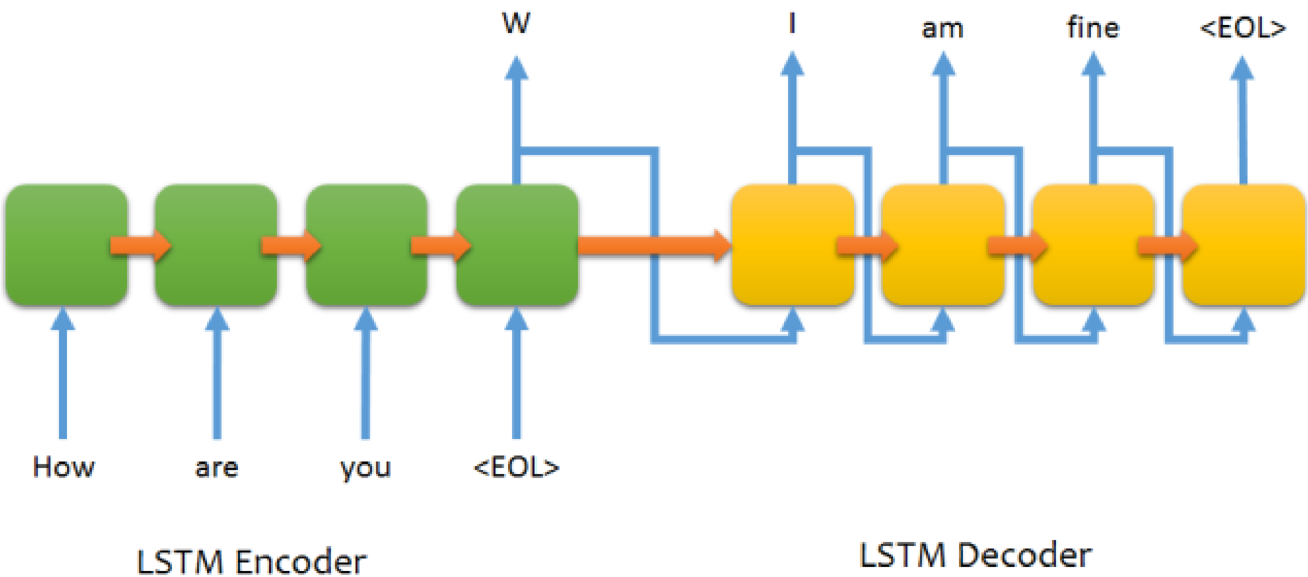
[*Vinyals and Le, 2015*]



Human: *what do you think about tesla ?*
Machine: *he 's a good conductor .*
Human: *what do you think about bill gates ?*
Machine: *he 's a good man .*
Human: *What do you think about messi ?*
Machine: *he 's a great player .*
Human: *what do you think about cleopatra ?*
Machine: *oh , she 's very regal .*
Human: *what do you think about england during the reign of elizabeth ?*
Machine: *it was a great place .*

Human: *what is the purpose of life ?*
Machine: *to serve the greater good .*
Human: *what is the purpose of living ?*
Machine: *to live forever .*
Human: *what is the purpose of existence ?*
Machine: *to find out what happens when we get to the planet earth .*
Human: *where are you now ?*
Machine: *i 'm in the middle of nowhere .*
Human: *what is the purpose of dying ?*
Machine: *to have a life .*
Human: *what is the purpose of being intelligent ?*
Machine: *to find out what it is .*
Human: *what is the purpose of emotions ?*
Machine: *i don 't know .*

Chit-Chat Bot



電視影集 (~40,000 sentences)、美國總統大選辯論

Sci-Fi Short Film - SUNSPRING

https://www.youtube.com/watch?v=LY7x2Ihqj

Concluding Remarks

- The three pillars of deep learning for NLG
  - **Distributed representation** – generalization
  - **Recurrent connection** – long-term dependency
  - **Conditional RNN** – flexibility/creativity
- Useful techniques in deep learning for NLG
  - Learnable gates
  - Attention mechanism
- Generating longer/complex sentences
- Phrase dialogue as conditional generation problem
  - Conditioning on raw input sentence → chit-chat bot
  - Conditioning on both structured and unstructured sources → task-completing dialogue system