End-to-End Memory Networks on bAbI

Ekrem Guzelyel - Hasan Rizvi
A20384767 A20374805

What do you see in these pictures?

1.

Attention!

because, we don't need the clutter.

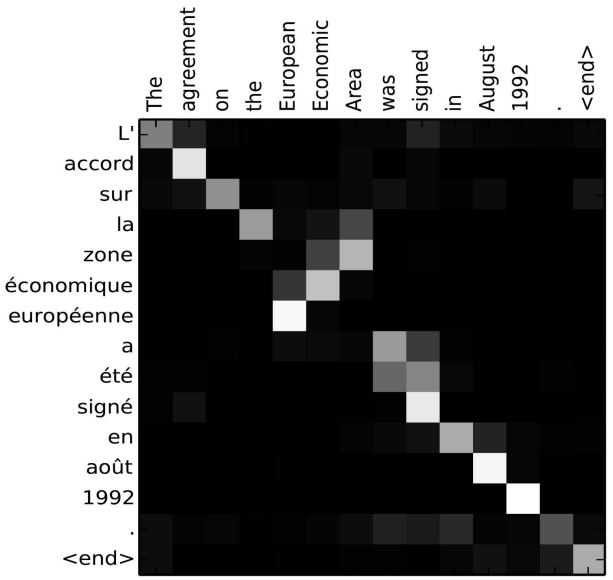
Attention!

Human brain doesn't have to render the full image.

Whereas the machines do? or do they?

Attention!

Embedding Layer Association



Attention is all you need.

$$w_t^c(i) = \text{softmax}(\beta_t \cdot \text{cosine}[\mathbf{k}_t, \mathbf{M}_t(i)]) = \frac{\exp(\beta_t \frac{\mathbf{k}_t \cdot \mathbf{M}_t(i)}{\|\mathbf{k}_t\| \cdot \|\mathbf{M}_t(i)\|})}{\sum_{j=1}^{N} \exp(\beta_t \frac{\mathbf{k}_t \cdot \mathbf{M}_t(j)}{\|\mathbf{k}_t\| \cdot \|\mathbf{M}_t(j)\|})}$$

Remember the animals? What are they doing?

2.

Memory Networks

If you don't want to do the same mistake twice.

Memory Networks

What is Memory?

Facebook Research – Weston Team

Towards AI Complete Question Answering: A Set of Prerequisite Toy Tasks

bAbI tasks

Memory Networks

Single Layer

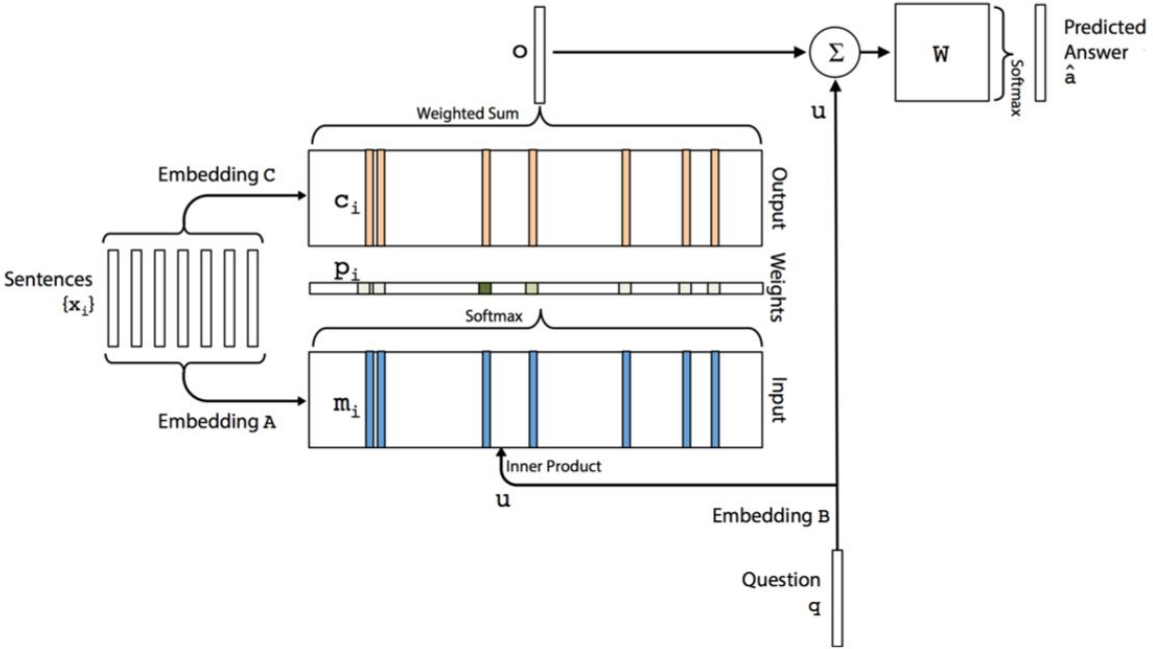


Image Credit: jhui.github.io/2017/03/15/Memory-network/

Memory Networks

Single Layer

$$m_i = embedding_A(x_i)$$

$$u = embedding_B(q)$$

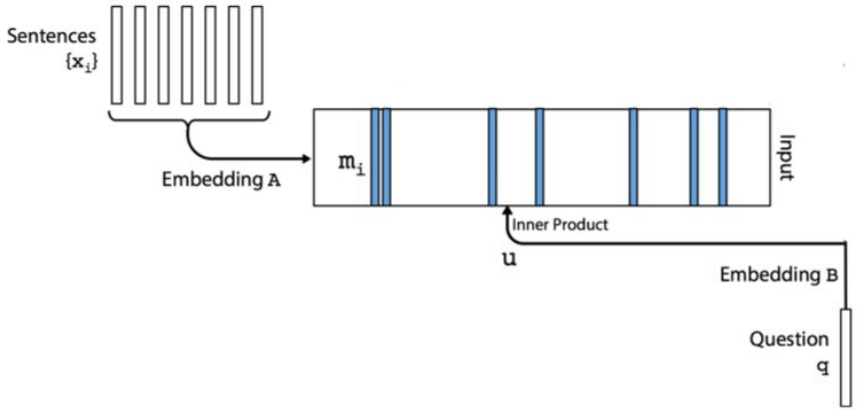


Image Credit: jhui.github.io/2017/03/15/Memory-network/

Memory Networks

Single Layer Output

$$\hat{a} = softmax(W(o + u))$$

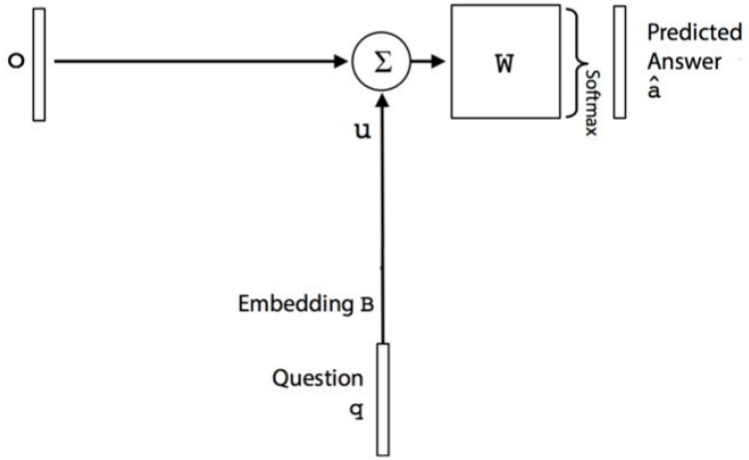


Image Credit: jhui.github.io/2017/03/15/Memory-network/

Memory Networks

Single Layer

$$p_i = softmax(u^T m_i).$$

$$u = embedding_B(q)$$

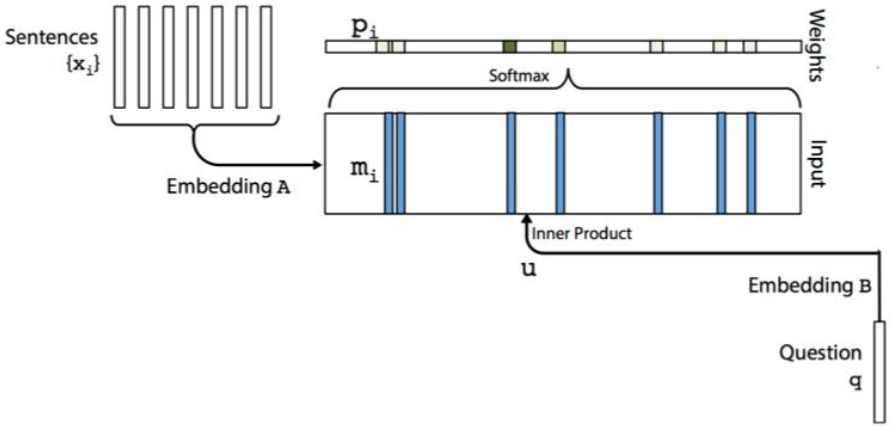


Image Credit: jhui.github.io/2017/03/15/Memory-network/

Memory Networks

Single Layer

$$o = \sum_i p_i c_i.$$

$$c_i = embedding_C(x_i)$$

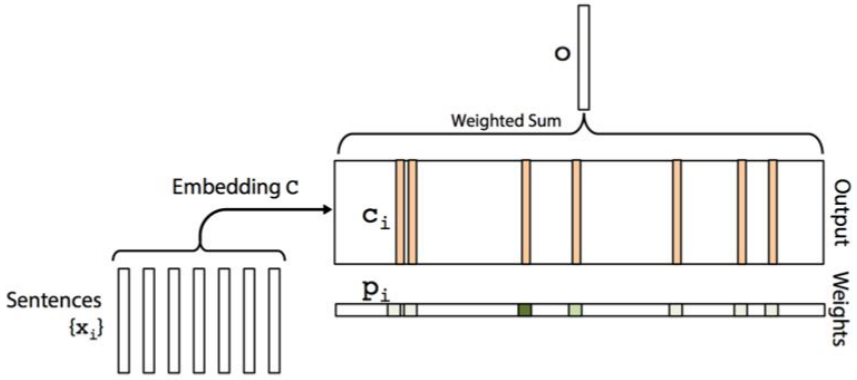


Image Credit: jhui.github.io/2017/03/15/Memory-network/

Memory Networks

Single Layer

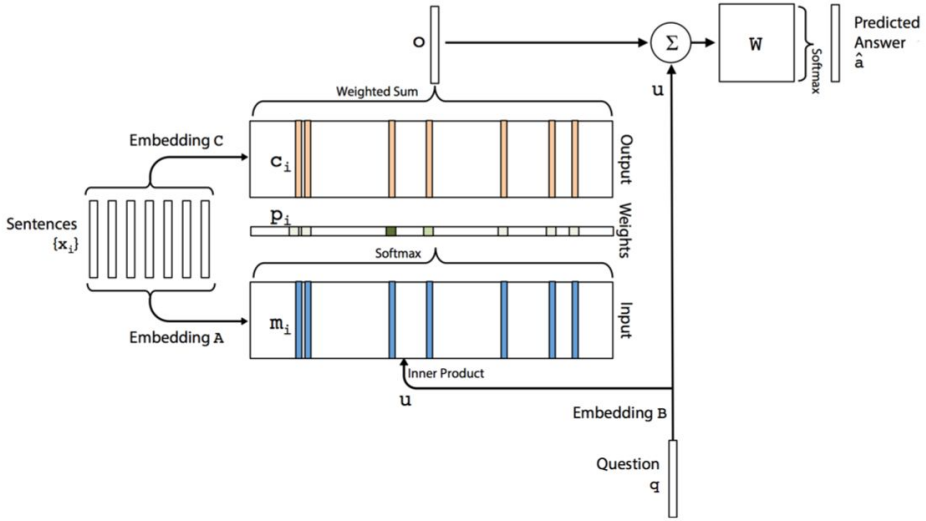


Image Credit: jhui.github.io/2017/03/15/Memory-network/

Memory Networks

Multi Layer (Hops) MemN2N

$$u^{k+1} = u^k + o^k.$$

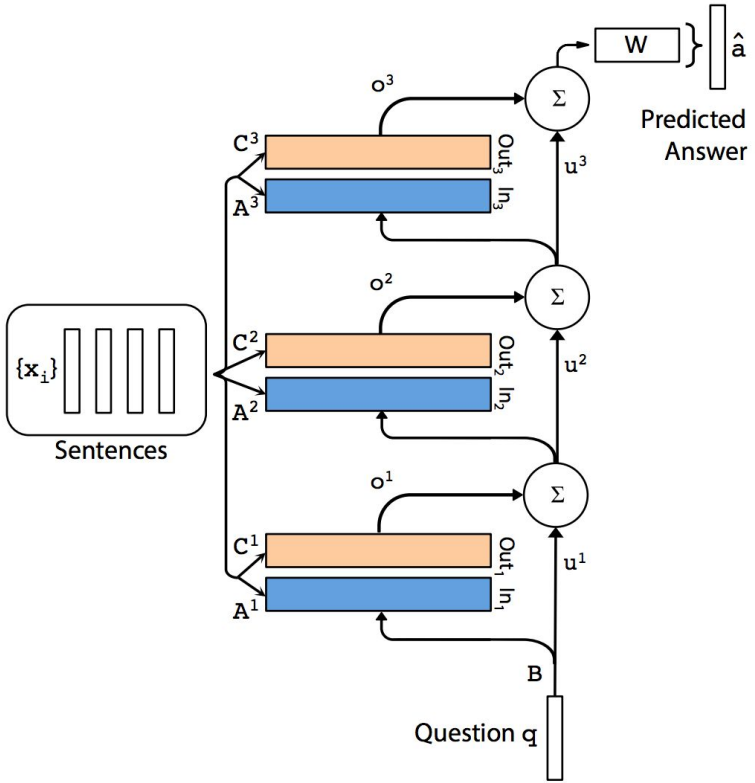


Image Credit: jhui.github.io/2017/03/15/Memory-network/

Memory Networks

Multi Layer (Hops)

Adjacent

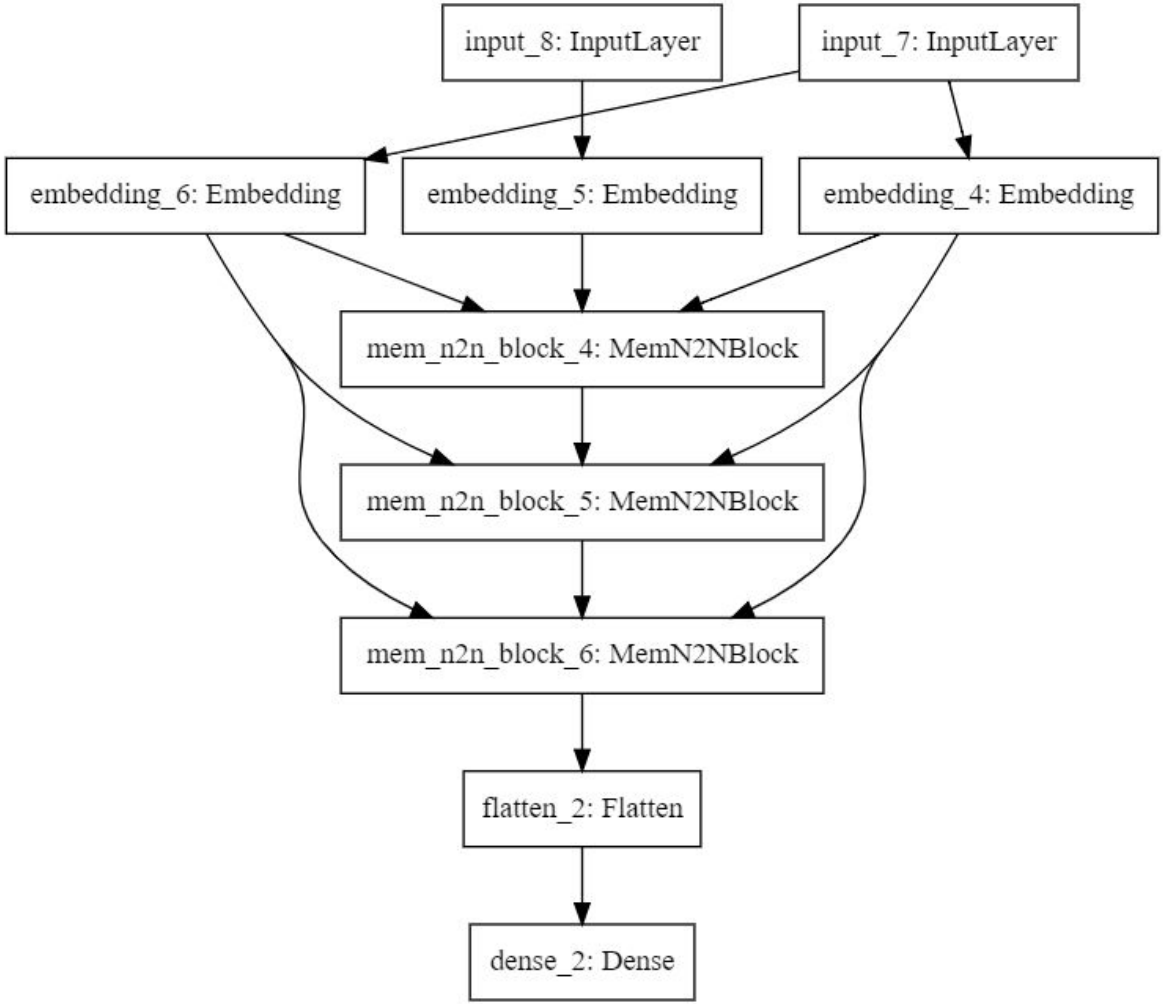
$$A^{k+1} = C^k$$

Layer-wise

$$A^1 = A^2 = ... = A^K$$
$$C^1 = C^2 = ... = C^K.$$

$$u^{k+1} = Hu^k + o^k.$$

Image Credit: jhui.github.io/2017/03/15/Memory-network/


input_8: InputLayer
input_7: InputLayer
embedding_6: Embedding
embedding_5: Embedding
embedding_4: Embedding
mem_n2n_block_4: MemN2NBlock
mem_n2n_block_5: MemN2NBlock
mem_n2n_block_6: MemN2NBlock
flatten_2: Flatten
dense_2: Dense

bAbI Tasks

1. Single Supporting Fact
7. Counting
16. Basic Induction
18. Size Reasoning

bAbI Examples

```
1 John travelled to the hallway.
2 Mary journeyed to the bathroom.
3 Where is John?        hallway 1
4 Daniel went back to the bathroom.
5 John moved to the bedroom.
6 Where is Mary?        bathroom        2
7 John went to the hallway.
8 Sandra journeyed to the kitchen.
9 Where is Sandra?      kitchen 8
```

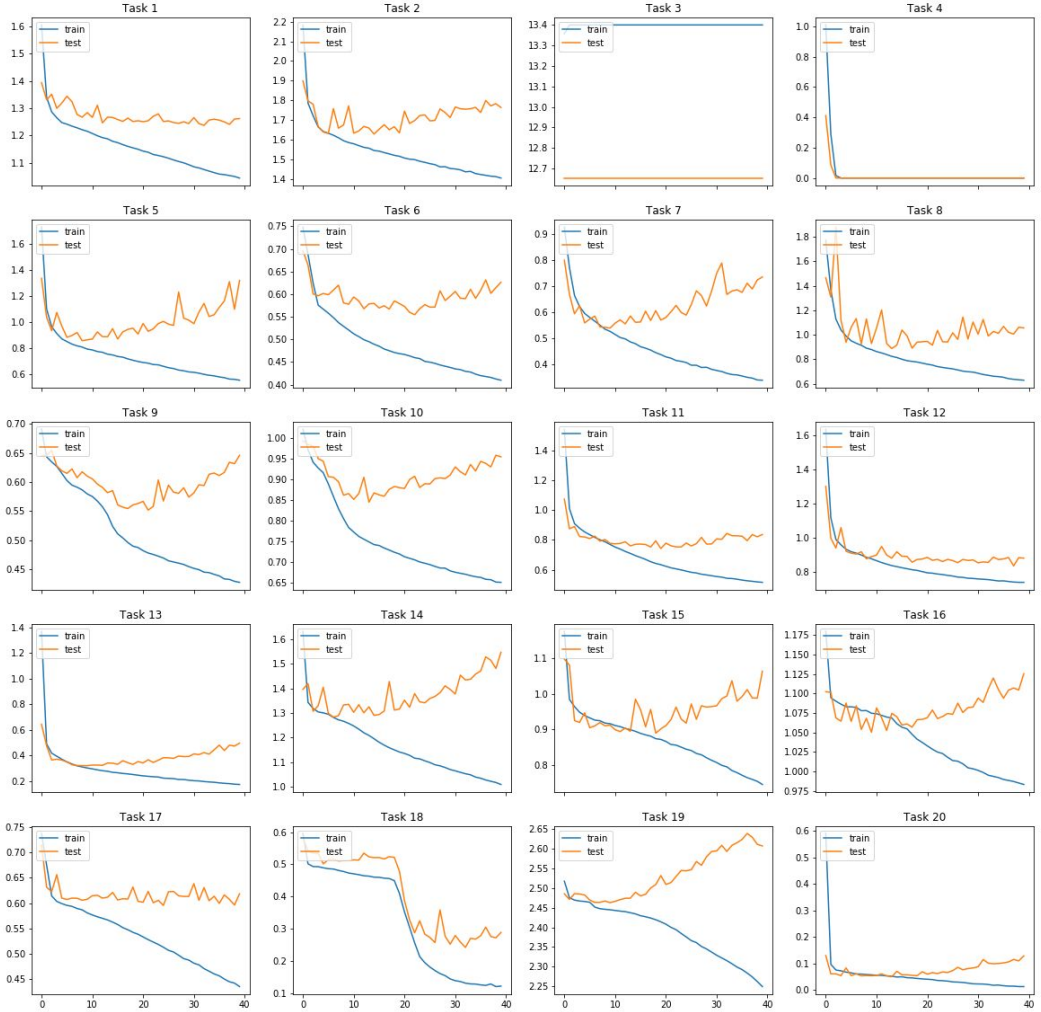
```
1 Sandra went to the garden.
2 Daniel journeyed to the bedroom.
3 Mary picked up the apple there.
4 Sandra got the milk there.
5 How many objects is Sandra carrying?  one     4
6 Mary went to the kitchen.
7 Sandra discarded the milk.
8 How many objects is Sandra carrying?  none    4 7
```

```
1 Lily is a frog.
2 Bernhard is a frog.
3 Bernhard is green.
4 Brian is a lion.
5 Brian is white.
6 Julius is a swan.
7 Julius is green.
8 Lily is green.
9 Greg is a swan.
10 What color is Greg?  green   9 6 7
```

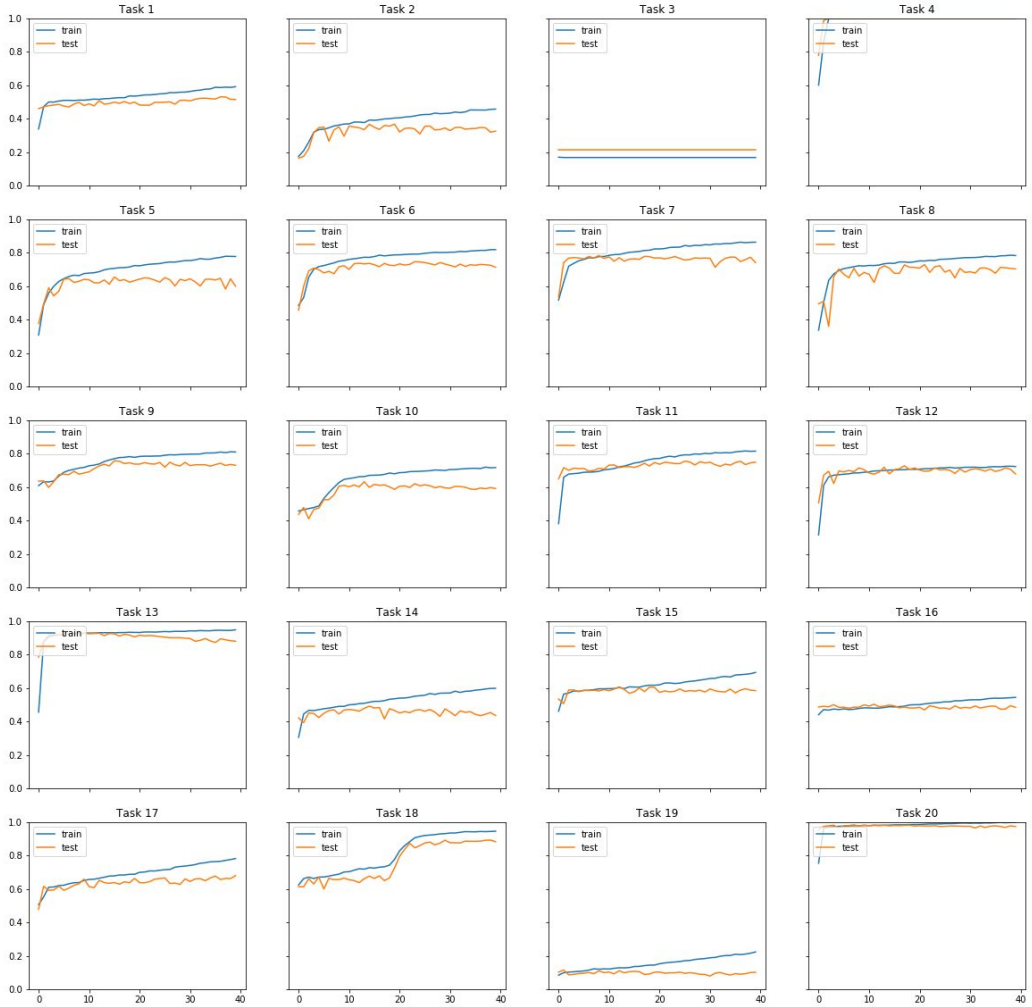
```
1 The box of chocolates fits inside the chest.
2 The box is bigger than the chest.
3 The box is bigger than the suitcase.
4 The suitcase fits inside the box.
5 The container is bigger than the box of chocolates.
6 Does the box fit in the box of chocolates?    no      1 2
7 Is the box of chocolates bigger than the box? no      1 2
8 Is the box bigger than the box of chocolates? yes     2 1
9 Does the box of chocolates fit in the box?    yes     2 1
10 Does the box fit in the box of chocolates?   no      1 2
```

	Baseline			MemN2N								
Task	Strongly Supervised MemNN [22]	LSTM [22]	MemNN WSH	BoW	PE	PE LS	PE LS RN	1 hop PE LS joint	2 hops PE LS joint	3 hops PE LS joint	PE LS RN joint	PE LS LW joint
1: 1 supporting fact	0.0	50.0	0.1	0.6	0.1	0.2	0.0	0.8	0.0	0.1	0.0	0.1
2: 2 supporting facts	0.0	80.0	42.8	17.6	21.6	12.8	8.3	62.0	15.6	14.0	11.4	18.8
3: 3 supporting facts	0.0	80.0	76.4	71.0	64.2	58.8	40.3	76.9	31.6	33.1	21.9	31.7
4: 2 argument relations	0.0	39.0	40.3	32.0	3.8	11.6	2.8	22.8	2.2	5.7	13.4	17.5
5: 3 argument relations	2.0	30.0	16.3	18.3	14.1	15.7	13.1	11.0	13.4	14.8	14.4	12.9
6: yes/no questions	0.0	52.0	51.0	8.7	7.9	8.7	7.6	7.2	2.3	3.3	2.8	2.0
7: counting	15.0	51.0	36.1	23.5	21.6	20.3	17.3	15.9	25.4	17.9	18.3	10.1
8: lists/sets	9.0	55.0	37.8	11.4	12.6	12.7	10.0	13.2	11.7	10.1	9.3	6.1
9: simple negation	0.0	36.0	35.9	21.1	23.3	17.0	13.2	5.1	2.0	3.1	1.9	1.5
10: indefinite knowledge	2.0	56.0	68.7	22.8	17.4	18.6	15.1	10.6	5.0	6.6	6.5	2.6
11: basic coreference	0.0	38.0	30.0	4.1	4.3	0.0	0.9	8.4	1.2	0.9	0.3	3.3
12: conjunction	0.0	26.0	10.1	0.3	0.3	0.1	0.2	0.4	0.0	0.3	0.1	0.0
13: compound coreference	0.0	6.0	19.7	10.5	9.9	0.3	0.4	6.3	0.2	1.4	0.2	0.5
14: time reasoning	1.0	73.0	18.3	1.3	1.8	2.0	1.7	36.9	8.1	8.2	6.9	2.0
15: basic deduction	0.0	79.0	64.8	24.3	0.0	0.0	0.0	46.4	0.5	0.0	0.0	1.8
16: basic induction	0.0	77.0	50.5	52.0	52.1	1.6	1.3	47.4	51.3	3.5	2.7	51.0
17: positional reasoning	35.0	49.0	50.9	45.4	50.1	49.0	51.0	44.4	41.2	44.5	40.4	42.6
18: size reasoning	5.0	48.0	51.3	48.1	13.6	10.1	11.1	9.6	10.3	9.2	9.4	9.2
19: path finding	64.0	92.0	100.0	89.7	87.4	85.6	82.8	90.7	89.9	90.2	88.0	90.6
20: agent's motivation	0.0	9.0	3.6	0.1	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.2
Mean error (%)	6.7	51.3	40.2	25.1	20.3	16.3	13.9	25.8	15.6	13.3	12.4	15.2
Failed tasks (err. $>$ 5%)	4	20	18	15	13	12	11	17	11	11	11	10
On 10k training data												
Mean error (%)	3.2	36.4	39.2	15.4	9.4	7.2	6.6	24.5	10.9	7.9	7.5	11.0
Failed tasks (err. $>$ 5%)	2	16	17	9	6	4	4	16	7	6	6	6

Losses


Task 1
Task 2
Task 3
Task 4
Task 5
Task 6
Task 7
Task 8
Task 9
Task 10
Task 11
Task 12
Task 13
Task 14
Task 15
Task 16
Task 17
Task 18
Task 19
Task 20
train
test

Accuracies


Task 1
Task 2
Task 3
Task 4
Task 5
Task 6
Task 7
Task 8
Task 9
Task 10
Task 11
Task 12
Task 13
Task 14
Task 15
Task 16
Task 17
Task 18
Task 19
Task 20
train
test

Methods

Implemented the Paper Explanation

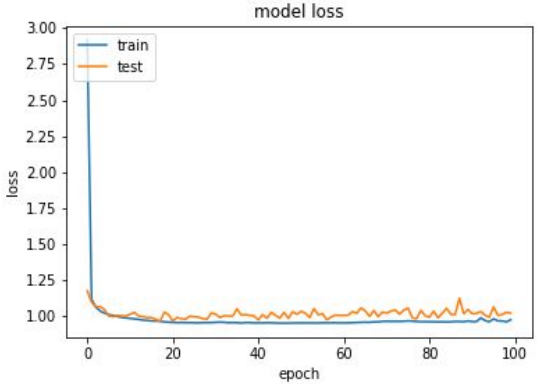
Train: 9000 * 20

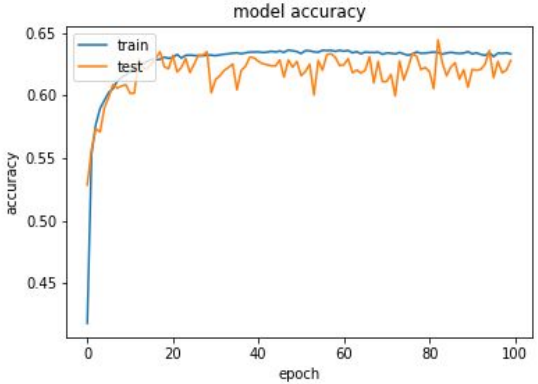
Validation: 5000

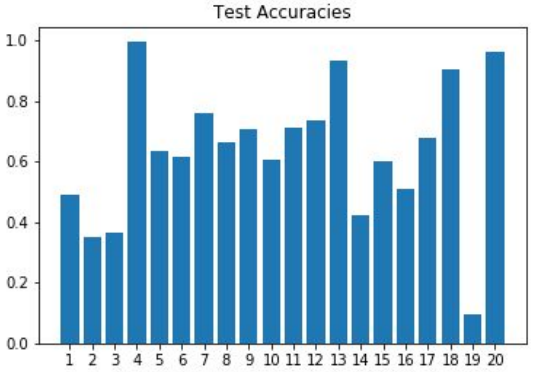
Test: 1000 * 20

Used Keras

Joint Training







Results

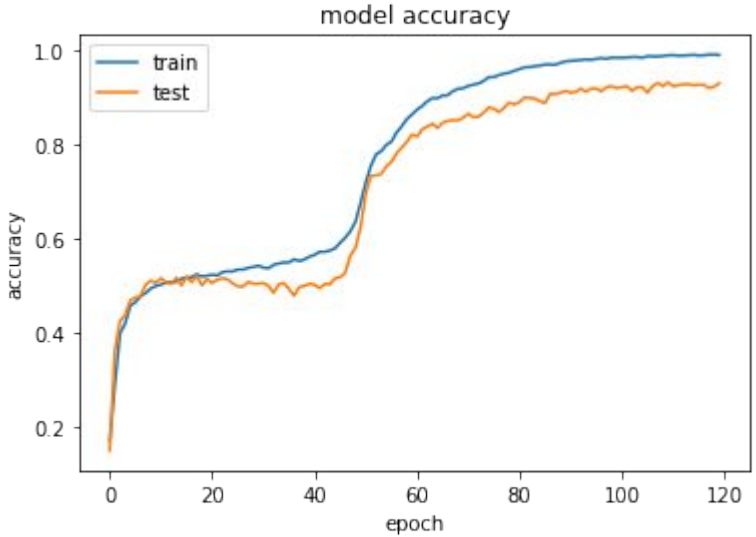
Adding LSTM/Dense

Result | Orig.	Task 1	Task 7	Task 16	Task 18
MemNN	**93% | 98%**	**34%**	**92%**	**~92% | 88%**
MemN2N	**94% | 98%**	**36%**	**50%**	**~91%**

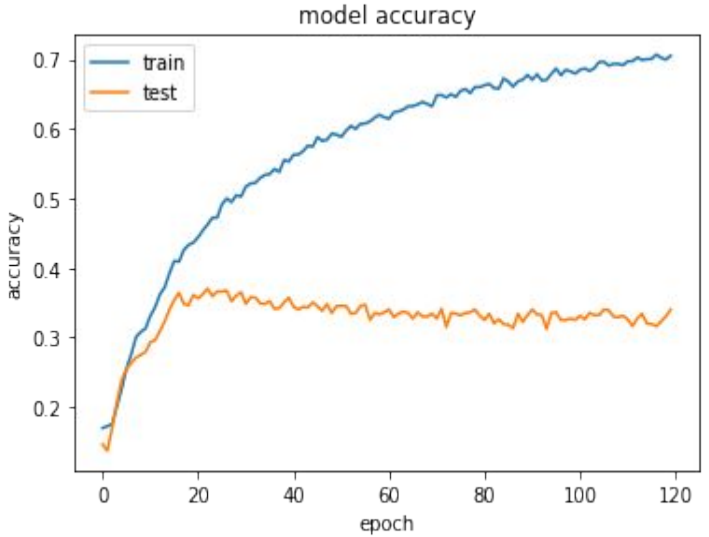
Results

Adding LSTM/Dense

Task 1: 94% Acc



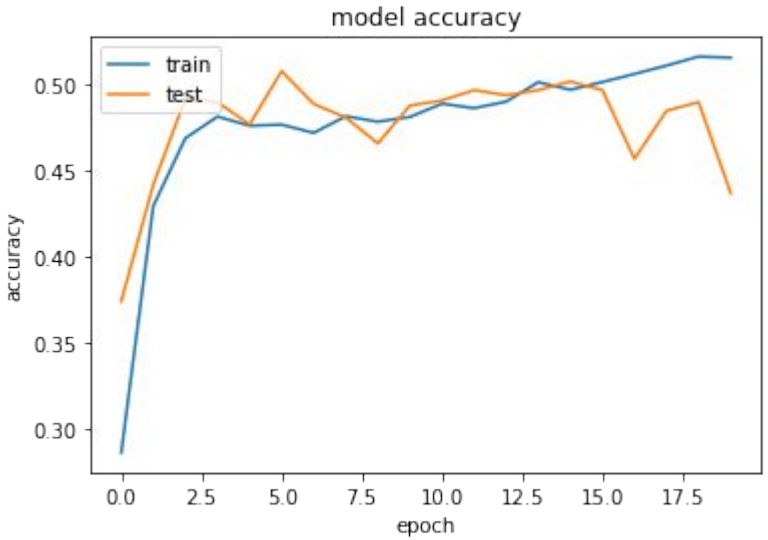
Task 7: 34% Acc



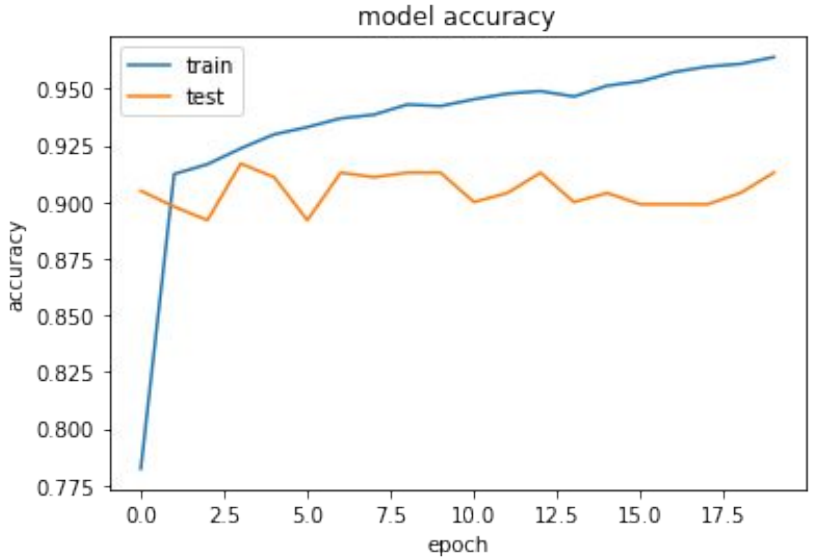
Results

Adding LSTM/Dense

Task 16: 49% Acc



Task 18: 92% Acc



Conclusion

Overall good results when combined w/o* different layers.

Memory Networks is a new era.

THANKS!

Any questions?

You can find the full code @
github.com/Eguzelyel/MemN2N_QA
github.com/hizvi/MemN2N

References

1. End-To-End Memory Networks by Sukhbaatar, Szlam, Weston, Fergus
2. Memory Networks by Weston, Chopra, Bordes
3. Towards AI Complete Question Answering: A Set of Prerequisite Toy Tasks by Weston, Border, Chopra
4. jhui.github.io/2017/03/15/Memory-network/
5. lilianweng.github.io/lil-log/2018/06/24/attention-attention.html