Comparison of Different Attention and Output Layer Strategies for Question Answering

Ashton Teng
Microsoft Bing Core Relevance
08/17/2017

The SQuAD Dataset

- The Stanford Question Answering Dataset for reading comprehension
- 100,000+ question-answer pairs on 500+ Wikipedia articles
- 1 datapoint: (paragraph, question, answer start idx, answer end idx)

The Yuan dynasty (Chinese: 元朝; pinyin: Yuán Cháo), officially the Great Yuan (Chinese: 大元; pinyin: Dà Yuán; Mongolian: Yehe Yuan Ulus[a]), was the empire or ruling dynasty of China established by Kublai Khan, leader of the Mongolian Borjigin clan. Although the Mongols had ruled territories including today's North China for decades, it was not until 1271 that Kublai Khan officially proclaimed the dynasty in the traditional Chinese style. His realm was, by this point, isolated from the other khanates and controlled most of present-day China and its surrounding areas, including modern Mongolia and Korea. It was the first foreign dynasty to rule all of China and lasted until 1368, after which its Genghisid rulers returned to their Mongolian homeland and continued to rule the Northern Yuan dynasty. Some of the Mongolian Emperors of the Yuan mastered the Chinese language, while others only used their native language (i.e. Mongolian) and the 'Phags-pa script.

What is the Chinese name for the Yuan dynasty?
Ground Truth Answers: Yuán Cháo | Yuán Cháo | 元朝

What is the Yuan dynasty's official name?
Ground Truth Answers: the Great Yuan | the Great Yuan | the Great Yuan

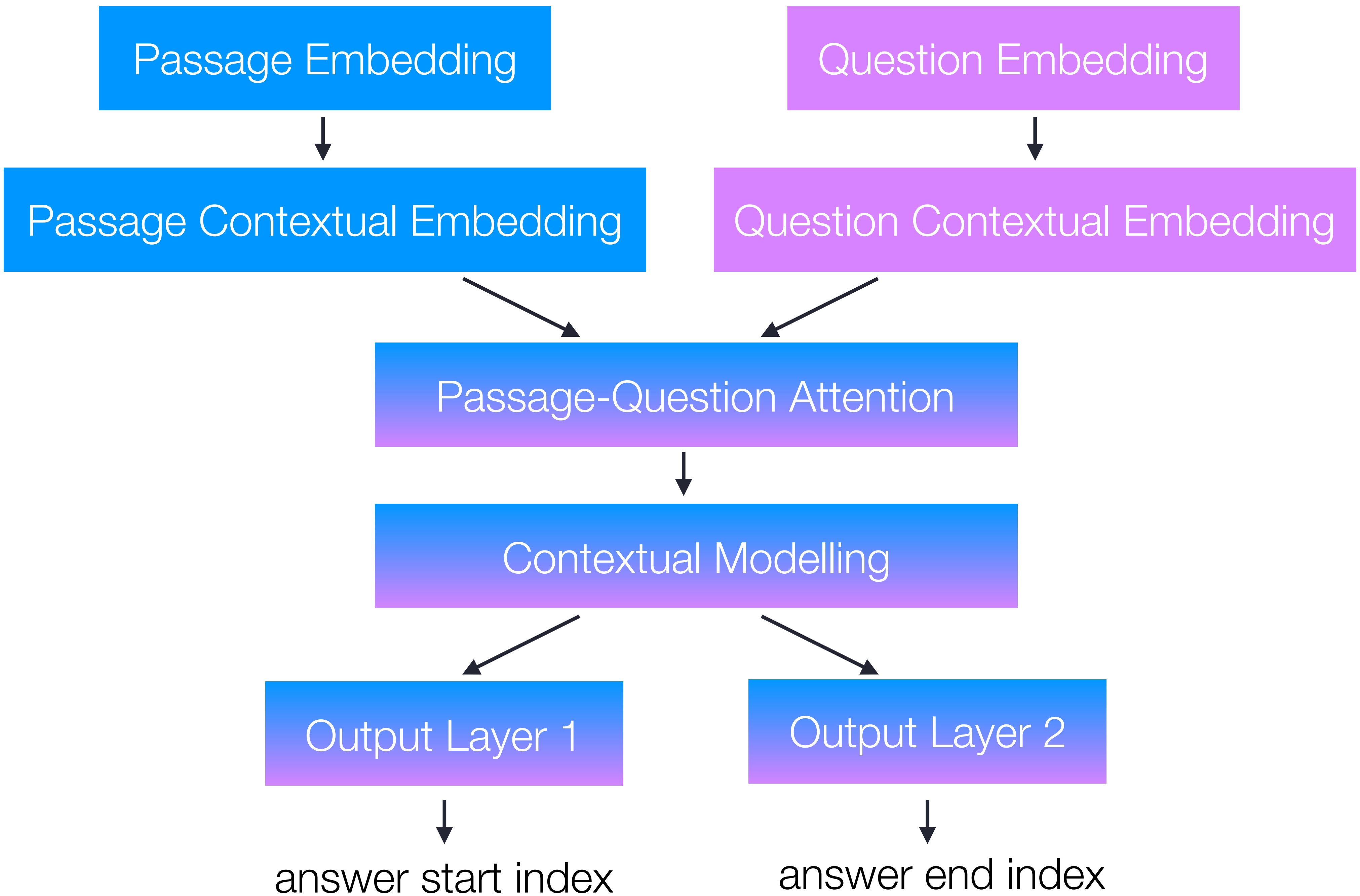
Who started the Yuan dynasty?
Ground Truth Answers: Kublai Khan | Kublai Khan | Kublai Khan

Who led the Mongolian Borjigin clan?
Ground Truth Answers: Kublai Khan | Kublai Khan | Kublai Khan

When did Khan formally declare the Yuan dynasty?
Ground Truth Answers: 1271 | 1271 | 1271

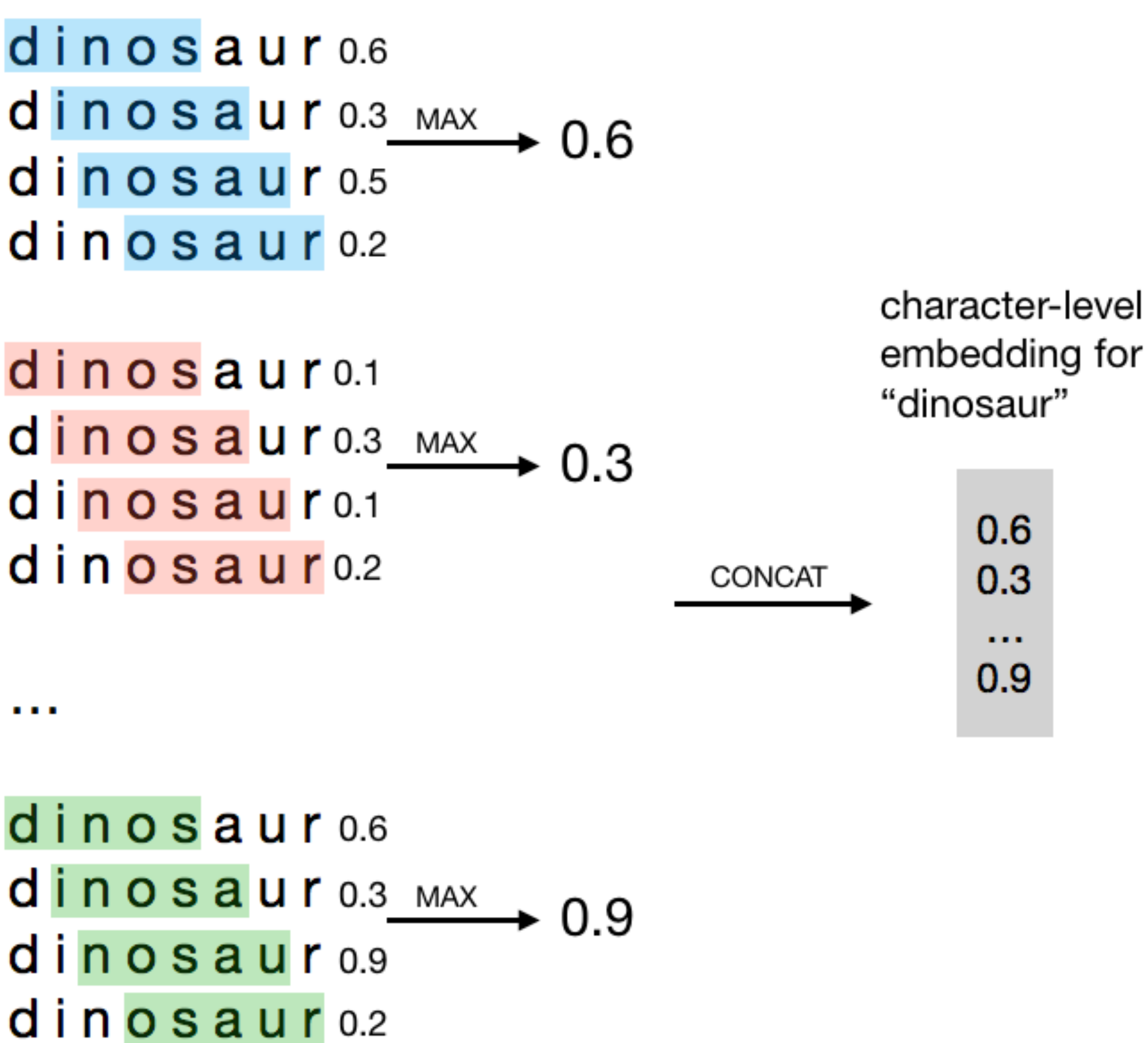
http://rajpurkar.github.io/SQuAD-explorer/

General Model Structure



Passage and Question Character Embedding

- Character-level embeddings with CNNs
    - [N, L, W] —> [N, L, W, d]
    - Each filter color acts like a detector for a different feature



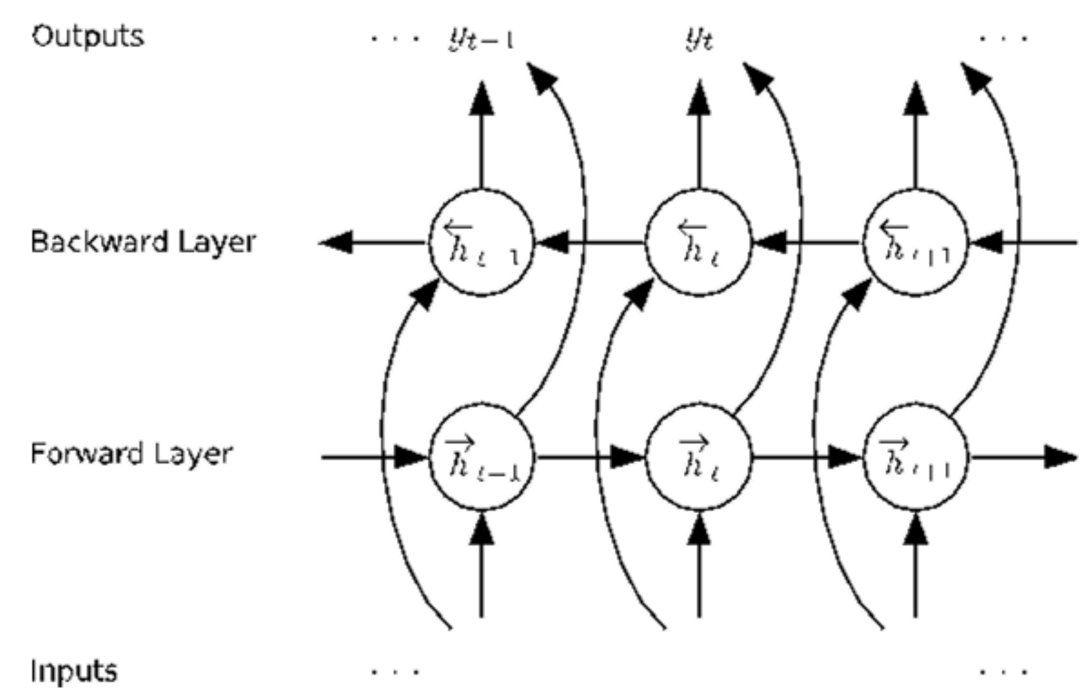
- I used 100 filters, all with length 5

Passage and Question Word Embedding

- Word-level embeddings with GLoVE (Global Vectors for Word Representation)
  - Pre-trained word vectors from unsupervised algorithms crawling Wikipedia. Available in d = 50, 100, 200, 300. I used 100.
  - Each word in the vocabulary is assigned an integer and a corresponding row in the embedding matrix.
  - Give a passage of integers, use tf.embedding_lookup to embed it into a passage of vectors. [N, L] —> [N, L, d]

Passage and Question Contextual Embedding

- Provide positional information to each word's vector, about where it is in the passage/question
- Bidirectional RNN
    - Input: $P_{emb} = [p_{emb1}, p_{emb2} \ldots p_{embp}] \in \mathbb{R}^{p \times 2d}$
      $Q_{emb} = [q_{emb1}, q_{emb2} \ldots q_{embq}] \in \mathbb{R}^{q \times 2d}$
      *concatenate word+char representations
    - Output: $P = [p_1, p_2 \ldots p_p] \in \mathbb{R}^{p \times d}$
      $Q = [q_1, q_2 \ldots q_q] \in \mathbb{R}^{q \times d}$



Attention Layer

- Fuses information about the passage and question together, determining the strength between each passage word and each question word.
- Given the question, which words in the passage are important?
- Two types:
  - Dynamic attention: step though the passage words with RNN, drawing attention over question words at each step.
  - Static attention: *func*(q_i, p_j) = *score*, produce similarity matrix then normalize across rows and columns to get weights.

Dynamic Attention with RNNs

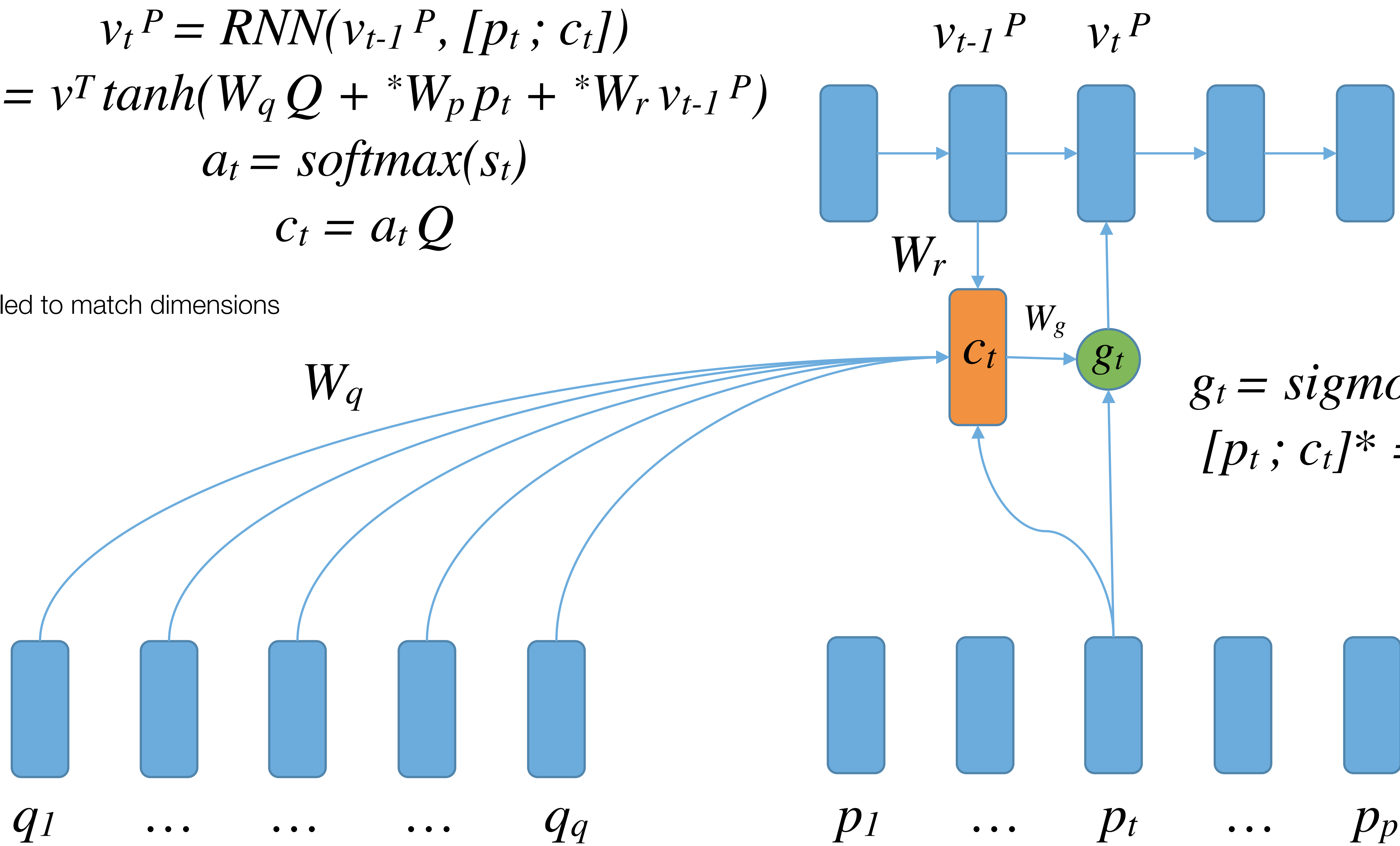
$$v_t^{P} = RNN(v_{t-1}^{P}, [p_t \, ; c_t])$$

$$s_t = v^T \tanh(W_q Q + {}^{*}W_p p_t + {}^{*}W_r v_{t-1}^{P})$$

$$a_t = softmax(s_t)$$

$$c_t = a_t Q$$

* tiled to match dimensions



$$g_t = sigmoid(W_g[p_t \, ; c_t])$$

$$[p_t \, ; c_t]^* = g_t \odot [p_t \, ; c_t]$$

$$\boxed{V^P = [v_1^{P}, v_2^{P}, \ldots v_p^{P}] \in \mathbb{R}^{p \times d}}$$

https://arxiv.org/abs/1611.01604

Static Attention 1: Dynamic Co-attention Network

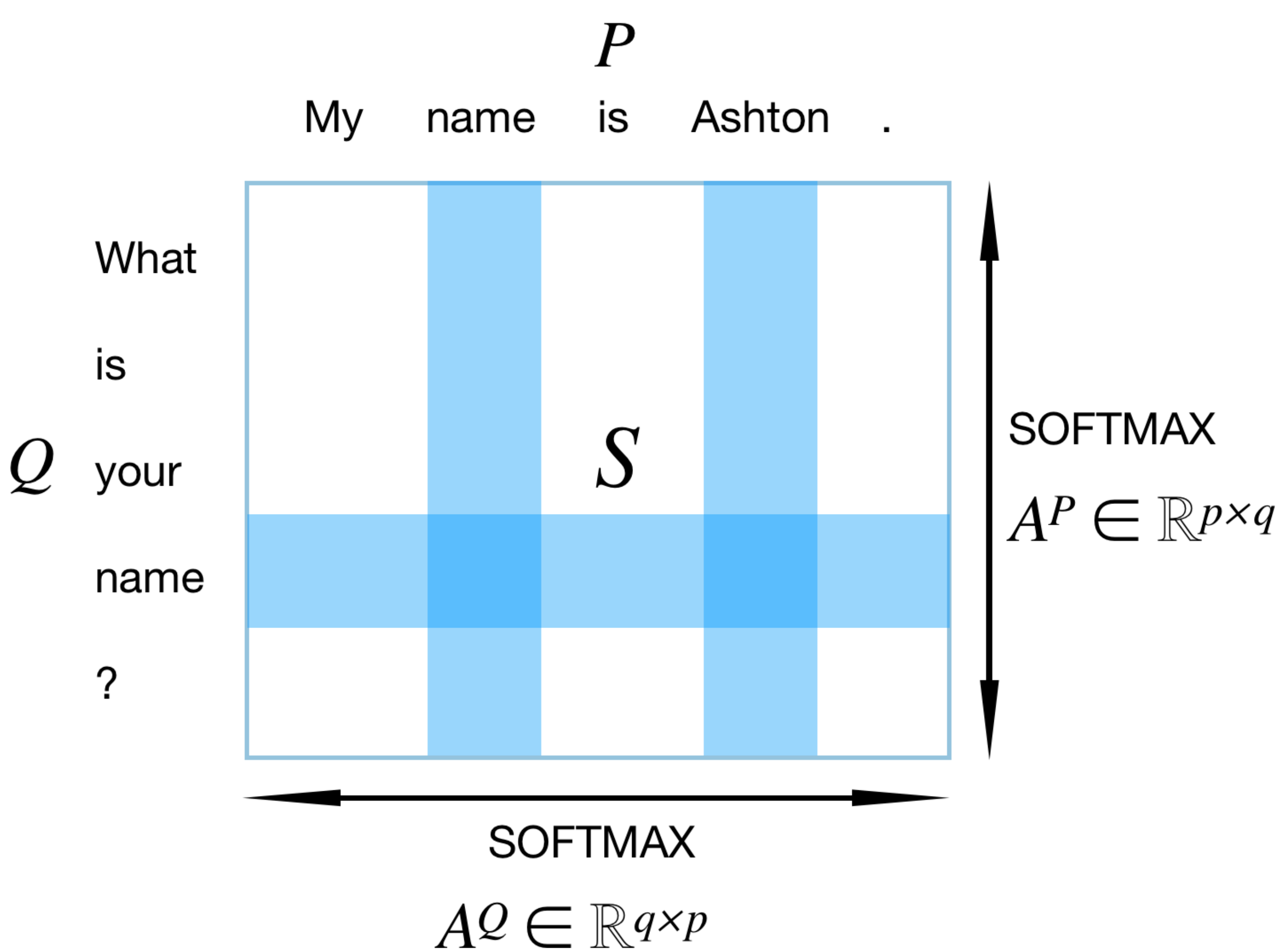
$$P = [p_1, p_2 \ldots p_p] \in \mathbb{R}^{p \times d}$$

$$Q = [q_1, q_2 \ldots q_q] \in \mathbb{R}^{q \times d}$$

$$S_{ij} = p_i \cdot q_j$$

A^Q: for each question word, weight all passage words by importance.

A^P: for each passage word, weight all question words by importance.



$$C^Q = A^Q P \in \mathbb{R}^{q \times d}$$

For each question word, sum up all the passage word representations according to their importance to this question word.
"passage summarization for each question word"

$$C^P = A^P Q \in \mathbb{R}^{p \times d}$$

For each passage word, sum up all the question word representations according to their importance to this passage word.
"question summarization for each passage word"

$$A^P \cdot C^Q \in \mathbb{R}^{p \times d}$$

For each passage word, summarize C^Q, weighting the question-dependent passage representations

$$\boxed{V^P = A^P [C^Q; Q] \in \mathbb{R}^{p \times 2d}}$$

https://arxiv.org/abs/1611.01604

Static Attention 2: Bi-directional Attention Flow

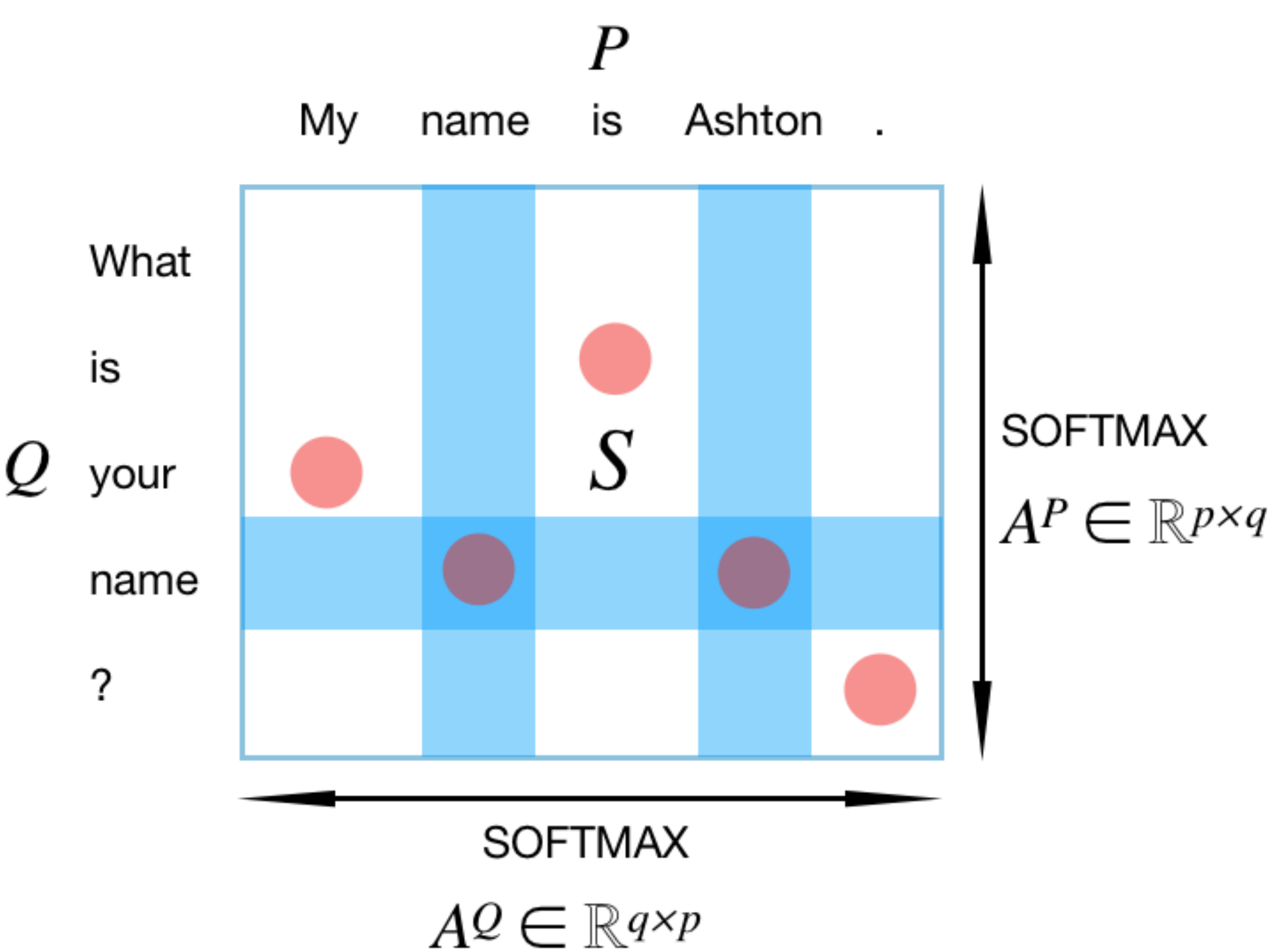
$$P = [p_1, p_2 \ldots p_p] \in \mathbb{R}^{p \times d}$$

$$Q = [q_1, q_2 \ldots q_q] \in \mathbb{R}^{q \times d}$$

$$S_{ij} = w^T [p_i \,;\, q_j \,;\, p_i \odot q_j]$$

A^Q: for each question word, weight all passage words by importance.

A^P: for each passage word, weight all question words by importance.



$$C^P = A^P Q \in \mathbb{R}^{p \times d}$$

For each passage word, sum up all the question word representations according to their importance to this passage word. "question summarization for each passage word"

$$b = softmax\,(max_Q(S)) \in \mathbb{R}^{1 \times p}$$

For each passage word, find the question word with the strongest similarity. Do this for all passage words, and form a normalized vector.

$$G^p = b\,P \in \mathbb{R}^{p \times d}$$

Weight each passage word representation by its global importance - is it important for at least one question word?

$$\boxed{V^P = [P \,;\, C^P \,;\, P \odot C^P ; P \odot G^p] \in \mathbb{R}^{p \times 4d}}$$

https://arxiv.org/abs/1611.01603

Attention Comparison

- bidirectional attention flow:
  - EM = 66.48%, F1 = 76.64%
- co-attention:
  - EM = 66.12%, F1 = 75.87%
- dynamic attention:
  - EM = 62.23%, F1 = 73.04%

Gated Combination Attention Layer

dynamic co-attention: $V^P = A^P\,[C^Q;Q] \in \mathbb{R}^{p\times 2d}$

bidirectional attention: $V^P=[P\,;\,C^P\,;\,P \odot C^P;P \odot G^p] \in \mathbb{R}^{p\times 4d}$

def X|Y:

  gate = sigmoid($W_1X + W_2Y + b_1$*)*

  transform = tanh($W_3X + W_4Y + b_2$*)*

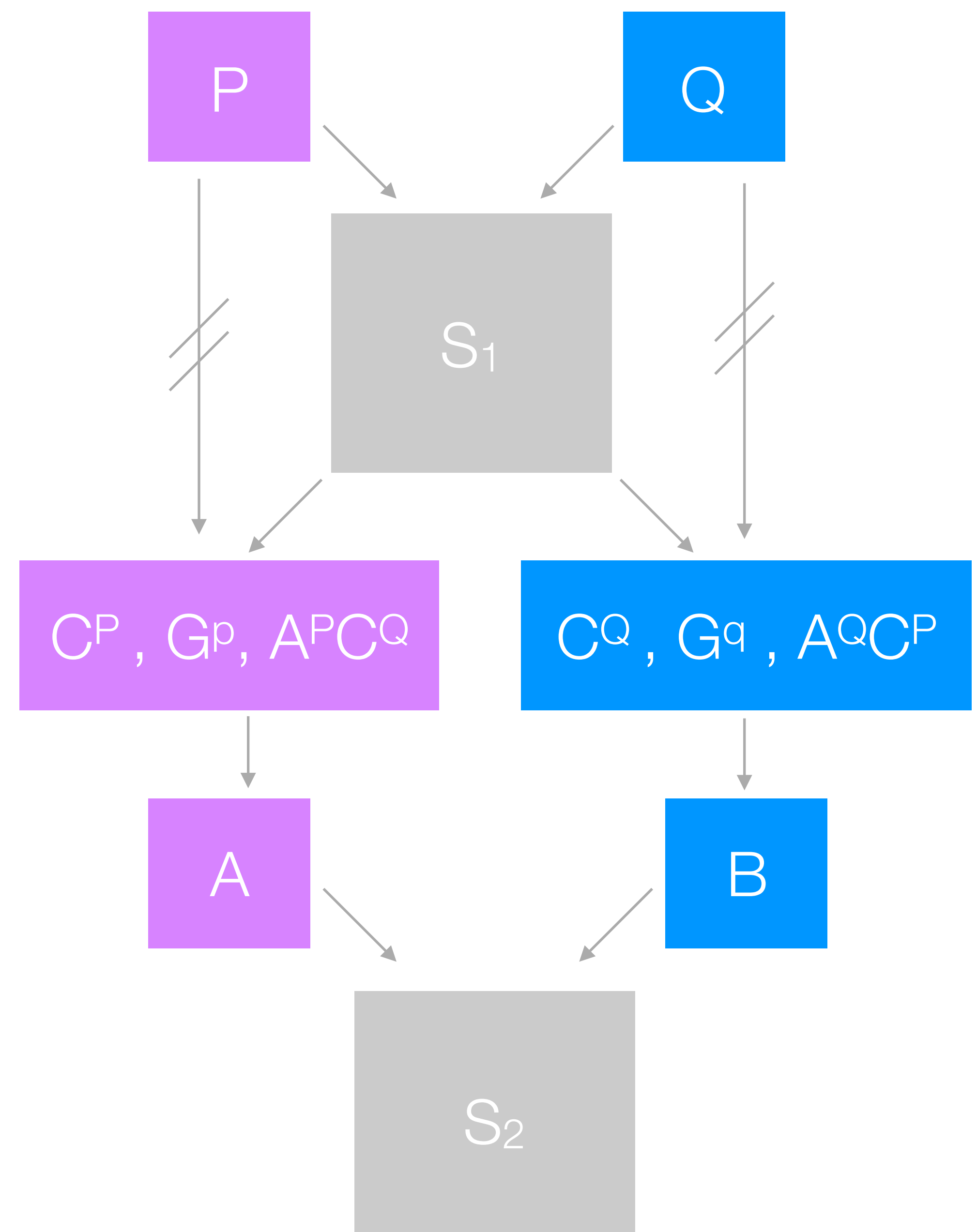
  return gate $\odot$ *X + (1-gate)* $\odot$ *transform*

$$A = [P|C^P\,;\,P|G^p;\,P|A^PC^Q] \in \mathbb{R}^{p\times 3d}$$

$$B = [Q|C^Q\,;\,Q|G^q\,;\,Q|A^QC^P] \in \mathbb{R}^{q\times 3d}$$

$$A^A, C^B, B = \textit{co-attention}\,(A, B)$$

$$\boxed{V^P= A^A\,[C^B;B] \in \mathbb{R}^{p\times 6d}}$$



Highway Networks https://arxiv.org/abs/1505.00387
Ruminating Reader https://arxiv.org/abs/1704.07415

Contextual Modeling

- Pass V^P as input into another bidirectional RNN, to reinforce positional information, outputting $M \in \mathbb{R}^{p \times 2d}$

Output Layer

- Bidirectional Attention Flow approach:

$$p_1 = softmax(w_{p1}^T\, [V^P; M]) \in \mathbb{R}^{1\times p}$$

$$M_2 = BiRNN(M)$$

$$p_2 = softmax(w_{p2}^T\, [V^P; M_2]) \in \mathbb{R}^{1\times p}$$

- My approach, multilayer perception + maxout:

def mlp_maxout(X):

$$layer_1 = ReLU(XW_1 + b_1),\ W_1 \in \mathbb{R}^{8d\times 8d}$$

$$layer_2 = ReLU(layer_1 W_2 + b_2),\ W_2 \in \mathbb{R}^{8d\times 8d}$$

$$all\_models = layer_2 W_3 + b_3,\ W_3 \in \mathbb{R}^{8d\times 16}$$

return max(all_models, dim=-1)

$$p_1 = mlp\_maxout([V^P; M]),\ p2 = mlp\_maxout([V^P; M]), \in \mathbb{R}^{1\times p}$$

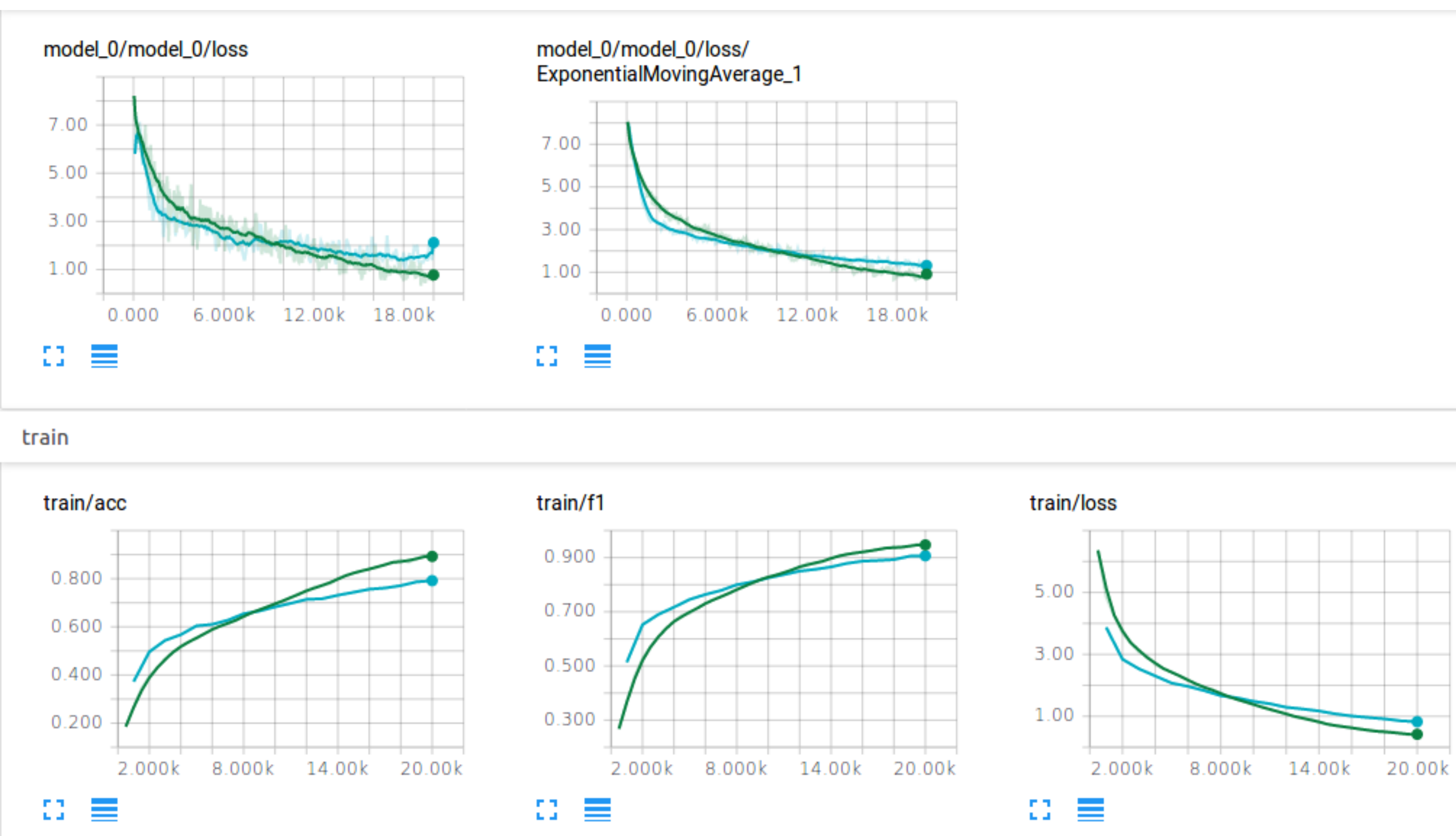
- Loss = softmax_cross_entropy_with_logits(p_1, $labels_1$) + tf.softmax_cross_entropy_with_logits(p_2, $labels_2$)

Parameters

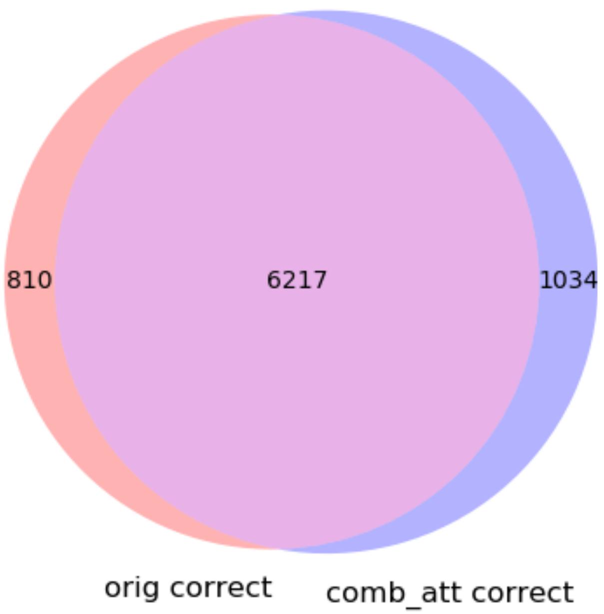
- batch_size = 64
- initial_learning_rate = 0.001
- exp_var_decay = 0.999
- optimizer = AdamOptimizer
- dropout = 0.2

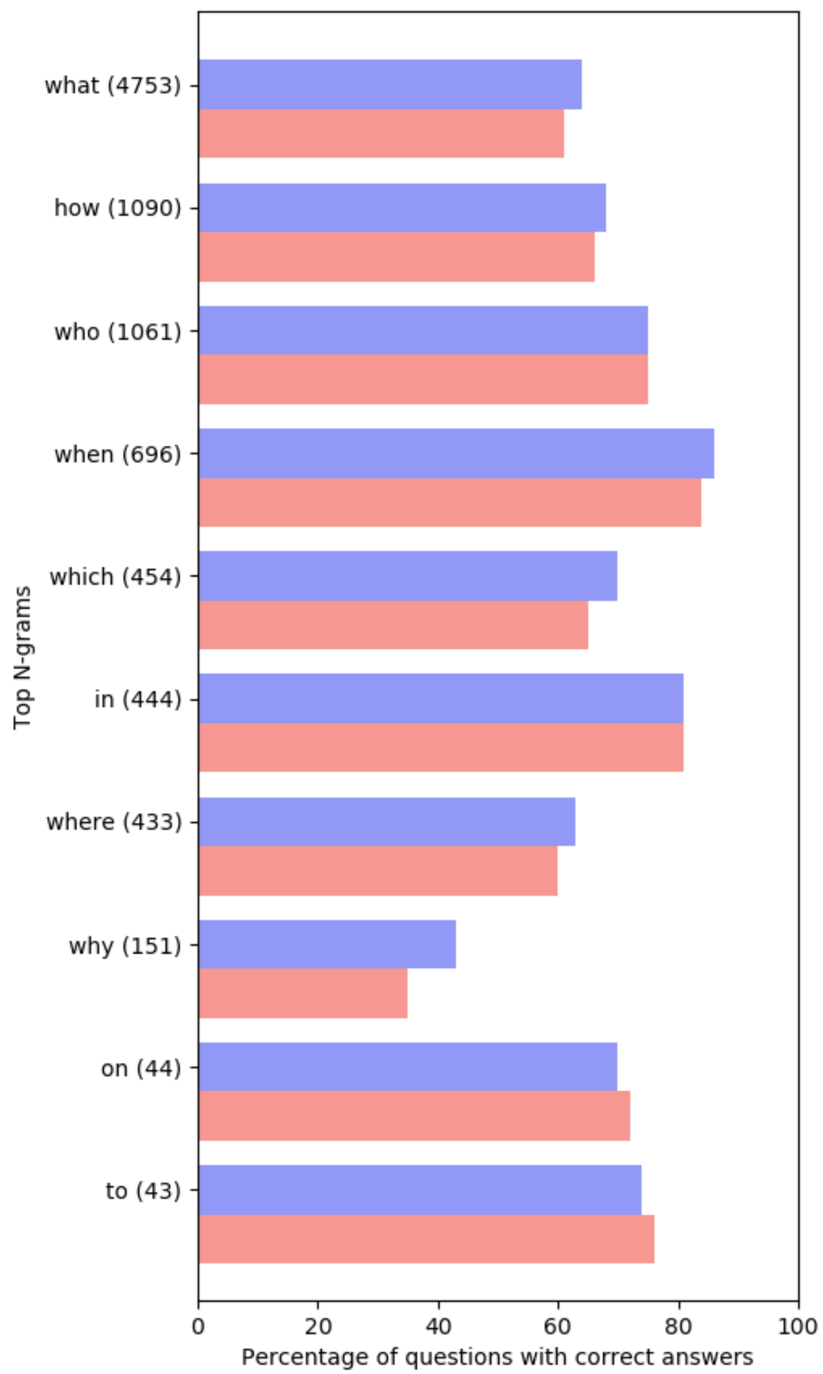
Results

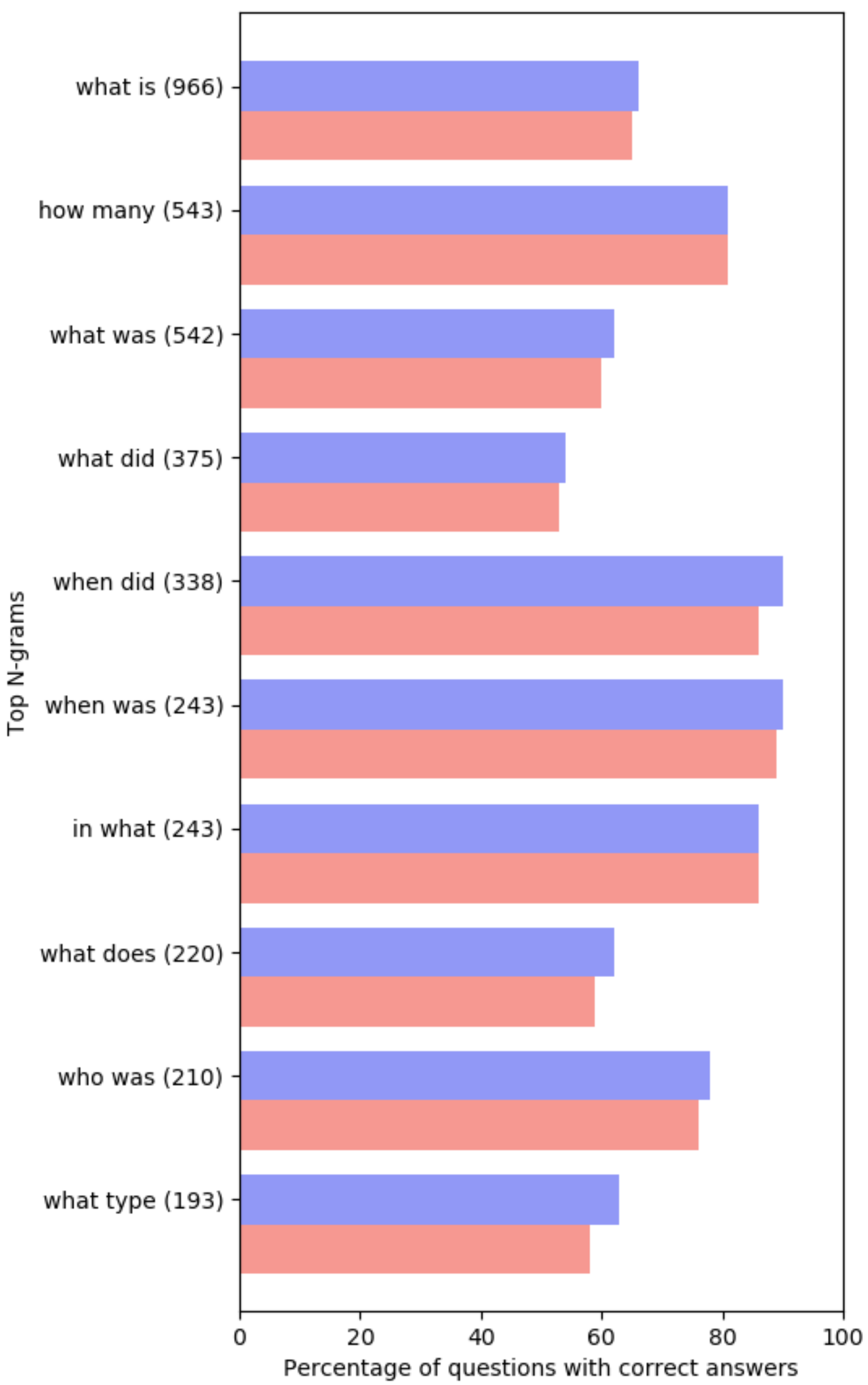
- bidirectional attention flow:
    - EM = 66.48%, F1 = 76.64%
- me:
    - EM = 68.69%, F1 = 77.99%



Results


810
6217
1034
orig correct
comb_att correct


what (4753)
how (1090)
who (1061)
when (696)
which (454)
in (444)
where (433)
why (151)
on (44)
to (43)
Top N-grams
0
20
40
60
80
100
Percentage of questions with correct answers


what is (966)
how many (543)
what was (542)
what did (375)
when did (338)
when was (243)
in what (243)
what does (220)
who was (210)
what type (193)
Top N-grams
0
20
40
60
80
100
Percentage of questions with correct answers

Things I failed at

- dynamic decoder + highway maxout in Dynamic Contention Networks
- self-matching attention in R-Net

Thank You!