Ask Me Anything:
Dynamic Memory Networks for Natural Language Processing

조용래
dreamgonfly@gmail.com

2019.03.16
모두의 연구소 풀잎스쿨 NLP Bootcamp

Let's start with the meaning of the title

Ask Me Anything:
Dynamic Memory Networks for Natural Language Processing

What problems does it solve?

Question Answering

Text Classification

Sequence Tagging

Other NLP tasks

Ask Me Anything: Dynamic Memory Networks for Natural Language Processing

How does it solve?

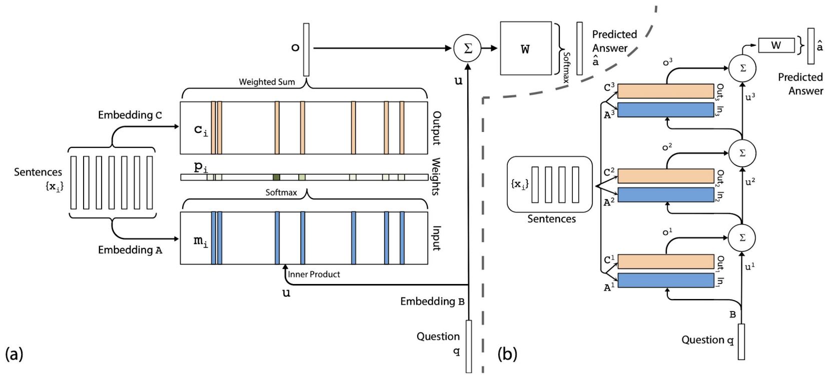
View any NLP task as question answering

Ask Me Anything: Dynamic Memory Networks for Natural Language Processing

Ask Me Anything:
Dynamic Memory Networks for Natural Language Processing

Based on

End-To-End Memory Networks



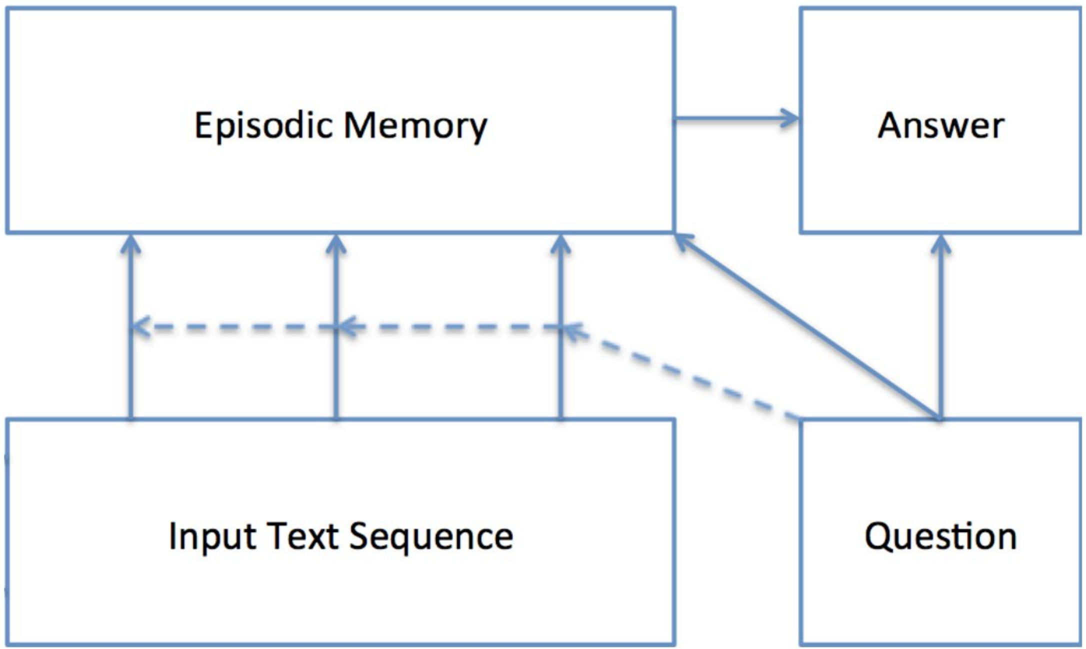
Ask Me Anything:
Dynamic Memory Networks for Natural Language Processing

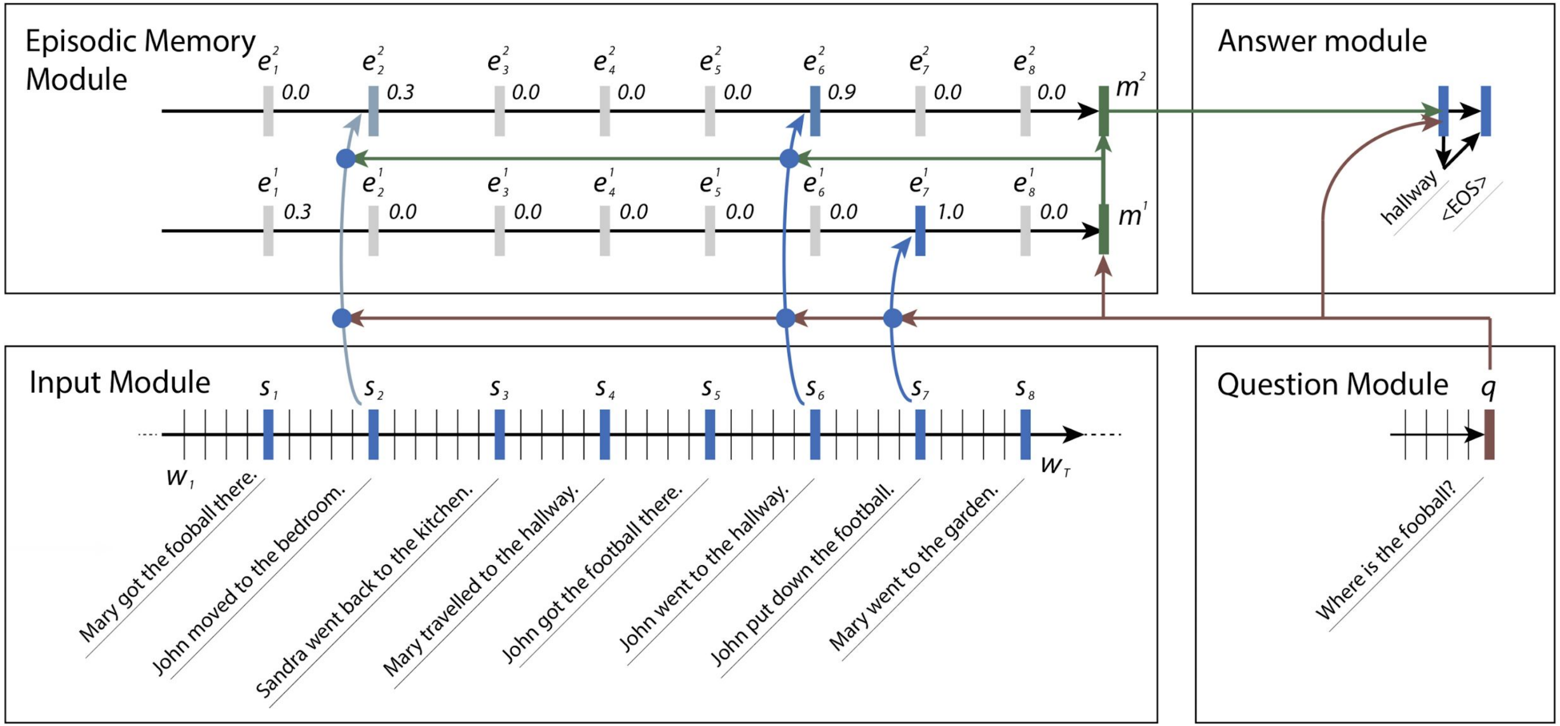
Contribution

Episodic Memory Module

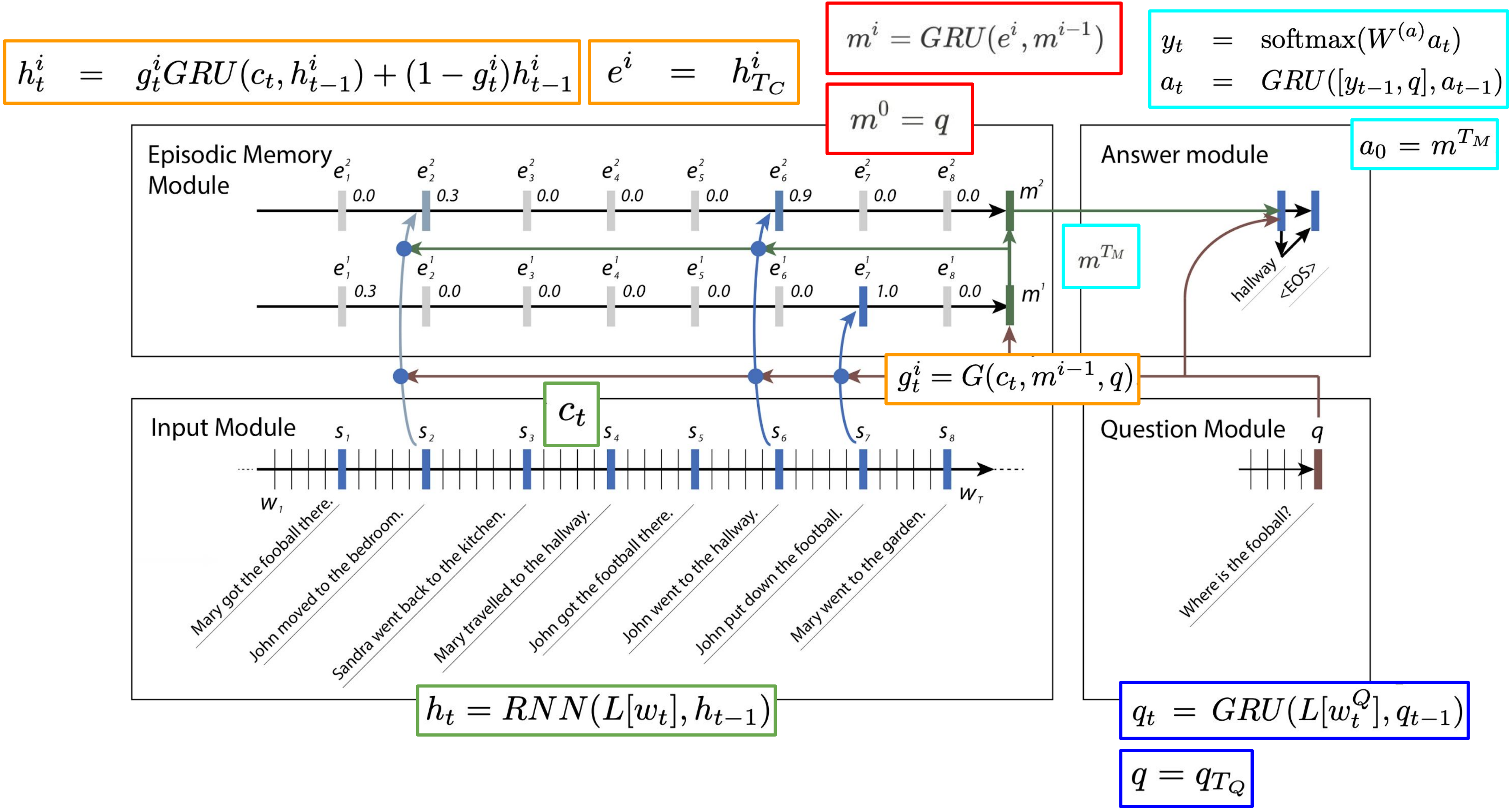
(IMHO, memory over memory)

Model Overview





Episodic Memory Module
Answer module
Input Module
Question Module
0.0
0.3
0.9
1.0
m^2
m^1
q
w_1
w_T
hallway
<EOS>
Mary got the fooball there.
John moved to the bedroom.
Sandra went back to the kitchen.
Mary travelled to the hallway.
John got the football there.
John went to the hallway.
John put down the football.
Mary went to the garden.
Where is the fooball?



$h_t^i = g_t^i GRU(c_t, h_{t-1}^i) + (1 - g_t^i) h_{t-1}^i$
$e^i = h_{T_C}^i$
$m^i = GRU(e^i, m^{i-1})$
$m^0 = q$
$y_t = \text{softmax}(W^{(a)} a_t)$
$a_t = GRU([y_{t-1}, q], a_{t-1})$
$a_0 = m^{T_M}$
Episodic Memory Module
e^2_1 0.0 e^2_2 0.3 e^2_3 0.0 e^2_4 0.0 e^2_5 0.0 e^2_6 0.9 e^2_7 0.0 e^2_8 0.0 m^2
e^1_1 0.3 e^1_2 0.0 e^1_3 0.0 e^1_4 0.0 e^1_5 0.0 e^1_6 0.0 e^1_7 1.0 e^1_8 0.0 m^1
m^{T_M}
Answer module
hallway
<EOS>
$g_t^i = G(c_t, m^{i-1}, q)$
c_t
Input Module
s_1 s_2 s_3 s_4 s_5 s_6 s_7 s_8
w_1
w_T
Mary got the fooball there.
John moved to the bedroom.
Sandra went back to the kitchen.
Mary travelled to the hallway.
John got the football there.
John went to the hallway.
John put down the football.
Mary went to the garden.
Question Module
q
Where is the fooball?
$h_t = RNN(L[w_t], h_{t-1})$
$q_t = GRU(L[w_t^Q], q_{t-1})$
$q = q_{T_Q}$

A few details

Input Module

- Single sentence : Output is hidden state of each token

This is a sentence

- Multiple sentences: Output is hidden state of each end-of-sentence token

This is a sentence EOS This is another EOS

Episodic Memory Module

- Need for Multiple Episodes

Where is the football?

John put down the football.

John went to the hallway.

- Attention Mechanism (actually, gate mechanism)

$$G(c, m, q) = \sigma(W^{(2)} \tanh(W^{(1)} z(c, m, q) + b^{(1)}) + b^{(2)})$$

$$z(c, m, q) = [c, m, q, c \cdot q, c \cdot m, |c - q|, |c - m|, c^T W^{(b)} q, c^T W^{(b)} m]$$

  - Used softmax instead for question answering, achieving better result.
  - Can be trained with supervision for bAbI dataset.

- Criteria for Stopping
  - Append a special end-of-passes token to the input, and stop if it is chosen by the gate function
  - Maximum number of iterations for others

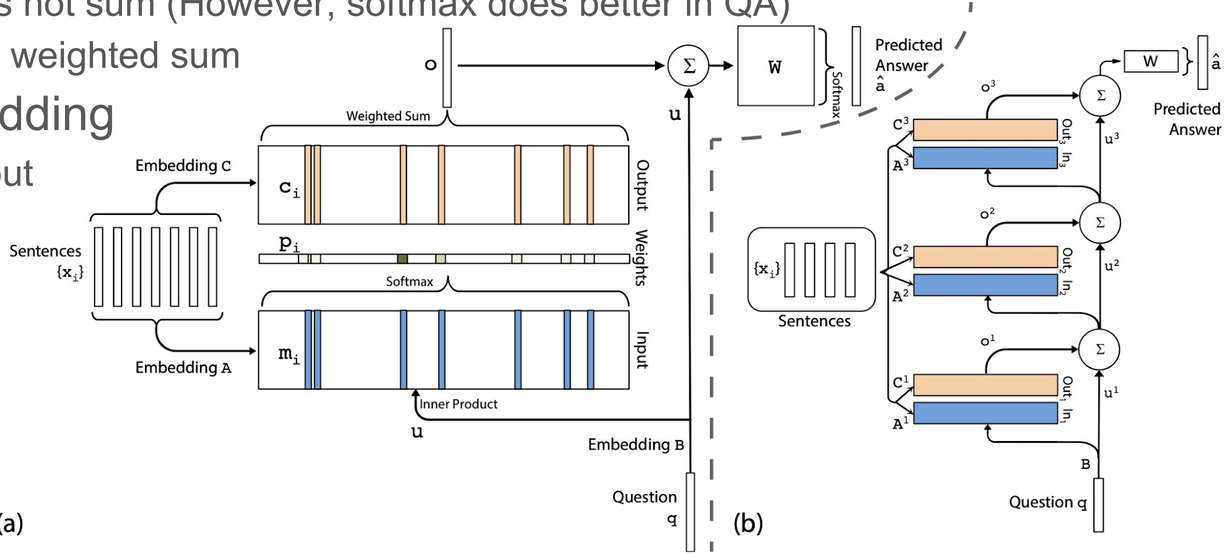
Answer Module

- For sequence labeling, we need answer for each input word.

$$e^i = h_t^i$$

vs. Memory Networks

- Episodic Memory Module vs. Summing
  - DMN runs GRU over episodes
  - MemNN sums output (memory attention) & input (question)
- Gate vs. Attention
  - DMN uses sigmoid, does not sum (However, softmax does better in QA)
  - MemNN uses softmax & weighted sum
- RNN on input vs. embedding
  - DMN runs GRU over input



Experiments: Question Answering

Task	MemNN	DMN
1: Single Supporting Fact	100	100
2: Two Supporting Facts	100	98.2
3: Three Supporting Facts	100	95.2
4: Two Argument Relations	100	100
5: Three Argument Relations	98	99.3
6: Yes/No Questions	100	100
7: Counting	85	96.9
8: Lists/Sets	91	96.5
9: Simple Negation	100	100
10: Indefinite Knowledge	98	97.5
11: Basic Coreference	100	99.9
12: Conjunction	100	100
13: Compound Coreference	100	99.8
14: Time Reasoning	99	100
15: Basic Deduction	100	100
16: Basic Induction	100	99.4
17: Positional Reasoning	65	59.6
18: Size Reasoning	95	95.3
19: Path Finding	36	34.5
20: Agent's Motivations	100	100
Mean Accuracy (%)	93.3	**93.6**

Table 1. Test accuracies on the bAbI dataset. MemNN numbers taken from Weston et al. (Weston et al., 2015a). The DMN passes (accuracy $>$ 95%) 18 tasks, whereas the MemNN passes 16.

- DMN does worse on tasks 2 and 3
  - Recurrent input sequence model has trouble modeling very long inputs
- DMN does better on tasks 7 and 8
  - Both tasks require the model to iteratively retrieve facts and store them in a representation that slowly incorporates more of the relevant information of the input sequence.
  - Power of episodic memory module
- Both do poorly on tasks 17 and 19
  - But MemNN does better
  - Maybe due to the MemNN using n-gram vectors and sequence position features.

Quantitative Analysis of Episodic Memory Module

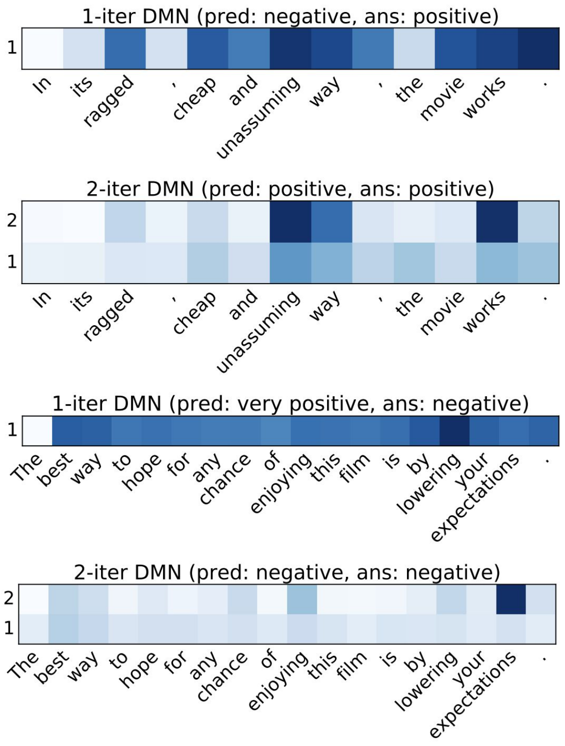
Max passes	task 3 three-facts	task 7 count	task 8 lists/sets	sentiment (fine grain)
0 pass	0	48.8	33.6	50.0
1 pass	0	48.8	54.0	51.5
2 pass	16.7	49.1	55.6	**52.1**
3 pass	64.7	83.4	83.4	50.1
5 pass	**95.2**	**96.9**	**96.5**	N/A

Qualitative Analysis of Episodic Memory Module

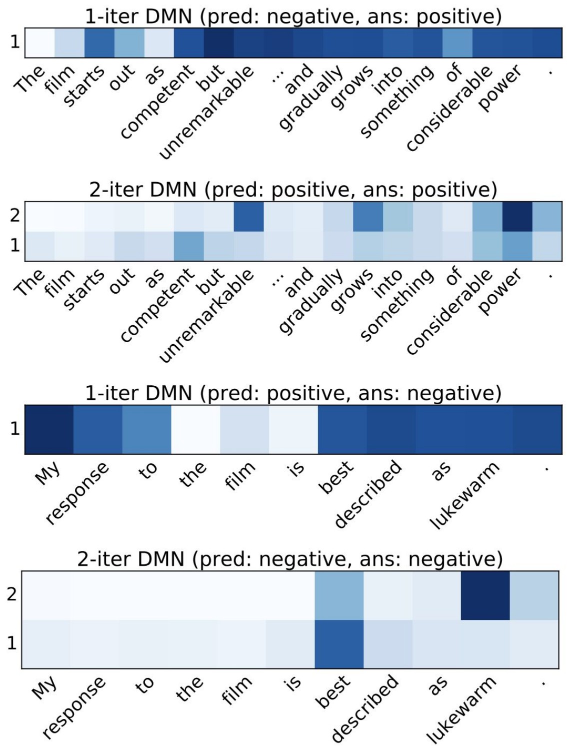
Question: Where was Mary before the Bedroom?
Answer: Cinema.

Facts	Episode 1	Episode 2	Episode 3
Yesterday Julie traveled to the school.			
Yesterday Marie went to the cinema.			
This morning Julie traveled to the kitchen.			
Bill went back to the cinema yesterday.			
Mary went to the bedroom this morning.			
Julie went back to the bedroom this afternoon.			
[done reading]			

Qualitative Analysis of Episodic Memory Module



Second thoughts led to correct answer



Second thoughts changed attention

Parting thoughts

- 그 많던 메모리 네트워크는 누가 다 먹었을까
  - Transformer & Pretraining의 시대에 Input을 여러번 본다는 아이디어는 여전히 유효할까?
  - Self-attention은 메모리의 필요성을 없앴는가?

Thank you