# End to End Memory Network

모두연 NLP Bootcamp
김 성 운

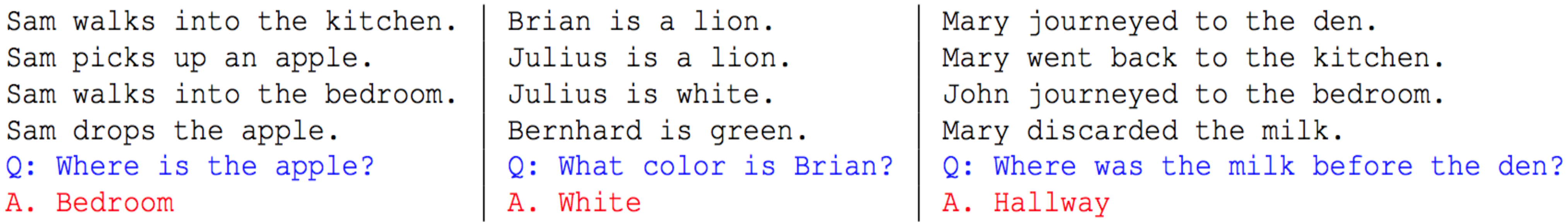
# Contents

1. Task
2. Model
3. Result
4. Conclusion

# Task

## 1. Question and Answering

- bAbI dataset
  - 20가지 종류의 문제로 구성
  - Supporting facts, Yes/No Question, Counting etc.



Sam walks into the kitchen.
Sam picks up an apple.
Sam walks into the bedroom.
Sam drops the apple.
Q: Where is the apple?
A. Bedroom

Brian is a lion.
Julius is a lion.
Julius is white.
Bernhard is green.
Q: What color is Brian?
A. White

Mary journeyed to the den.
Mary went back to the kitchen.
John journeyed to the bedroom.
Mary discarded the milk.
Q: Where was the milk before the den?
A. Hallway

## 2. Language Modeling

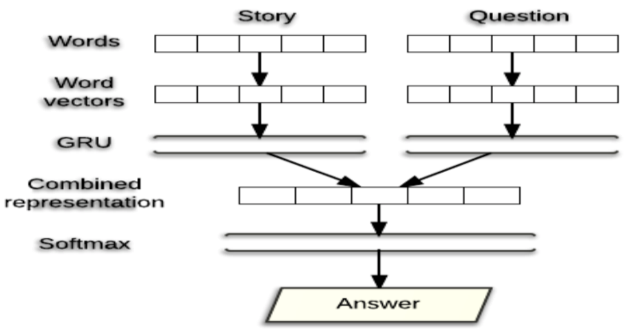
- Penn Treebank, Text8 dataset

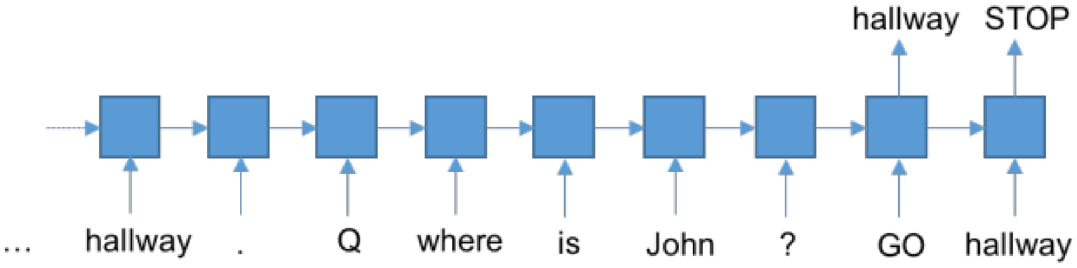
# Model

## 1. Motivation

- 기존 RNN 기반 모델 대체 (Memory를 활용하자!)

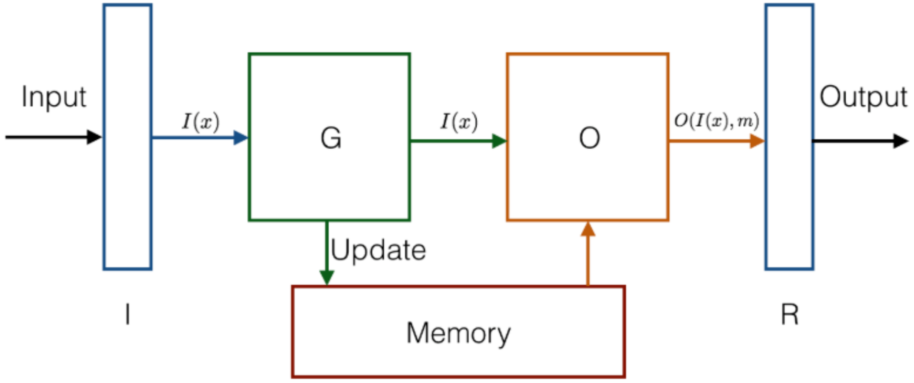
(The memory in these models is the state of the network, which is latent and inherently unstable over long timescales.)





https://cs224d.stanford.edu/reports/StrohMathur.pdf

- End to End learning (MemNN의 supporting facts 없이 학습하자!)
  ☞ Weakly supervised, Soft attention



Memory Networks

https://persagen.com/resources/biokdd-review-nlu.html

# Model

## 2. Architecture

- Main idea

- 각 sentences를 embedding한 memory 벡터들(m)과, 질문 q의 embedding 벡터(u)를 inner product 한 후, softmax.

☞ 지문의 문장들과 질문의 similarity를 확률로 표현

☞ 문장 별로 주목할 정도를 학습 (soft attention)

- Trainable varibles:

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

  V: number of words in training set (number of classes),

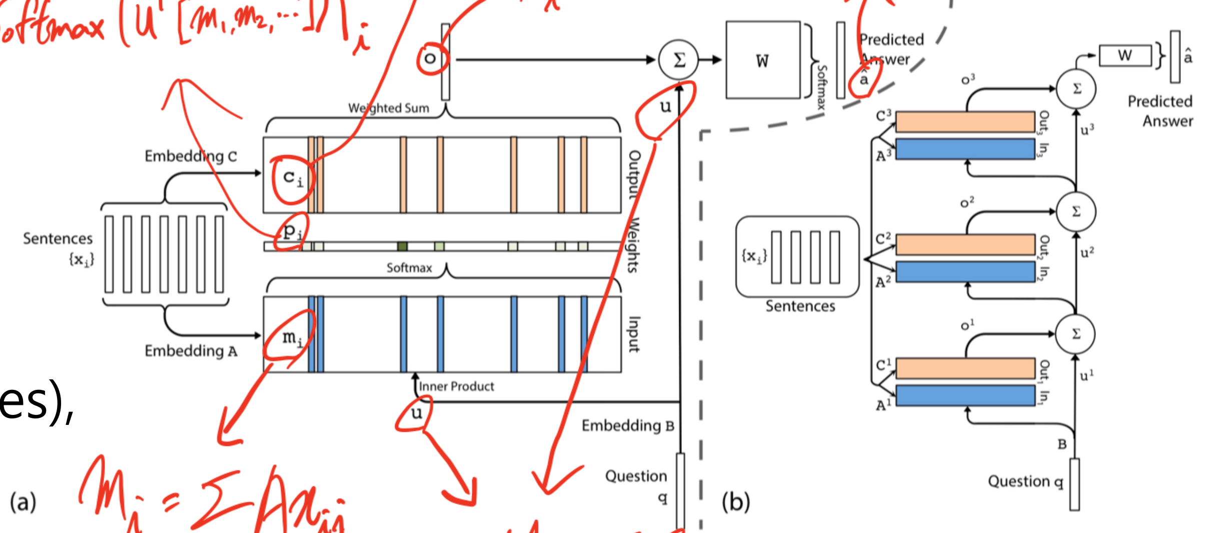
  d: dimension of the embedding.

$c_i = \sum_j C x_{ij}$

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

$p_i = \{\text{Softmax}(u^T[m_1, m_2, \cdots])\}_i$

$\sum_i p_i c_i$



$m_i = \sum_j A x_{ij}$

$u = \sum_j B q_j$, $q = [q_1, q_2, \cdots]$

i: 문장 index

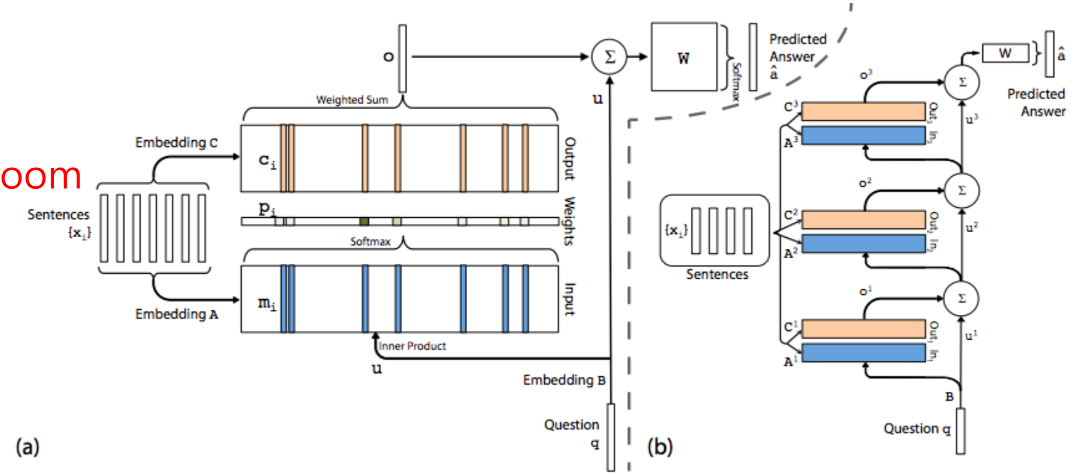
j: 문장 내 단어 index

# Model

## 3. Single layer example

a: bathroom

x1: Mary moved to the bathroom
x2: John went to the hallway



q: Where is Mary

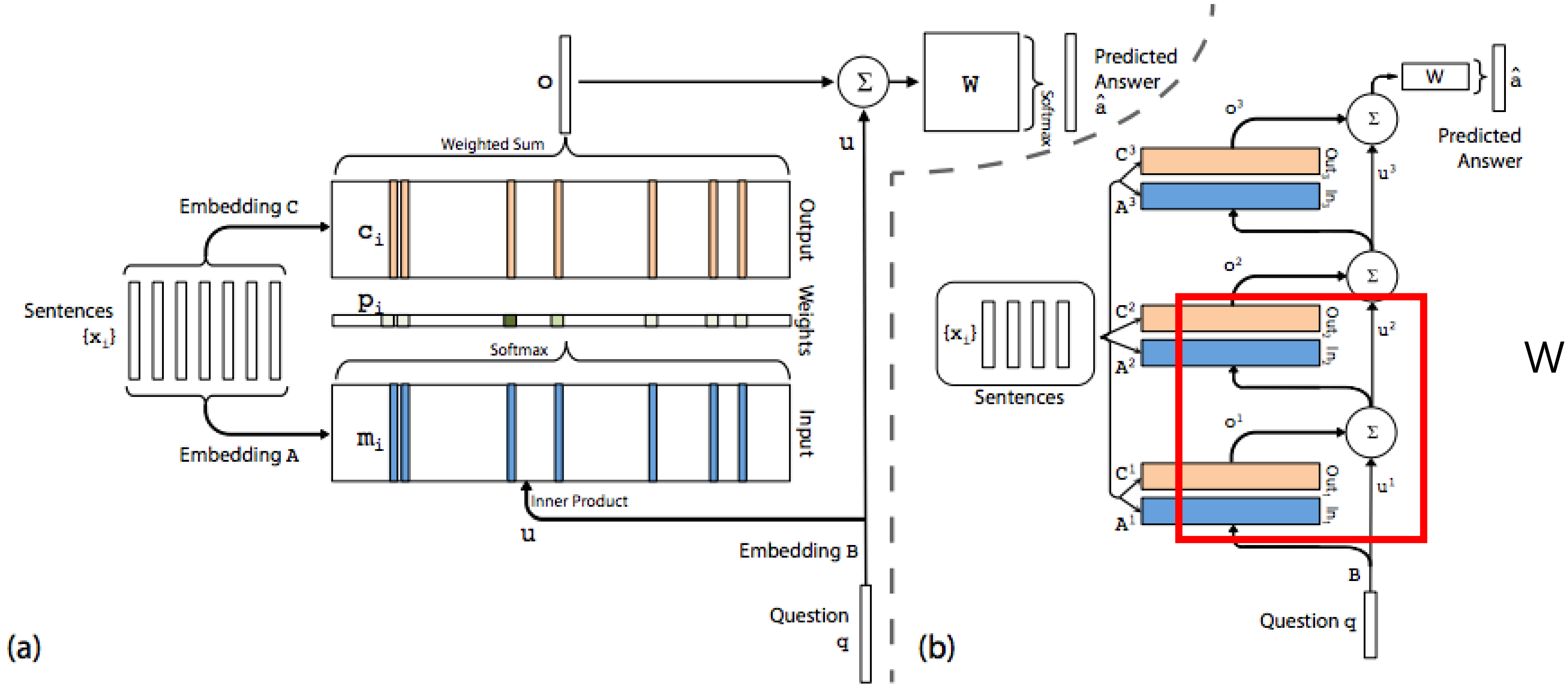
- vocab ={Mary: 0, moved: 1, to: 2, the: 3, bathroom: 4, John: 5, went: 6, hallway: 7, where: 8, is: 9}
- |vocab| = V = 10 (= number of classes)
- $x_1 = \{\text{Mary, moved, to, the, bathroom}\} = \{x_{11}, x_{12}, x_{13}, x_{14}, x_{15}\} = \{0, 1, 2, 3, 4\} = \{[1,0,0,0,0,0,0,0,0,0]^T, \dots\} \in\ R^{10\times 5}$,

  $x_2 = \{[0,0,0,0,0,1,0,0,0,0]^T, \dots\}$,

  $q = \{q_1, q_2, q_3\} = \{8, 9, 0\} = \{[0,0,0,0,0,0,0,0,1,0]^T, \dots\}$
- $A, B, C\ \in\ R^{3\times 10}$ (ex. embedding dim = 3)
- $m_1 = Ax_{11} + Ax_{12} + Ax_{13} + Ax_{14} + Ax_{15} \in\ R^{3\times 1}$, $u = Bq_1 + Bq_2 + Bq_3 \in\ R^{3\times 1}$, $c_1 = Cx_{11} + Cx_{12} + Cx_{13} + Cx_{14} + Cx_{15} \in\ R^{3\times 1}$,
- $p_1 = \{\ Softmax([u^T m_1, u^T m_2])\ \}_1 \in\ R^1$, $\text{o} \in\ R^{3\times 1}$
- $W\ \in\ R^{10\times 3}$

# Model

## 4. Multi layers



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

With K hop operations,

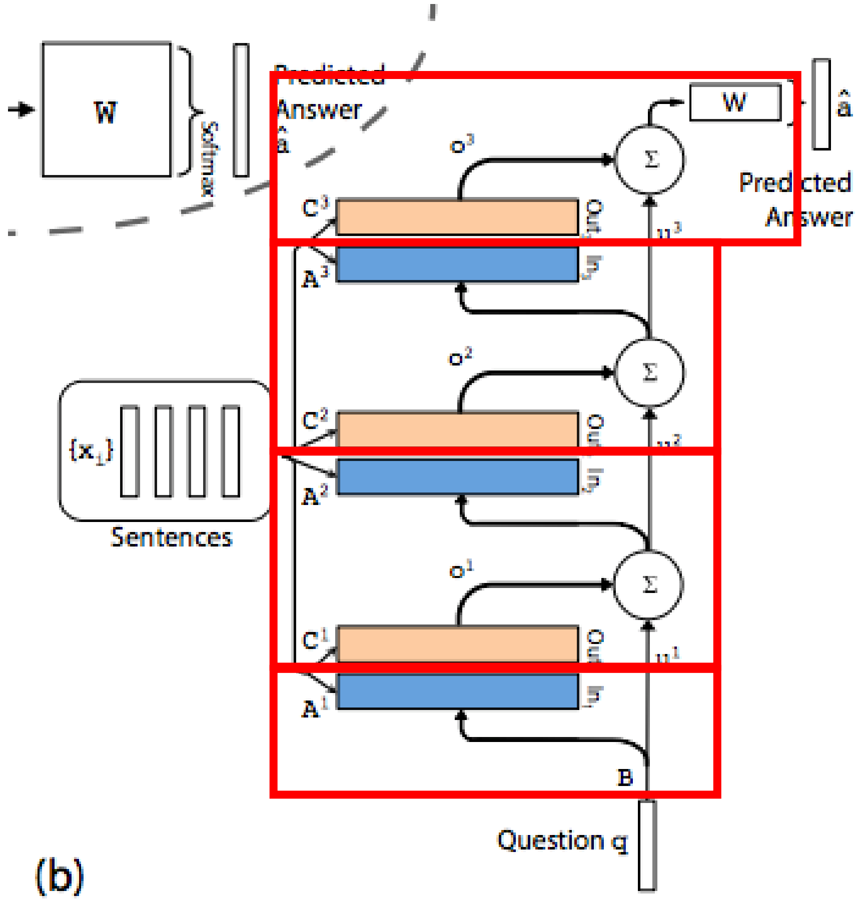
$$\hat{a} = \text{Softmax}(Wu^{K+1})$$

$$= \text{Softmax}(W(o^K + u^K)).$$

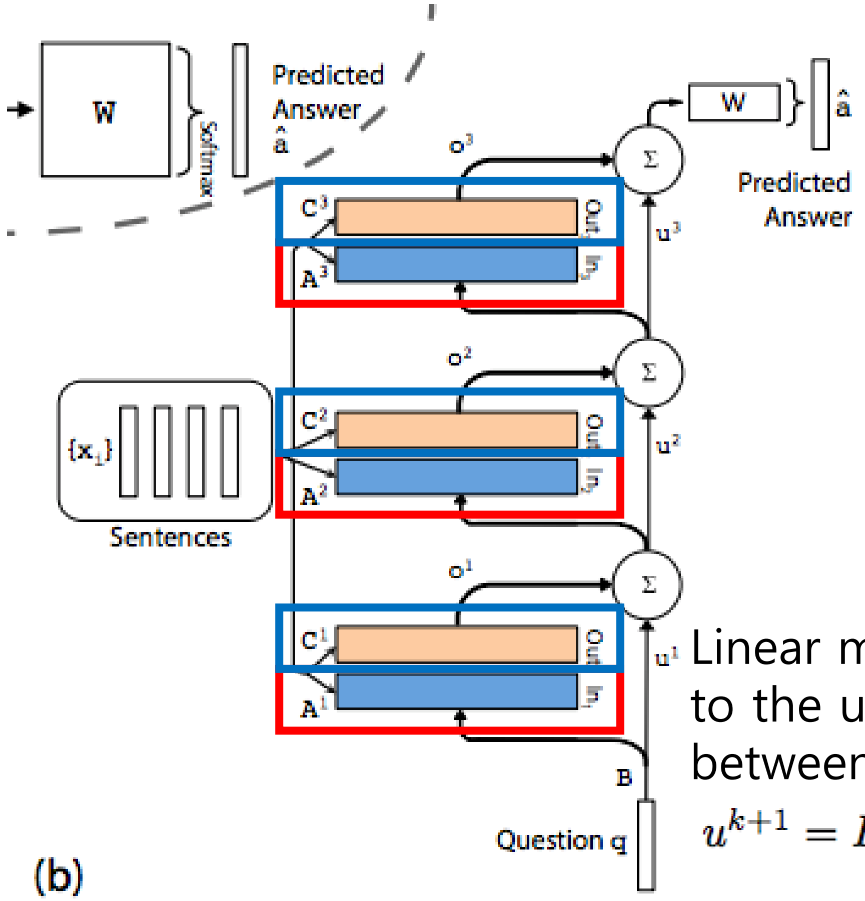
# Model

## 4. Multi layers(weight tying strategy)

- Adjacent: $A^{k+1} = C^k$



- Layer-wise: $A^1 = A^2 = \cdots = A^k$,
  $C^1 = C^2 = \cdots = C^k$



Linear mapping 'H' to the update of 'u' between hops

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

# Model

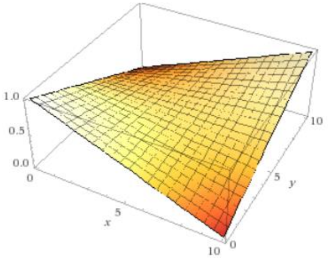
## 5. Model Details

- Multi layers (3 hops)
- Position encoding (PE)

: word embedding 결과에 서로 다른 가중치를 곱해 줌으로써 단어가 문장 안의 단어 순서가 바뀌면 메모리 벡터도 달라짐.

d: embedding 차원, k: embedding된 벡터에서 몇 번째 원소?(1~d),
j: 문장 내 word 순서(1~J) (문장 내 단어 순서를 고려), i: 문장 순서

$$m_i = \sum_j l_j \cdot Ax_{ij}$$
$$l_{kj} = (1 - j/J) - (k/d)(1 - 2j/J)$$



- Temporal Encoding

: 문장의 순서를 고려할 수 있는 time embedding matrix를 별도로 학습(문장 간 순서를 고려)

$$m_i = \sum_j Ax_{ij} + T_A(i), \quad c_i = \sum_j Cx_{ij} + T_C(i)$$

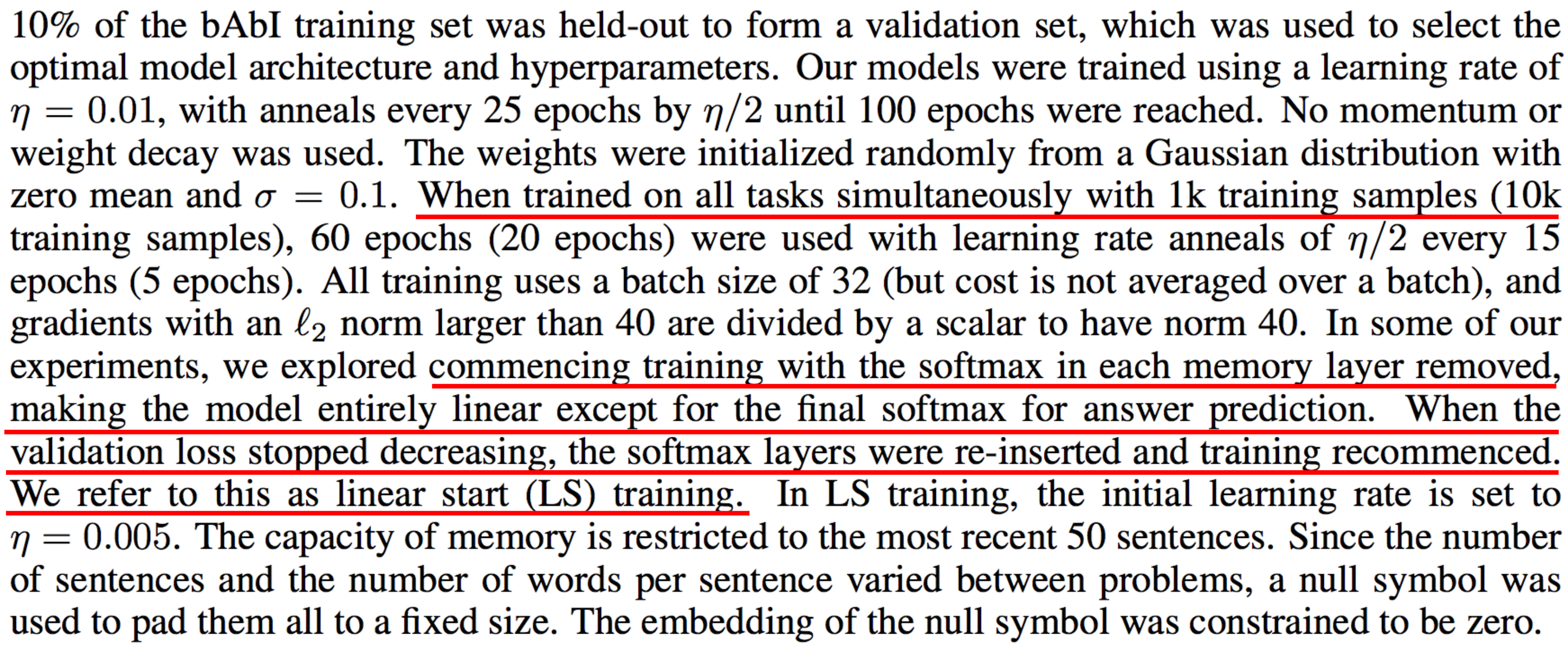
- Learning time invariance by injecting random noise (RN)

: 학습과정에서 10%의 empty memories를 삽입하여 regularization 효과

# Model

## 6. Training Details

- Joint training
- Linear start traing: avoid local minima



10% of the bAbI training set was held-out to form a validation set, which was used to select the optimal model architecture and hyperparameters. Our models were trained using a learning rate of $\eta = 0.01$, with anneals every 25 epochs by $\eta/2$ until 100 epochs were reached. No momentum or weight decay was used. The weights were initialized randomly from a Gaussian distribution with zero mean and $\sigma = 0.1$. When trained on all tasks simultaneously with 1k training samples (10k training samples), 60 epochs (20 epochs) were used with learning rate anneals of $\eta/2$ every 15 epochs (5 epochs). All training uses a batch size of 32 (but cost is not averaged over a batch), and gradients with an $\ell_2$ norm larger than 40 are divided by a scalar to have norm 40. In some of our experiments, we explored commencing training with the softmax in each memory layer removed, making the model entirely linear except for the final softmax for answer prediction. When the validation loss stopped decreasing, the softmax layers were re-inserted and training recommenced. We refer to this as linear start (LS) training. In LS training, the initial learning rate is set to $\eta = 0.005$. The capacity of memory is restricted to the most recent 50 sentences. Since the number of sentences and the number of words per sentence varied between problems, a null symbol was used to pad them all to a fixed size. The embedding of the null symbol was constrained to be zero.

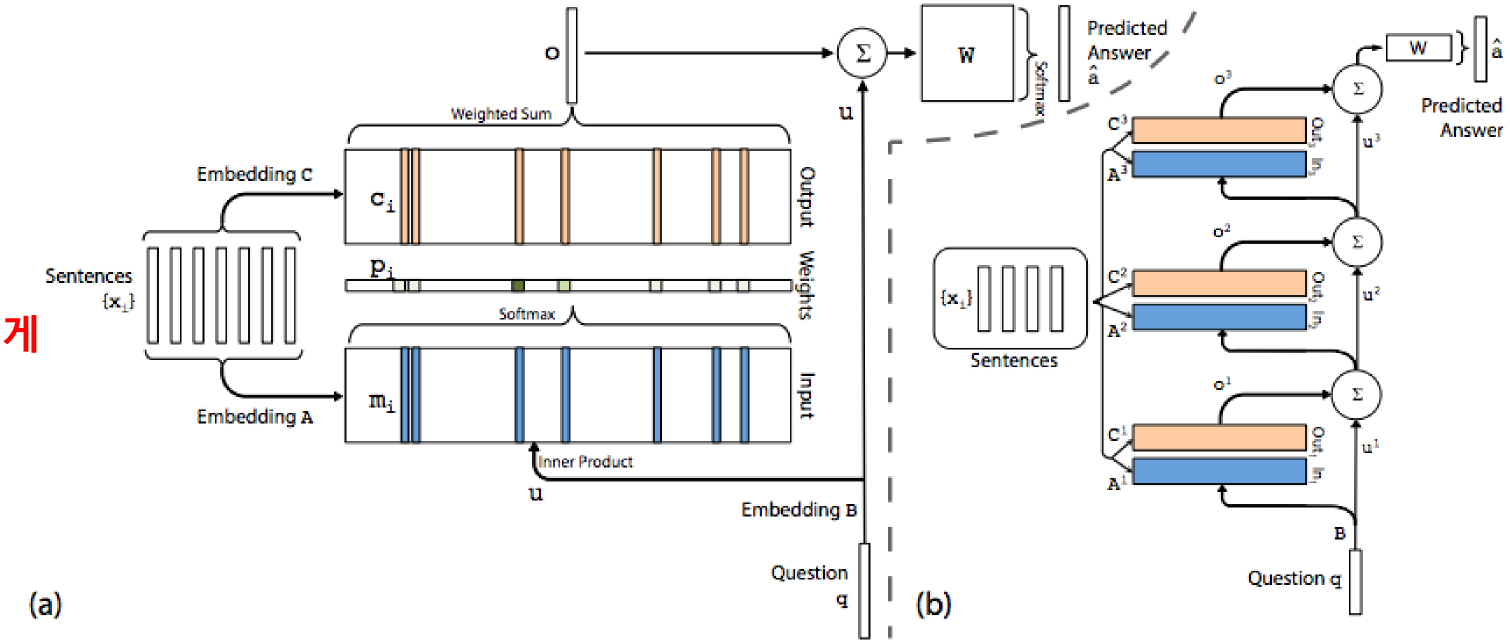
# Model

## 7. Language Modeling

- memory vector는 각 word 별로,
- question vector는 0.1로 이뤄진 constant vector

a: 배출된다

x1: 중국에서
x2: 미세먼지가
x3: 어마어마하게
...

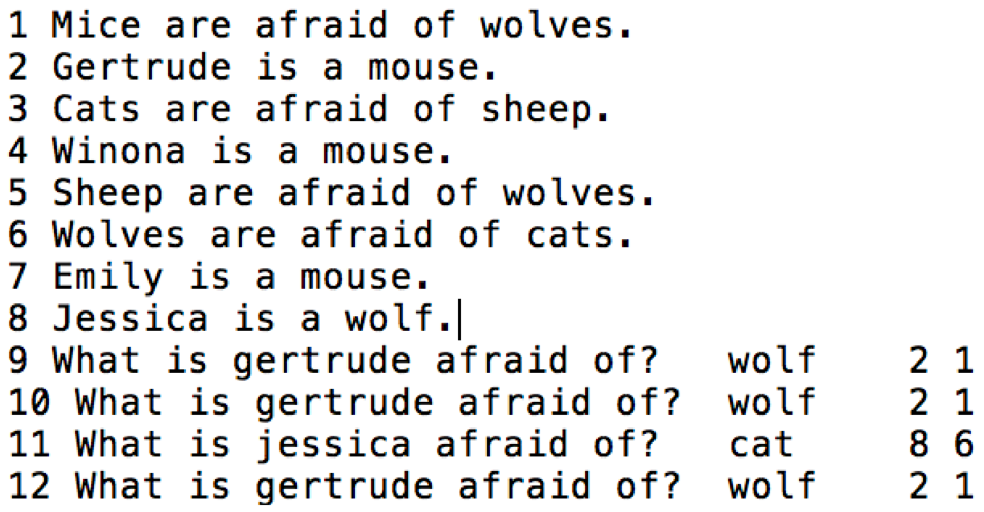


q: Constant vector = [0.1, 0.1, ...]

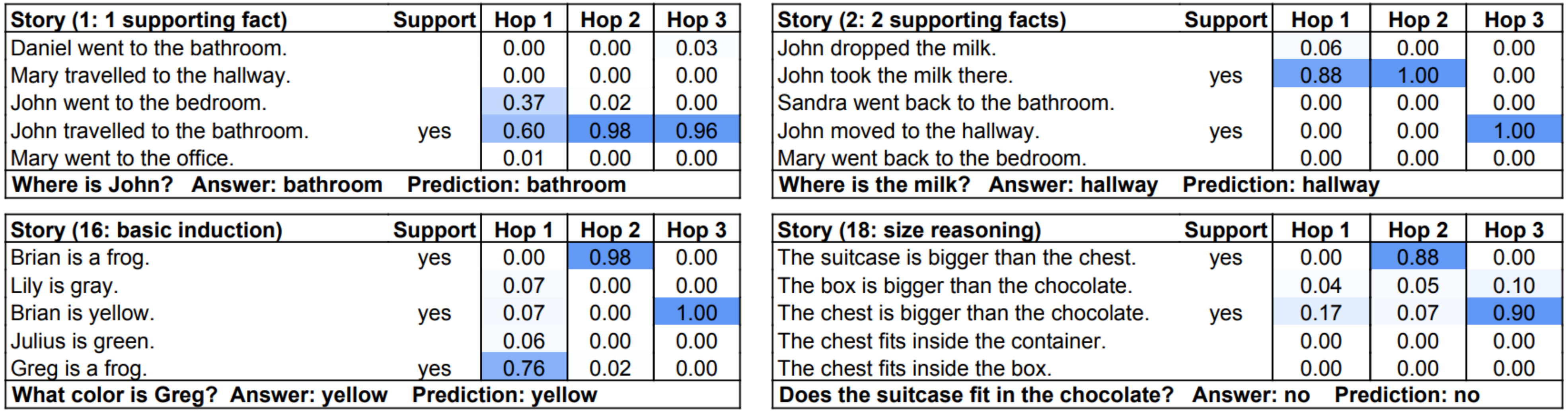
# Result (QA)

- 적용 방법에 따른 효과 분석
    - Position encoding(PE)
    - Linear start (LS)
    - Random noise (RN)
    - Joint training (Joint training on all tasks helps.)
    - Multi layers

Task that word ordering is particularly important.
Ex) qa15: basic deduction



```
1 Mice are afraid of wolves.
2 Gertrude is a mouse.
3 Cats are afraid of sheep.
4 Winona is a mouse.
5 Sheep are afraid of wolves.
6 Wolves are afraid of cats.
7 Emily is a mouse.
8 Jessica is a wolf.|
9 What is gertrude afraid of?   wolf     2 1
10 What is gertrude afraid of?  wolf     2 1
11 What is jessica afraid of?   cat      8 6
12 What is gertrude afraid of?  wolf     2 1
```



| 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** | | | | |

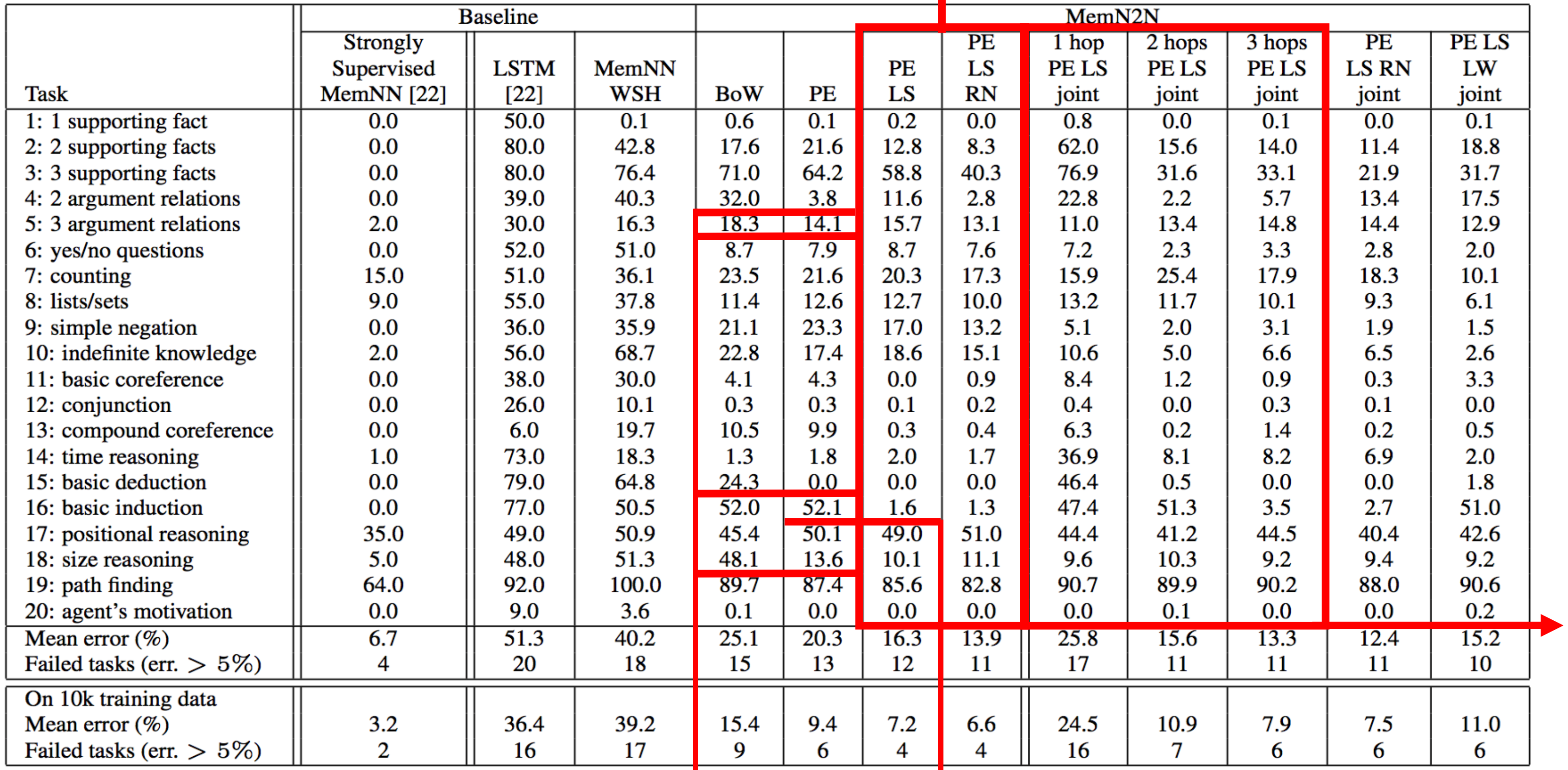
| Story (2: 2 supporting facts) | Support | Hop 1 | Hop 2 | Hop 3 |
|---|---|---|---|---|
| John dropped the milk. | | 0.06 | 0.00 | 0.00 |
| John took the milk there. | yes | 0.88 | 1.00 | 0.00 |
| Sandra went back to the bathroom. | | 0.00 | 0.00 | 0.00 |
| John moved to the hallway. | yes | 0.00 | 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** | | | | |

# Result (QA)

RN: small but consistent boost



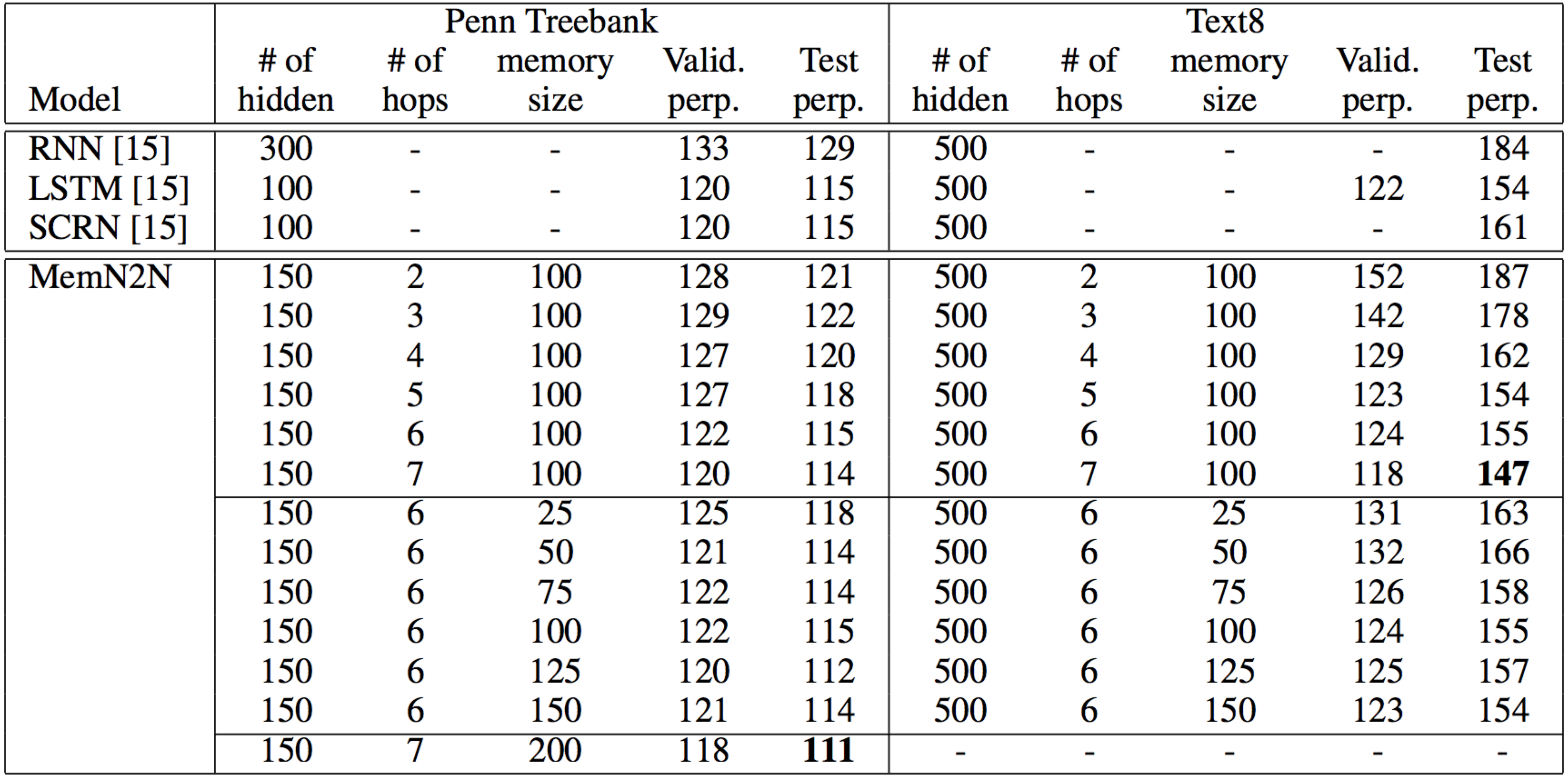
| | Baseline | | | MemN2N | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Task | Strongly Supervised MemNN [22] | LSTM [22] | MemNN WSH | BoW | PE | PE LS | PE LS RN | 1 hop PE LS joint | 2 hops PE LS joint | 3 hops PE LS joint | PE LS RN joint | PE LS LW joint |
| 1: 1 supporting 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 |

PE: improvements on tasks 4, 5, 15 and 18, where word ordering is particularly important.

LS seems to help avoid local minima.

More computational hops give improved performance.

# Result (LM)



| | Penn Treebank | | | | | Text8 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Model | # of hidden | # of hops | memory size | Valid. perp. | Test perp. | # of hidden | # of hops | memory size | Valid. perp. | Test perp. |
| RNN [15] | 300 | - | - | 133 | 129 | 500 | - | - | - | 184 |
| LSTM [15] | 100 | - | - | 120 | 115 | 500 | - | - | 122 | 154 |
| SCRN [15] | 100 | - | - | 120 | 115 | 500 | - | - | - | 161 |
| MemN2N | 150 | 2 | 100 | 128 | 121 | 500 | 2 | 100 | 152 | 187 |
| | 150 | 3 | 100 | 129 | 122 | 500 | 3 | 100 | 142 | 178 |
| | 150 | 4 | 100 | 127 | 120 | 500 | 4 | 100 | 129 | 162 |
| | 150 | 5 | 100 | 127 | 118 | 500 | 5 | 100 | 123 | 154 |
| | 150 | 6 | 100 | 122 | 115 | 500 | 6 | 100 | 124 | 155 |
| | 150 | 7 | 100 | 120 | 114 | 500 | 7 | 100 | 118 | **147** |
| | 150 | 6 | 25 | 125 | 118 | 500 | 6 | 25 | 131 | 163 |
| | 150 | 6 | 50 | 121 | 114 | 500 | 6 | 50 | 132 | 166 |
| | 150 | 6 | 75 | 122 | 114 | 500 | 6 | 75 | 126 | 158 |
| | 150 | 6 | 100 | 122 | 115 | 500 | 6 | 100 | 124 | 155 |
| | 150 | 6 | 125 | 120 | 112 | 500 | 6 | 125 | 125 | 157 |
| | 150 | 6 | 150 | 121 | 114 | 500 | 6 | 150 | 123 | 154 |
| | 150 | 7 | 200 | 118 | **111** | - | - | - | - | - |

# Conclusion

- Contribution
  - Weakly supervised
  - Soft attention
  - 범용성 (diverse tasks)
  - Weakly supervised baseline 모델에 비해 뛰어난 성능

- Limitation
  - Strong supervised model을 능가하지는 못함
  - Fail on several of the 1k QA tasks
  - Large momory가 필요한 경우 문제