Deep Reasoning

2016-03-16

Taehoon Kim
carpedm20@gmail.com

References

1. **[Sukhbaatar, 2015]** Sukhbaatar, Szlam, **Weston**, Fergus. *"End-To-End Memory Networks"* Advances in Neural Information Processing Systems. 2015.
2. **[Hill, 2015]** Hill, Bordes, Chopra, **Weston**. *"The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations" arXiv preprint arXiv:1511.02301* (2015).
3. **[Kumar, 2015]** Kumar, Irsoy, Ondruska, Iyyer, Bradbury, Gulrajani, Zhong, Paulus, **Socher**. "*Ask Me Anything: Dynamic Memory Networks for Natural Language Processing" arXiv preprint arXiv:1511.06038* (2015).
4. **[Xiong, 2016]** Xiong, Merity, **Socher**. "*Dynamic Memory Networks for Visual and Textual Question Answering" arXiv preprint arXiv:1603.01417* (2016).
5. **[Yin, 2015]** Yin, Schütze, Xiang, **Zhou**. *"ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs" arXiv preprint arXiv:1512.05193* (2015).
6. **[Yu, 2015]** Yu, Zhang, Hang, Xiang, **Zhou**. *"Empirical Study on Deep Learning Models for Question Answering" arXiv preprint arXiv:1510.07526* (2015).
7. **[Hermann, 2015]** Hermann, Kočiský, Grefenstette, Espeholt, Will Kay, Suleyman, Blunsom. *"Teaching Machines to Read and Comprehend" arXiv preprint arXiv:1506.03340* (2015).
8. **[Kadlec, 2016]** Kadlec, Schmid, Bajgar, Kleindienst. *"Text Understanding with the Attention Sum Reader Network" arXiv preprint arXiv:1603.01547* (2016).

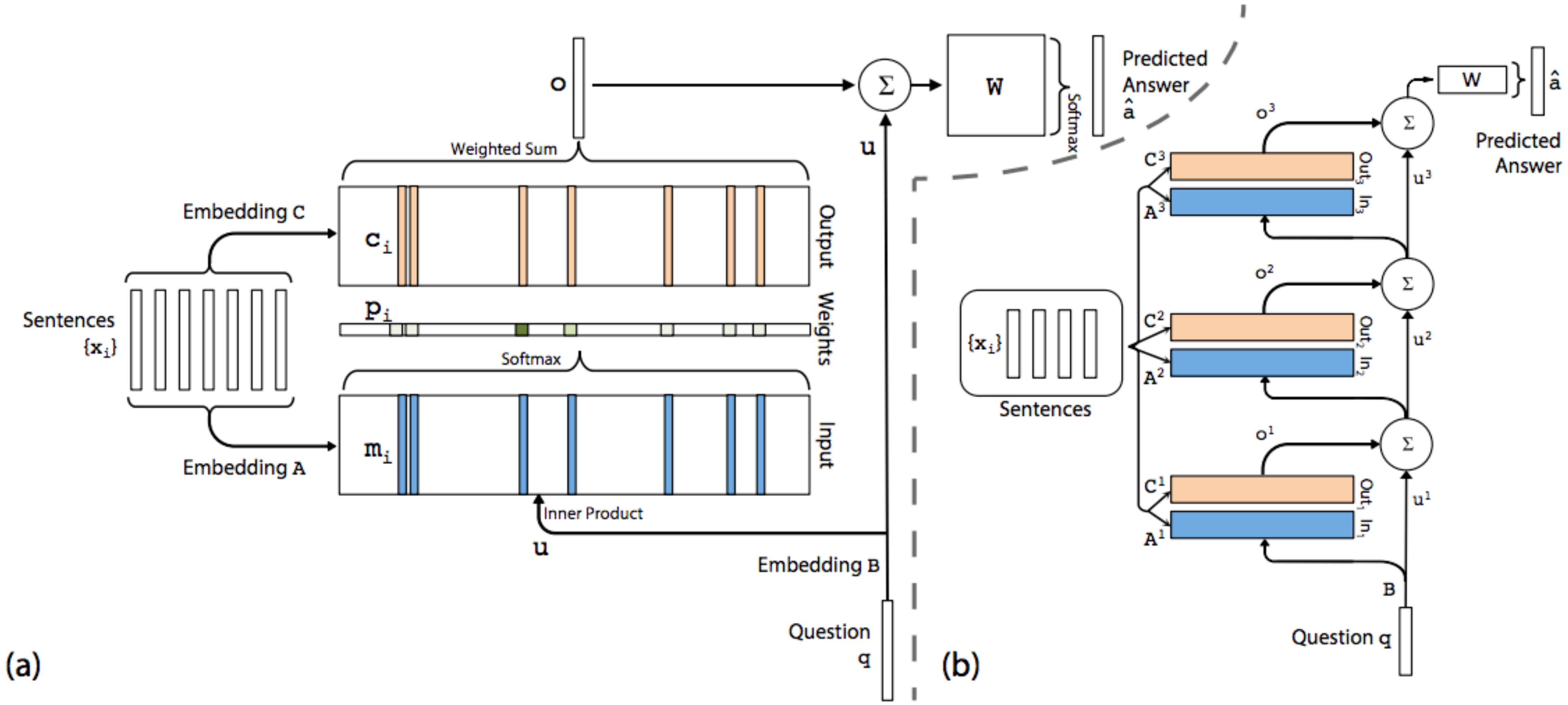
References

9. **[Miao, 2015]** Miao, Lei Yu, Blunsom. *“Neural Variational Inference for Text Processing” arXiv preprint arXiv:1511.06038* (2015).
10. **[Kingma, 2013]** Kingma, Diederik P., and Max Welling. *"Auto-encoding variational bayes" arXiv preprint arXiv:1312.6114* (2013).
11. **[Sohn, 2015]** Sohn, Kihyuk, Honglak Lee, and Xinchen Yan. *"Learning Structured Output Representation using Deep Conditional Generative Models."* Advances in Neural Information Processing Systems. 2015.

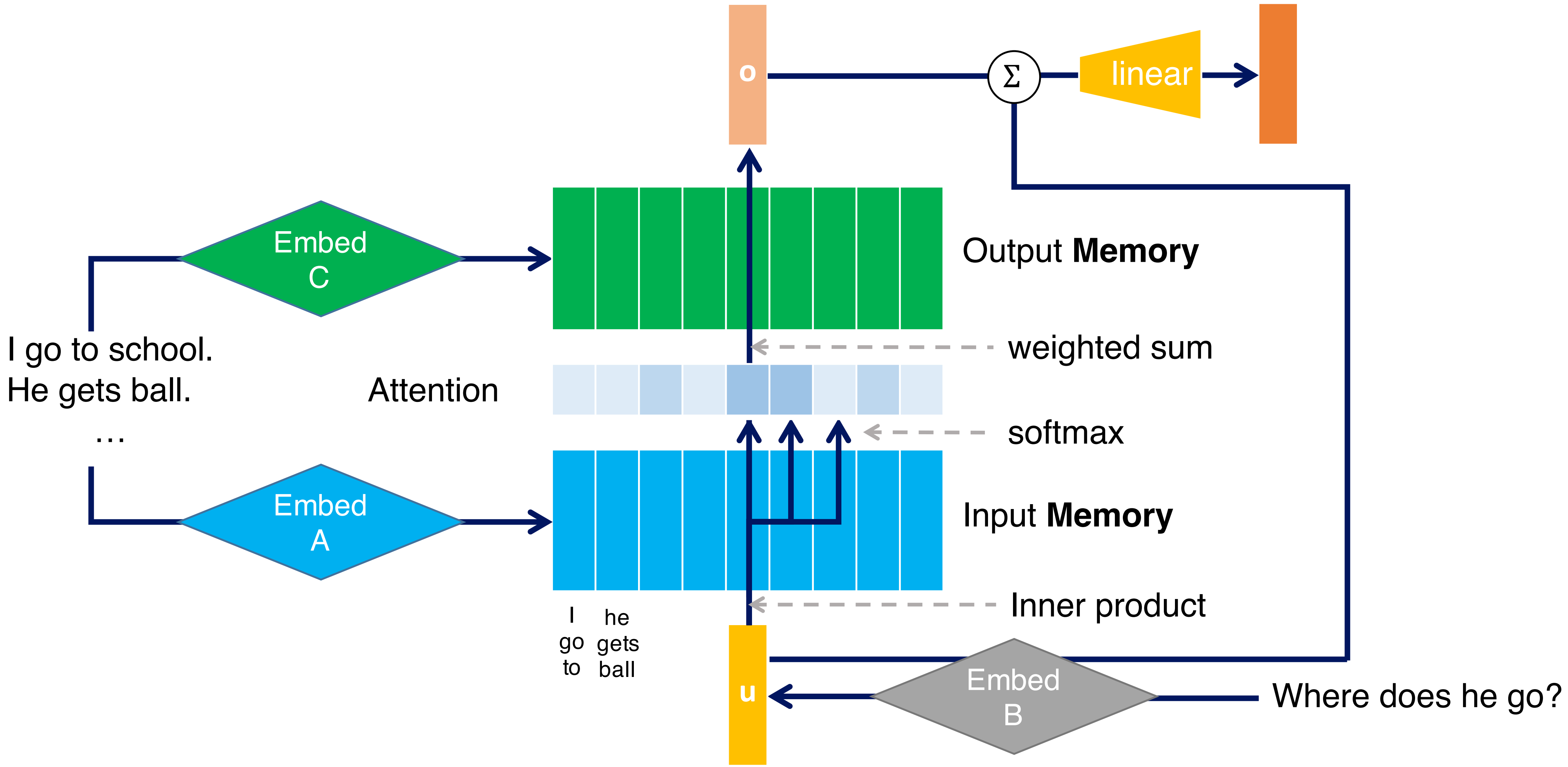
Models

Answer selection (WikiQA)	General QA (CNN)	Considered transitive inference (bAbI)
ABCNN	E2E MN	E2E MN
Variational	Impatient Attentive Reader	DMN
Attentive Pooling	Attentive (Impatient) Reader	ReasoningNet
	Attention Sum Reader	NTM

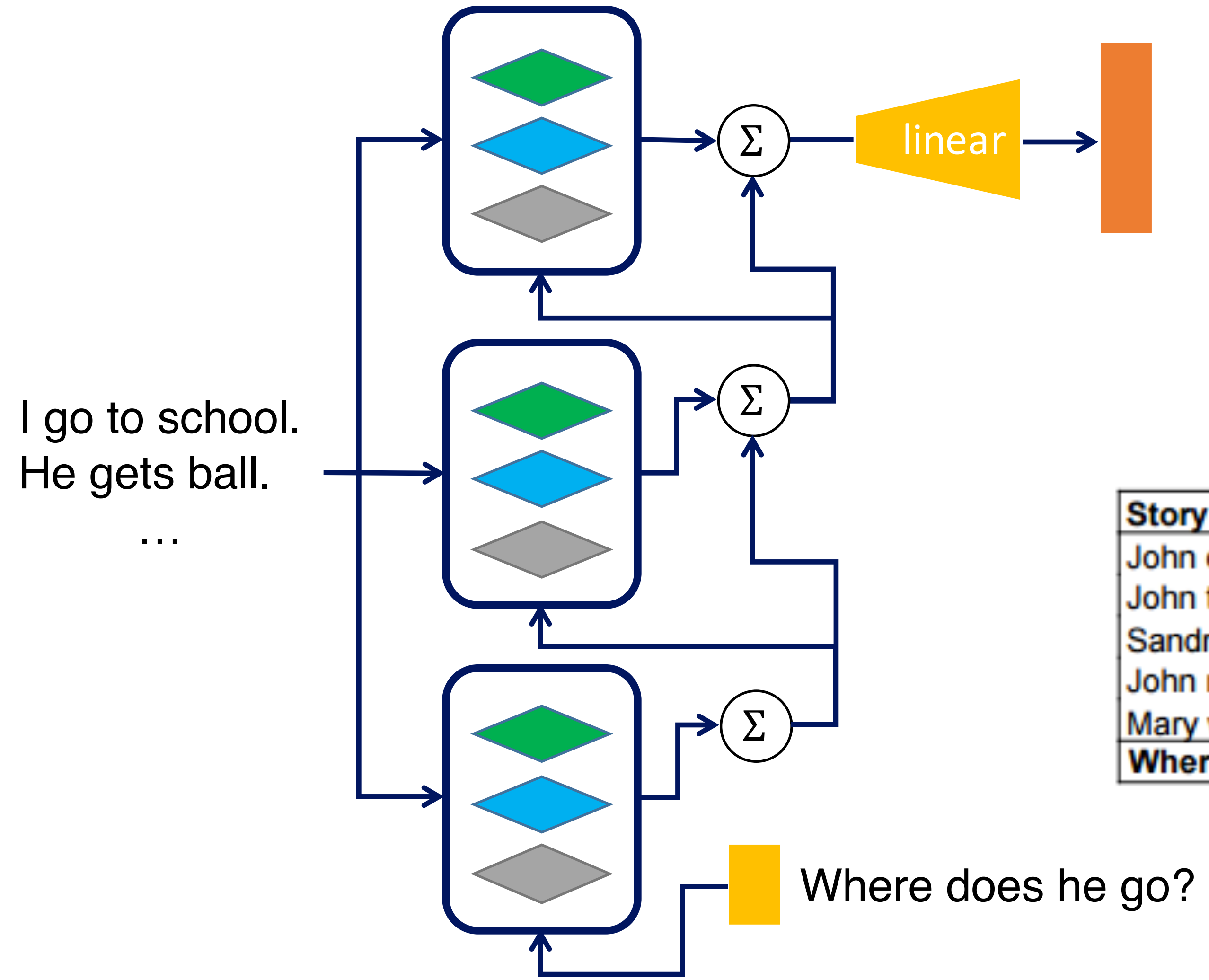
End-to-End Memory Network [Sukhbaatar, 2015]



End-to-End Memory Network [Sukhbaatar, 2015]



End-to-End Memory Network [Sukhbaatar, 2015]



Sentence representation :

i th sentence : $x_i = \{x_{i1}, x_{i2}, \dots, x_{in}\}$

BoW : $m_i = \sum_j Ax_{ij}$

Position Encoding : $m_i = \sum_j l_j \cdot Ax_{ij}$

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

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				

Training details

Linear Start (LS) help avoid local minima

- First train with softmax in each memory layer removed, making the model entirely linear except for the final softmax
- When the validation loss stopped decreasing, the softmax layers were re-inserted and training recommenced

RNN-style layer-wise weight tying

- The input and output embeddings are the same across different layers

Learning **time invariance** by injecting random noise

- Jittering the time index with random empty memories
- Add "dummy" memories to regularize $T_A(i)$

Example of bAbI tasks

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				

The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations [Hill, 2016]

S: 1 Mr. Cropper was opposed to our hiring you .
2 Not , of course , that he had any personal objection to you , but he is set against female teachers , and when a Cropper is set there is nothing on earth can change him .
3 He says female teachers ca n't keep order .
4 He 's started in with a spite at you on general principles , and the boys know it .
5 They know he 'll back them up in secret , no matter what they do , just to prove his opinions .
6 Cropper is sly and slippery , and it is hard to corner him . ''
7 `` Are the boys big ? ''
8 queried Esther anxiously .
9 `` Yes .
10 Thirteen and fourteen and big for their age .
11 You ca n't whip 'em -- that is the trouble .
12 A man might , but they 'd twist you around their fingers .
13 You 'll have your hands full , I 'm afraid .
14 But maybe they 'll behave all right after all . ''
15 Mr. Baxter privately had no hope that they would , but Esther hoped for the best.
16 She could not believe that Mr. Cropper would carry his prejudices into a personal application .
17 This conviction was strengthened when he overtook her walking from school the next day and drove her home .
18 He was a big , handsome man with a very suave , polite manner .
19 He asked interestedly about her school and her work , hoped she was getting on well , and said he had two young rascals of his own to send soon .
20 Esther felt relieved .

q: She thought that Mr. ________ had exaggerated matters a little .

C: Baxter, Cropper, Esther, course, fingers, manner, objection, opinion, right, spite.

a: Baxter

The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations [Hill, 2016]

- Context sentences : $S = \{s_1, s_2, \ldots, s_n\}$, s_i : BoW word representation
- Encoded memory : $\mathrm{m_i} = \phi(s)\ \forall s \in S$

- Lexical memory
  - Each word occupies a separate slot in the memory
  - s is a single word and $\phi(s)$ has only one non-zero feature
  - Multiple hop only beneficial in this memory model
- **Window memory (best)**
  - s corresponds to a window of text from the context S centered on an individual mention of a candidate c in S

$$\mathrm{m_i} = \left\{w_{i-(b-1)/2} \ \ldots\ w_i\ \ldots w_{i+(b-1)/2}\right\}$$

  - Where $w_i \in C$ which is an instance of one of the candidate words
- Sentential memory
  - Same as original implementation of Memory Network

The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations [Hill, 2016]

Self-supervision for window memories

- Memory supervision (knowing which memories to attend to) is not provided at training time
- Making gradient steps using SGD to **force** the model to give a **higher score to the supporting memory $\tilde{m}$** relative to any other memory from any other candidate using:

Hard attention (training and testing) : $m_{o1} = \underset{i=1,\ldots,n}{\operatorname{argmax}} c_i^T q$

Soft attention (testing) : $m_{o1} = \sum_{i=1\ldots n} \alpha_i m_i , with\ \alpha_i = \frac{e^{c_i^T q}}{\sum_j e^{c_i^T q}}$

- If m_{o1} happens to be different from $\tilde{m}$ (memory contain true answer), then model is updated
- Can be understood as **a way of achieving *hard attention* over memories** (no need any new label information beyond the training data)

The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations [Hill, 2016]

S: 1 So they had to fall a long way .
2 So they got their tails fast in their mouths .
3 So they could n't get them out again .
4 That 's all . '
5 ` Thank you , ' said Alice , ` it 's very interesting .
6 I never knew so much about a whiting before . ' '
7 I can tell you more than that , if you like , ' said the Gryphon .
8 ` Do you know why it 's called a whiting ? ''
9 I never thought about it , ' said Alice .
10 ` Why ? '
11 ` IT DOES THE BOOTS AND SHOES . '
12 the Gryphon replied very solemnly .
13 Alice was thoroughly puzzled .
14 ` Does the boots and shoes ! '
15 she repeated in a wondering tone .
16 ` Why , what are YOUR shoes done with ? '
17 said the Gryphon . '
18 I mean , what makes them so shiny ? '
19 Alice looked down at them , and considered a little before she gave her answer .
20 They 're done with blacking , I believe .

q: `Boots and shoes under the sea , ' the _______ went on in a deep voice , are done with a whiting .

C: Alice, BOOTS, Gryphon, SHOES, answer, fall, mouths, tone, way, whiting.

MemNNs (window + self-sup.): **Gryphon**

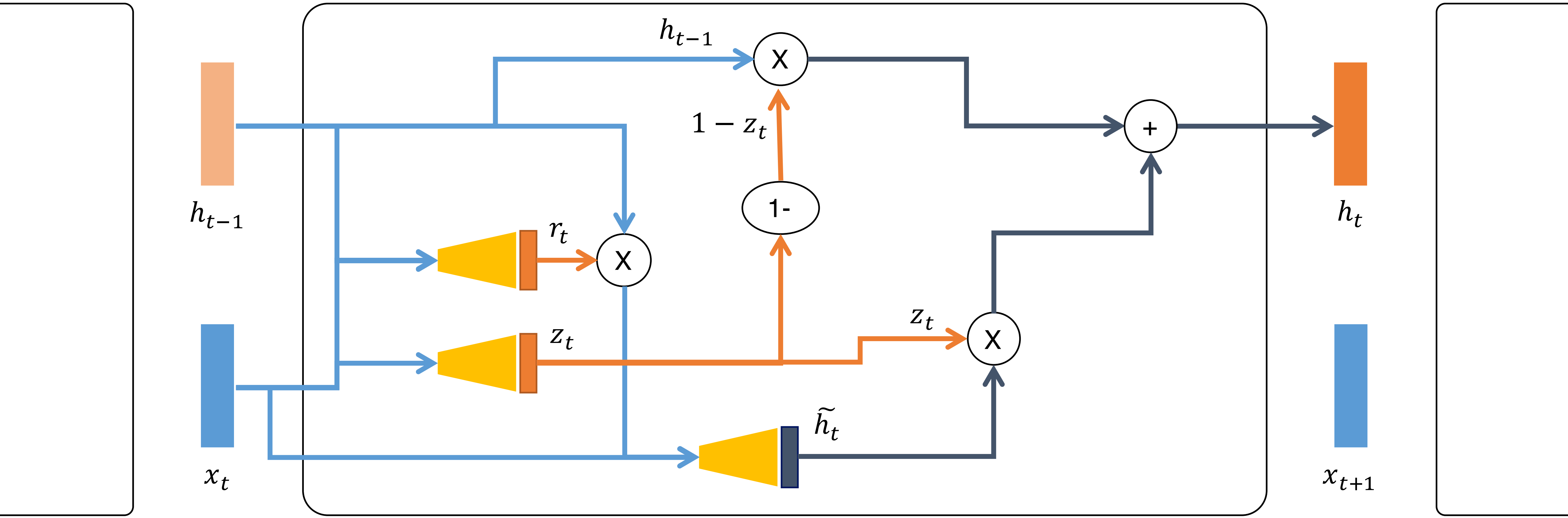
S: 1 He thought that Old Mr. Toad was trying to fool him .
2 Presently Peter Rabbit came along .
3 He found Jimmy Skunk sitting in a brown study .
4 He had quite forgotten to look for fat beetles , and when he forgets to do that you may make up your mind that Jimmy is doing some hard thinking .
5 `` Hello , old Striped-coat , what have you got on your mind this fine morning ? ''
6 cried Peter Rabbit .
7 `` Him , '' said Jimmy simply , pointing down the Lone Little Path .
8 Peter looked .
9 `` Do you mean Old Mr. Toad ! ''
10 he asked .
11 Jimmy nodded .
12 `` Do you see anything queer about him ? ''
13 he asked in his turn .
14 `` Do you see anything queer about him ? ''
15 he asked .
16 Peter stared down the Lone Little Path .
17 `` No , '' he replied , `` except that he seems in a great hurry . ''
18 `` That 's just it , '' Jimmy returned promptly .
19 `` Did you ever see him hurry unless he was frightened ? ''
20 Peter confessed that he never had

q: `` Well , he is n't _______ now , yet just look at him go '' retorted Jimmy .

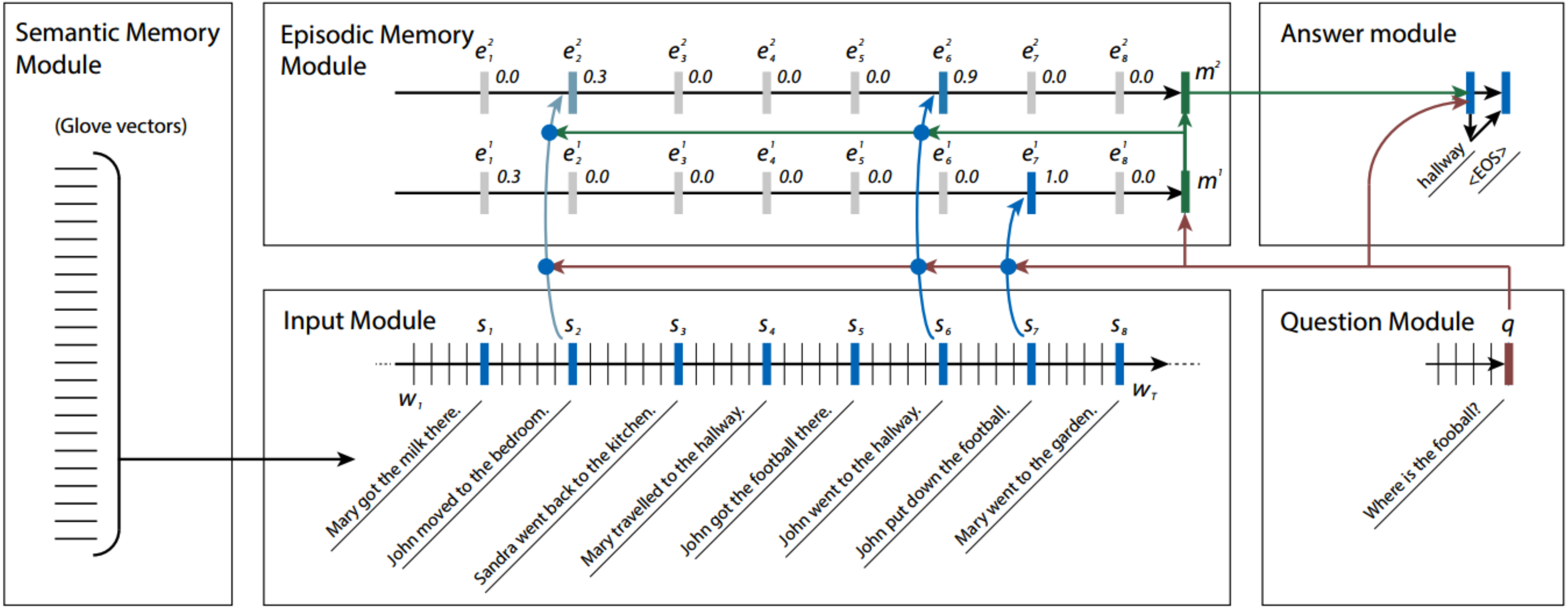
C: Do, came, confessed, frightened, mean, replied, returned, said, see, thought.

MemNNs (window +self-sup.): **frightened**

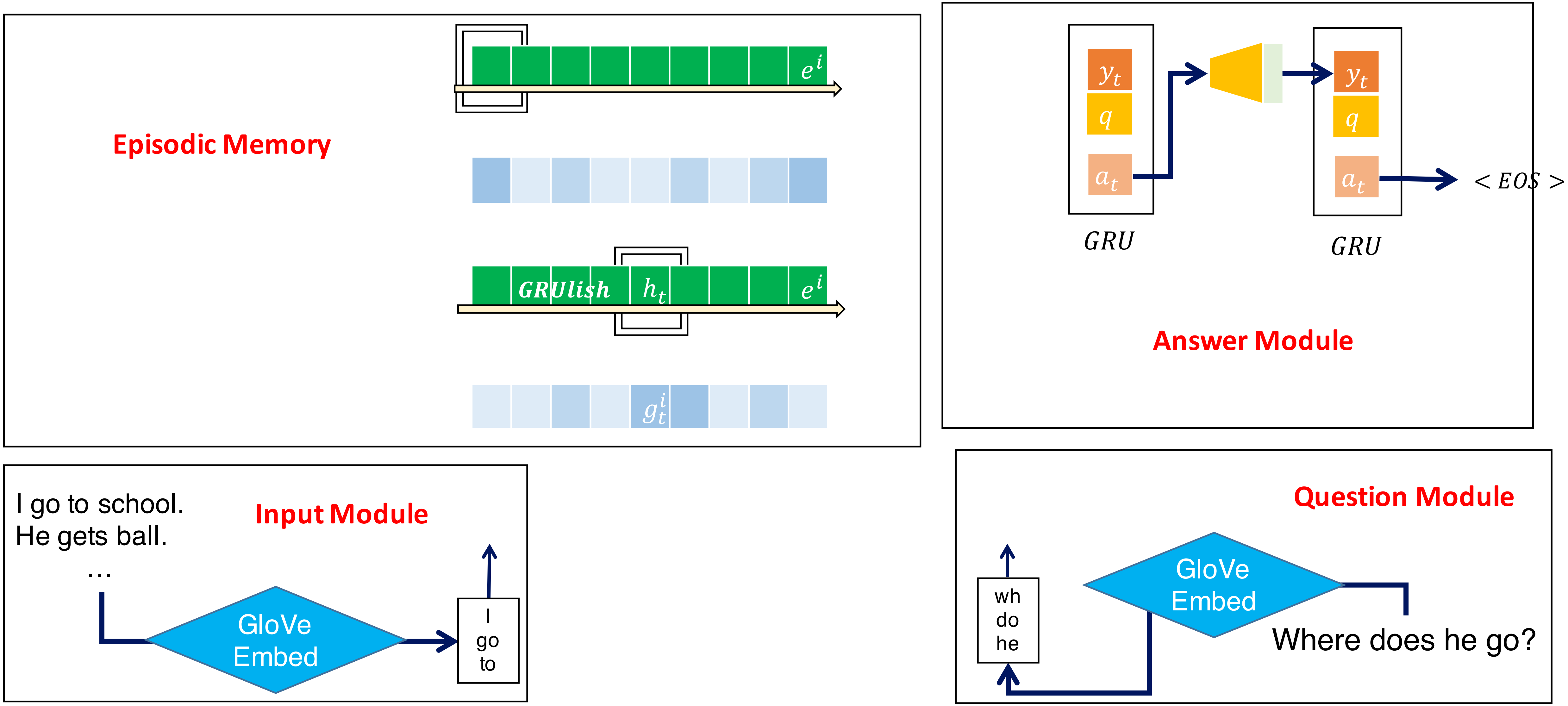
Gated Recurrent Network (GRU)



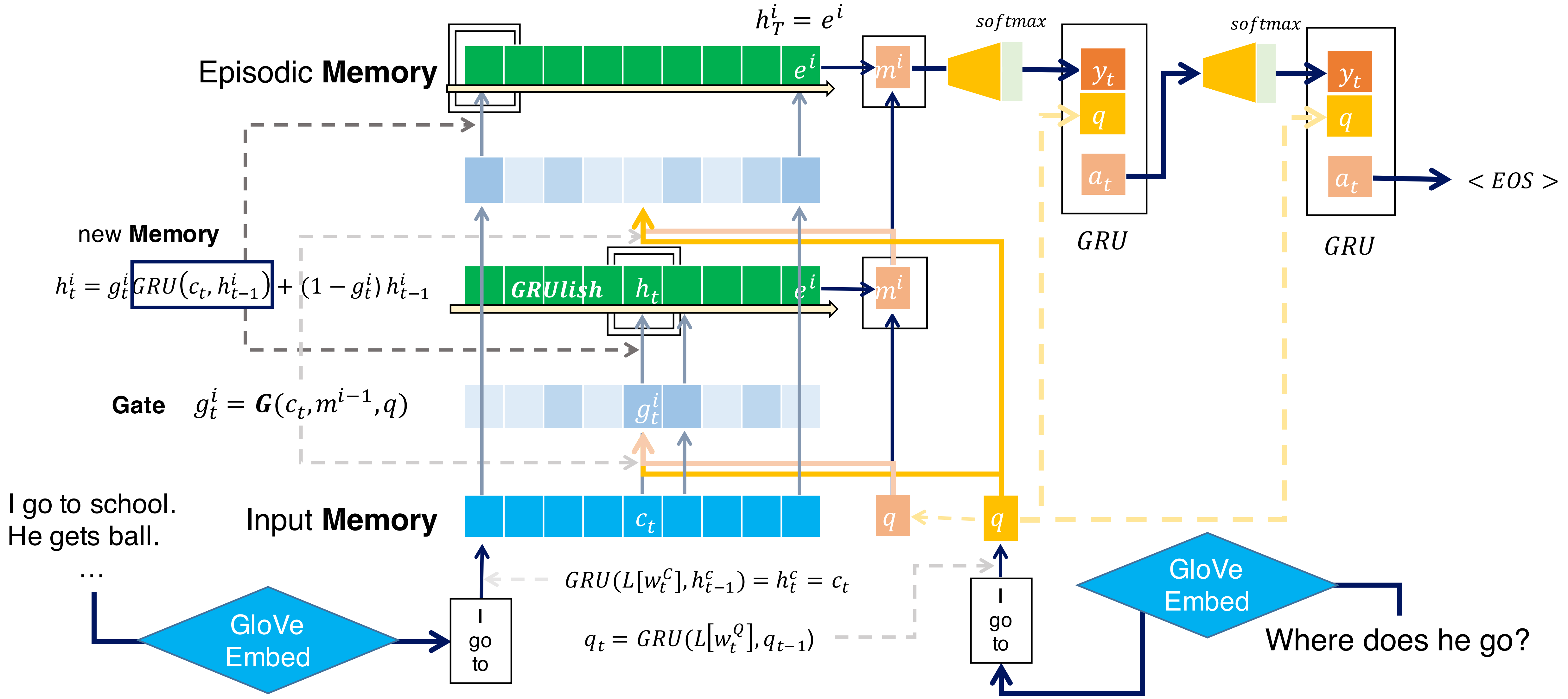
Ask Me Anything: Dynamic Memory Networks for Natural Language Processing [Kumar, 2015]



Ask Me Anything: Dynamic Memory Networks for Natural Language Processing [Kumar, 2015]



Ask Me Anything: Dynamic Memory Networks for Natural Language Processing [Kumar, 2015]



Ask Me Anything: Dynamic Memory Networks for Natural Language Processing [Kumar, 2015]

G : two-layer feed forward neural network

$$G(c,m,q) =$$

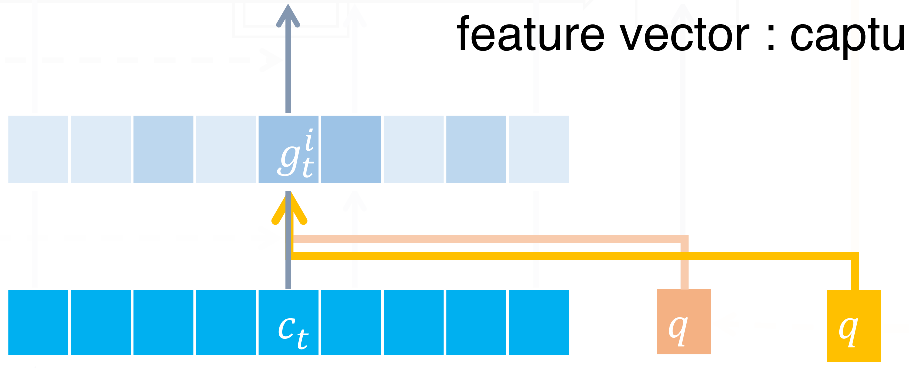
$$\sigma\left(W^{(2)}\tanh\left(W^{(1)}\boxed{z(c,m,q)}+b^{(1)}\right)+b^{(2)}\right)$$

$$\left[c, m, q, c \circ q, c \circ m, |c-q|, |c-m|, c^T W^{(b)} q, c^T W^{(b)} m\right]$$

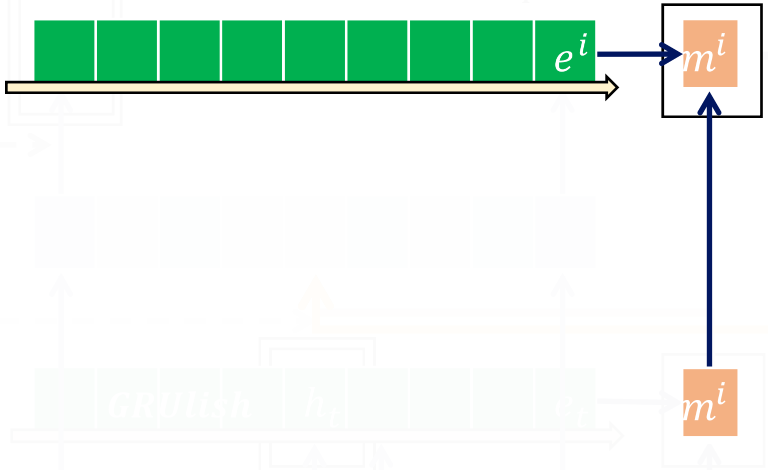
feature vector : captures a similarities between c, m, q

Attention Mechanism

Gate $g_t^i = \boldsymbol{G}(c_t, m^{i-1}, q)$



Ask Me Anything: Dynamic Memory Networks for Natural Language Processing [Kumar, 2015]



$$m^i = GRU(e^i, m^{i-1})$$

Memory update

new **Memory**

$$h_t^i = g_t^i \boxed{GRU(c_t, h_{t-1}^i)} + (1 - g_t^i)\, h_{t-1}^i$$

$$e^i = h_{T_C}^i$$

Episodic memory update

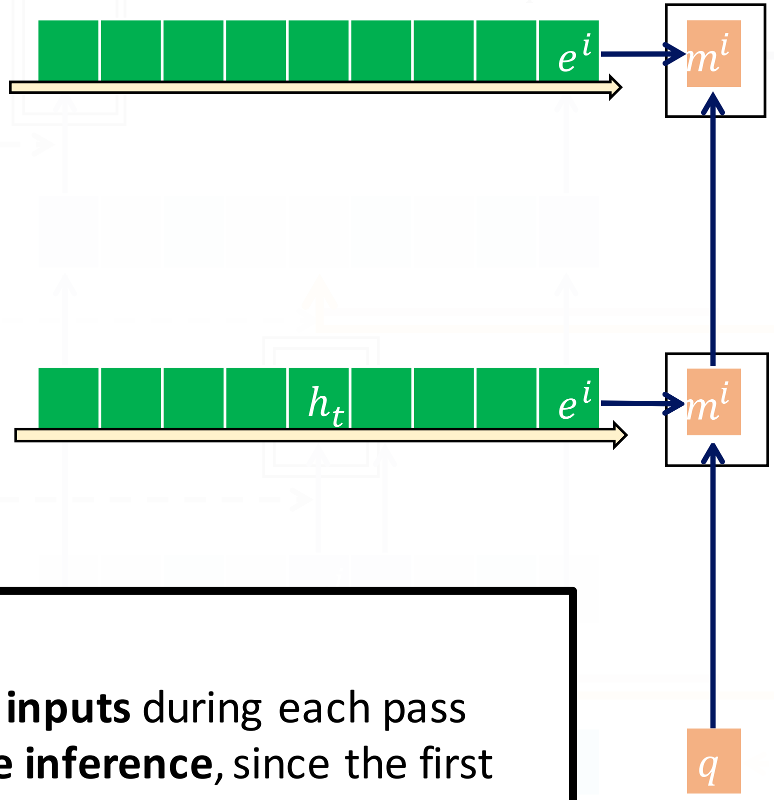
Episodic Memory Module

- **Iterates** over **input** representations, while updating episodic memory $\boldsymbol{e^i}$
- Attention mechanism + Recurrent network → Update memory $\boldsymbol{m^i}$

Ask Me Anything: Dynamic Memory Networks for Natural Language Processing [Kumar, 2015]

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

Table 4. Effectiveness of episodic memory module across tasks. Each row shows the final accuracy in term of percentages with a different maximum limit for the number of passes the episodic memory module can take. Note that for the 0-pass DMN, the network essential reduces to the output of the attention module.



Criteria for Stopping

- Append a special end-of-passes representation to the input $\boldsymbol{c}$
- Stop if this representation is **chosen** by the **gate** function
- Set a maximum number of iterations
- This is why called **Dynamic** MM

Multiple Episodes

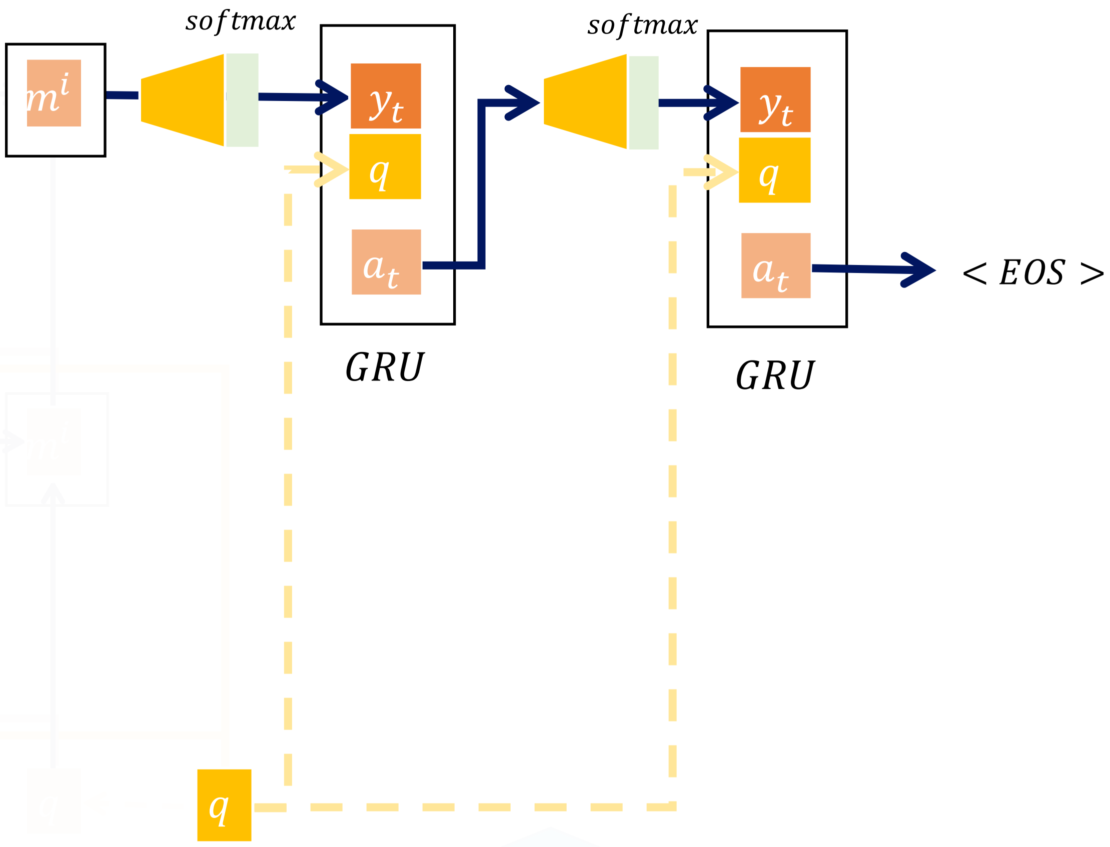
- Allows to **attend** to **different inputs** during each pass
- Allows for a type of **transitive inference**, since the first pass may uncover the need to retrieve additional facts.

Q : Where is the football?
C1 : John put down the football.

Only once the model sees C1, John is relevant, can reason that the second iteration should retrieve where John was.

Ask Me Anything: Dynamic Memory Networks for Natural Language Processing [Kumar, 2015]

$$y_t = \text{softmax}(W^{(a)} a_t)$$
$$a_t = GRU([y_{t-1}, q], a_{t-1}),$$



Answer Module

- **Triggered** once at **the end of the episodic memory** or at each time step
- Concatenate the **last** generated **word** and the **question** vector as the input at each time step
- **Cross-entropy error**

Training Details

- Adam optimization
- L_2 regularization, dropout on the word embedding (GloVe)

bAbI dataset

- Objective function : $J = \alpha E_{CE}(Gates) + \beta E_{CE}(Answers)$
- **Gate supervision** aims to select **one sentence** per pass
  - Without supervision : GRU of c_t, h_t^i and $e^i = h_{T_C}^i$
  - With supervision (simpler) : $e^i = \sum_{t=1}^{T} softmax(g_t^i) c_t$, where $softmax(g_t^i) = \frac{\exp(g_t^i)}{\sum_{j=1}^{T} \exp(g_j^i)}$ and g_t^i is the value before sigmoid
  - Better results, because softmax encourages **sparsity** & suited to **picking one** sentence

Training Details

Stanford Sentiment Treebank (Sentiment Analysis)

- Use all full sentences, subsample 50% of phrase-level labels every epoch
- Only evaluated on the full sentences
- Binary classification, neutral phrases are removed from the dataset
- Trained with GRU sequence models

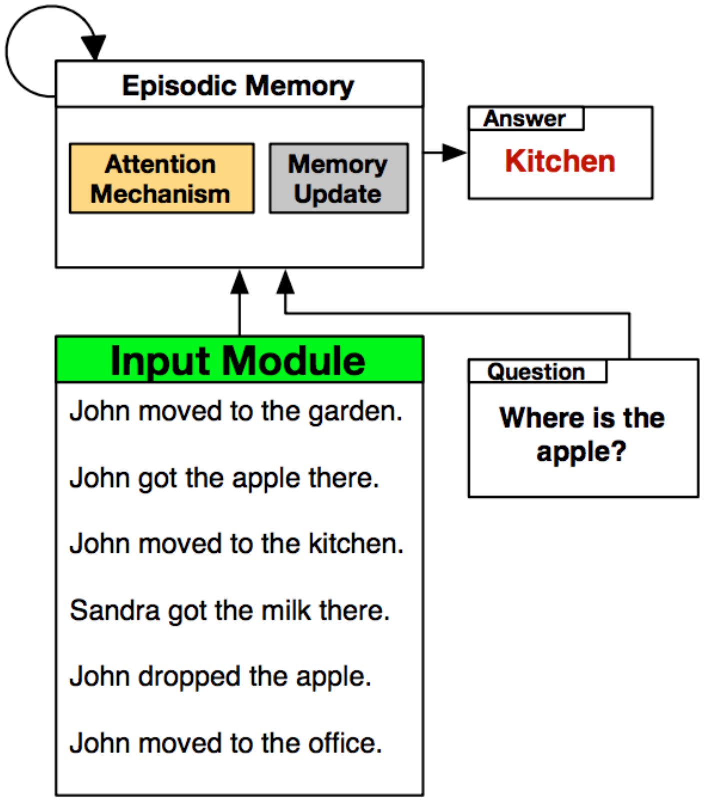
Task	Binary	Fine-grained
MV-RNN	82.9	44.4
RNTN	85.4	45.7
DCNN	86.8	48.5
PVec	87.8	48.7
CNN-MC	88.1	47.4
DRNN	86.6	49.8
CT-LSTM	88.0	51.0
DMN	**88.6**	**52.1**

Training Details

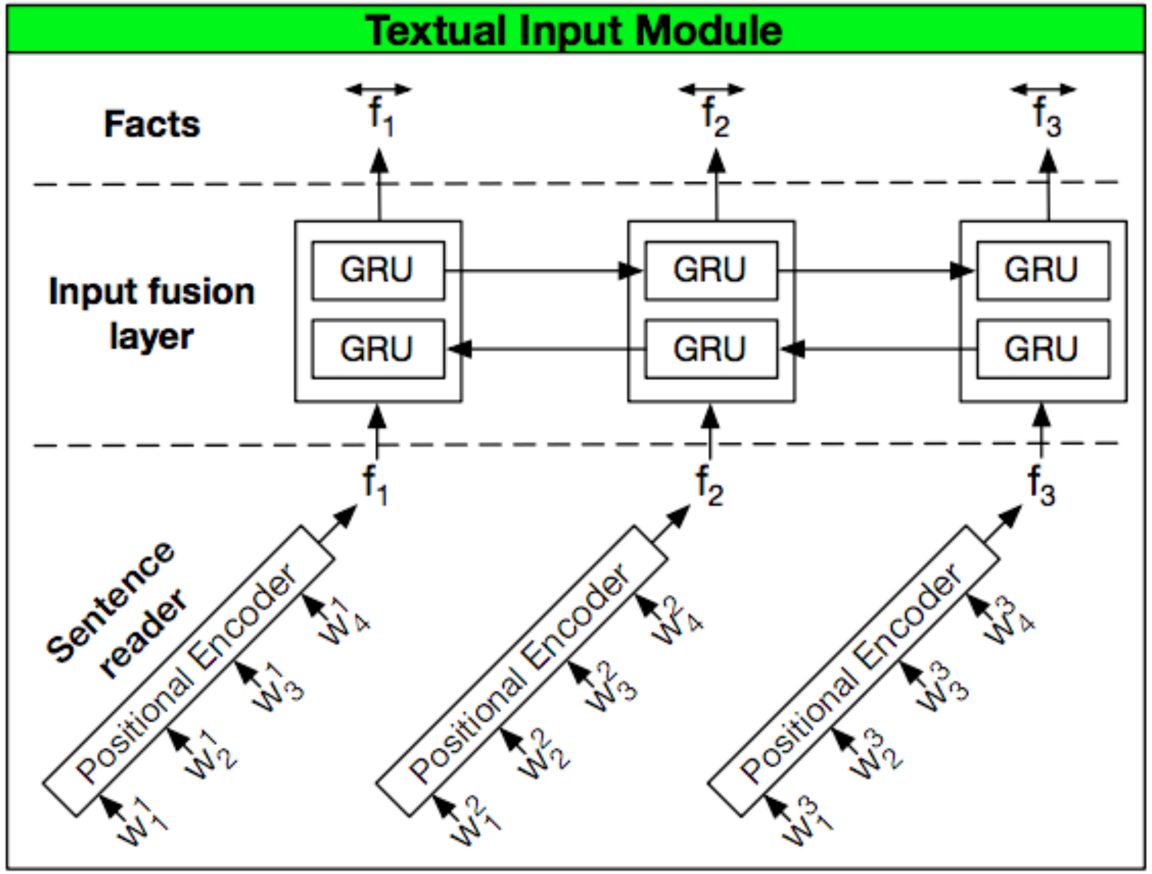
Question: Where was Mary before the Bedroom?
Answer: Cinema.

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]			

Dynamic Memory Networks for Visual and Textual Question Answering [Xiong 2016]



(a) Text Question-Answering

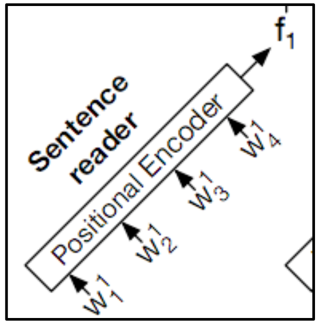


Several design choices are **motivated by intuition** and **accuracy improvements**

Input Module in DMN

- A single GRU for embedding story and store the hidden states
- GRU provides **temporal component** by allowing a sentence to know the **content of** the sentences that came **before them**
- **Cons:**
  - GRU only allows sentences to have context from sentences **before** them, but **not after them**
  - **Supporting sentences** may be too **far** away from each other
- Here comes **Input fusion** layer

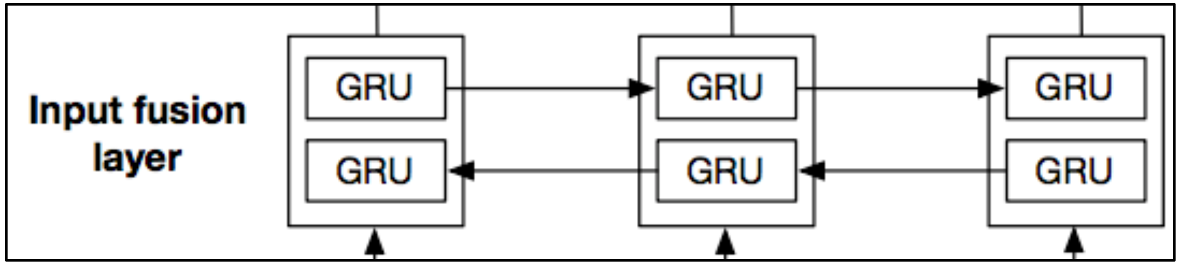
Input Module in DMN+



Replacing a single GRU with two different components

1. **Sentence reader** : responsible only for encoding the **words into a sentence embedding**
   - Use positional encoder (used in E2E) : $f_i = \sum_j l_j \cdot Ax_{ij}$
   - Considered GRUs LSTMs, but required more computational resources, prone to overfitting
2. **Input fusion layer** : interactions between sentences, allows **content interaction** between sentences
   - **bi-directional** GRU to allow information from both past and future sentences
   - gradients do not need to propagate through the words between sentences
   - **distant supporting sentences** can have a more **direct interaction**

$$\overrightarrow{f_i} = GRU_{fwd}(f_i, \overrightarrow{f_{i-1}})$$
$$\overleftarrow{f_i} = GRU_{bwd}(f_i, \overleftarrow{f_{i+1}})$$
$$\overleftrightarrow{f_i} = \overleftarrow{f_i} + \overrightarrow{f_i}$$



Input Module for DMN+

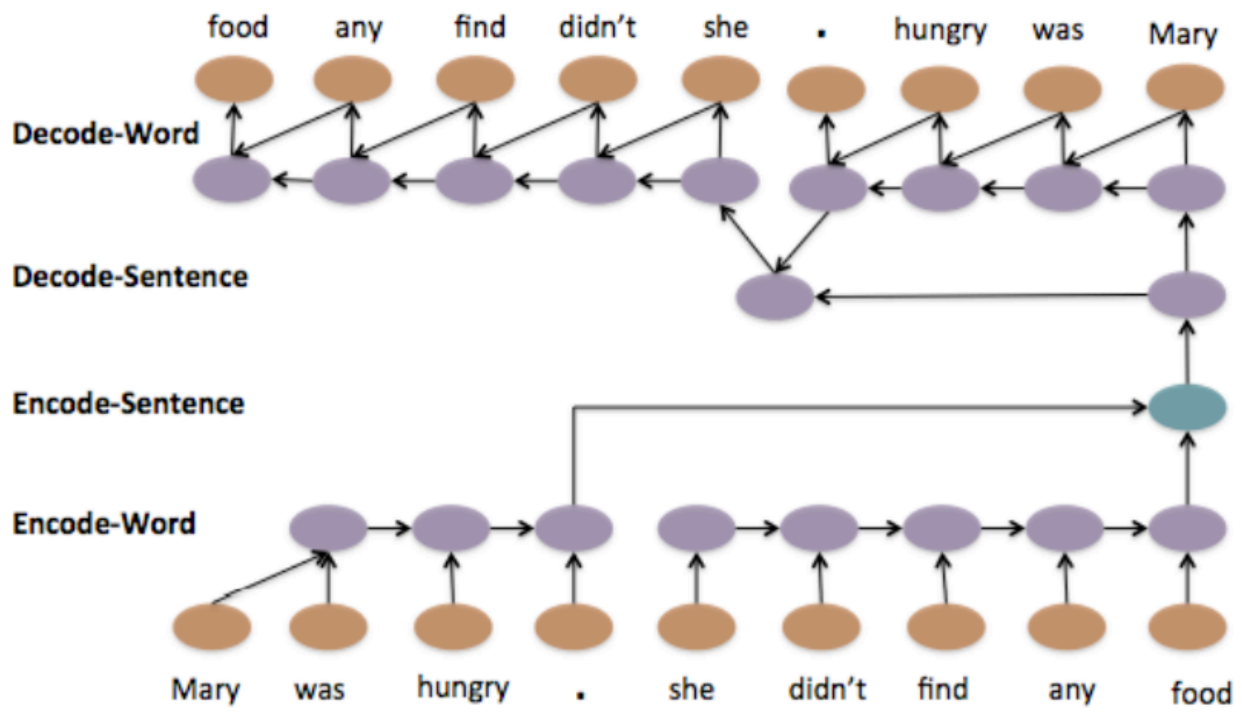
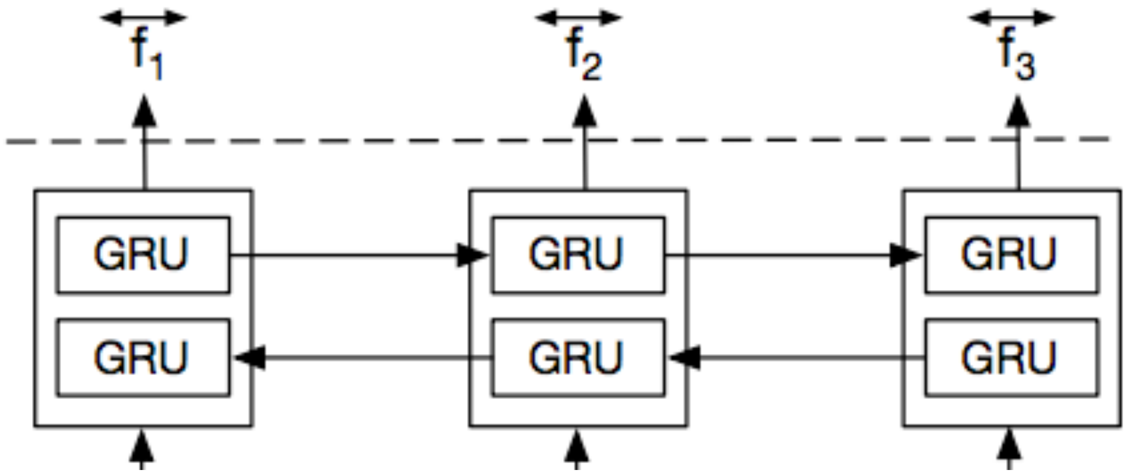


Figure 2: Hierarchical Sequence to Sequence Model.



Referenced paper : [A Hierarchical Neural Autoencoder for Paragraphs and Documents] [Li, 2015]

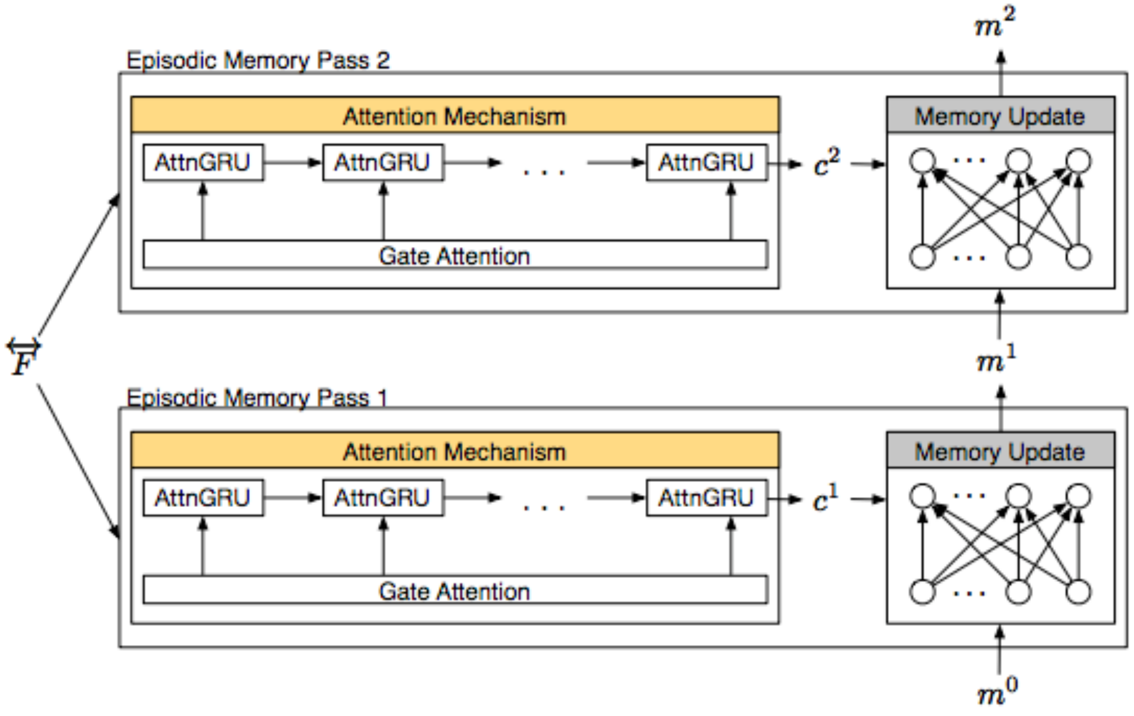
Episodic Memory Module in DMN+

- $\overleftrightarrow{F} = [\overleftrightarrow{f_1}, \overleftrightarrow{f_2}, \dots, \overleftrightarrow{f_N}]$: output of the input module
- Interactions between the fact $\overleftrightarrow{\boldsymbol{f}_i}$ and both the question $\boldsymbol{q}$ and episode memory state $\boldsymbol{m}^t$

$$z_i^t = [\overleftrightarrow{f_i} \circ q; \overleftrightarrow{f_i} \circ m^{t-1}; |\overleftrightarrow{f_i} - q|; |\overleftrightarrow{f_i} - m^{t-1}|]$$

$$Z_i^t = W^{(2)} \tanh\left(W^{(1)} z_i^t + b^{(1)}\right) + b^{(2)}$$

$$g_i^t = \frac{\exp(Z_i^t)}{\sum_{k=1}^{M_i} \exp(Z_k^t)}$$



Attention Mechanism in DMN+

Use attention to extract contextual vector c^t based on the current focus

1. Soft attention

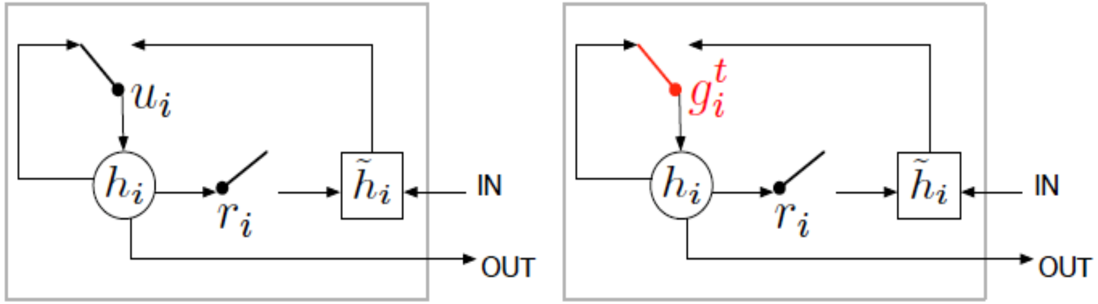
- A weighted summation of $\overleftrightarrow{F}$: $c^t = \sum_{i=1}^{N} g_i^t \overleftrightarrow{f_i}$
- Can approximate a hard attention by selecting a single fact $\overleftrightarrow{f_i}$
- Cons: losses positional and ordering information
    - Attention passes can retrieve some of this information, but inefficient

Attention Mechanism in DMN+

2. Attention based GRU (best)

- position and ordering information : RNN is proper but can't use g_i^t
- u_i: update, r_i: how much retain from h_{i-1}
- Replace u_i (vector) to g_i^t (scalar)
- Allows us to **easily visualize** how the attention gates activate
- Use final hidden state as $\boldsymbol{c_t}$, which is used to update episodic memory $\boldsymbol{m^t}$

$$\cancel{u_i = \sigma\left(W^{(u)}x_i + U^{(u)}h_{i-1} + b^{(u)}\right)}$$
$$r_i = \sigma\left(W^{(r)}x_i + U^{(r)}h_{i-1} + b^{(r)}\right)$$
$$\tilde{h}_i = \tanh\left(Wx_i + r_i \circ Uh_{i-1} + b^{(h)}\right)$$
$$h_i = \underbrace{u_i}_{g_i^t} \circ \tilde{h}_i + (1 - \underbrace{u_i}_{g_i^t}) \circ h_{i-1}$$



(a) (b)

Episode Memory Updates in DMT+

1. Untied and **Tied** (better) GRU

$$m^t = GRU(C^t, m^{t-1})$$

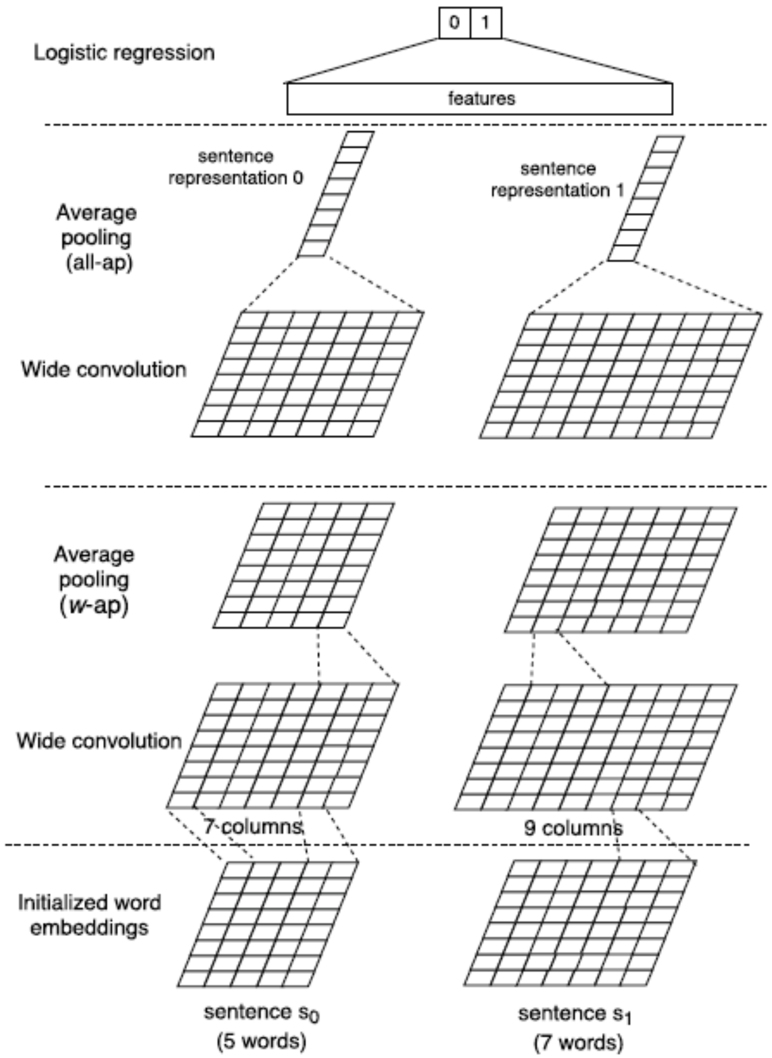
2. Untied ReLU layer (best)

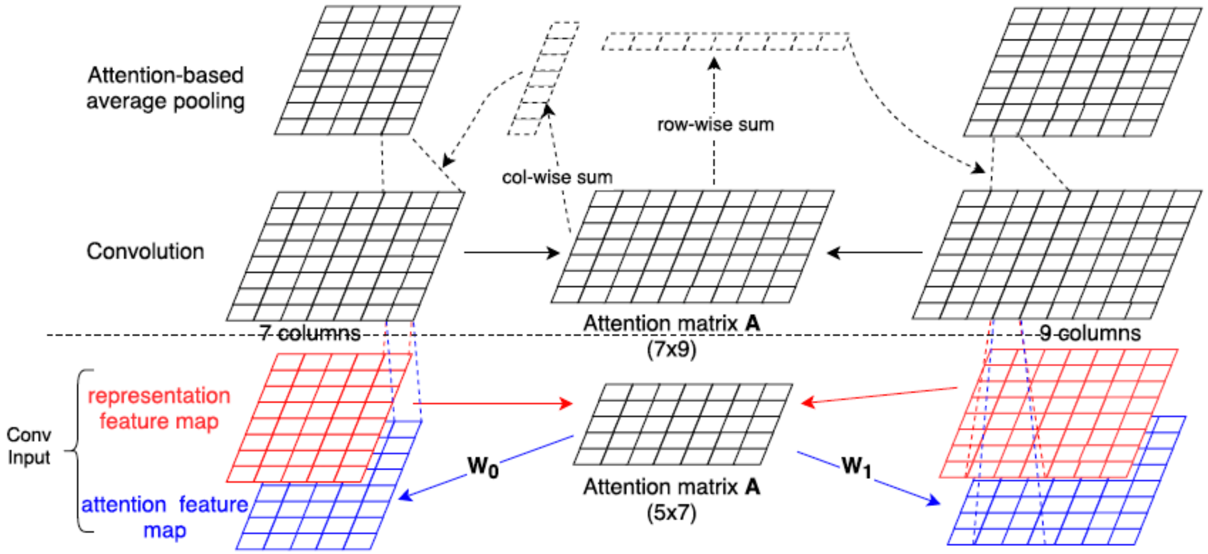
$$m^t = ReLU(W^t[m^{t-1}; c^t; q] + b)$$

Training Details

- Adam optimization
- **Xavier** initialization is used for all weights except for the word embeddings
- L_2 regularization on all weights except bias
- Dropout on the word embedding (GloVe) and answer module with $p = 0.9$

ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs [Yin 2015]





ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs [Yin 2015]

- Most prior work on answer selection model each sentence separately and neglects mutual influence
- Human **focus on key parts** of s_0 by extracting parts from s_1 related by identity, synonymy, antonym etc.
- **ABCNN** : taking into account the interdependence between s_0 and s_1
- Convolution layer : increase abstraction of a phrase from words

ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs [Yin 2015]

1. Input embedding with word2vec

2-1. Convolution layer with **wide convolution**

- To make each word v_i to be detected by all weights in W

2-2. Average pooling layer

- ***all-ap*** : column-wise averaging over all columns
- ***w-ap*** : column-wise averaging over windows of w

3. Output layer with logistic regression

- Forward all-ap to all non-final ap layer + final ap layer

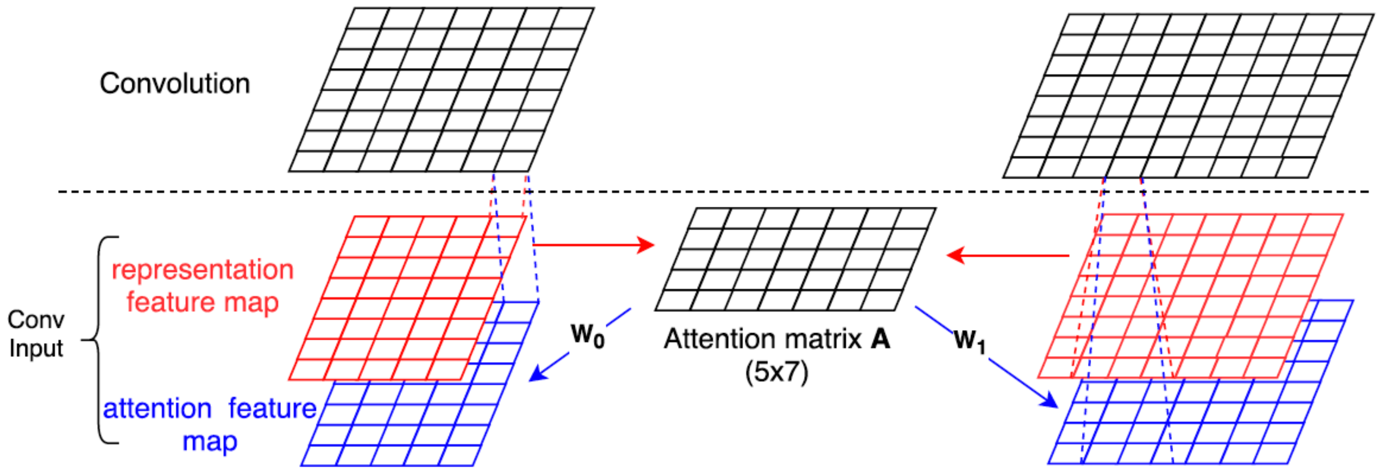
ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs [Yin 2015]

Attention on feature map (ABCNN-1)

- Attention values of row i in $\boldsymbol{A}$: attention distribution of the i–th unit of s_0 with respect to s_1
- $A_{i,j} = matchscore(F_{0,r}[:,i], F_{1,r}[:,j])$
- $matchscore = 1/(1 + |x - y|)$
- Generate the attention feature map $F_{i,a}$ for s_i

$$\begin{aligned} \mathbf{F}_{0,a} &= \mathbf{W}_0 \cdot \mathbf{A}^\intercal \\ \mathbf{F}_{1,a} &= \mathbf{W}_1 \cdot \mathbf{A} \end{aligned}$$

- Cons : need more parameters

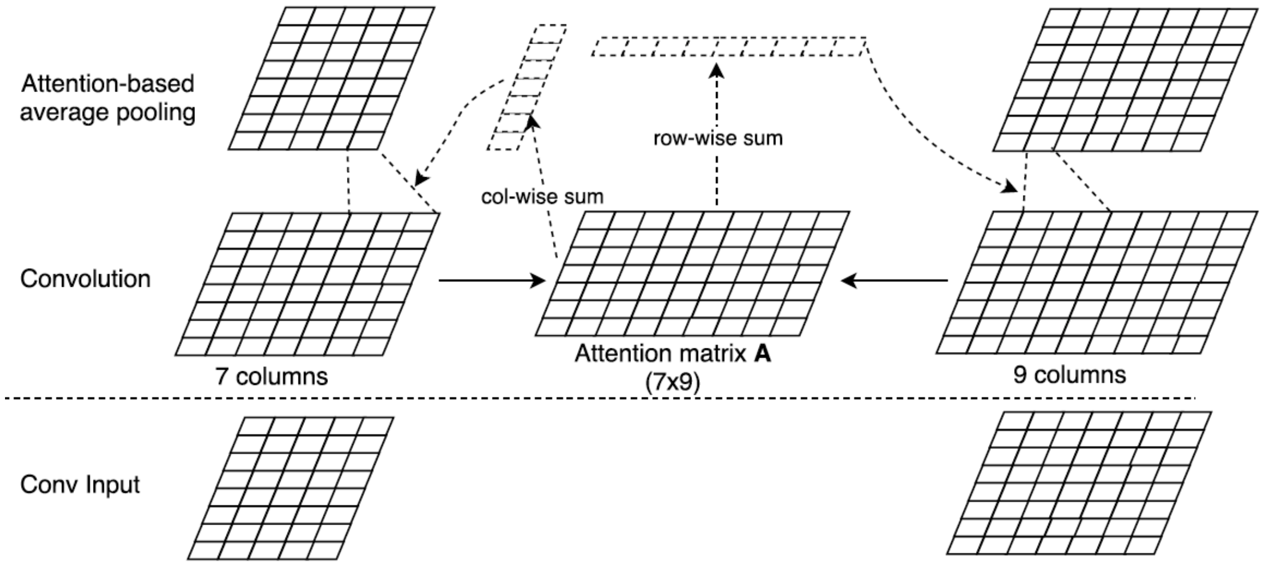


ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs [Yin 2015]

Attention after convolution (ABCNN-2)

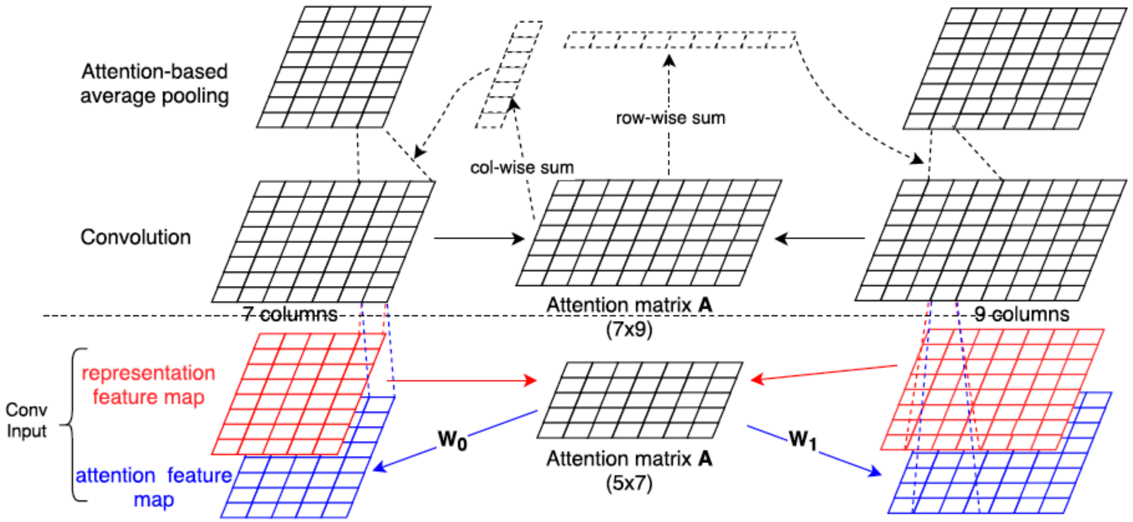
- Attention weights directly on the representation with the aim of improving the features computed by convolution
- $a_{0,j} = \sum A[j,:]$ $\rightarrow$ col-wise, row-wise sum
- ***w-ap*** on convolution feature

$$\mathbf{F}^{p}_{i,r}[:,j] = \sum_{k=j:j+w} a_{i,k} \cdot \mathbf{F}^{c}_{i,r}[:,k], \quad j = 1 \dots s_i$$



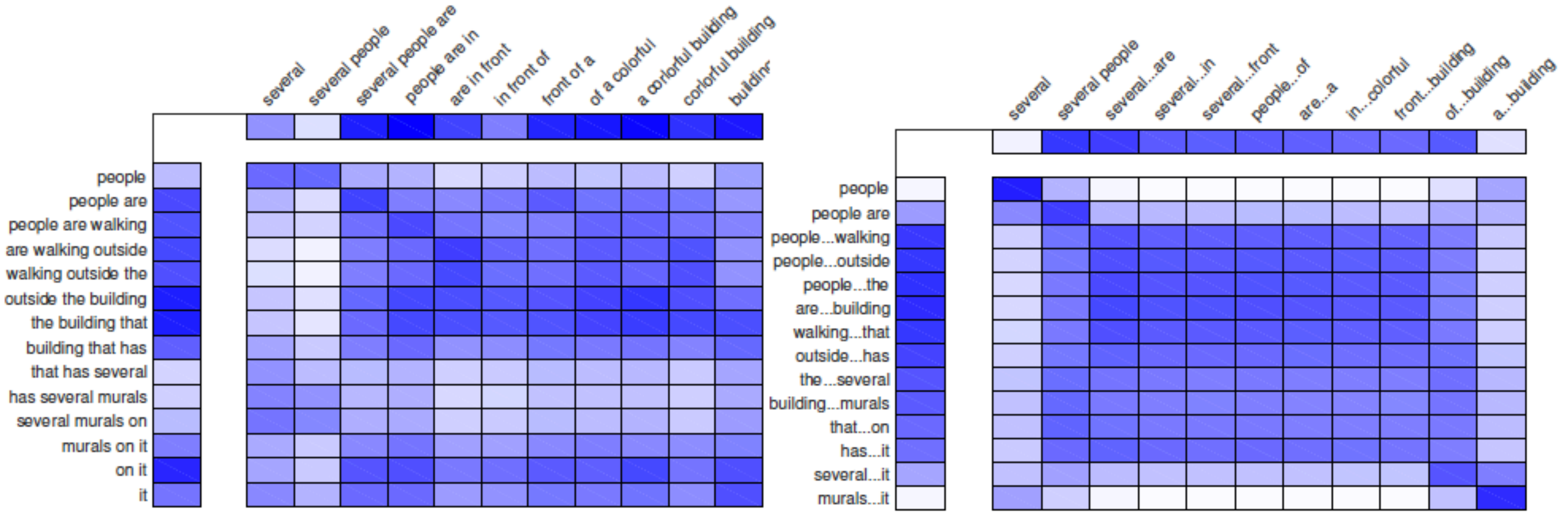
ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs [Yin 2015]

ABCNN-3

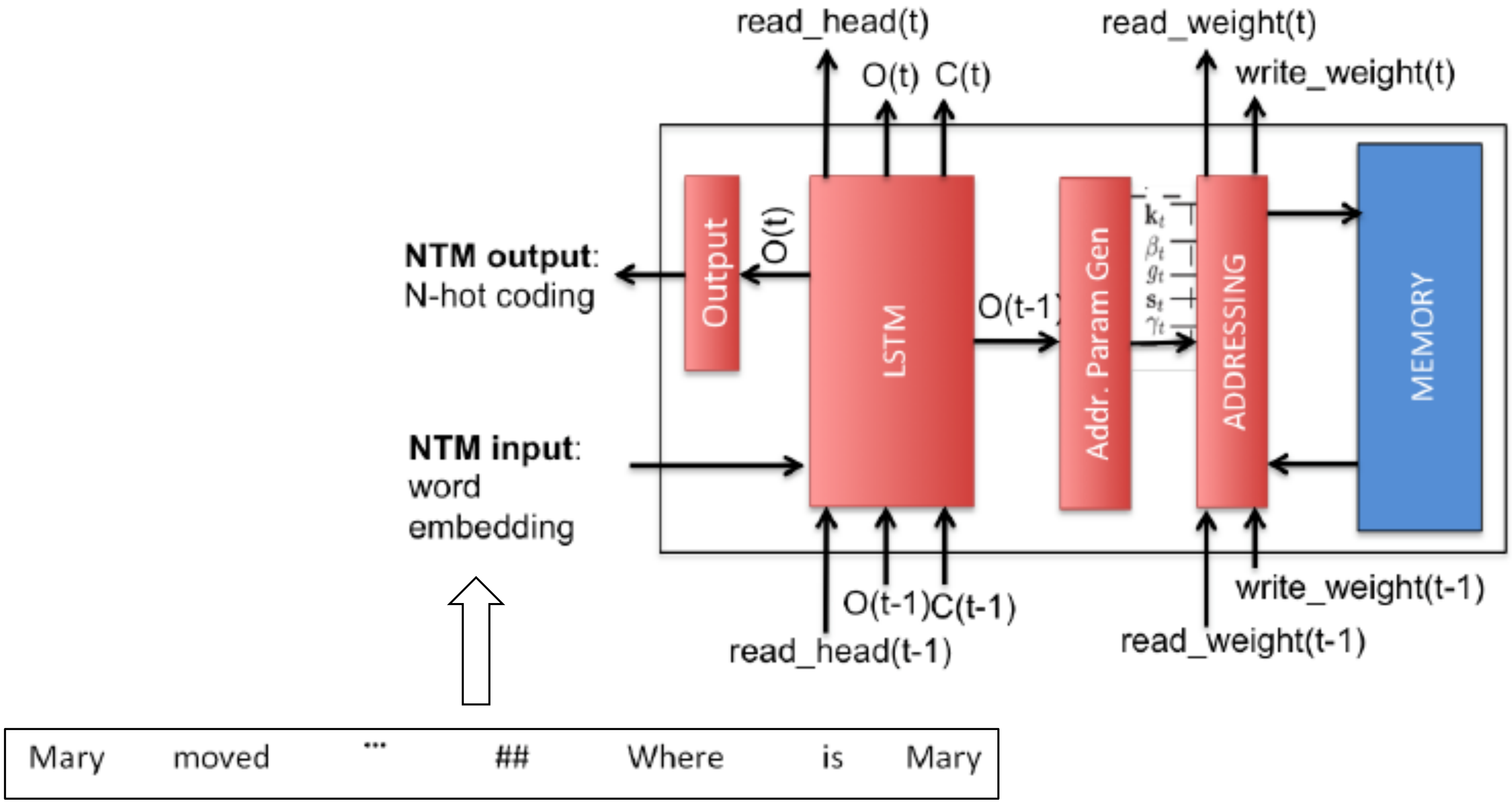


ABCNN-1	ABCNN-2
Indirect impact to convolution	Direct convolution via pooling (weighted attention)
Need more features Vulnerable to overfitting	No need features
handles smaller-granularity units (ex. Word level)	handles larger-granularity units (ex. Phrase level, phrase size = window size)

ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs [Yin 2015]



Empirical Study on Deep Learning Models for QA [Yu 2015]



Empirical Study on Deep Learning Models for QA [Yu 2015]

The first to examine **Neural Turing Machines** on QA problems

Split QA into two step

1. search supporting facts
2. Generate answer from relevant pieces of information

NTM

- Single-layer LSTM network as controller
- Input : word embedding
  1. Support fact only
  2. Fact highlighted : user marker to annotate begin and end of supporting facts
- Output : softmax layer (multiclass classification) for answer

(ii) Support fact only		(iii) Sup. fact highlighted	
d NMT	e NTM	f NMT	g NTM
100	100	100	100
100	100	99.6	100
100	100	99.5	100
99.1	100	97.5	100
99.3	79.2	90.6	73.7
100	100	99.8	100
98.5	100	96.6	100
99	100	92.7	98
100	100	99.7	100
98.9	94.6	96.8	85.9
100	100	100	100
100	100	100	100
100	100	100	100
99.8	100	97.5	100
100	100	92.7	100
100	100	88.1	100
64.2	69.3	58	61.2
97.8	93	91.8	93
80.7	100	29.7	100
100	100	93.3	100
96.9	96.7	91.2	95.6

Teaching Machines to Read and Comprehend [Hermann 2015]

	Original Version	Anonymised Version
Context		
	The BBC producer allegedly struck by Jeremy Clarkson will not press charges against the "Top Gear" host, his lawyer said Friday. Clarkson, who hosted one of the most-watched television shows in the world, was dropped by the BBC Wednesday after an internal investigation by the British broadcaster found he had subjected producer Oisin Tymon "to an unprovoked physical and verbal attack." …	the *ent381* producer allegedly struck by *ent212* will not press charges against the " *ent153* " host , his lawyer said friday . *ent212* , who hosted one of the most - watched television shows in the world , was dropped by the *ent381* wednesday after an internal investigation by the *ent180* broadcaster found he had subjected producer *ent193* " to an unprovoked physical and verbal attack . " …
Query		
	Producer **X** will not press charges against Jeremy Clarkson, his lawyer says.	producer **X** will not press charges against *ent212* , his lawyer says .
Answer		
	Oisin Tymon	*ent193*

Table 3: Original and anonymised version of a data point from the Daily Mail validation set. The anonymised entity markers are constantly permuted during training and testing.

Teaching Machines to Read and Comprehend [Hermann 2015]

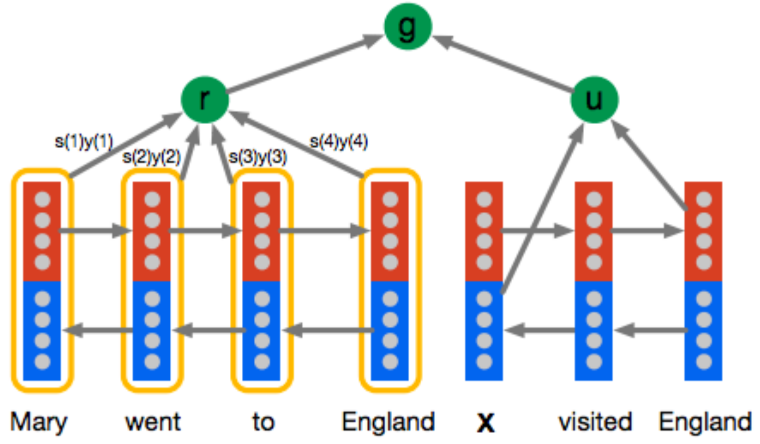
$$u = \overrightarrow{y_q}(|q|) \;||\; \overleftarrow{y_q}(1)$$

$$m(t) = \tanh\left(W_{ym} y_d(t) + W_{um} u\right),$$
$$s(t) \propto \exp\left(\mathrm{w}_{ms}^{\intercal} m(t)\right),$$

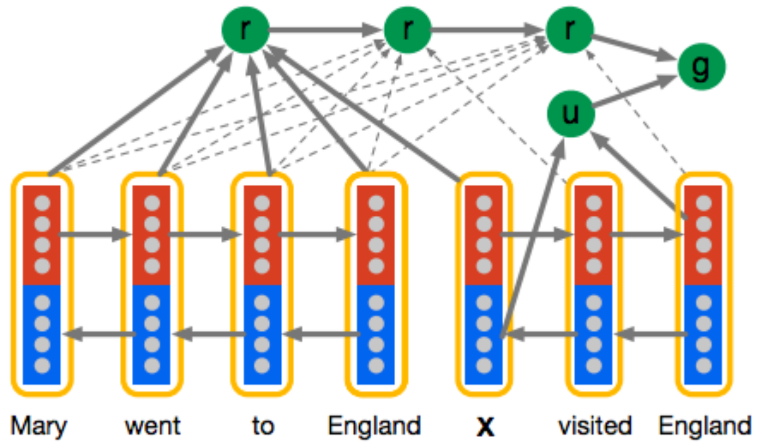
$$r = \sum_i s_i f_i$$

where $f_i = y_d(t)$

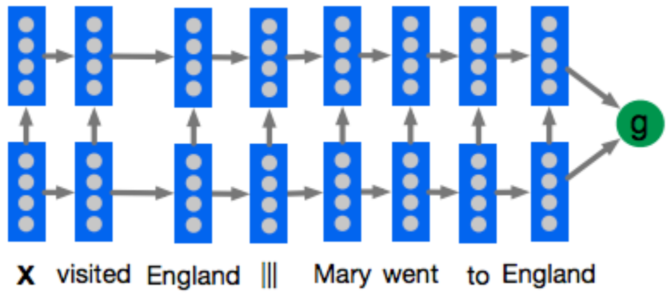
$s(t)$: degree to which the network attends to a particular token in the document when answering the query (soft attention)



(a) Attentive Reader.



(b) Impatient Reader.



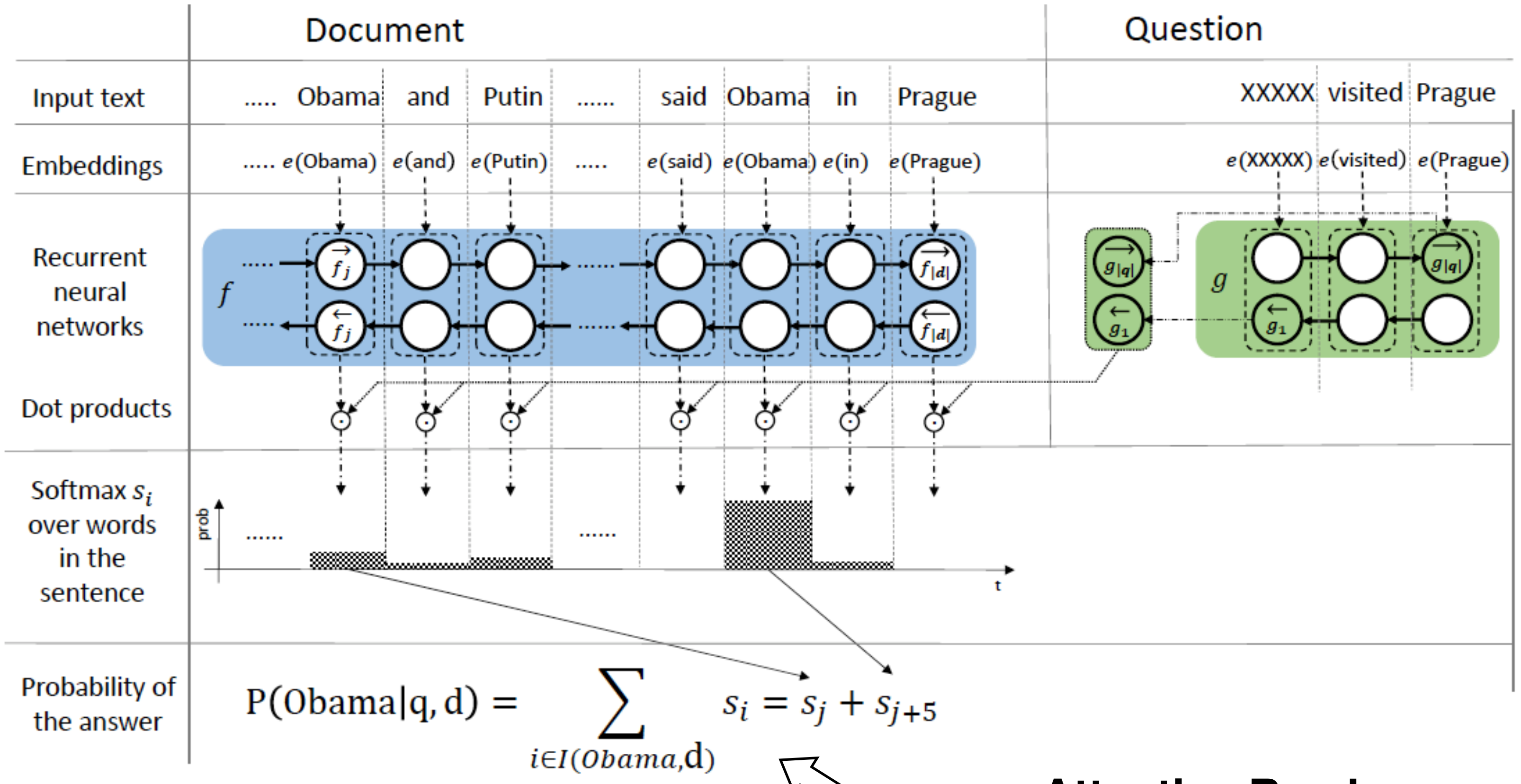
(c) A two layer Deep LSTM Reader with the question encoded before the document.

Text Understanding with the Attention Sum Reader Network [Kadlec 2016]

Answer should be in context

$$s_i \propto \exp\left(f_i(\mathbf{d}) \cdot g(\mathbf{q})\right)$$

$$P(w|\mathbf{q}, \mathbf{d}) = \sum_{i \in I(w, \mathbf{d})} s_i$$



Inspired by Pinter Network

Contrast to Attentive Reader:

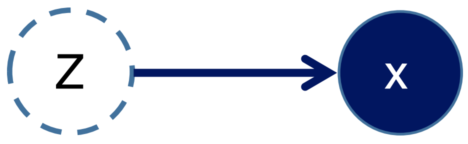
- We select answer from context directly using weighted sum of individual representation

Attentive Reader

$$P(a'|\mathbf{q}, \mathbf{d}) \propto \exp\left(e(a') \cdot r\right).$$

Stochastic Latent Variable

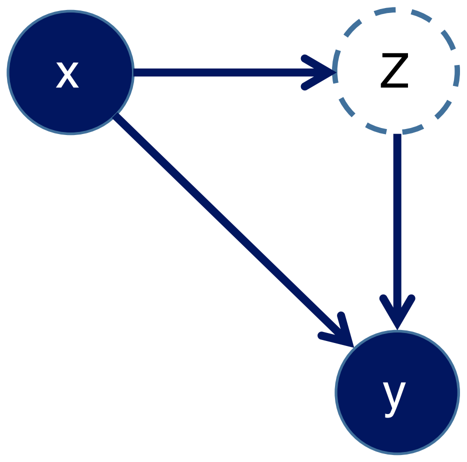
Generative Model



$$p(x) = \sum_z p(x,z) = \sum_z p(x|z)p(z)$$

$$p(x) = \int_z p(x,z) = \int_z p(x|z)p(z)$$

Conditional Generative Model



$$p(y|x) = \sum_z p(y|z,x)p(z|x)$$

$$p(y|x) = \int_z p(y|z,x)p(z|x)$$

Variational Inference Framework

$$p(x, z) = p(x|z)p(z) = \sum_h p(x|h)p(h|z)p(z)$$

$$\log p_\theta(x, z) = \log \int_h \frac{q(h)}{q(h)} p(x|h)p(h|z)p(z)dh \geq \int_h q(h) \log \frac{p(x|h)p(h|z)p(z)}{q(h)} dh$$

$$= \int_h q(h) \log \frac{p(x|h)p(h|z)}{q(h)} dh + \int_h q(h) \log \frac{p(z)}{q(h)} dh$$

$$= E_{q(h)}[\log p(x|h)p(h|z) - \log q(h)] - D_{KL}(q(h)||p(z))$$

$$= E_{q(h)}[\log p(x|h)p(h|z)p(z) - \log q(h)]$$

Variational Inference Framework

$$p_\theta(x,z) = p_\theta(x|z)p(z) = \sum_h p_\theta(x|h)p_\theta(h|z)p(z)$$

Jensen's Inequality

$$\log p_\theta(x,z) = \log \int_h \frac{q(h)}{q(h)} p_\theta(x|h)p_\theta(h|z)p(z)dh \geq \int_h q(h) \log \frac{p_\theta(x|h)p_\theta(h|z)p(z)}{q(h)} dh$$

$$= \int_h q(h) \log \frac{p_\theta(x|h)p_\theta(h|z)}{q(h)} dh + \int_h q(h) \log \frac{p(z)}{q(h)} dh$$

$$= E_{q(h)}[\log p_\theta(x|h)p_\theta(h|z) - \log q(h)] - D_{KL}(q(h)||p(z))$$

$$= E_{q(h)}[\log p_\theta(x|h)p_\theta(h|z) - \log q(h)]$$ a tight lower bound if $q(h) = p(h|x,z)$

Conditional Variational Inference Framework

$$p_\theta(y|x) = \sum_z p_\theta(y,z|x) = \sum_z p_\theta(y|x,z)p_\pi(z|x)$$

Jensen's Inequality

$$\log p(y|x) = \log \int_z \frac{q(z)}{q(z)} p(y|z,x)p(z|x)dz \geq \int_z q(z) \log \frac{p(y|z,x)p(z|x)}{q(z)} dz$$

$$= \int_z q(z) \log \frac{p(y|z,x)}{q(z)} dz + \int_h q(z) \log \frac{p(z|x)}{q(z)} dz$$

$$= \int_z q(z) \log p(y|z,x)\, dz - \int_z q(z) \log q(z)\, dz + \int_h q(z) \log \frac{p(z|x)}{q(z)} dz$$

$$= E_{q(z)}[\log p(y|z,x) - \log q(z)] - D_{KL}(q(z) \parallel p(z|x))$$

$$= E_{q(z)}[\log p(y|z,x) - \log q(z)]$$ a tight lower bound if $q(z) = p(z|x)$

Neural Variational Inference Framework

$$\log p_\theta(x, z) \geq E_{q(z)}[\log p(y|z, x) - \log q(z)] - D_{KL}\big(q(z) \parallel p(z|x)\big) = \mathcal{L}$$

1. Vector representations of the observed variables

$$u = f_z(z), v = f_x(x)$$

2. Joint representation (concatenation)

$$\pi = g(u, v)$$

3. Parameterize the variational distribution

$$\mu = l_1(\pi), \sigma = l_2(\pi)$$

Neural Variational Document Model [Miao, 2015]

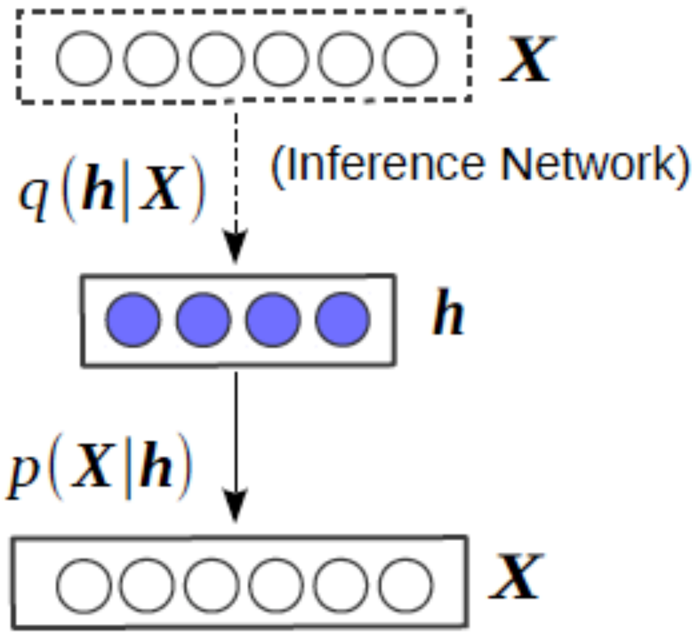


Figure 1: NVDM for document modelling.

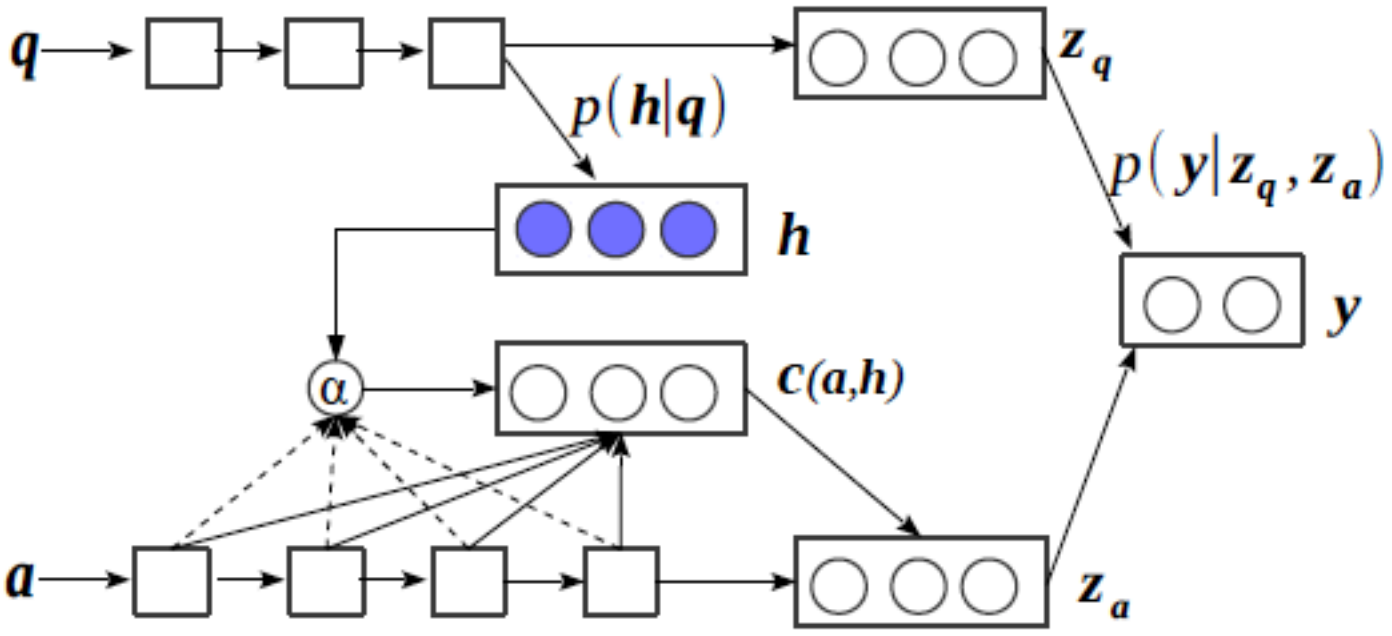


Figure 2: NASM for question answer selection.