A Multi-View Fusion Neural Network for Answer Selection

Lei Sha*, Xiaodong Zhang*, Feng Qian, Baobao Chang, Zhifang Sui

* Contributed equally

Peking University EECS

{shalei, zxdcs, nickqian, chbb, szf} @ pku.edu.cn

Task Description

- Community question answering aims at choosing the most appropriate answer for a given question.

Motivation

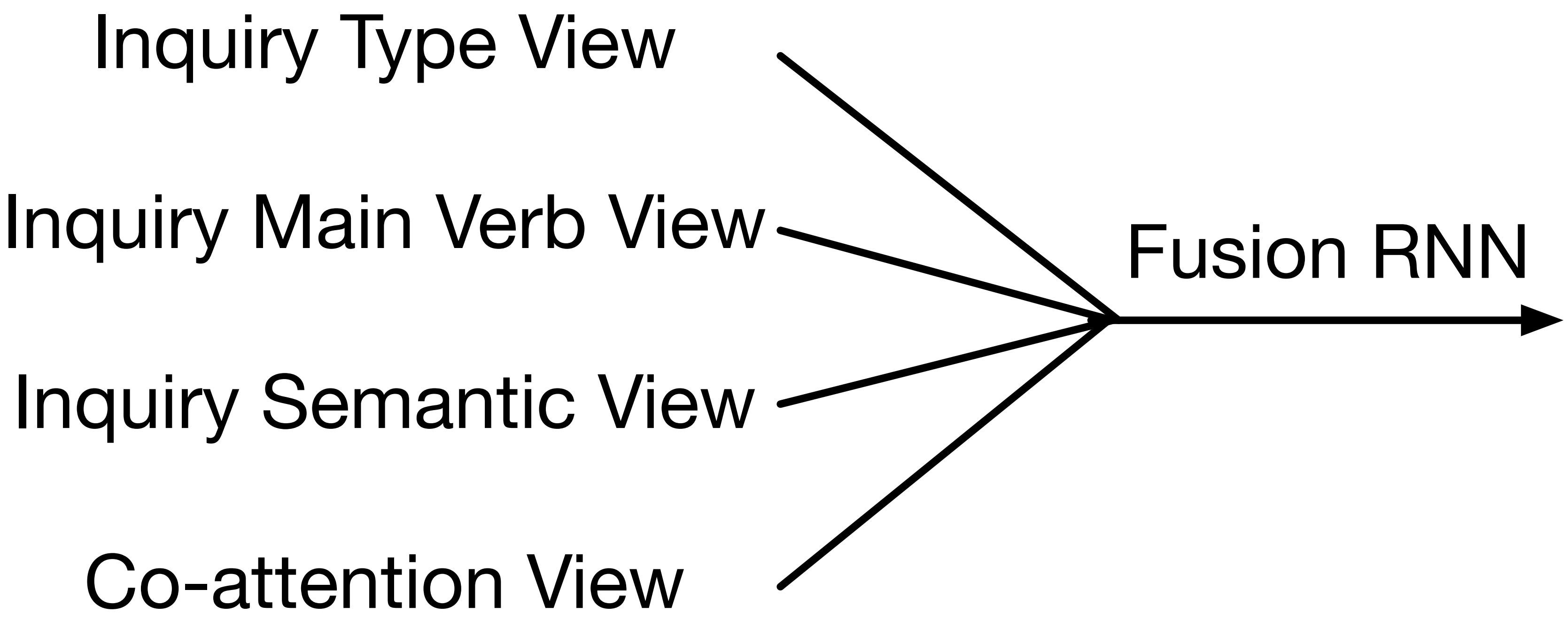
- Attentions from different aspects are always simply summed up and can be seen as a “single view”, causing severe information loss.

Multi-View Fusion Neural Network (MVFNN)

- we propose a Multi-View Fusion Neural Network (NVFNN), where each attention component generates a “view” of the QA pair. We utilized totally Four attention views

- A fusion RNN integrates the generated views to form a more holistic representation.

Multi-View Fusion Neural Network (MVFNN)



Inquiry Type View

Inquiry Main Verb View

Inquiry Semantic View

Co-attention View

Fusion RNN

Notations

- In this work, each word is represented using an embedding vector:

$$x_i \in R^d$$

- We denote the question and the answer as:

$$X_Q = \{x_{q_1}, x_{q_2}, \dots ,x_{q_{|Q|}}\} \in R^{d*|Q|}$$

$$X_A = \{x_{a_1}, x_{a_2}, \dots ,x_{a_{|A|}}\} \in R^{d*|A|}$$

- |Q| and |A| represent the length of the question and answer, respectively.

Inquiry Type View

- For typical question sentences, for example those in the WikiQA dataset, we used interrogative word (‘what’, ‘how’, ‘why’) as inquiry type.

- In Semeval-2016 CQA dataset, there is an inquiry type annotated for each question. So we just take this annotated type to calculate our inquiry type view.

Inquiry Type View

- We denote the interrogative word as

$$x_t \in R^d$$

- Inquiry Type View attention is calculated as:

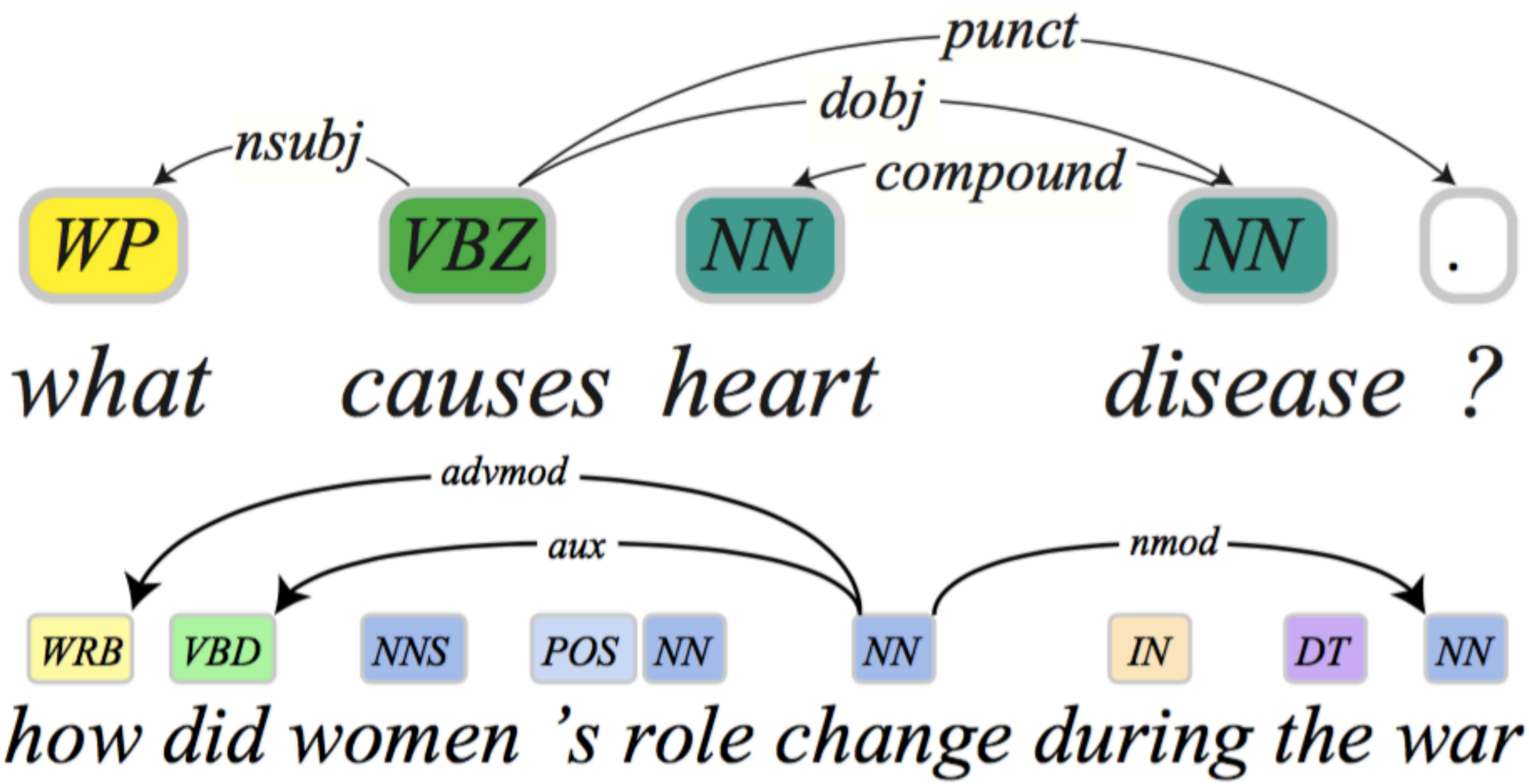
$$Att_t = softmax(w_t^T tanh(W_t x_t \oplus W_{ta} X_A))$$

- We generate the Inquiry Type View as:

$$V_t = Att_t \otimes X_A, V_t \in R^{d*|A|}$$

Main Inquiry verb View

- We use the root of dependency relationship as main inquiry verb.



Main Inquiry Verb View

- We denote the main verb as

$$X_c \in R^d$$

- Inquiry Type View attention is calculated as:

$$Att_c = softmax(w_c^T tanh(W_c x_c \oplus W_{ca} X_A))$$

- We generate the Inquiry Type View as:

$$V_c = Att_c \otimes X_A, V_c \in R^{d*|A|}$$

Inquiry Semantic View

- To understand the meaning of the whole question, we need to build the question’s semantic information into the inquiry semantic view.

Inquiry Semantic View

- We denote the semantic information of the question as:

$$x_s = Average(LSTM(x_{q1}, x_{q2}, \dots\ , x_{q|Q|}))$$

- Inquiry semantic View attention is calculated as:

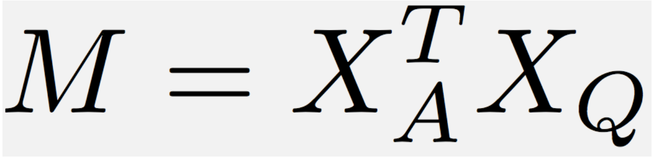
$$Att_s = softmax(w_x^T tanh(W_s x_s \oplus W_{sa} X_A))$$

- We generate the Inquiry Type View as:

$$V_s = Att_s \otimes X_A, V_s \in R^{d*|A|}$$

Co-attention View

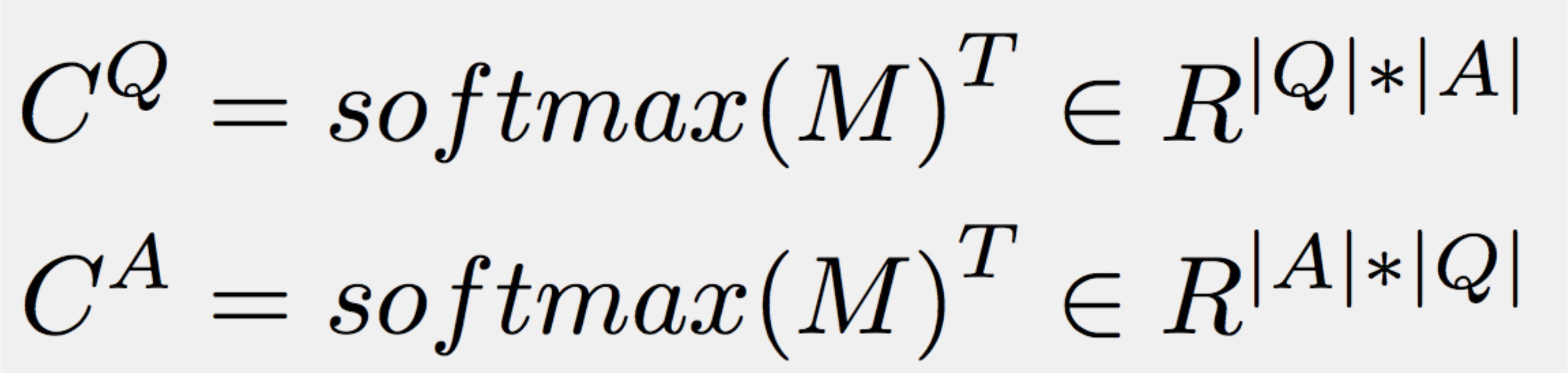
- Inspired by previous work on two-way attention from paired aspects (Santos et al. 2016; Xiong, Zhong, and Socher 2016)
- we introduce a co-attention view in this work, focusing more on the interaction between the question and the answers.
- We first compute the affinity matrix, which contains affinity scores that correspond to all pairs of question words and answer words:



$$M = X_A^T X_Q$$

Co-attention View

- Then we normalize M row-wise and column-wise to obtain the attention weights:



$$C^Q = softmax(M)^T \in R^{|Q|*|A|}$$

$$C^A = softmax(M)^T \in R^{|A|*|Q|}$$

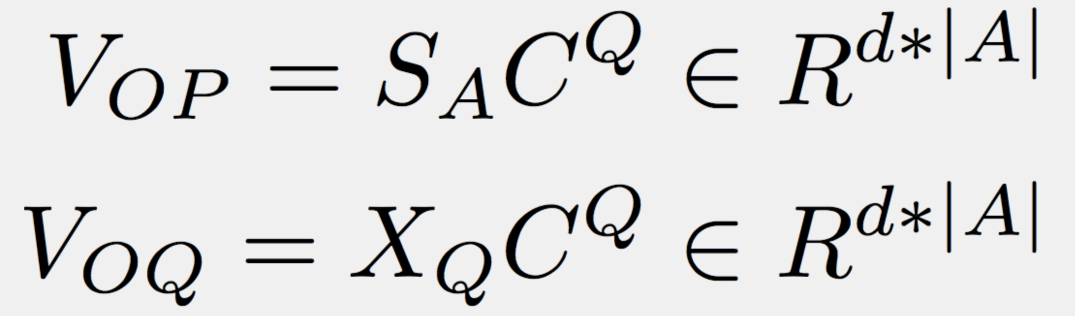
Co-attention View

- we directly multiply X_A and C^A to obtain the summaries of the answer for each word in the question:

$$S_A = X_A C^A \in R^{d*|Q|}$$

Co-attention View

- We compute the summary of the question and the summary of the previous attention context S_A in light of each word of the answer, then we get the co-attention QA pair view V_{CP} and co-attention question view V_{CQ} :

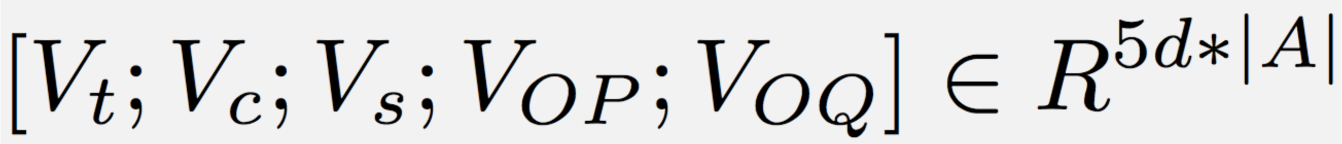


$$V_{OP} = S_A C^Q \in R^{d*|A|}$$

$$V_{OQ} = X_Q C^Q \in R^{d*|A|}$$

Concatenated View Matrix

- We concatenated five view vectors from four types of views

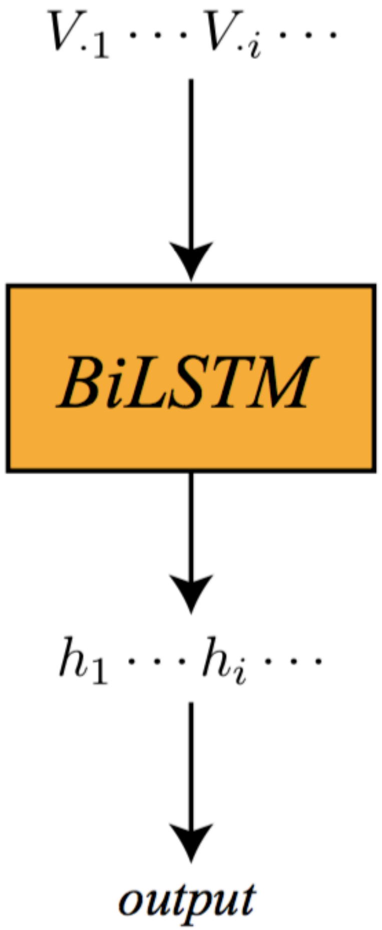


$$[V_t; V_c; V_s; V_{OP}; V_{OQ}] \in R^{5d*|A|}$$

Fusion multiple views

- Simple Bi-LSTM Fusion
- Simple Bi-LSTM Fusion + ResNet
- Fusion RNN for Building A Holistic View

Simple Bi-LSTM Fusion



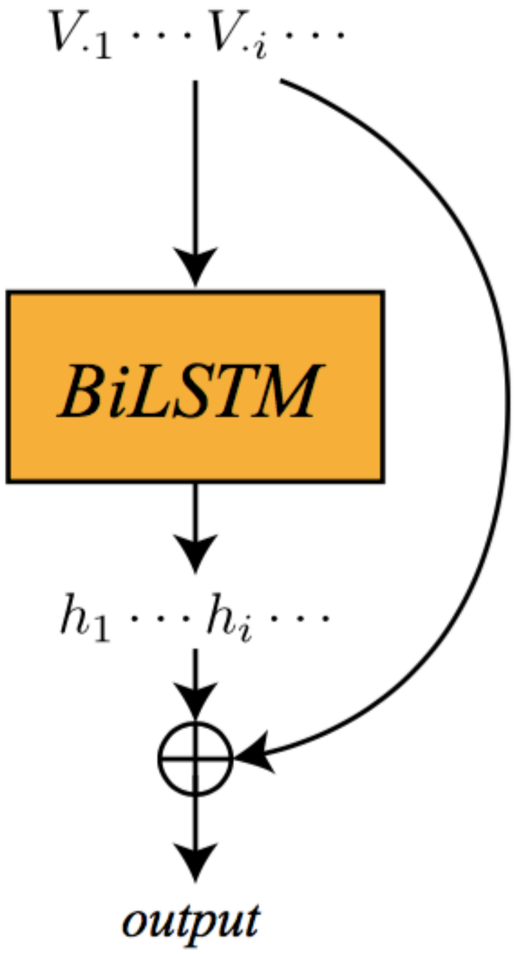
- Read the word-level view sequence $[V_{\cdot 1}, V_{\cdot 2}, \cdots, V_{\cdot |A|}]$ with Bi-LSTM:

$$h_1, \ldots\ h_{|A|} = BiLSTM(V_1, \ldots\ V_{|A|})$$

- The matching score is calculated as:

$$s(X_Q, X_A) = w^T Average(h_1, \ldots, h_{|A|})$$

Simple Bi-LSTM Fusion + ResNet

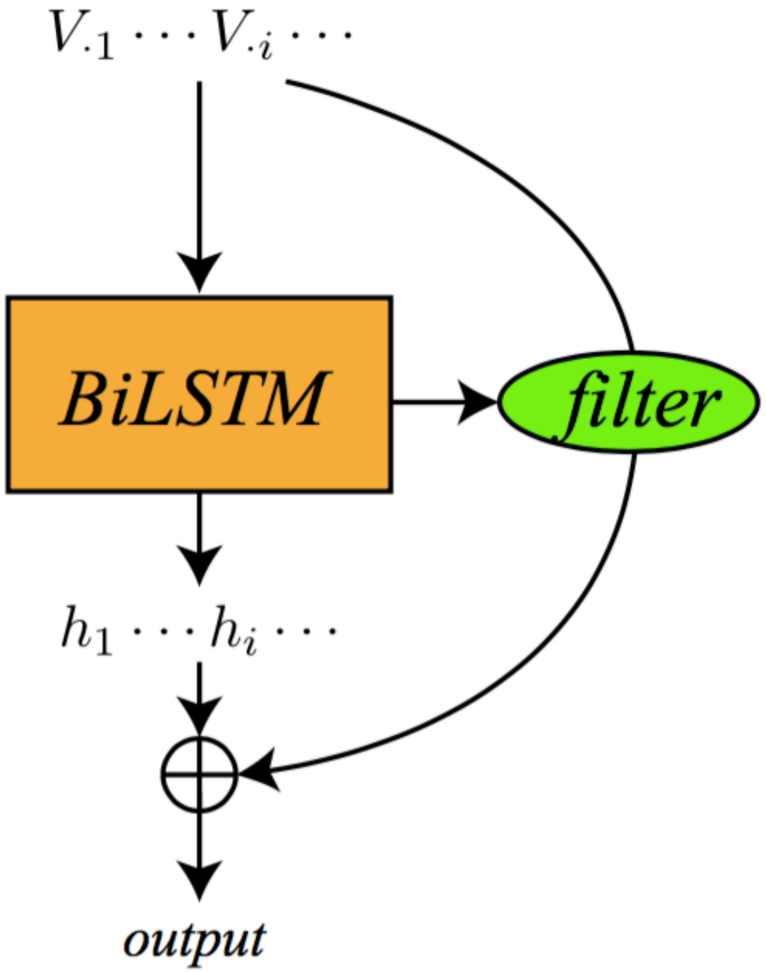


- The difference to simple BiLSTM is that the inputs of the BiLSTM are directly linked to the output:

$$v_{in} = Average(V_1, \dots, V_{|A|})$$

$$s(X_Q, X_A) = w^T(h_{ave} + W_{res} v_{in})$$

Fusion RNN for Building A Holistic View

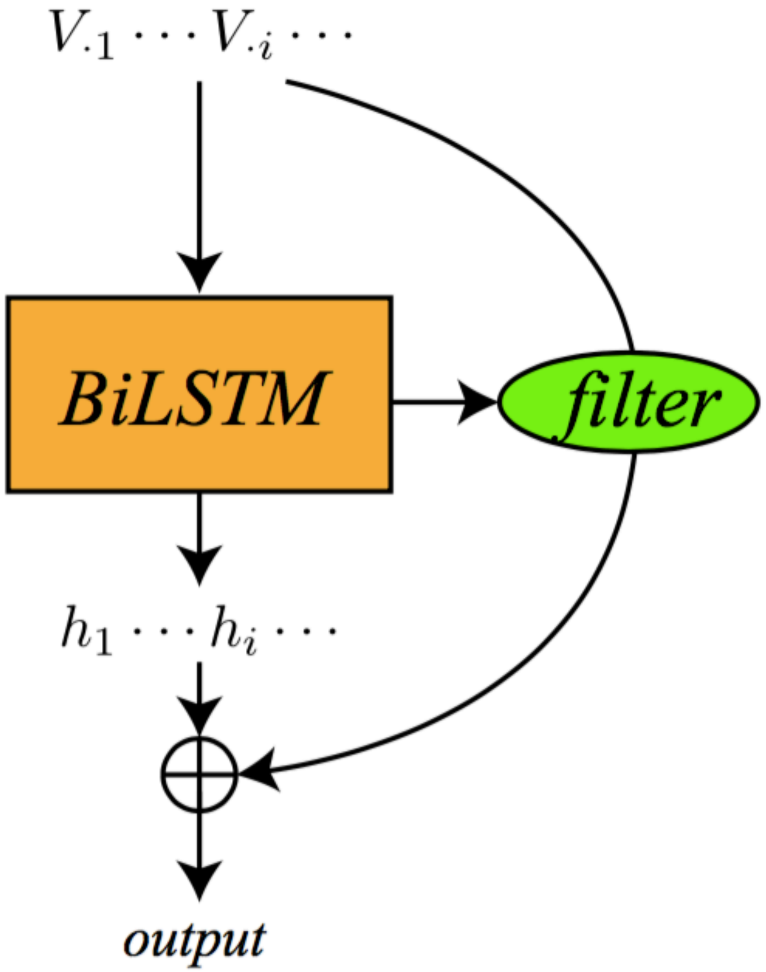


$$z = sigmoid(W_h h^{\rightarrow}_{t-1}) \in R^{d_M}$$

$$I_{t-1} = W_i V_{t-1} + b_i \in R^{d_M}$$

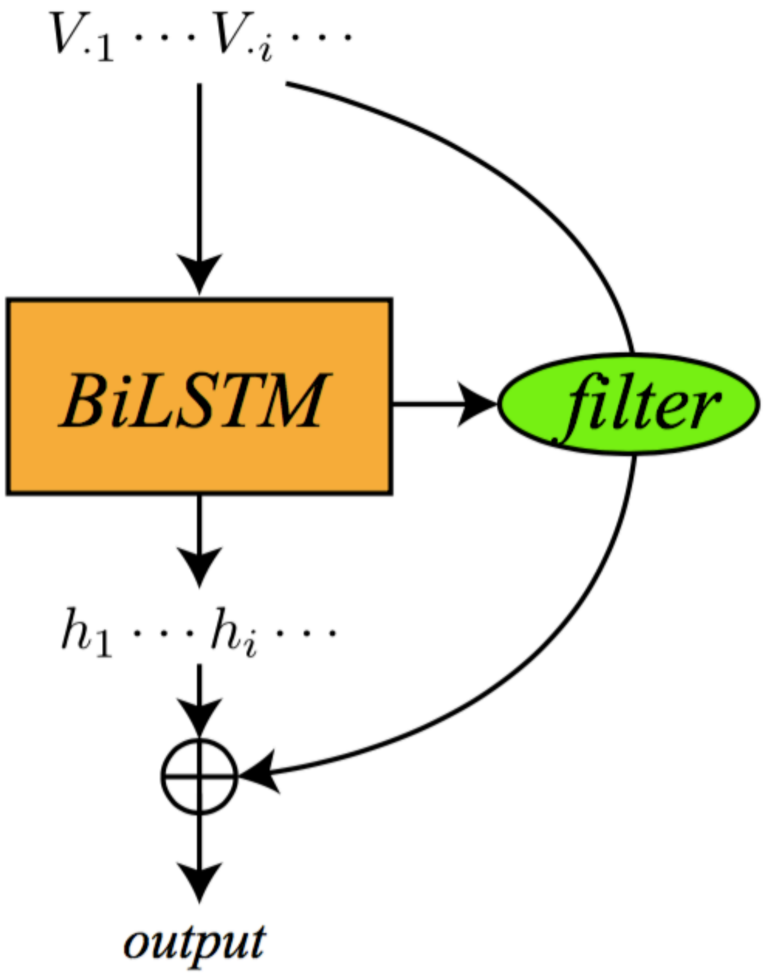
$$M^{\rightarrow}_t = (1 - z) \odot M^{\rightarrow}_{t-1} + z \odot I_{t-1}$$

Fusion RNN for Building A Holistic View



$$M_{|A|} = [M_{\overrightarrow{|A|}}; M_{\overleftarrow{|A|}}]$$
$$h_t = [h_{\overrightarrow{t}}; h_{\overleftarrow{t}}]$$

Fusion RNN for Building A Holistic View



- Since the external memory keeps the important information of the input view, we add external memory to the average pooling of the BiLSTM's output as inspired by deep residual network (He et al. 2016)

$$h_{ave} = Average(h_1, \dots , H_{|A|})$$

$$F = h_{ave} + W_{mh} M_{|A|}$$

$$s(X_Q, X_A) = w^T F$$

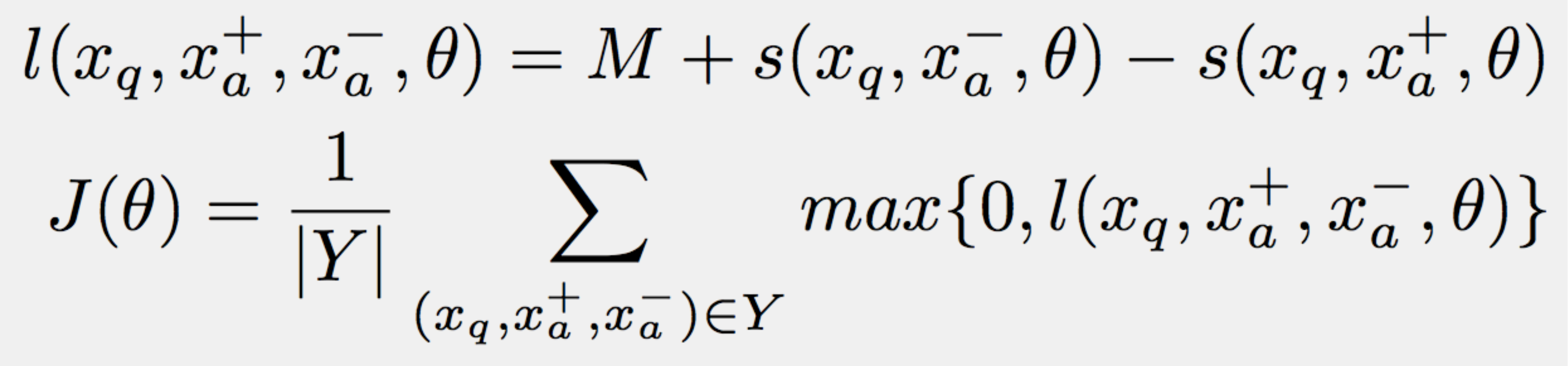
Training

- Target: The score of the correct answer-question pair should be larger than any other pairs.

$$s(x_q, x_a^+, \theta) \geq s(x_q, x_a^-, \theta) + M$$

Training

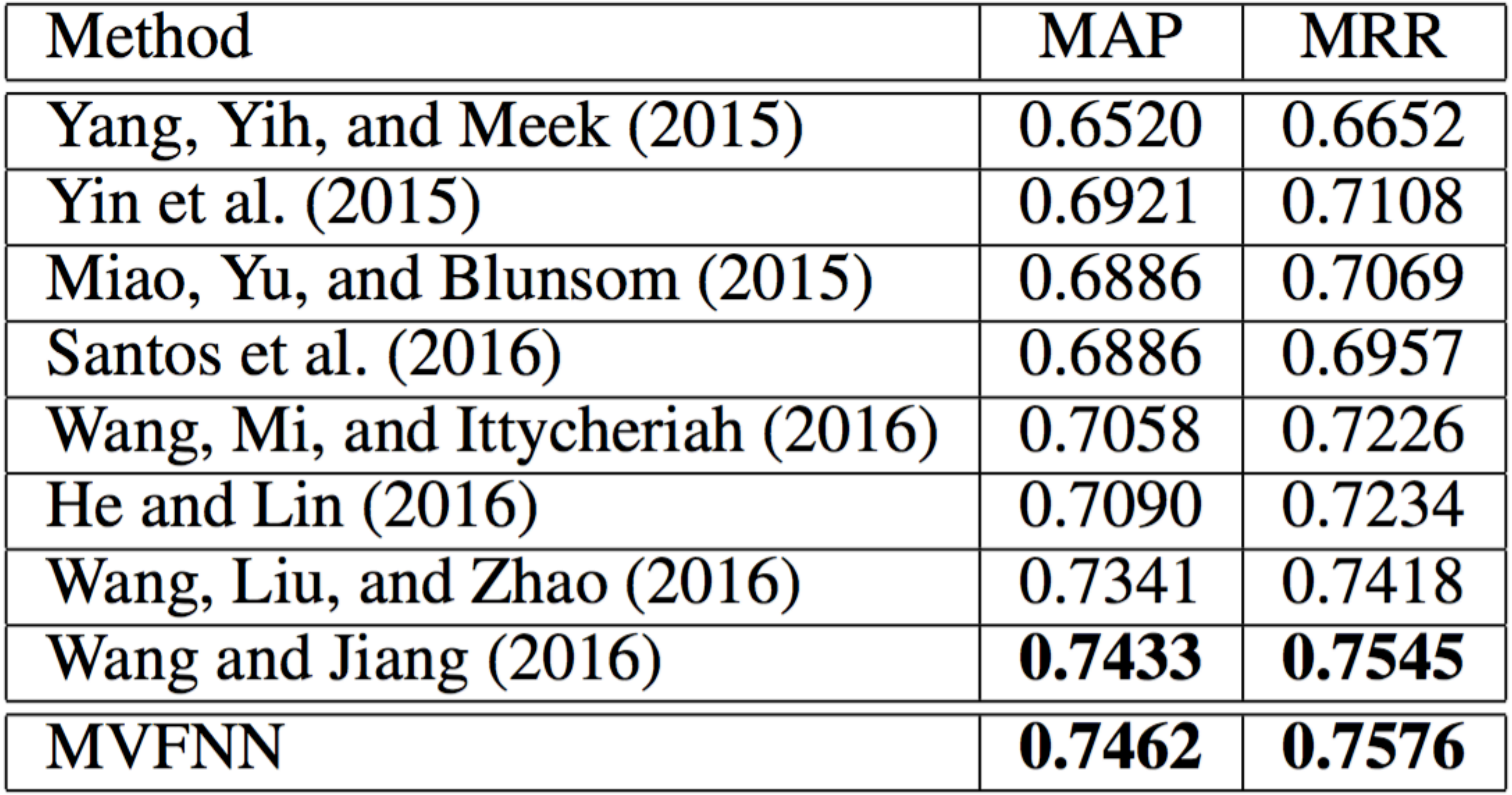
- Minimize the target:



$$l(x_q, x_a^+, x_a^-, \theta) = M + s(x_q, x_a^-, \theta) - s(x_q, x_a^+, \theta)$$

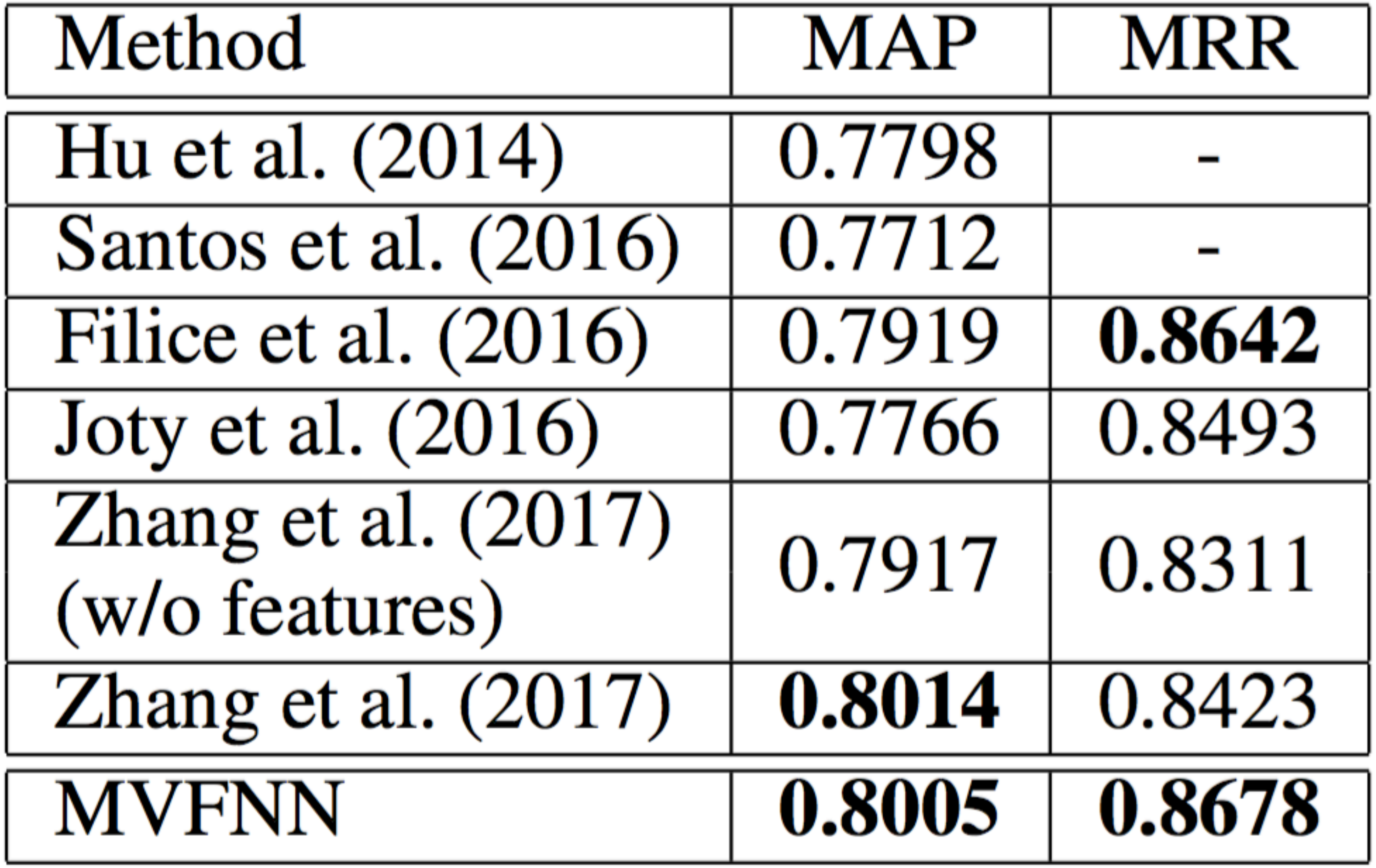
$$J(\theta) = \frac{1}{|Y|} \sum_{(x_q, x_a^+, x_a^-) \in Y} max\{0, l(x_q, x_a^+, x_a^-, \theta)\}$$

Experiments : Wiki-QA



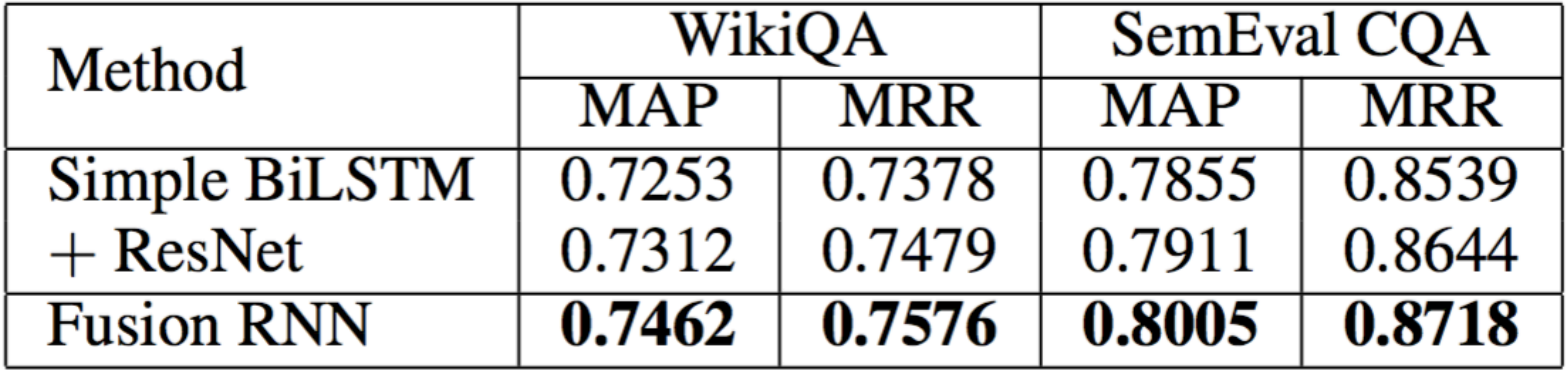
Method	MAP	MRR
Yang, Yih, and Meek (2015)	0.6520	0.6652
Yin et al. (2015)	0.6921	0.7108
Miao, Yu, and Blunsom (2015)	0.6886	0.7069
Santos et al. (2016)	0.6886	0.6957
Wang, Mi, and Ittycheriah (2016)	0.7058	0.7226
He and Lin (2016)	0.7090	0.7234
Wang, Liu, and Zhao (2016)	0.7341	0.7418
Wang and Jiang (2016)	**0.7433**	**0.7545**
MVFNN	**0.7462**	**0.7576**

Experiments: Sem-Eval-2016



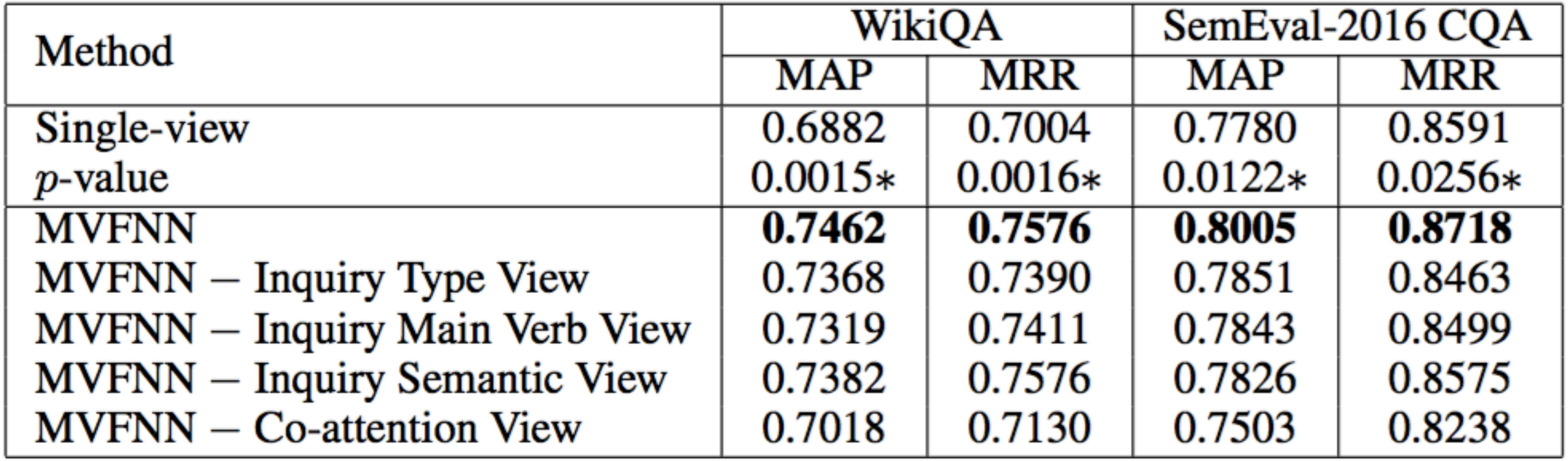
Method	MAP	MRR
Hu et al. (2014)	0.7798	-
Santos et al. (2016)	0.7712	-
Filice et al. (2016)	0.7919	**0.8642**
Joty et al. (2016)	0.7766	0.8493
Zhang et al. (2017) (w/o features)	0.7917	0.8311
Zhang et al. (2017)	**0.8014**	0.8423
MVFNN	**0.8005**	**0.8678**

Experiments: Fusion Methods



Method	WikiQA		SemEval CQA	
	MAP	MRR	MAP	MRR
Simple BiLSTM	0.7253	0.7378	0.7855	0.8539
+ ResNet	0.7312	0.7479	0.7911	0.8644
Fusion RNN	**0.7462**	**0.7576**	**0.8005**	**0.8718**

Experiments: Multi-View



Method	WikiQA		SemEval-2016 CQA	
	MAP	MRR	MAP	MRR
Single-view	0.6882	0.7004	0.7780	0.8591
p-value	0.0015∗	0.0016∗	0.0122∗	0.0256∗
MVFNN	**0.7462**	**0.7576**	**0.8005**	**0.8718**
MVFNN − Inquiry Type View	0.7368	0.7390	0.7851	0.8463
MVFNN − Inquiry Main Verb View	0.7319	0.7411	0.7843	0.8499
MVFNN − Inquiry Semantic View	0.7382	0.7576	0.7826	0.8575
MVFNN − Co-attention View	0.7018	0.7130	0.7503	0.8238

- Thanks!

A Multi-View Fusion Neural Network for Answer Selection

Lei Sha*, Xiaodong Zhang*, Feng Qian, Baobao Chang, Zhifang Sui

* Contributed equally

Peking University EECS

{shalei, zxdcs, nickqian, chbb, szf} @ pku.edu.cn