Memory Networks for Language Understanding

QA problem

John is in the playground.
John picked up the football.
Bob went to the kitchen.
Where is the football? A:playground

Supporting facts

⬇

Intuition

- Large long term memory ➔ read and written to
- Attention over memory ➔ reasoning

MemNet Framework

1. Convert x to an internal feature representation $I(x)$.
2. Update memories $\mathbf{m}_i$ given the new input: $\mathbf{m}_i = G(\mathbf{m}_i, I(x), \mathbf{m}),\ \forall i$.
3. Compute output features o given the new input and the memory: $o = O(I(x), \mathbf{m})$.
4. Finally, decode output features o to give the final response: $r = R(o)$.

Basic Model

- O (output) component: support facts

$$o_1 = O_1(x, \mathbf{m}) = \underset{i=1,\dots,N}{\arg\max}\ s_O(x, \mathbf{m}_i)$$

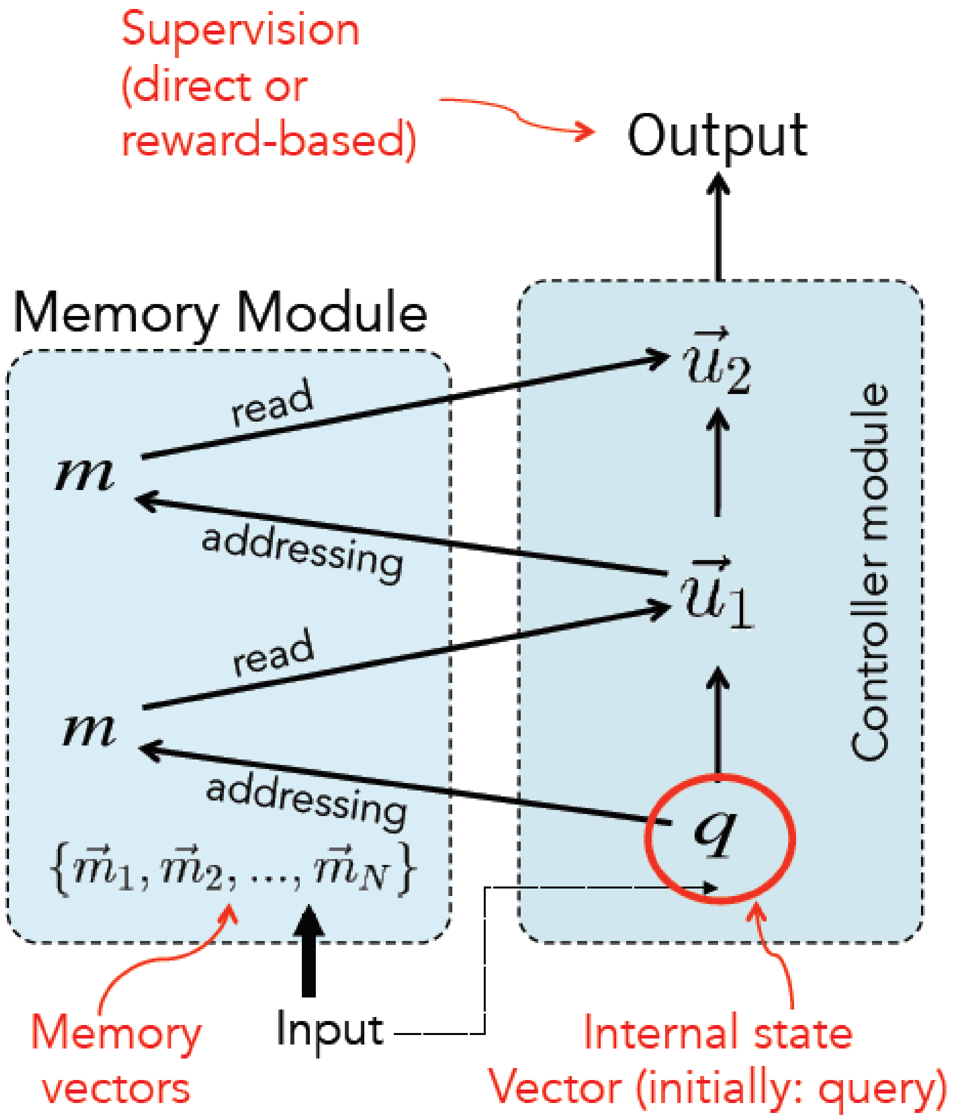
$$o_2 = O_2(x, \mathbf{m}) = \underset{i=1,\dots,N}{\arg\max}\ s_O([x, \mathbf{m}_{o_1}], \mathbf{m}_i)$$

- R (response) component:

$$r = \text{argmax}_{w \in W}\ s_R([x, \mathbf{m}_{o_1}, \mathbf{m}_{o_2}], w)$$

- Scoring function:

$$s(x, y) = \Phi_x(x)^\top U^\top U \Phi_y(y)$$



Objective function

Minimize:

$$\sum_{\bar{f}\neq\mathbf{m}_{o_1}} \max(0, \gamma - s_O(x, \mathbf{m}_{o_1}) + s_O(x, \bar{f})) +$$

$$\sum_{\bar{f}'\neq\mathbf{m}_{o_2}} \max(0, \gamma - s_O([x, \mathbf{m}_{o_1}], \mathbf{m}_{o_2}]) + s_O([x, \mathbf{m}_{o_1}], \bar{f}'])) +$$

$$\sum_{\bar{r}\neq r} \max(0, \gamma - s_R([x, \mathbf{m}_{o_1}, \mathbf{m}_{o_2}], r) + s_R([x, \mathbf{m}_{o_1}, \mathbf{m}_{o_2}], \bar{r}]))$$

Where: S_O is the matching function for the Output component.
S_R is the matching function for the Response component.
x is the input question.
m_{O1} is the first true supporting memory (fact).
m_{O2} is the first second supporting memory (fact).
r is the response
True facts and responses $m_{O1,}$ m_{O2} and r should have higher scores than all other facts and responses by a given margin.

Variants of the class

- **Representation of inputs:** bag of words, RNN style reading at word or character level, etc. → **DMN**
- **Different possibilities for output module:** e.g. multiclass classifier or uses an RNN to output sentences.
- **If the memory is huge** (e.g. Wikipedia): hash the memories to store in buckets (topics). Then, memory addressing and reading doesn't operate on *all* memories.
- **If the memory is full**, there could be a way of removing one it thinks is most useless; *i.e*. it ``forgets'' somehow. That would require a scoring function of the utility of each memory..

bAbI dataset

- To measure understanding in several ways:
  - answer questions via chaining facts
  - simple induction
  - simple deduction

- Working with larger amount of read data tends to lead researchers to simpler models

- The dataset contains 20 nearly independent tasks, each checking one skill that the system must have (1k questions per task; About 15 sentences per story)

Simple Tasks

Task 1: Single Supporting Fact
Mary went to the bathroom.
John moved to the hallway.
Mary travelled to the office.
Where is Mary? A:office

Task 2: Two Supporting Facts
John is in the playground.
John picked up the football.
Bob went to the kitchen.
Where is the football? A:playground

Task 3: Three Supporting Facts
John picked up the apple.
John went to the office.
John went to the kitchen.
John dropped the apple.
Where was the apple before the kitchen? A:office

Task 4: Two Argument Relations
The office is north of the bedroom.
The bedroom is north of the bathroom.
The kitchen is west of the garden.
What is north of the bedroom? A: office
What is the bedroom north of? A: bathroom

→ **Bag of words will not work**

Task 5: Three Argument Relations
Mary gave the cake to Fred.
Fred gave the cake to Bill.
Jeff was given the milk by Bill.
Who gave the cake to Fred? A: Mary
Who did Fred give the cake to? A: Bill

Task 6: Yes/No Questions
John moved to the playground.
Daniel went to the bathroom.
John went back to the hallway.
Is John in the playground? A:no
Is Daniel in the bathroom? A:yes

Task 7: Counting
Daniel picked up the football.
Daniel dropped the football.
Daniel got the milk.
Daniel took the apple.
How many objects is Daniel holding? A: two

Task 8: Lists/Sets
Daniel picks up the football.
Daniel drops the newspaper.
Daniel picks up the milk.
John took the apple.
What is Daniel holding? milk, football

→ **harder ones**

Task 9: Simple Negation
Sandra travelled to the office.
Fred is no longer in the office.
Is Fred in the office? A:no
Is Sandra in the office? A:yes

Task 10: Indefinite Knowledge
John is either in the classroom or the playground.
Sandra is in the garden.
Is John in the classroom? A:maybe
Is John in the office? A:no

Harder ones

- (17) Positional Reasoning

> The triangle is to the right of the blue square.
> The red square is on top of the blue square.
> The red sphere is to the right of the blue square.
> Is the red sphere to the right of the blue square? A:yes
> Is the red square to the left of the triangle? A:yes

- (19) Path Finding

> The kitchen is north of the hallway.
> The den is east of the hallway.
> How do you go from den to kitchen? A:west,north

Contributions of MemNet

- Global long term memory

- Attention mechanism for inference

- Make use of strong supervision information (supporting facts)

End-to-end Memory Network (MemN2N)

- New end-to-end (MemN2N) model (Sukhbaatar '15):
  - Reads from memory with **soft attention**
  - Performs **multiple lookups** (hops) on memory
  - End-to-end training with **backpropagation**
  - Only need supervision on the final output

performance diminish

- It is based on "Memory Networks" by [Weston, Chopra & Bordes ICLR 2015] but that had:
  - Hard attention
  - requires explicit supervision of attention during training
  - Only feasible for simple tasks

Model

- Single Hop

$m_i, u \in R^d \quad A, B, C \in R^{d \times V} \quad W \in R^{V \times d}$

J_i: *The length of the* i^{th} *Sentence*

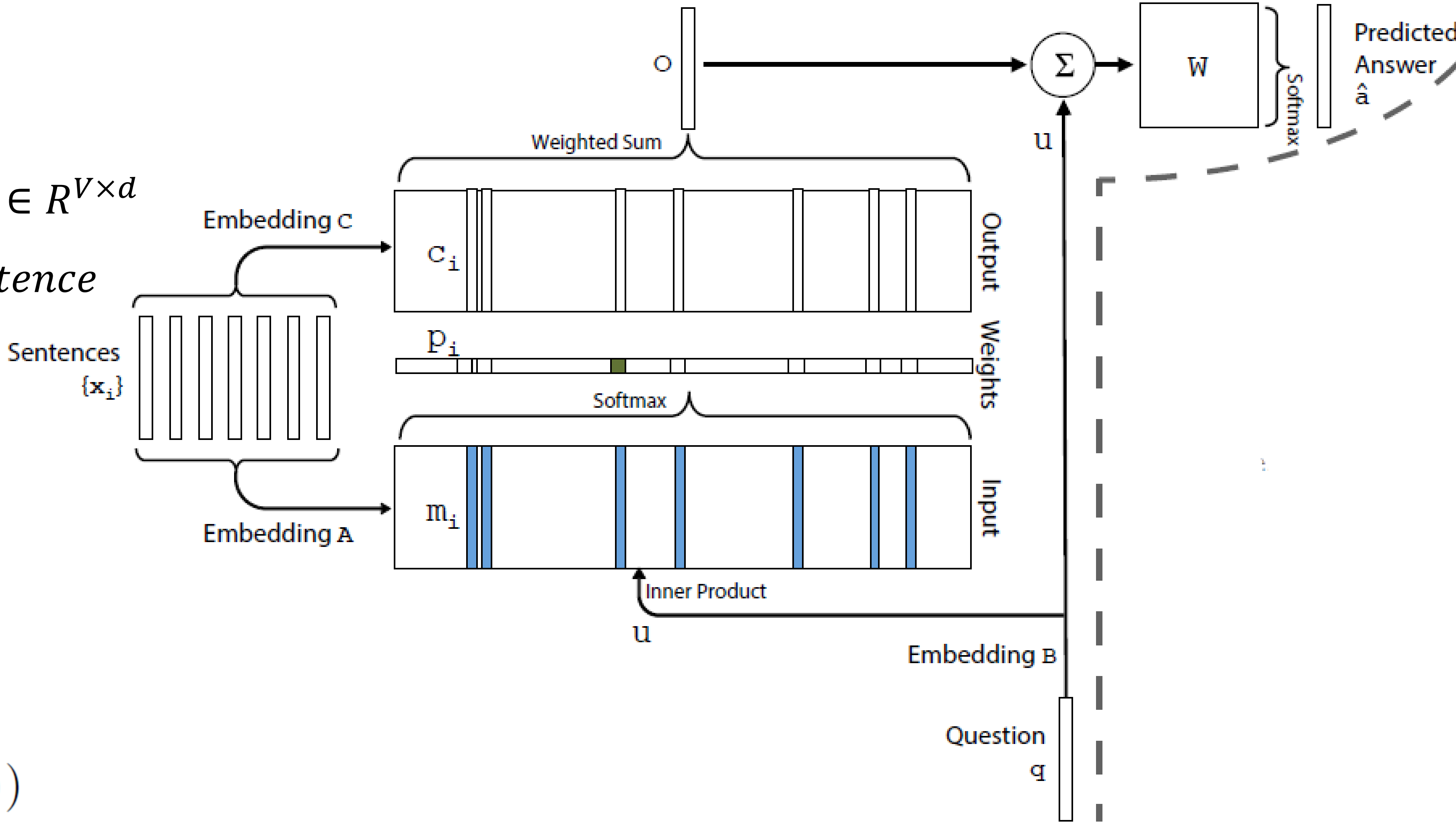
$$x_i = \{x_{i1}, \ldots, x_{iJ_i}\}$$

$$m_i = \sum_j A x_{ij}$$

$$p_i = \text{Softmax}(u^T m_i).$$

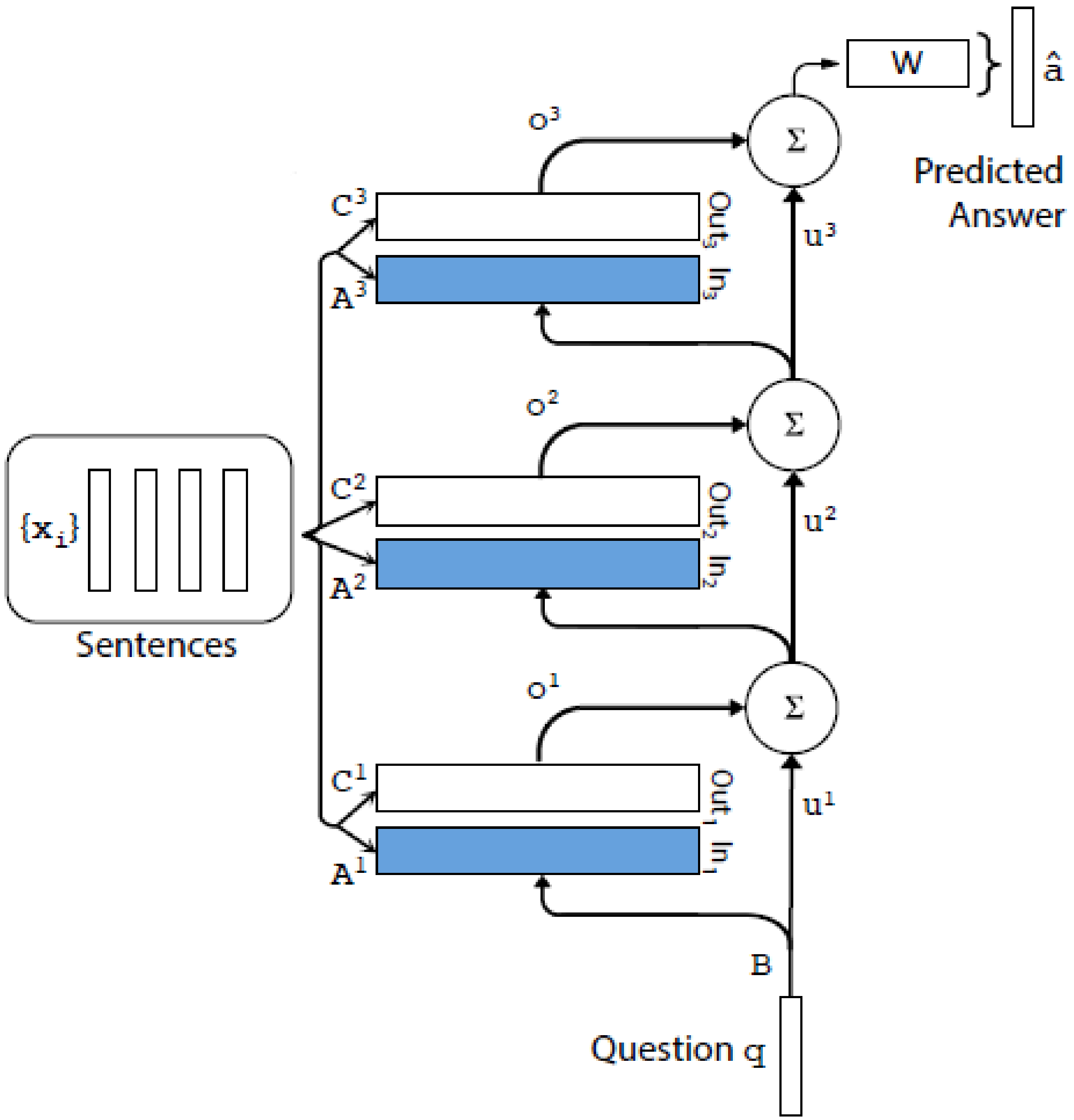
$$o = \sum_i p_i c_i.$$

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



Model

- Multi Hops

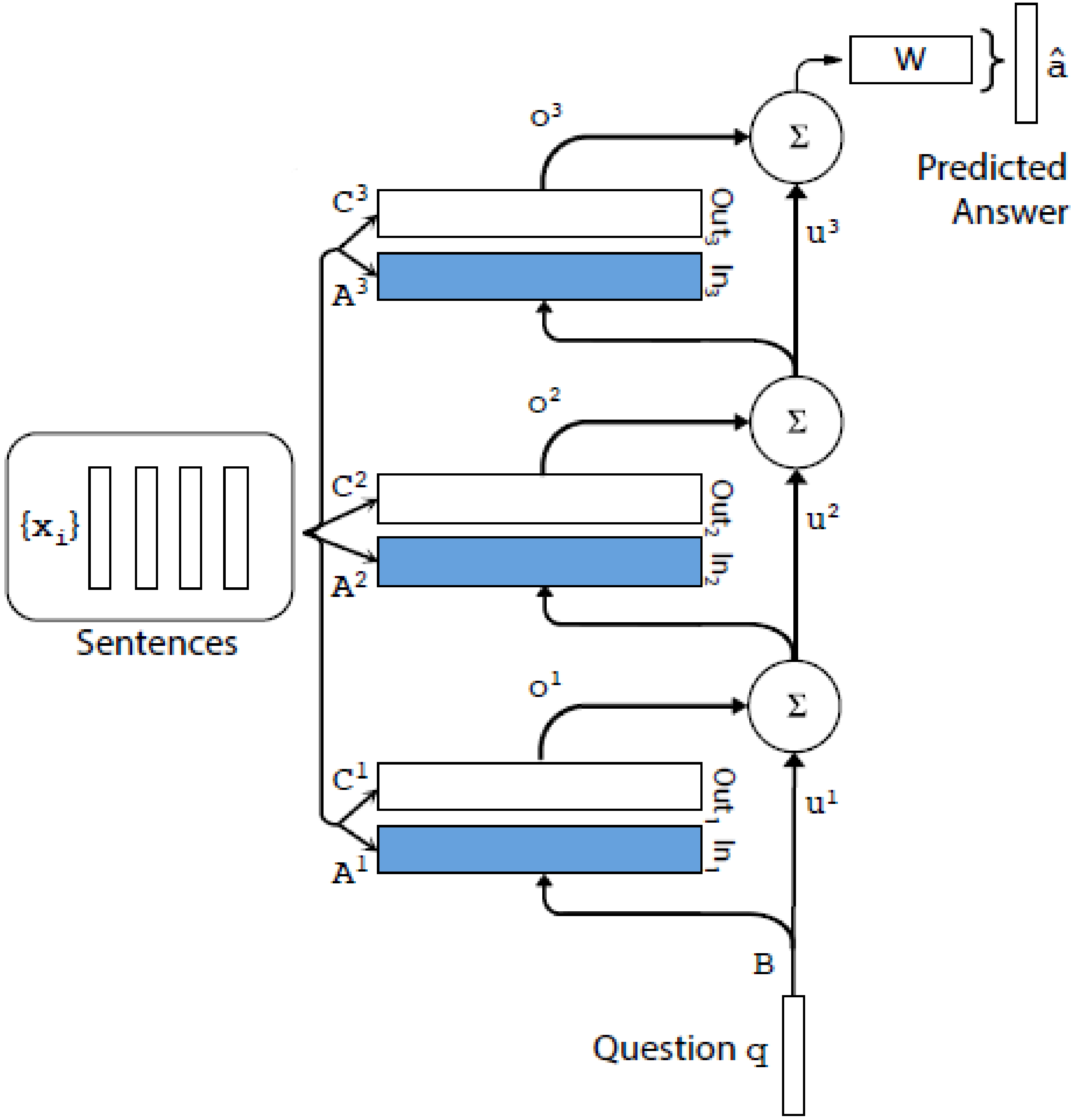


Model details

- Weight Sharing：

Adjacent：
$B = A^1$
$A^{k+1} = C^k, k = 1, \dots, K - 1$
$C^K = W$

Layer-wise (RNN-like):
$A^1 = \cdots = A^K$
$C^1 = \cdots = C^K$
$u^{k+1} = Hu^k + o^k$



Model details

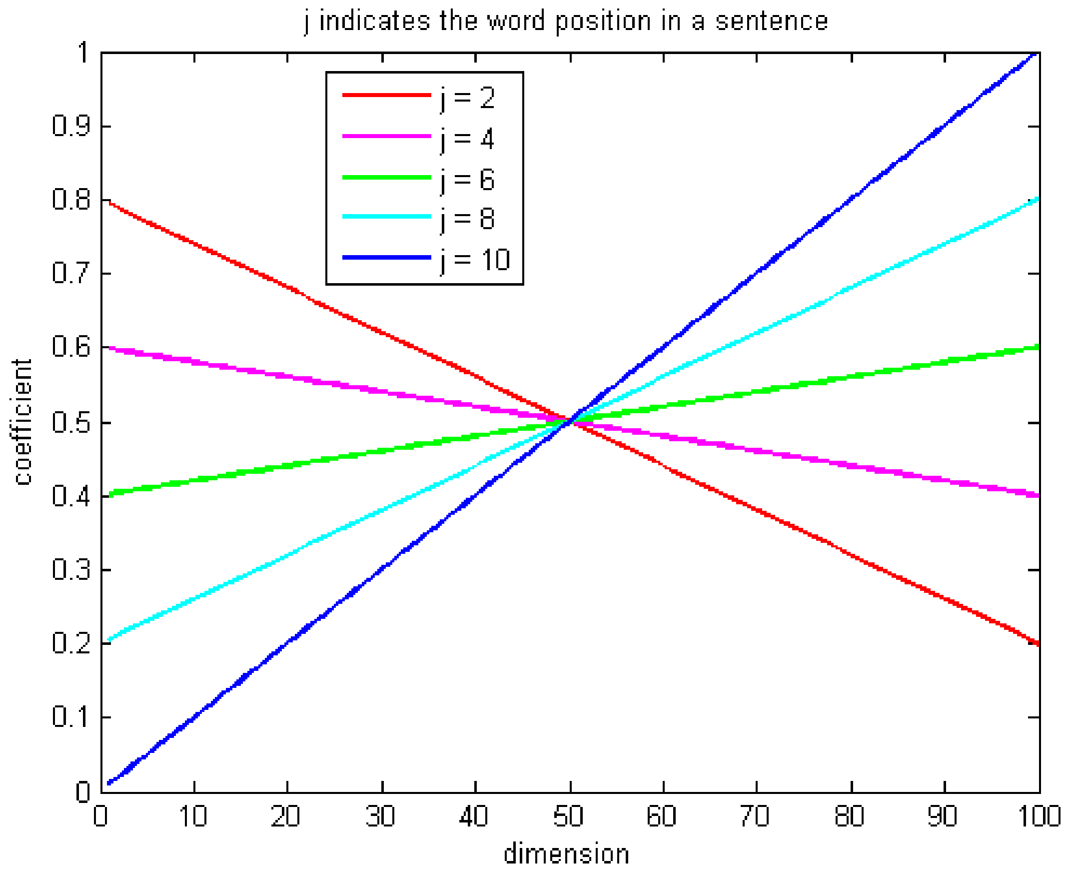
- Position encoding other than BoW:

$$x_i = \{x_{i1}, \dots, x_{iJ_i}\}$$

$$\left(m_i = \sum_j A x_{ij} \right)$$

$$m_i = \sum_j l_j \cdot A x_{ij}$$

$$l_j \in R^d, l_{jk} = \left(1 - \frac{j}{J_i}\right) - \left(\frac{k}{d}\right)\left(1 - \frac{2j}{J_i}\right)$$



$d = 100, J_i = 10$

Model details

- Temporal Encoding:

$$m_i = \sum_j Ax_{ij} + T_A(i)$$

- Random Noise:

Randomly add 10% of empty memories to the stories when training

- Linear Start:

At the beginning of training, remove the softmax layer when calculating p_i. When the validation loss stopped decreasing, the softmax layers were re-inserted.

Experiments

Training on 1k stories

TASK	N-grams	LSTMs	MemN2N	Memory Networks
T1. Single supporting fact	36	50	PASS	PASS
T2. Two supporting facts	2	20	87	PASS
T3. Three supporting facts	7	20	60	PASS
T4. Two arguments relations	50	61	PASS	PASS
T5. Three arguments relations	20	70	87	PASS
T6. Yes/no questions	49	48	92	PASS
T7. Counting	52	49	83	85
T8. Sets	40	45	90	91
T9. Simple negation	62	64	87	PASS
T10. Indefinite knowledge	45	44	85	PASS
T11. Basic coreference	29	72	PASS	PASS
T12. Conjunction	9	74	PASS	PASS
T13. Compound coreference	26	PASS	PASS	PASS
T14. Time reasoning	19	27	PASS	PASS
T15. Basic deduction	20	21	PASS	PASS
T16. Basic induction	43	23	PASS	PASS
T17. Positional reasoning	46	51	49	65
T18. Size reasoning	52	52	89	PASS
T19. Path finding	0	8	7	36
T20. Agent's motivation	76	91	PASS	PASS

Experiments

- The position encoding representation improves over bag-of-words on tasks where word ordering is particularly important.
- The linear start to training seems to help avoid local minima.
- Random empty memories gives a small but consistent boost in performance, especially for the smaller 1k training set.
- Joint training on all tasks helps.
- More computational hops give improved performance.

Attention during memory lookups

Story (1: 1 supporting fact)	Support	Hop 1	Hop 2	Hop 3
Daniel went to the bathroom.		0.00	0.00	0.03
Mary travelled to the hallway.		0.00	0.00	0.00
John went to the bedroom.		0.37	0.02	0.00
John travelled to the bathroom.	yes	0.60	0.98	0.96
Mary went to the office.		0.01	0.00	0.00
Where is John? Answer: bathroom Prediction: bathroom				

Story (2: 2 supporting facts)	Support	Hop 1	Hop 2	Hop 3
John dropped the milk.		0.06	0.00	0.00
John took the milk there.	yes	0.88	1.00	0.00
Sandra went back to the bathroom.		0.00	0.00	0.00
John moved to the hallway.	yes	0.00	0.00	1.00
Mary went back to the bedroom.		0.00	0.00	0.00
Where is the milk? Answer: hallway Prediction: hallway				

Story (16: basic induction)	Support	Hop 1	Hop 2	Hop 3
Brian is a frog.	yes	0.00	0.98	0.00
Lily is gray.		0.07	0.00	0.00
Brian is yellow.	yes	0.07	0.00	1.00
Julius is green.		0.06	0.00	0.00
Greg is a frog.	yes	0.76	0.02	0.00
What color is Greg? Answer: yellow Prediction: yellow				

Story (18: size reasoning)	Support	Hop 1	Hop 2	Hop 3
The suitcase is bigger than the chest.	yes	0.00	0.88	0.00
The box is bigger than the chocolate.		0.04	0.05	0.10
The chest is bigger than the chocolate.	yes	0.17	0.07	0.90
The chest fits inside the container.		0.00	0.00	0.00
The chest fits inside the box.		0.00	0.00	0.00
Does the suitcase fit in the chocolate? Answer: no Prediction: no				

Language Modeling

The goal is to predict the next word in a text sequence given the previous words. Results on the Penn Treebank and Text8 (Wikipedia-based) corpora.

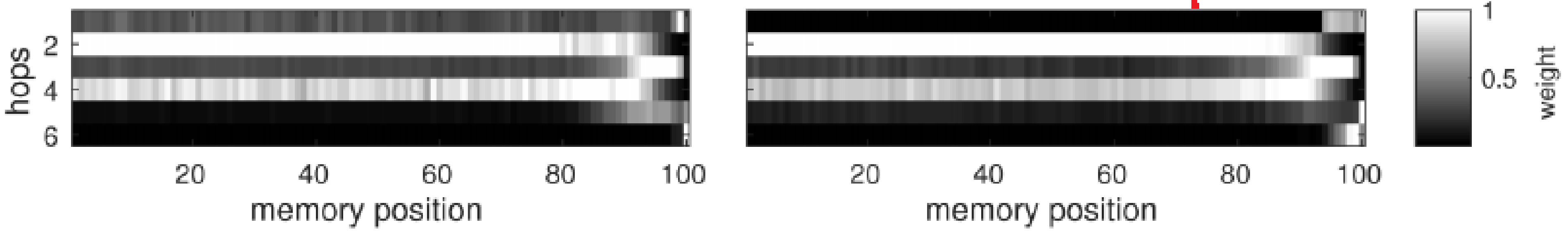
	Penn Tree	Text8
RNN	129	184
LSTM	115	154
MemN2N 2 hops	121	187
5 hops	118	154
7 hops	111	147

Test perplexity

periodic

Hops vs. Attention:
Average over (PTB)

Average over (Text8)



Q & A

Task	Baseline			MemN2N								
	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