ReasoNet: Learning to Stop Reading in Machine Comprehension

Yelong Shen, Po-Sen Huang, Jianfeng Gao, Weizhu Chen

Microsoft Research

Motivation

- **Task**

  Cloze-style dataset (CNN datasets)

Step	Termination Probability	Attention Sum
1	0.0011	0.4916
2	0.5747	0.5486
3	0.9178	0.5577

Query: passenger @placeholder , 36 , died at the scene

Passage: (@entity0) what was supposed to be a fantasy sports car ride at @entity3 turned deadly when a @entity4 crashed into a guardrail . the crash took place sunday at the @entity8 , which bills itself as a chance to drive your dream car on a racetrack . the @entity4 's passenger , 36 - year - old @entity14 of @entity15 , @entity16 , died at the scene , @entity13 said . the driver of the @entity4 , 24 - year - old @entity18 of @entity19 , @entity16 , lost control of the vehicle , the @entity13 said . he was hospitalized with minor injuries . @entity24 , which operates the @entity8 at @entity3 , released a statement sunday night about the crash . " on behalf of everyone in the organization , it is with a very heavy heart that we extend our deepest sympathies to those involved in today 's tragic accident in @entity36 , " the company said . @entity24 also operates the @entity3 -- a chance to drive or ride in @entity39 race cars named for the winningest driver in the sport 's history . @entity0 's @entity43 and @entity44 contributed to this report .

Answer: @entity14

Step 1 Step 2 Step 3

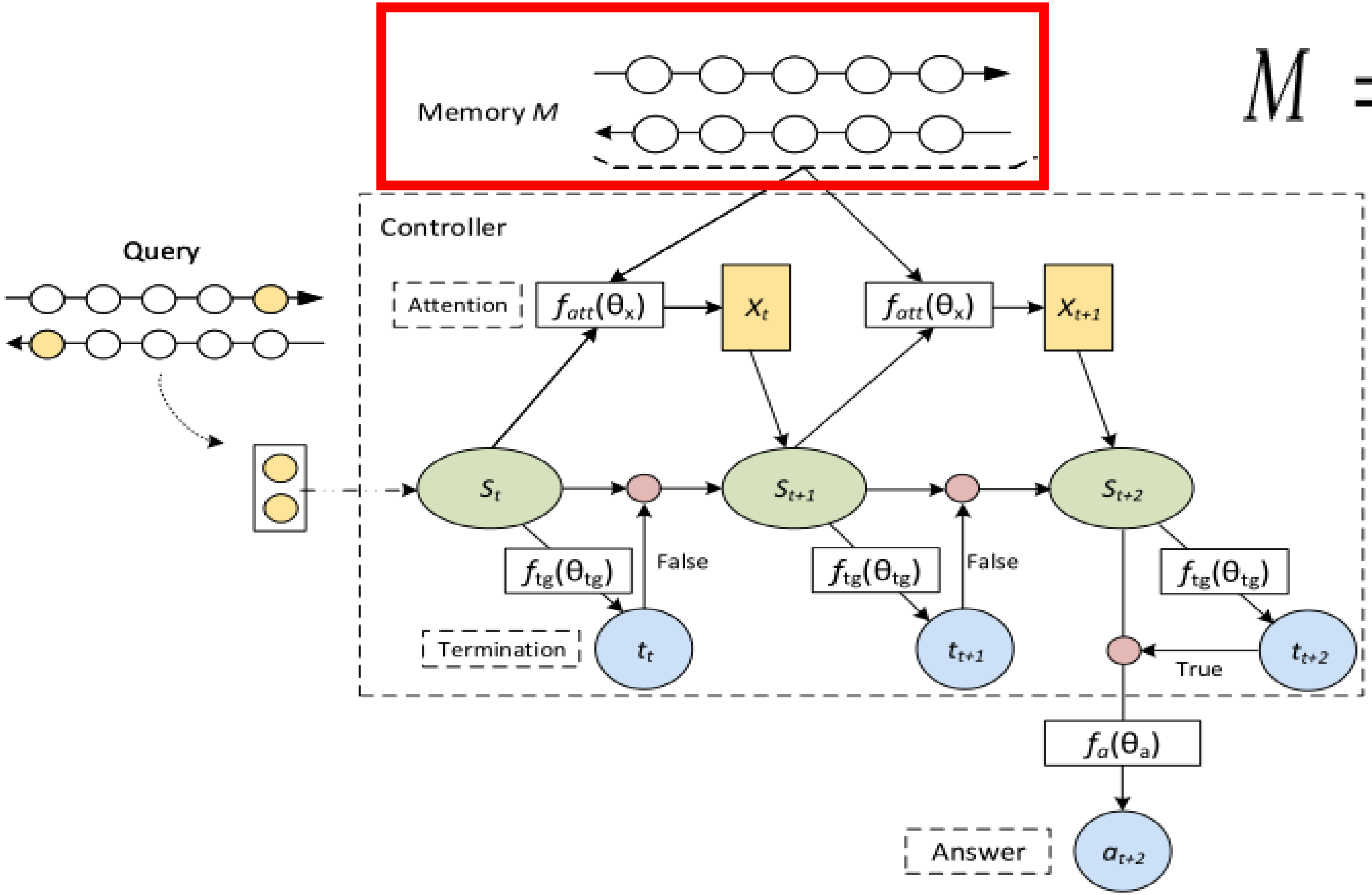
Motivation

- **Related work**

1. Single-turn reasoning
   - utilize attention mechanisms to emphasize specific parts of the document which are relevant to the query
2. Multi-turn reasoning
   - By repeatedly processing the document and the question after digesting intermediate information, multi-turn reasoning can generally produce a better answer
   - pre-defined number of hops or iterations in their inference without regard to the complexity of each individual query or document
3. ReasoNet

   ReasoNets introduce a termination state in the inference. This state can decide whether to continue the inference to the next turn

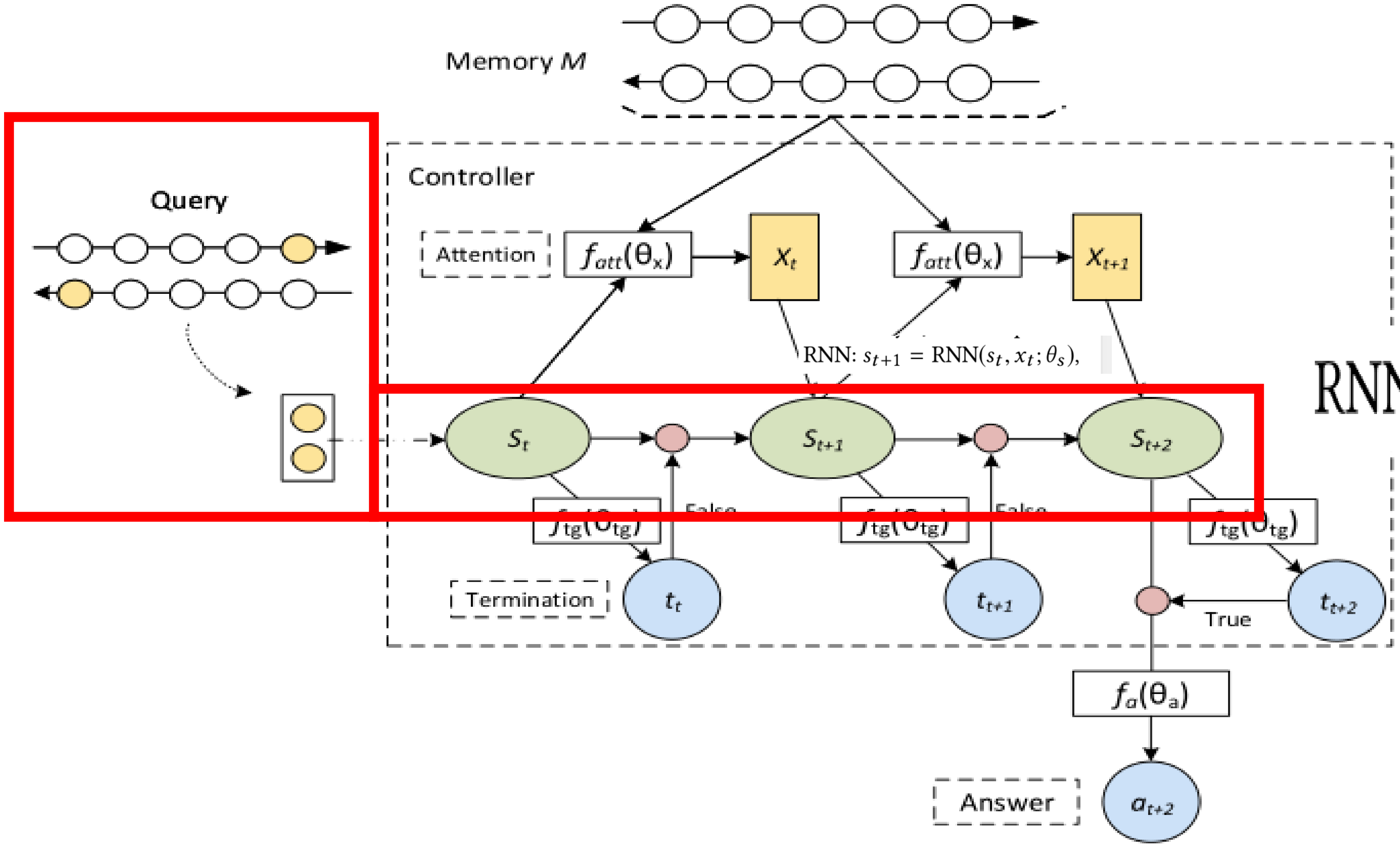
Model (Memory)



$$M = (M^{query}, M^{doc}),$$

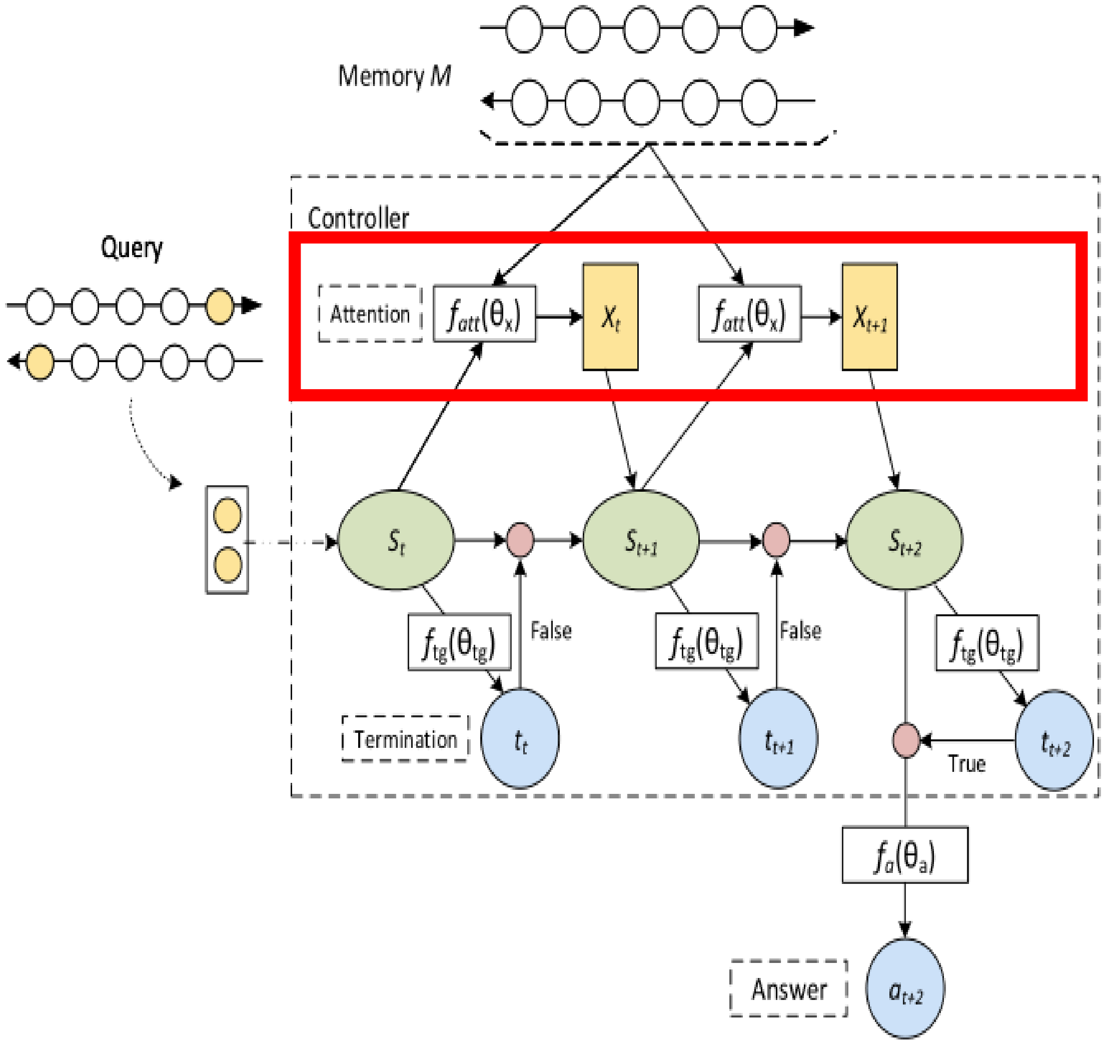
are extracted from query bidirectional-GRU encoder and passage bidirectional-GRU encoder respectively.

Model (Internal State)



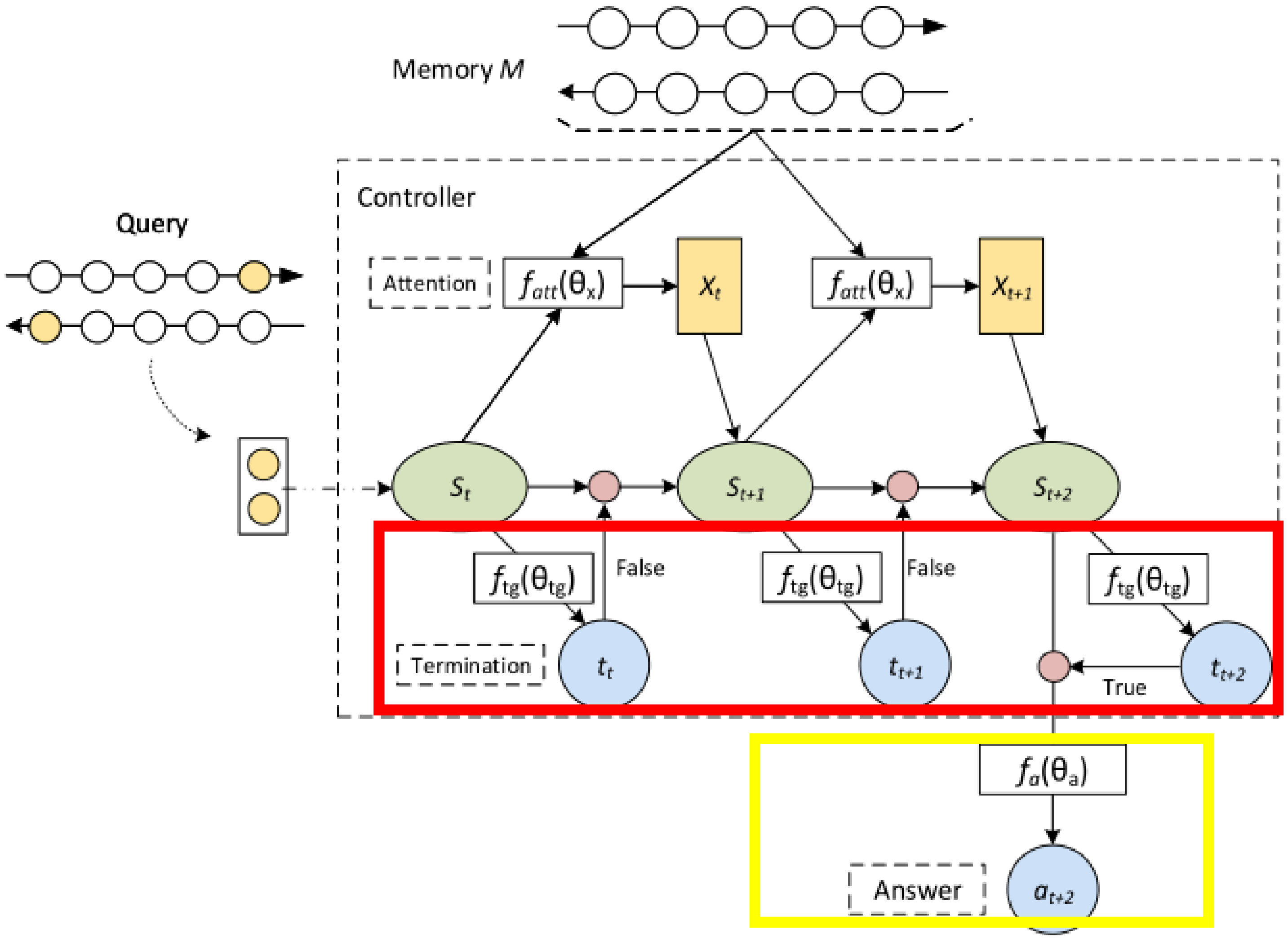
$$\text{RNN}: s_{t+1} = \text{RNN}(s_t, x_t; \theta_s),$$

Model (Attention)



sine similarity function as the attention module. The attention score $a_{t,i}^{doc}$ on memory m_i^{doc} given the state s_t is computed as follows: $a_{t,i}^{doc} = \text{softmax}_{i=1,\ldots,|M^{doc}|} \gamma \cos(W_1^{doc} m_i^{doc}, W_2^{doc} s_t)$, where γ is set to 10. W_1^{doc} and W_2^{doc} are weight vectors associated with m_i^{doc} and s_t, respectively, and are joint trained in the ReasoNet. Thus, the attention vector on passage is given by $x_t^{doc} = \sum_i^{|M^{doc}|} a_{t,i}^{doc} m_i^{doc}$. Similarly, the attention vector on query is $x_t^{query} = \sum_i^{|M^{query}|} a_{t,i}^{query} m_i^{query}$. The final attention vector is the concatenation of the query attention vector and the passage attention vector $x_t = (x_t^{query}, x_t^{doc})$. The attention module is parameterized by $\theta_x = (W_1^{query}, W_2^{query}, W_1^{doc}, W_2^{doc})$;

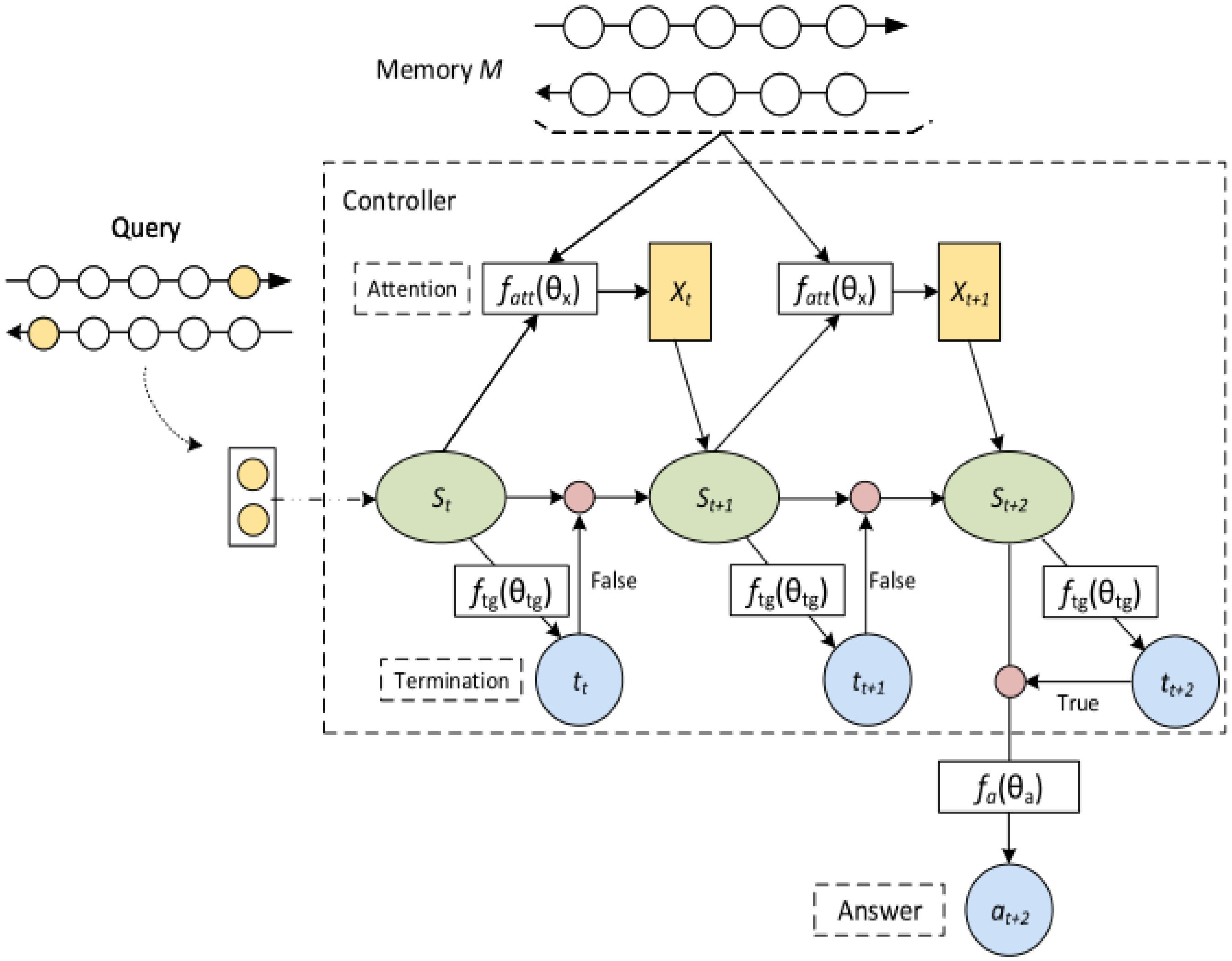
Model (Termination Gate and Answer)



$$t_t \sim p(\cdot | f_{tg}(s_t; \theta_{tg}))).$$

$$a_t \sim p(\cdot | f_a(s_t; \theta_a)).$$

Algorithm



Algorithm 1: Stochastic Inference in a ReasoNet

Input : Memory M; Initial state s_1; Step $t = 1$; Maximum Step $T_{\max}$

Output: Termination Step T, Answer a_T

1 Sample t_t from the distribution $p(\cdot|f_{tg}(s_t;\theta_{tg}))$;

2 if t_t is false, go to Step 3; otherwise Step 6;

3 Generate attention vector $x_t = f_{att}(s_t, M; \theta_x)$;

4 Update internal state $s_{t+1} = \text{RNN}(s_t, x_t; \theta_s)$;

5 Set $t = t + 1$; if $t < T_{\max}$ go to Step 1; otherwise Step 6;

6 Generate answer $a_t \sim p(\cdot|f_a(s_t;\theta_a))$;

7 Return $T = t$ and $a_T = a_t$;

Training Details

the total expect reward. The expected reward for an instance is defined as:

$$J(\theta) = \mathbb{E}_{\pi(t_{1:T}, a_T;\theta)}\left[\sum_{t=1}^{T} r_t\right]$$

The reward can only be received at the final termination step when an answer action a_T is performed. We define $r_T = 1$ if $t_T = 1$ and the answer is correct, and $r_T = 0$ otherwise. The rewards on intermediate steps are zeros, $\{r_t = 0\}_{t=1...T-1}$. J can be maximized by directly applying gradient based optimization methods. The

Motivated by the REINFORCE algorithm [31], we compute $\nabla_\theta J(\theta)$:

$$\mathbb{E}_{\pi(t_{1:T}, a_T;\theta)}\left[\nabla_\theta \log\pi(t_{1:T}, a_T;\theta) r_T\right] =$$

$$\sum_{(t_{1:T}, a_T)\in \mathbb{A}^{\dagger}} \pi(t_{1:T}, a_T;\theta)\left[\nabla_\theta \log\pi(t_{1:T}, a_T;\theta)(r_T - b_T)\right]$$

Experimental results

Table 1: The performance of Reasoning Network on CNN and Daily Mail dataset.

	CNN		Daily Mail	
	valid	test	valid	test
Deep LSTM Reader [7]	55.0	57.0	63.3	62.2
Attentive Reader [7]	61.6	63.0	70.5	69.0
MemNets [8]	63.4	66.8	-	-
AS Reader [9]	68.6	69.5	75.0	73.9
Stanford AR [3]	72.2	72.4	76.9	75.8
DER Network [12]	71.3	72.9	-	-
Iterative Attention Reader [21]	72.6	73.3	-	-
EpiReader [25]	**73.4**	74.0	-	-
GA Reader [6]	73.0	73.8	76.7	75.7
AoA Reader [5]	73.1	74.4	-	-
ReasoNet	72.9	**74.7**	**77.6**	**76.6**