Dynamic Coattention Networks for Question Answering

Caiming Xiong   Victor Zhong   Richard Socher

Saleforce Research

ICLR, 2017

Presenter: Tianlu Wang

Outline

1. Introduction
   - Question Answering
   - Related Work
2. Model
   - Overview
   - Document and Question Encoder
   - Coattention Encoder
   - Dynamic Pointing Decoder
3. Experiments
   - Qualitative Examples
   - Results
4. Summary

Outline

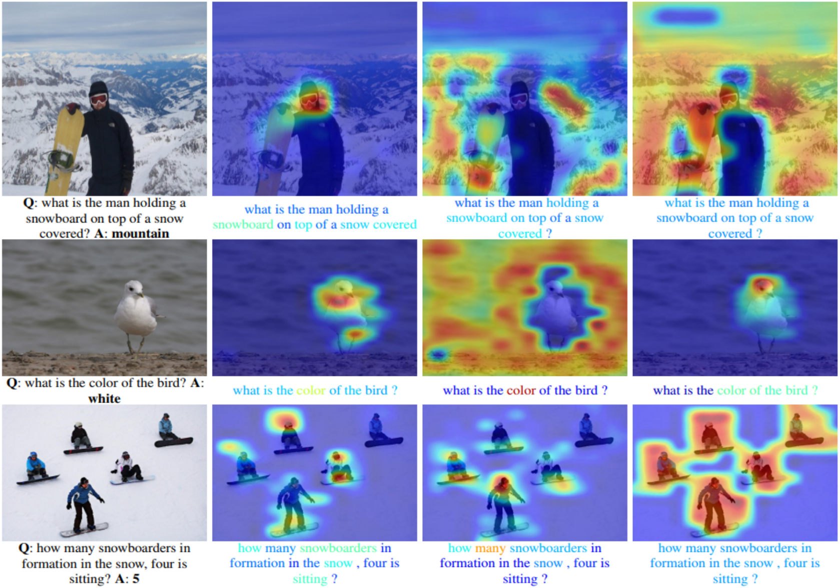
Question Answering

- Human annotated high quality but small dataset
- Large scale dataset through semi-annotated techniques but far from natural language
- Stanford Question Answering dataset(SQuAD)
  - Larger than all previous hand-annotated datasets
  - Various qualities
  - Answers are spans in a reference document

Outline

Related Work

- Statistical QA
  - Rule-based algorithms
  - Linear classifiers over feature sets: lexical features(bag of words), word distance, word order, pos_tag, dependency parse
- Neural QA
  - NLI(natural language inference) : match LSTM encoder + pointer network decoder,
  - dynamic chunk reader: extract answer candidates and rank
  - hierarchical co-attention model


Q: what is the man holding a snowboard on top of a snow covered? A: mountain
what is the man holding a snowboard on top of a snow covered
what is the man holding a snowboard on top of a snow covered ?
what is the man holding a snowboard on top of a snow covered ?
Q: what is the color of the bird? A: white
what is the color of the bird ?
what is the color of the bird ?
what is the color of the bird ?
Q: how many snowboarders in formation in the snow, four is sitting? A: 5
how many snowboarders in formation in the snow , four is sitting ?
how many snowboarders in formation in the snow , four is sitting ?
how many snowboarders in formation in the snow , four is sitting ?

Outline

Overview

End-to-end neural network for question answering:

- A coattention encoder captures the interaction between the question and the document
- A dynamic pointing decoder alternates between estimating the start and end of the answer span

Overview

End-to-end neural network for question answering:

- A coattention encoder captures the interaction between the question and the document
- A dynamic pointing decoder alternates between estimating the start and end of the answer span

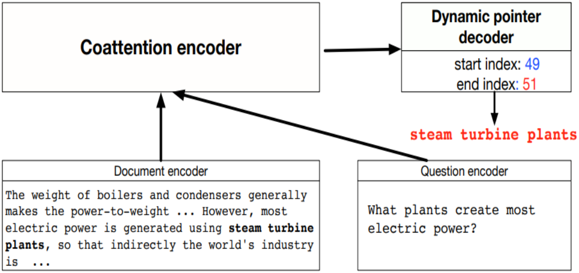


Figure 1: Overview of the Dynamic Coattention Network.

Outline

Document and Question Encoder

- Sequence of word vectors in document:
  $(x_1^D, x_2^D, \ldots, x_n^D)$
  $\Rightarrow d_t = LSTM_{enc}(d_{t-1}, x_t^D)$
  $\Rightarrow D = [d_1 \ldots d_m d_\phi] \in \mathbb{R}^{l \times (m+1)}$
- Sequence of word vectors in document:
  $(x_1^Q, x_2^Q, \ldots, x_m^Q)$
  $\Rightarrow q_t = LSTM_{enc}(q_{t-1}, x_t^Q)$
  $\Rightarrow Q' = [q_1 \ldots q_n q_\phi] \in \mathbb{R}^{l \times (n+1)}$
  $\Rightarrow Q = tanh(W^{(Q)} Q' + b^{(Q)}) \in \mathbb{R}^{l \times (n+1)}$
  (allow for variation between question encoding space and document encoding space)
- d_ϕ and q_ϕ: sentinel vector

Outline

Coattention Encoder

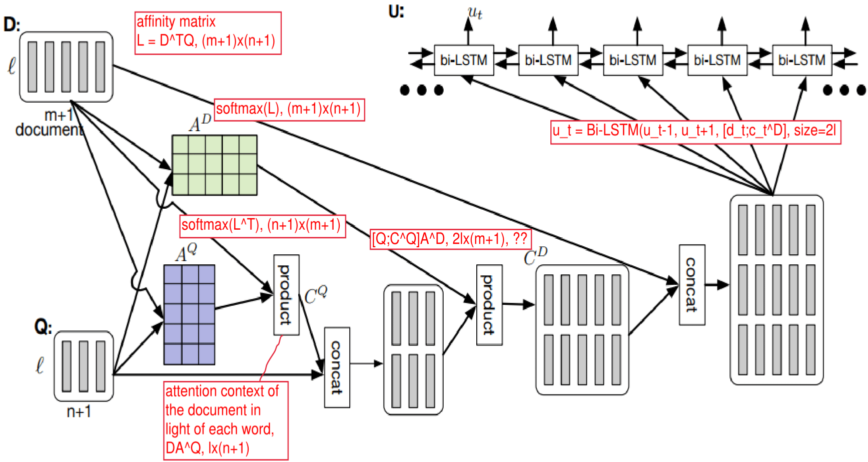


Figure 2: Coattention encoder. The affinity matrix L is not shown here. We instead directly show the normalized attention weights A^D and A^Q.

Outline

Dynamic Pointing Decoder

$h_i = LSTM_{dec}(h_{i-1}, [u_{s_{i-1}}; u_{e_{i-1}}])$, $U = [u_1, \ldots, u_m] \in \mathbb{R}^{2l \times m}$ from encoder

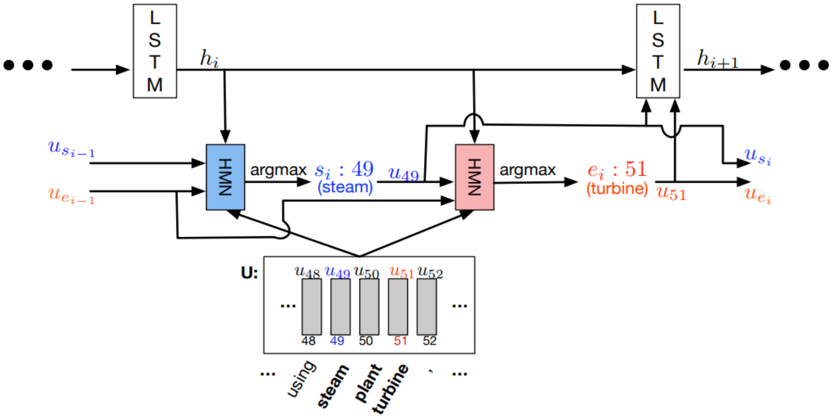


Figure 3: Dynamic Decoder. Blue denotes the variables and functions related to estimating the start position whereas red denotes the variables and functions related to estimating the end position.

Dynamic Pointing Decoder

$$h_i = LSTM_{dec}(h_{i-1}, [u_{s_{i-1}}; u_{e_{i-1}}])$$

Given current hidden state h_i, previous start position $u_{s_{i-1}}$ and previous end position $u_{e_{i-1}}$, how to estimate the current start position s_i and current end position e_i?

$$s_i = argmax(\alpha_1, \ldots, \alpha_m)$$

$$\alpha_t = \mathbf{HMN}_{start}(u_t, h_i, u_{s_{i-1}}, u_{e_{i-1}})$$

Highway Maxout Network(HMN)

$$HMN(u_t, h_i, u_{s_{i-1}}, u_{e_{i-1}}) = max(W^{(3)}[m_t^{(1)}; m_t^{(2)}] + b^{(3)}) \quad (1)$$

$$r = tanh(W^{(D)}[h_i; u_{s_{i-1}}; u_{e_{i-1}}]) \quad (2)$$

$$m_t^{(1)} = max(W^{(1)}[u_t; r] + b^{(1)}) \quad (3)$$

$$m_t^{(2)} = max(W^{(2)} m_t^{(1)} + b^{(2)}) \quad (4)$$

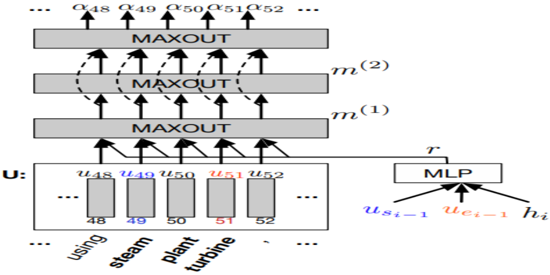
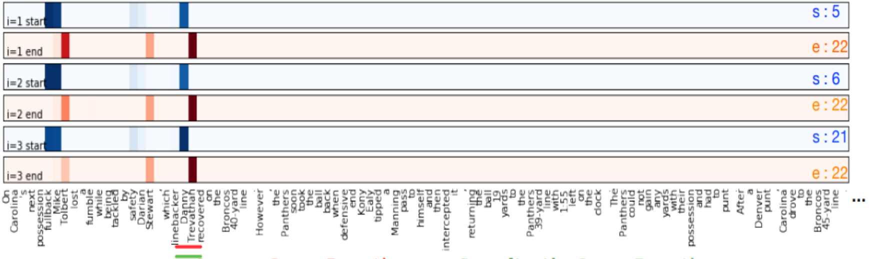


Figure 4: Highway Maxout Network. Dotted lines denote highway connections.

Outline

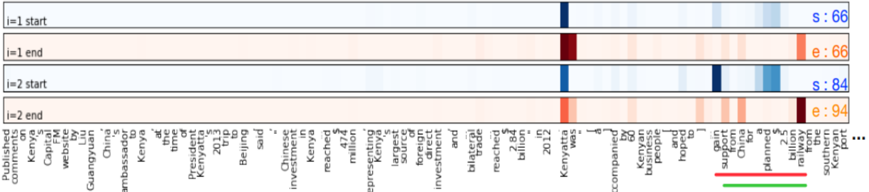
Qualitative Examples

Question 1: Who recovered Tolbert's fumble?



Answer: Danny Trevathan **Groundtruth:** Danny Trevathan

Question 2: What did the Kenyan business people hope for when meeting with the Chinese?



Answer: gain support from China for a planned $2.5 billion railway

Groundtruth: support from China for a planned $2.5 billion railway

Outline

Results

Model	Dev EM	Dev F1	Test EM	Test F1
Ensemble				
DCN (Ours)	**70.3**	**79.4**	**71.2**	**80.4**
Microsoft Research Asia *	–	–	69.4	78.3
Allen Institute *	69.2	77.8	69.9	78.1
Singapore Management University *	67.6	76.8	67.9	77.0
Google NYC *	68.2	76.7	–	–
Single model				
DCN (Ours)	65.4	**75.6**	**66.2**	**75.9**
Microsoft Research Asia *	65.9	75.2	65.5	75.0
Google NYC *	**66.4**	74.9	–	–
Singapore Management University *	–	–	64.7	73.7
Carnegie Mellon University *	–	–	62.5	73.3
Dynamic Chunk Reader (Yu et al., 2016)	62.5	71.2	62.5	71.0
Match-LSTM (Wang & Jiang, 2016b)	59.1	70.0	59.5	70.3
Baseline (Rajpurkar et al., 2016)	40.0	51.0	40.4	51.0
Human (Rajpurkar et al., 2016)	81.4	91.0	82.3	91.2

Table 1: Leaderboard performance at the time of writing (Nov 4 2016). * indicates that the model used for submission is unpublished. – indicates that the development scores were not publicly available at the time of writing.

Results

Model	Dev EM	Dev F1
Dynamic Coattention Network (DCN)		
pool size 16 HMN	**65.4**	**75.6**
pool size 8 HMN	64.4	74.9
pool size 4 HMN	65.2	75.2
DCN with 2-layer MLP instead of HMN	63.8	74.4
DCN with single iteration decoder	63.7	74.0
DCN with Wang & Jiang (2016b) attention	63.7	73.7

Table 2: Single model ablations on the development set.

Summary

- An end-to-end neural network architecture for question answering
- On the SQuAD dataset achieves the state of the art results at 75.9% F1 with a single model and 80.4% F1 with an ensemble.