Teach Machine to Comprehend Text

Feng Wang @ Tencent AI
wangfelix87@gmail.com
March 16, 2019

Outline

- Problem Definition
- Network Architecture
- Open Question

Different Tasks

- To identify a candidate answer from a set of candidates (MCTest)
- To identify a word from passage as the final answer (CNN/Daily Mail)
- To identify **a subsequence** words from the passage as the answer (SQuAD)
- To answer the question given a set of passages, and the answer is not necessarily sub-span of the passages (MS-MARCO)

An Example from the SQuAD dataset

Passage: In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, graupel and hail. .....

Question: *What causes precipitation to fall?*

Answer: gravity

Problem Definition

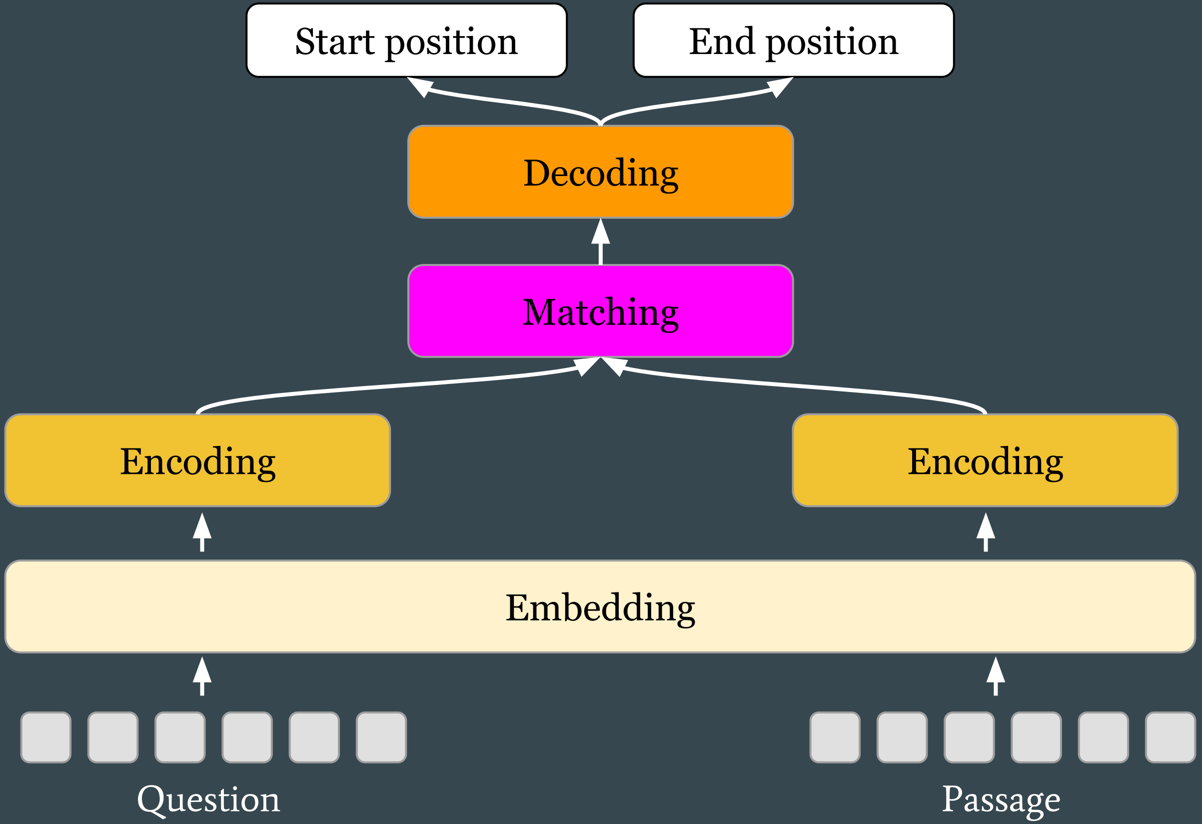
Given a pair of passage and question (p_i, q_i), construct a matching/mapping function $f(\cdot)$ to identify answer (s_i, e_i) (start and end position in passage p_i), which an be formally defined as:

$$(s_i, e_i) = f(p_i, q_i)$$

Our training data can be represented as $T = \{p_i, q_i, (s_i, e_i)\}_i^N$, the optimal solution f^* is defined as

$$f^* := \arg\min_f \sum_{i=1}^{N} \mathcal{L}(f(p_i, q_i), (s_i, e_i))$$

Network Architecture



Embedding Layer

- **Word-level embeddings**
  - A pre-trained embedding matrix
  - A trainable model initialized from a pre-trained embedding matrix
- **Character-level embeddings**
  - Generated by taking the final hidden states of a bidirectional RNN applied to embeddings of characters in the token.
  - Such character-level embeddings have been shown to be helpful to deal with out-of-vocab (OOV) tokens.

Encoding Layer

- ~~RNN~~
- ~~LSTM~~
- GRU
- CNN
- Transformer

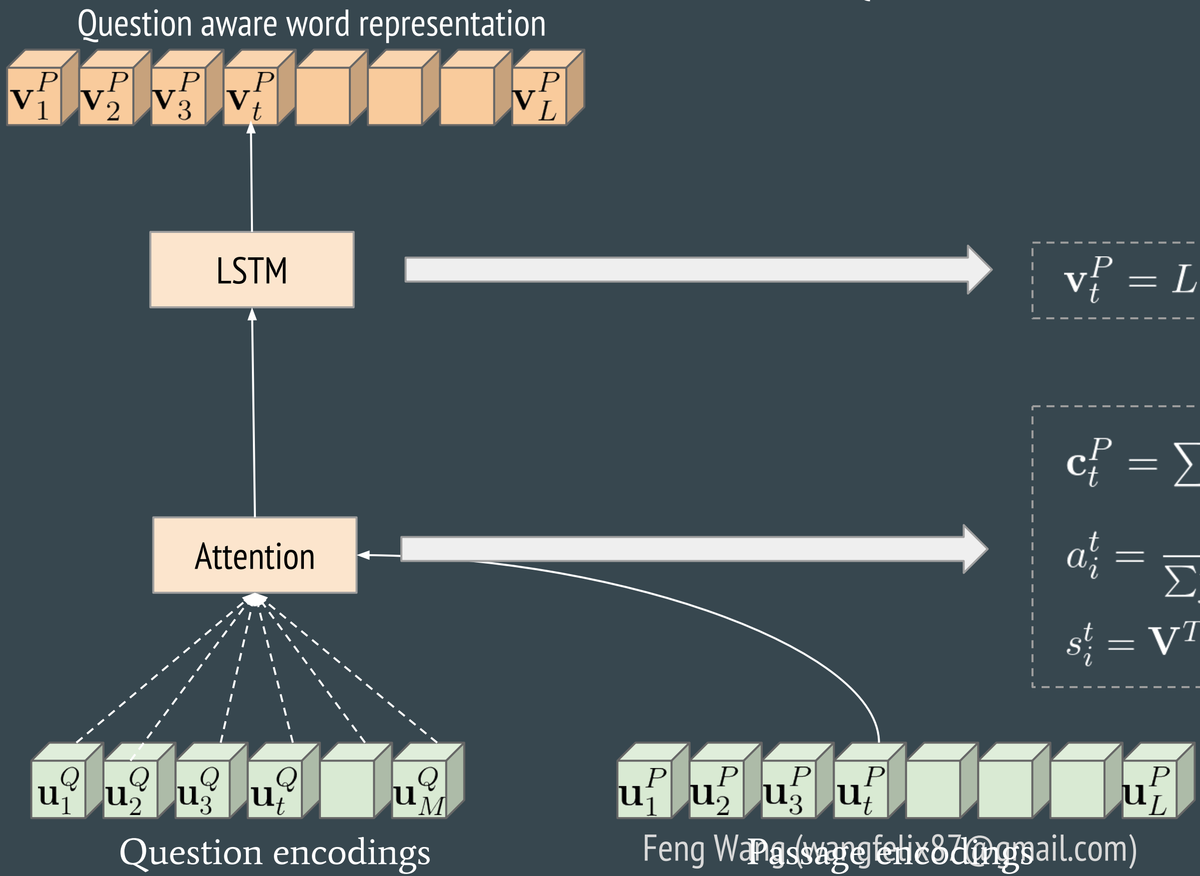
Matching Layer

Intuitively, not all words are equally useful for answering the question. Therefore, the sequence of passage vectors need to be weighted according to their relations to the question.

Attention mechanism:

- Word-by-Word Attention (*Rocktaschel* et al., 2015)
- Match-LSTM Layer (*Wang & Jiang* 2016)
- Gated attention-based Recurrent Network (*Wang, W., et al.*, 2017)
- Bi-Directional Attention Flow (BIDAF) (*Hasan & Fischer*, 2018)

#1: Word-by-Word Attention (Rocktaschel et al. 2015)



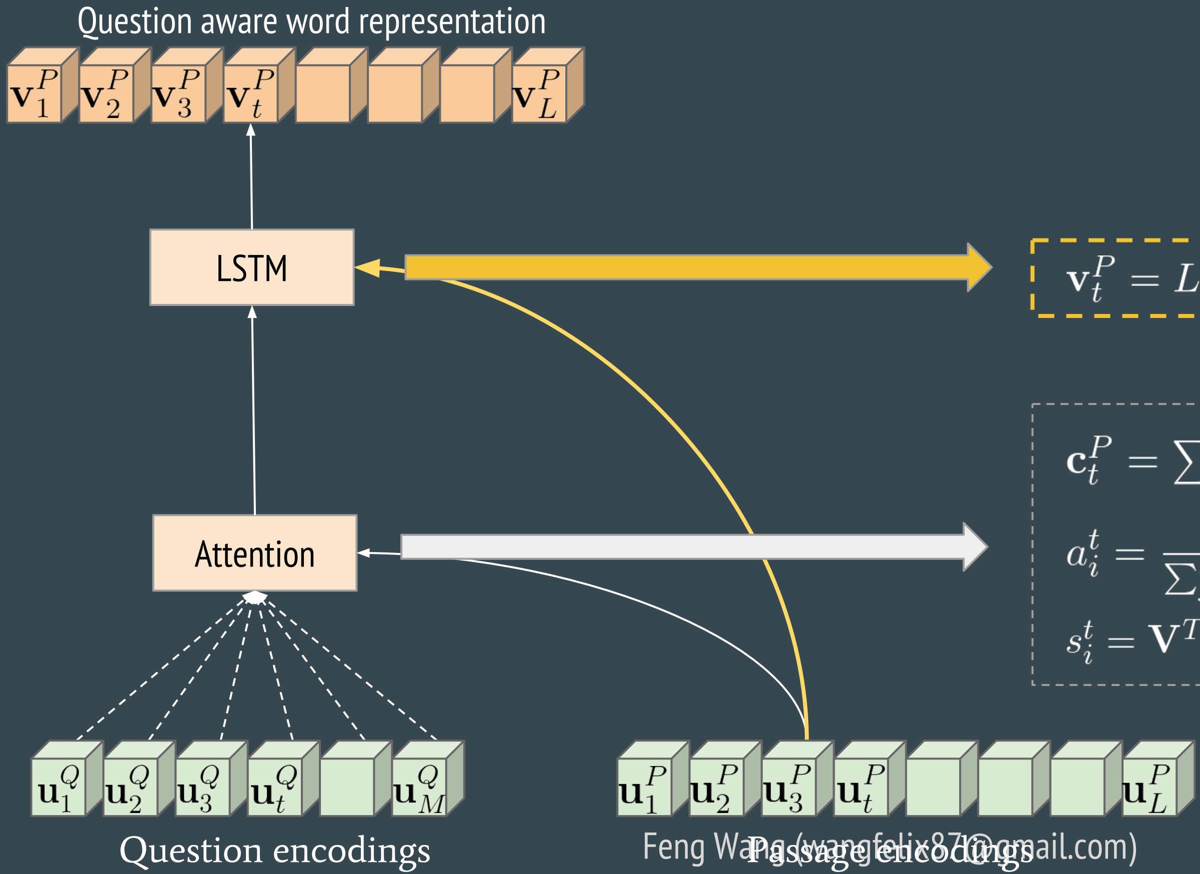
$$\mathbf{v}_t^P = LSTM(\mathbf{v}_{t-1}^P, \mathbf{c}_t^P)$$

$$\mathbf{c}_t^P = \sum_{i=1}^{M} a_i^t \mathbf{u}_i^Q$$

$$a_i^t = \frac{\exp(\mathbf{s}_i^t)}{\sum_{j=1}^{M} \exp(\mathbf{s}_j^t)}$$

$$s_i^t = \mathbf{V}^T \tanh(\mathbf{W}_u^Q \mathbf{u}_j^Q + \mathbf{W}_u^P \mathbf{u}_t^P + \mathbf{W}_v^P \mathbf{v}_{t-1}^P)$$

#2 Match-LSTM (Wang & Jiang 2016)



$$\mathbf{v}_t^P = LSTM(\mathbf{v}_{t-1}^P, [\mathbf{u}_t^P, \mathbf{c}_t^P])$$

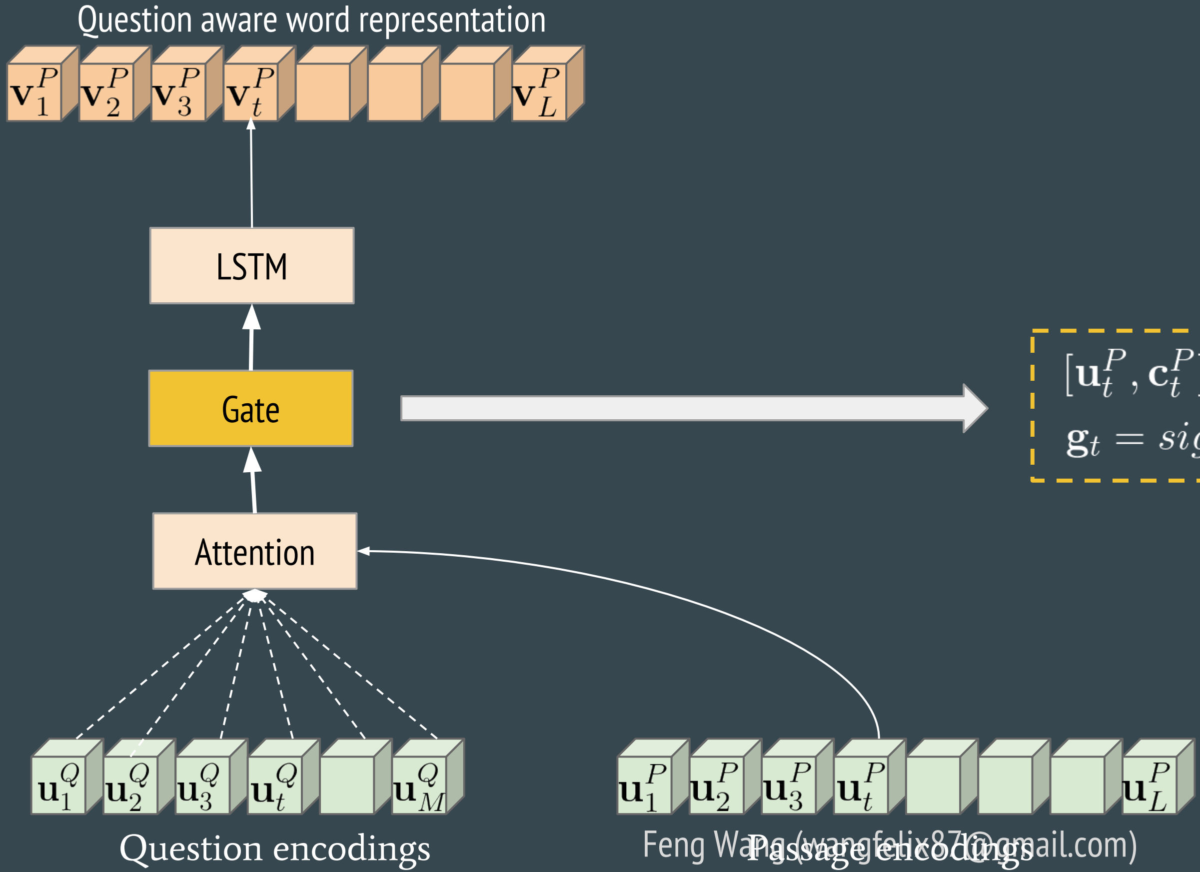
$$\mathbf{c}_t^P = \sum_{i=1}^M a_i^t \mathbf{u}_i^Q$$

$$a_i^t = \frac{\exp(\mathbf{s}_i^t)}{\sum_{j=1}^M \exp(\mathbf{s}_j^t)}$$

$$s_i^t = \mathbf{V}^T \tanh(\mathbf{W}_u^Q \mathbf{u}_j^Q + \mathbf{W}_u^P \mathbf{u}_t^P + \mathbf{W}_v^P \mathbf{v}_{t-1}^P)$$

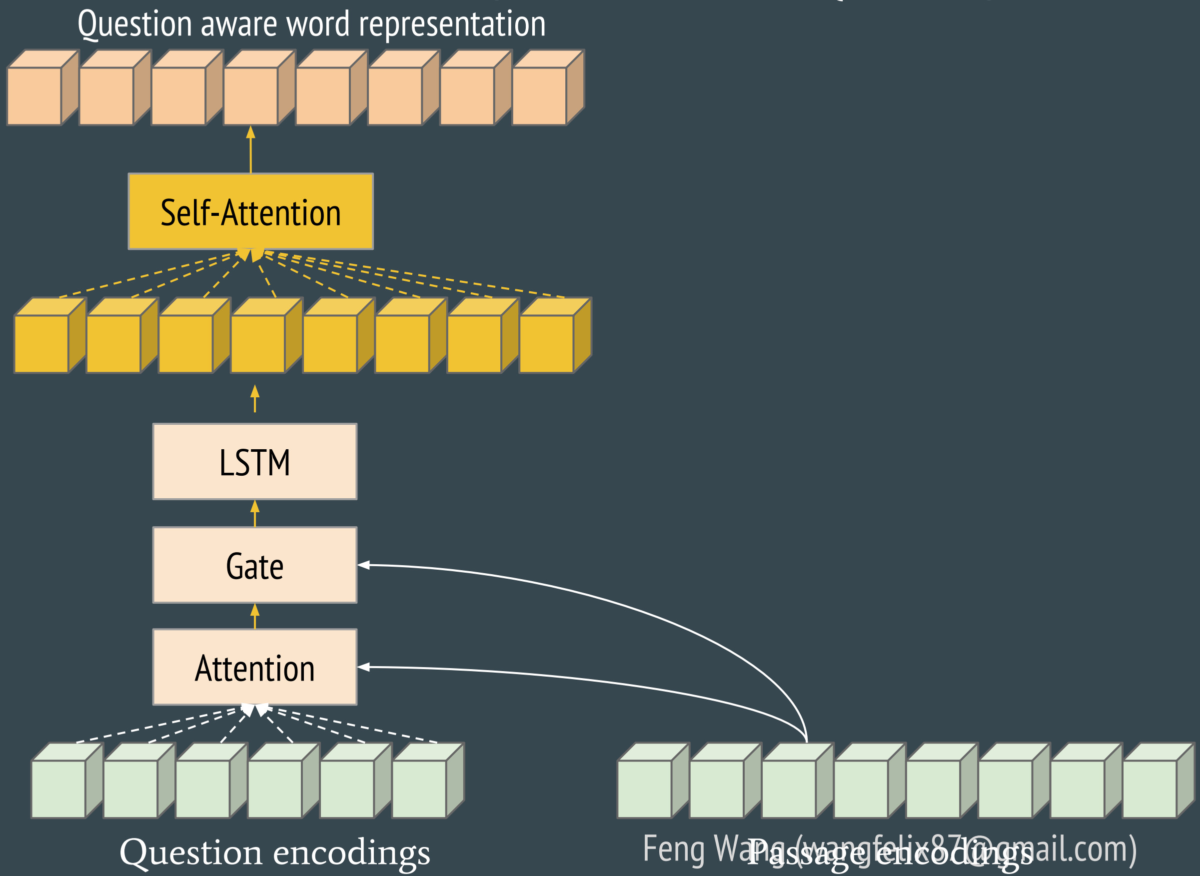
Feng Wang (wangfelix87@gmail.com)

#3 Gated Attention-based RNN (Wang, W., et al., 2017)



$$[\mathbf{u}_t^P, \mathbf{c}_t^P]* = \mathbf{g}_t \bigodot [\mathbf{u}_t^P, \mathbf{c}_t^P]$$
$$\mathbf{g}_t = sigmoid(\mathbf{W}_g[\mathbf{u}_t^P, \mathbf{c}_t^P])$$

#4 Self-Matching Attention (Wang, W., et al., 2017)

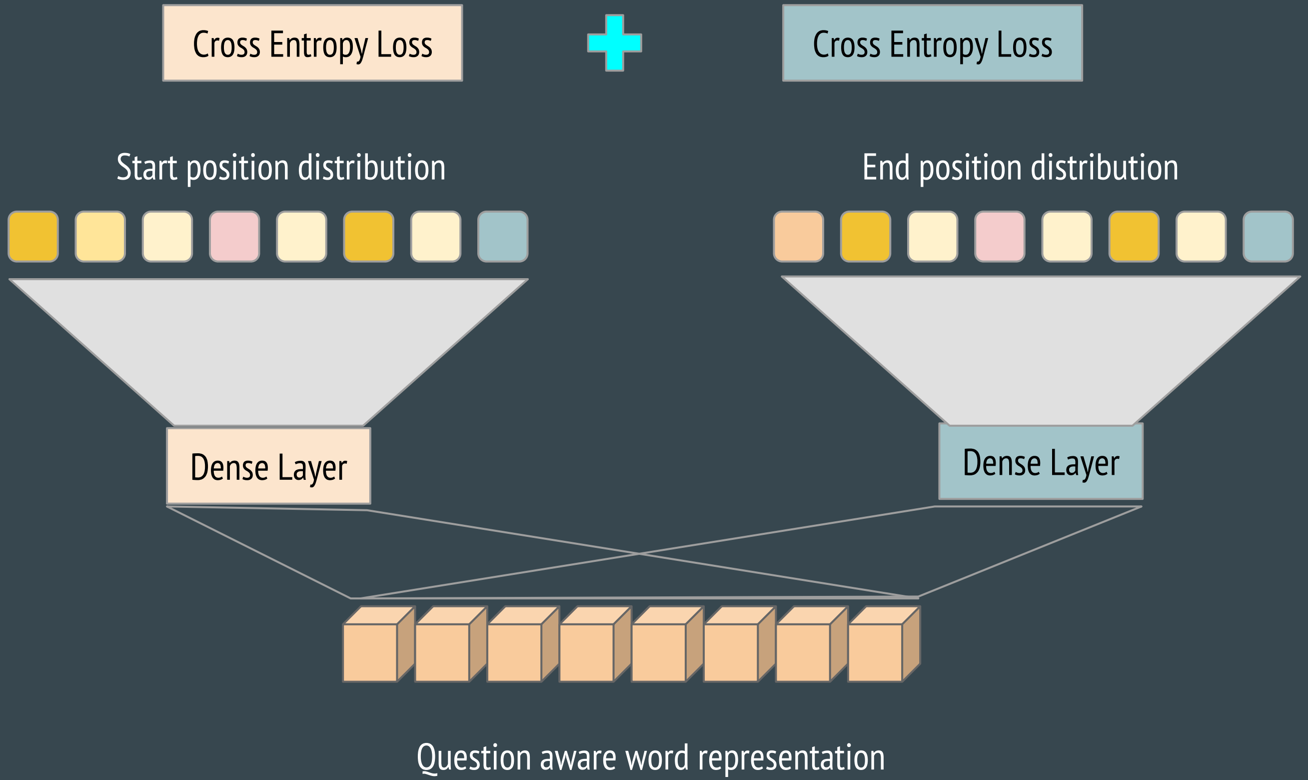


Decoding Layer

The final step is to decode output of match layer as an answer span, i.e., two discrete distributions $\mathbf{s}_i, \mathbf{e}_i$ over [0, L), which represent the start and end probability on every position of a L-length passage, respectively.

- Extraction vs Generation mechanism
- Extraction: Pointer network
- Generation: Seq-2-Seq transductive model

Decoding Layer & Loss Function



Feng Wang (wangfelix87@gmail.com)

The Open Question

- Does the model really understand the question and passage?
- Where is the future direction?
  - More good quality training data
  - Wide and Deep Model
  - Inferencing and Reasoning ability
  - Attractive approach vs Generative approach

About Me

Feng Wang (Felix)