Multi-Task Learning with Multi-View Attention for Answer Selection and Knowledge Base Question Answering

Yang Deng, Yuexiang Xie, Yaliang Li, Min Yang, Nan Du, Wei Fan, Kai Lei, Ying Shen

School of Electronics and Computer Engineering, Peking University Shenzhen Graduate School Tencent Medical AI Lab

Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences

(AAAI-19)

Presenter: **Happy Buzaaba**

University of Tsukuba: KDE Lab Mining Seminar

Date: September 20th, 2019

Outline

- **Introduction**
  - Answer Selection & Knowledge Base Question Answering
  - Multi-task Learning
- **Methodology**
  - Problem Definition
  - Multi-Task Learning for Question Answering
  - Multi-View Attention Scheme
- **Experiment**
  - Multi-Task Learning Results
  - Ablation Analysis of Multi-View Attention
  - Case Study of Multi-View Attention
- **Summary**

Outline

- **Introduction**
  - Answer Selection & Knowledge Base Question Answering
  - Multi-task Learning
- **Methodology**
  - Problem Definition
  - Multi-Task Learning for Question Answering
  - Multi-View Attention Scheme
- **Experiment**
  - Multi-Task Learning Results
  - Ablation Analysis of Multi-View Attention
  - Case Study of Multi-View Attention
- **Summary**

Introduction

❑ Answer Selection (AS)

Problem: Given a question, answer selection aims to pick out a correct Answer from a set of candidates.

Current Study: Deep Learning , Attention mechanisms, External Knowledge

❑ Knowledge Base Question Answering (KBQA)

Problem: Given a question, KBQA aims to pick out a fact from a given Knowledge base to answer the question

Current study: Semantic parsing, Deep Learning, Contextual Information

❖ **Existing methods solve these two tasks separately**. This requires large number of repetitive work and neglects the rich correlation information between tasks.

Introduction

❑ Multi-Task Learning (MTL)

Multi-task learning aims to jointly learn different related tasks

Note: In this paper, we tackle answer selection (AS) and KBQA tasks Simultaneously via multi-task learning.

Motivation:

- Both AS & KBQA can be regarded as a ranking problem, with one at text-level while the other at knowledge level.[1]
- Both tasks can benefit each other: AS can incorporate external knowledge from the knowledge base, while KBQA can be improved by learning contextual information from AS.[2]

[1] Savenkov, D., and Agichtein, E. 2017. Evinets: Neural networks for combining evidence signals for factoid question answering. In ACL, 299–304.
[2] Sorokin, D., and Gurevych, I. 2018. Modeling semantics with gated graph neural networks for knowledge base question answering. In COLING, 3306– 3317.

Main Contribution

❑ Multi-Task Question Answer Scheme

Propose a novel multi-task learning scheme that utilizes multi-view attention learned from various perspectives to enable these tasks to interact with each other as well as learn more comprehensive sentence representations.

Summary of contribution:

- We explore multi-task learning approaches for answer selection and knowledge base question answering.
- We propose a novel multi-task learning scheme that leverages multi-view attention mechanism to bridge different tasks.
- Experimental results show that multi-task learning of answer selection and KBQA outperforms state-of-the-art single-task learning methods. Besides, the multi-view attention based MTL scheme further enhance the performance.

Outline

- ➢ **Introduction**
  - Answer Selection & Knowledge Base Question Answering
  - Multi-task Learning
- ➢ **Methodology**
  - Problem Definition
  - Multi-Task Learning for Question Answering
  - Multi-View Attention Scheme
- ➢ **Experiment**
  - Multi-Task Learning Results
  - Ablation Analysis of Multi-View Attention
  - Case Study of Multi-View Attention
- ➢ **Summary**

Methodology

❑ Multi-Task Learning for Question Answering

▪ Problem Definition

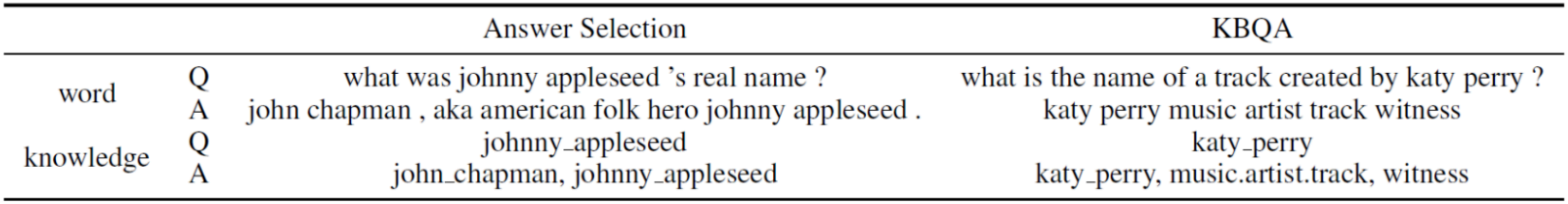
Ranking Problem: Given question $\boldsymbol{q}_{i.} \in Q$, the task is to rank a set of candidate answer sentences or facts $a_i \in A$.

Inputs:

- a word sequence $W = \{w_1, w_2, \ldots, w_L\}$
- a knowledge sequence $K = \{k_1, k_2, \ldots, k_L\}$
- D_t as the *t-th* preprocessed task dataset with N samples:

  $D_t = \{(W^{(t)}_{q_i}, K^{(t)}_{q_i}, W^{(t)}_{a_i}, K^{(t)}_{a_i}, Y^{(t)}_{i})\}_{i=1}^{N_t}$, where $Y^{(t)}_{i}$ denotes the label Of the *i-th* QA pair in the *t-th* task.

Outputs: a relevancy score $f(q,a) \in [0, 1]$ for each QA pair



		Answer Selection	KBQA
word	Q	what was johnny appleseed 's real name ?	what is the name of a track created by katy perry ?
	A	john chapman , aka american folk hero johnny appleseed .	katy perry music artist track witness
knowledge	Q	johnny_appleseed	katy_perry
	A	john_chapman, johnny_appleseed	katy_perry, music.artist.track, witness

Table 1: Examples of AS and KBQA Data

Methodology

- ❑ Multi-Task Learning for Question Answering
- ▪ Multi-Task QA network (MTQA-net)

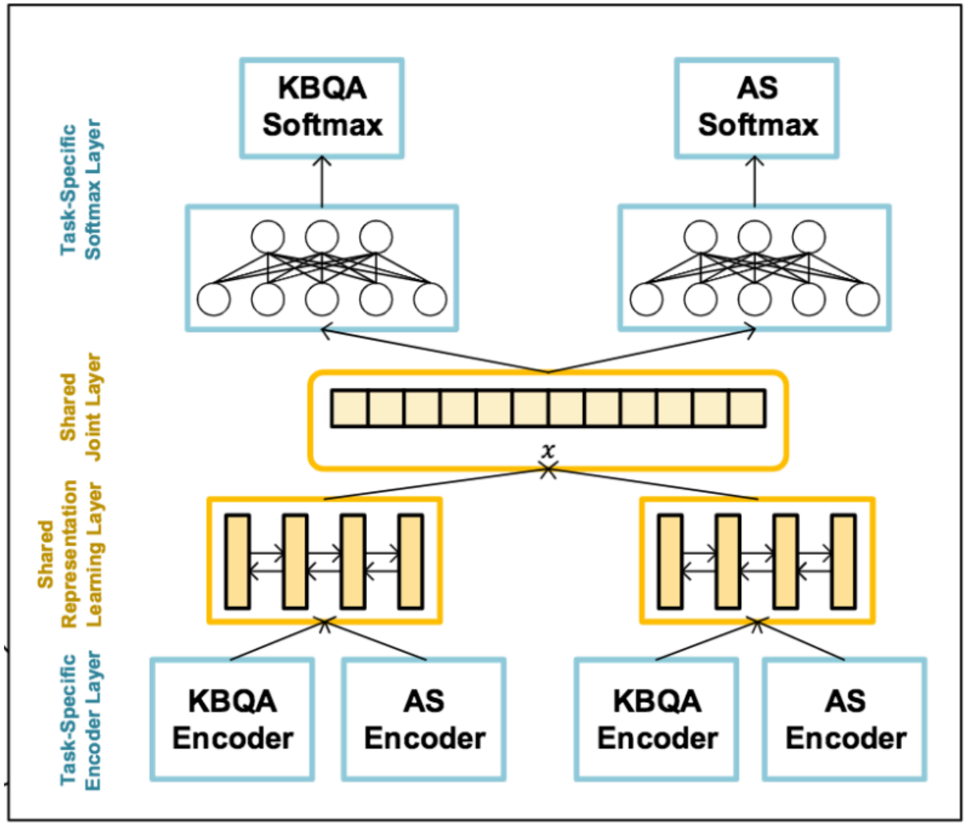

KBQA Softmax
AS Softmax
Task-Specific Softmax Layer
Shared Joint Layer
x
Shared Representation Learning Layer
Task-Specific Encoder Layer
KBQA Encoder
AS Encoder
KBQA Encoder
AS Encoder

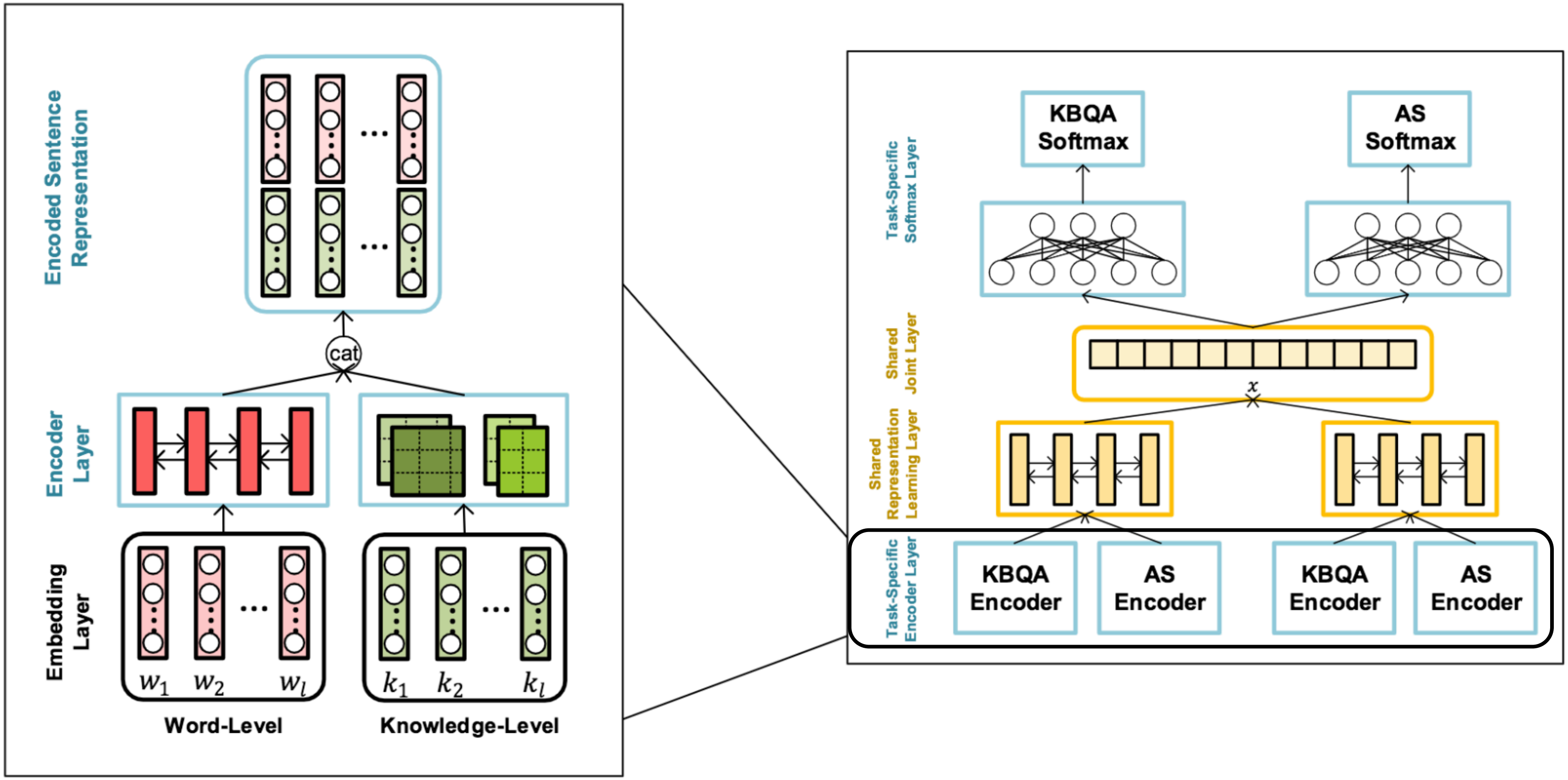
Figure 1: Basic multi-task QA Network

Guo, H.; Pasunuru, R.; and Bansal, M. 2018. Soft layer-specific multi-task summarization with entailment and question generation. In ACL

Methodology

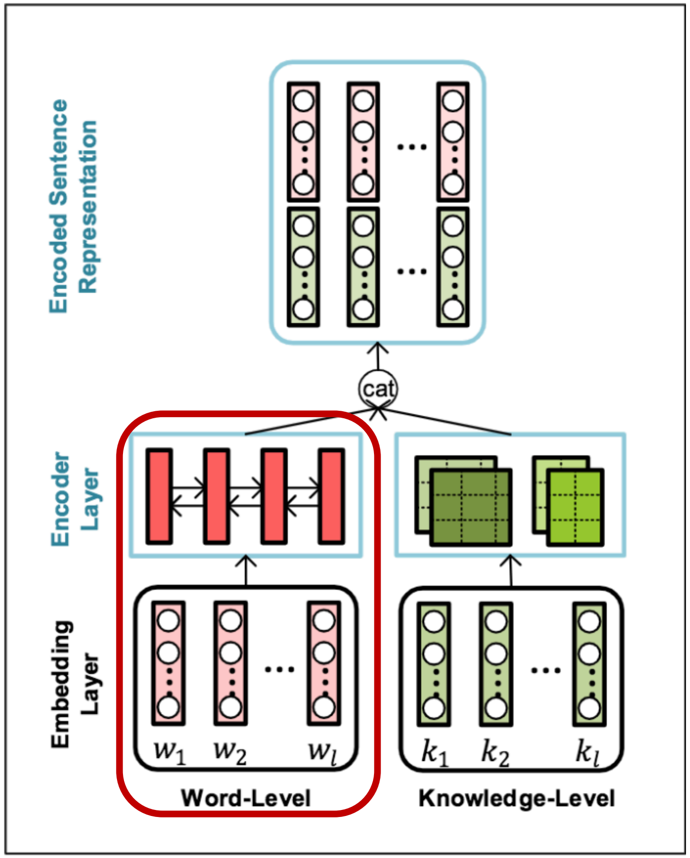
❑ Multi-Task QA network (MTQA-net)

Task-specific Encoder layer



Methodology

Task-specific Encoder layer



Word Encoder

$E_W = \{e_{w1}, e_{w2}, ..., e_{wL}\}$ Word embeddings

$h_l = \overrightarrow{h_l} : \overleftarrow{h_l}$ ***l-th*** **word hidden representation**

$H_W \in R^{L \times d_h}$: Sentence representation

L : length of sentences
d_h : size of hidden units

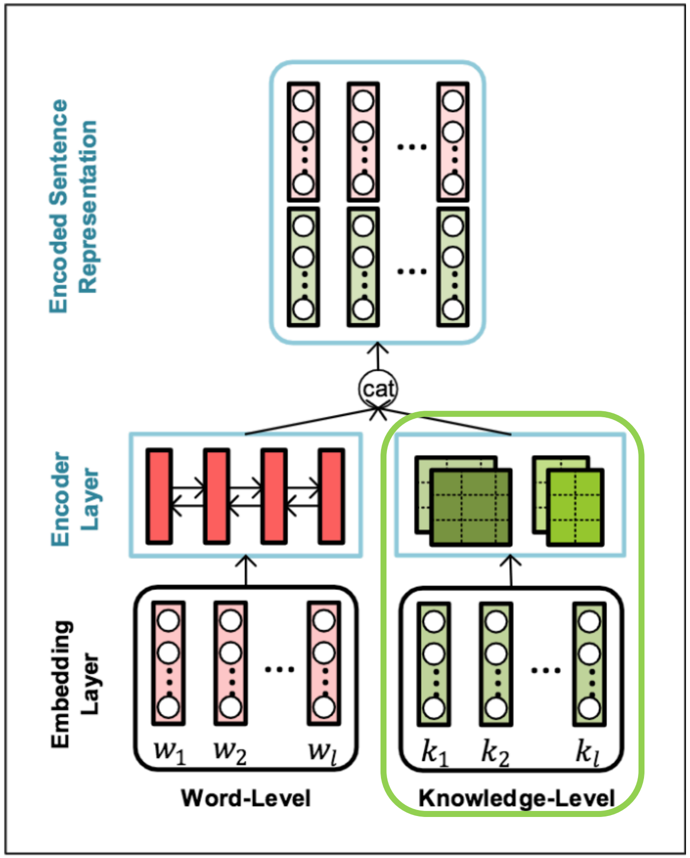
$$H_{Wq} = \textbf{BiLSTM}\ (E_{wq});\ H_{Wa} = \textbf{BiLSTM}\ (E_{wa})$$

Word-level
Question sentence
representation

Word-level
answer sentence
representation

Methodology

Task-specific Encoder layer



Knowledge Encoder

$E_K = \{e_{k1}, e_{k2}, \ldots, e_{kL}\}$ knowledge embeddings

Sliding filters

$$x_l = [e_{k_{l-\frac{n-1}{2}}}, \ldots, e_{k_l}, \ldots, e_{k_{l+\frac{n-1}{2}}}]$$

$h_l = tanh(W_c x_l + b_c)$: hidden layer vector
Wc : convolutional kernel
bc : bias

$\{H^{(1)}, H^{(2)}, \cdots, H^{(n)}$: $H^{(i)}$ *i-th* filter output vector

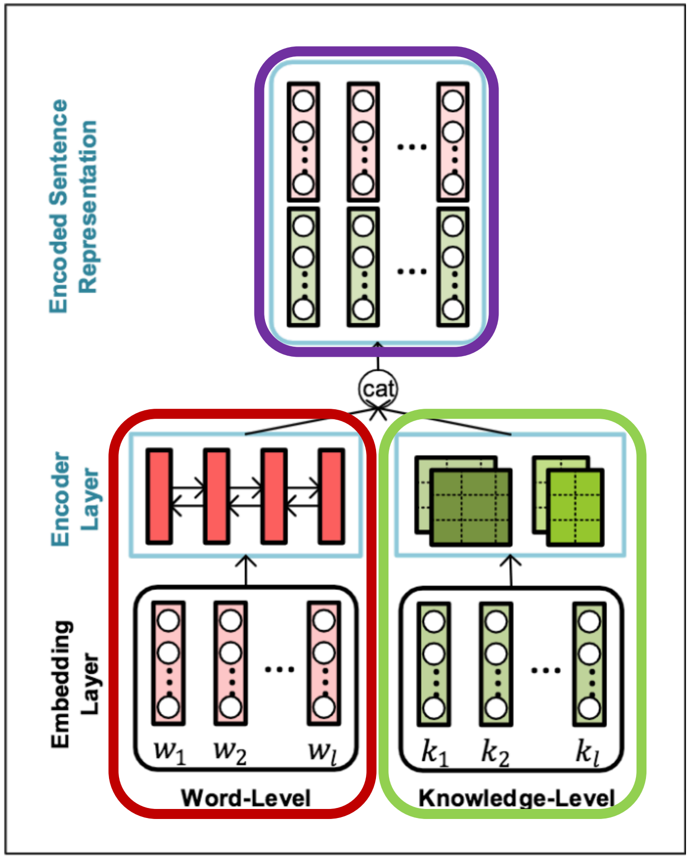
$H_K \in R^{L \times d_f}$: Sentence representation
L : length of sentences
d_f: total filter sizes of CNN

$$H_{Kq} = [H^{(1)}_{Kq} : H^{(2)}_{Kq} : \cdots : H^{(n)}_{Kq}],$$

$$H_{Ka} = [H^{(1)}_{Ka} : H^{(2)}_{Ka} : \cdots : H^{(n)}_{Ka}],$$

Methodology

Task-specific Encoder layer



Concatenated representations

$$H_q = [H_{Wq} : H_{Kq}] \text{ and } H_a = [H_{Wa} : H_{Ka}]$$

Knowledge-based Encoder

$$H_{Kq} = [H^{(1)}_{Kq} : H^{(2)}_{Kq} : \cdots : H^{(n)}_{Kq}],$$

$$H_{Kq} = [H^{(1)}_{Ka} : H^{(2)}_{Ka} : \cdots : H^{(n)}_{Ka}],$$

Word-based Encoder

$$H_{Wq} = \textbf{BiLSTM} (E_{wq}); \; H_{Wa} = \textbf{BiLSTM} (E_{wa})$$

Methodology

- Shared Representation Learning Layer.

Final QA representation

$$S_q = \textbf{BiLSTM}\ (H_q);\ S_a = \textbf{BiLSTM}\ (H_a)$$

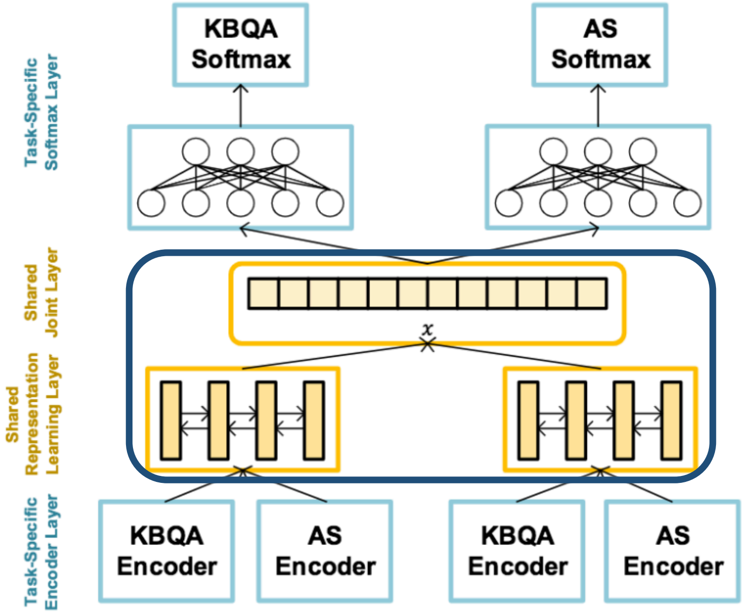
$$s_q = Average\ (S_q);\ s_a = Average\ (S_a)$$

$x_{ol} \in R^6$: overlap features [3]

$x = [s_q, s_a, x_{ol}]$: final feature space

overlap features:

- word overlap score
- non-stop word overlap score
- weighted word overlap score
- non-stop weighted word overlap score
- knowledge overlap score
- weighted knowledge overlap score


KBQA Softmax
AS Softmax
Task-Specific Softmax Layer
Shared Joint Layer
x
Shared Representation Learning Layer
Task-Specific Encoder Layer
KBQA Encoder
AS Encoder
KBQA Encoder
AS Encoder

[3] Severyn, A., and Moschitti, A. 2015. Learning to rank short text pairs with convolutional deep neural networks. In SIGIR, 373–382

[3] Tay, Y.; Phan, M. C.; Tuan, L. A.; and Hui, S. C. 2017. Learning to rank question answer pairs with holographic dual lstm architecture. In SIGIR.

Methodology

- Task Specific Softmax Layer.

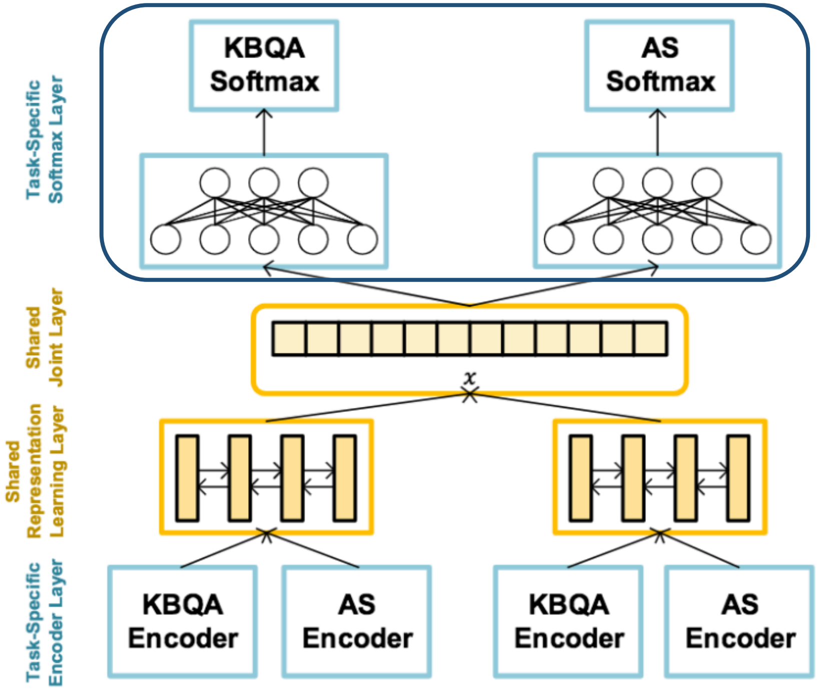
$q_i^{(t)}$, $a_i^{(t)}$ & $y_i^{(t)}$: *k-th* task

$$p^{(t)} = \text{softmax}\,(W_s^{(t)} x + b_s^{(t)})$$

$p^{(t)}$: predicted probability

$W_s^{(t)} \in R^{dx \times 2}$: weight matrix

$b_s^{(t)} \in R^2$: bias


Task-Specific Softmax Layer
KBQA Softmax
AS Softmax
Shared Joint Layer
x
Shared Representation Learning Layer
Task-Specific Encoder Layer
KBQA Encoder
AS Encoder
KBQA Encoder
AS Encoder

- Multi-Task Learning.

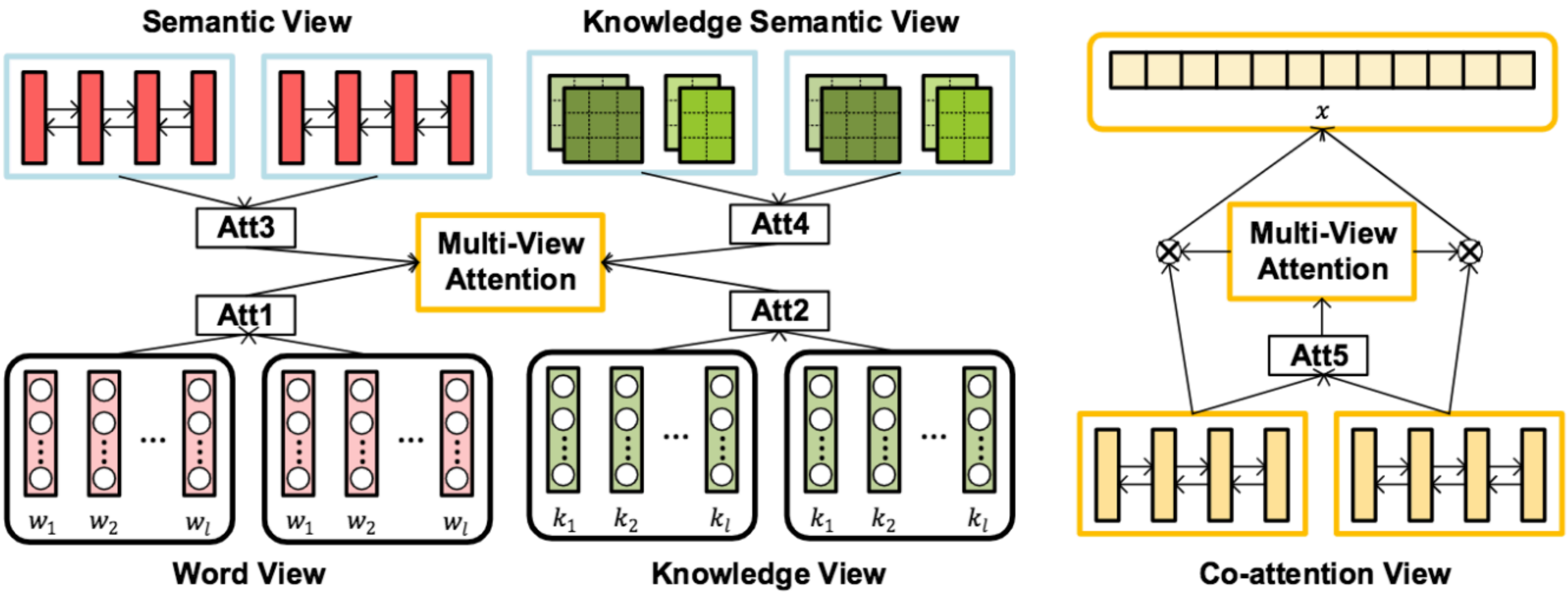
$$L = -\sum_{t=1}^{T} \lambda_t \sum_{i=1}^{N_t} [y_i^{(t)} \, logp_i^{(t)} + (1 - y_i^{(t)}) \log(1 - p_i^{(t)})]$$

λ_t : parameter to determine weight of *t-th* task,

$y_i^{(t)}$: ground truth label of question-answer pair ($q_i^{(t)}$, $a_i^{(t)}$)

Methodology

- Multi-View Attention Scheme.



Methodology

- Multi-View Attention Scheme: Word and Knowledge view

$M_W = \tanh(E_{Wa}^{T} U_W E_{Wa})$; $M_K = \tanh(E_{Ka}^{T} U_K E_{Ka})$; attention weights from embeddings

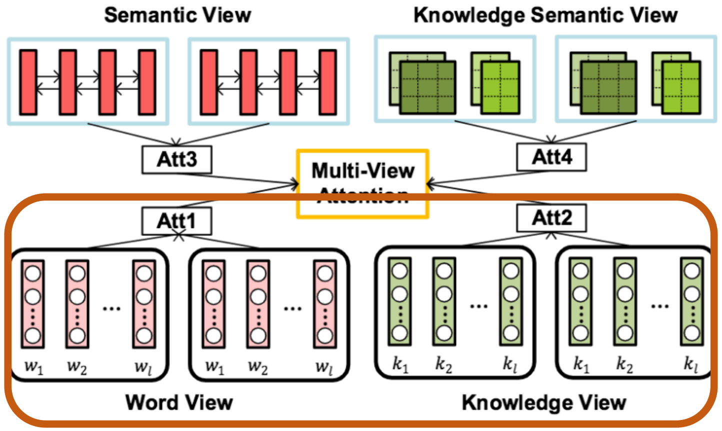
$U_W \in R^{dew \times dew}$, $U_K \in R^{dek \times dek}$: Matrices

d_{ew} : dimension of word embedding

d_{ek} : dimension of knowledge embedding

$$\alpha_q^{(1)} = \mathrm{softmax}(Max(M_W))\ ;\ \alpha_a^{(1)} = \mathrm{softmax}(Max(M_W^{T})),$$

$$\alpha_q^{(2)} = \mathrm{softmax}(Max(M_K));\ \alpha_a^{(2)} = \mathrm{softmax}(Max(M_K^{T})),$$



$\alpha_q^{(1)}$ & $\alpha_a^{(1)}$: attention weights from word view

$\alpha_q^{(2)}$ & $\alpha_a^{(2)}$: attention weights from knowledge view

dos Santos, C. N.; Tan, M.; Xiang, B.; and Zhou, B. 2016. Attentive pooling networks. CoRR, abs/1602.03609

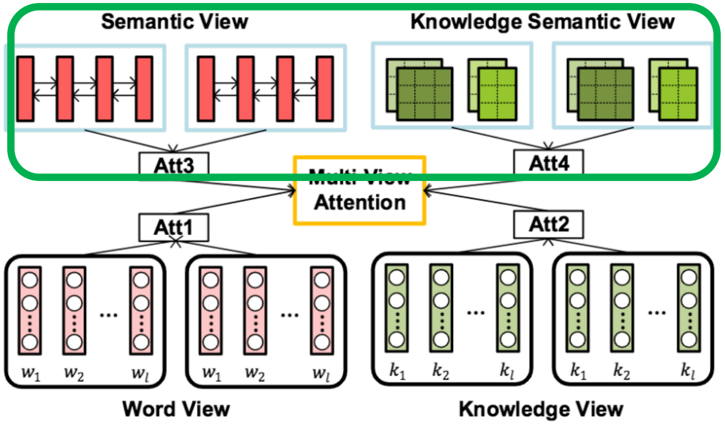
Methodology

- Multi-View Attention Scheme: Semantic and Knowledge sematic view

$$O_{wq} = \text{Average}(H_{Wq});\;\; O_{wa} = \text{Average}(H_{Wa}),$$

$$O_{kq} = \text{Max}(H_{Kq}));\;\;\; O_{ka} = \text{Max}(H_{Ka}),$$

$$\alpha_q^{(3)} = \text{softmax}(w_{wq}^T \tanh(W_{wa}O_{wa} + W_{wq}H_{wq})),$$
$$\alpha_a^{(3)} = \text{softmax}(w_{wa}^T \tanh(W_{wq}O_{wq} + W_{wa}H_{wa})),$$
$$\alpha_q^{(4)} = \text{softmax}(w_{kq}^T \tanh(W_{ka}O_{ka} + W_{kq}H_{kq})),$$
$$\alpha_a^{(4)} = \text{softmax}(w_{ka}^T \tanh(W_{kq}O_{kq} + W_{ka}H_{ka})),$$



$W_{wq,} W_{wa} \in R^{dh \times dh}$, $w_{wq,} w_{wa} \in R^{dh}$, $W_{kq,} W_{ka} \in R^{df \times df,}$ $w_{kq,} w_{ka} \in R^{df}$: attention parameters to be learned

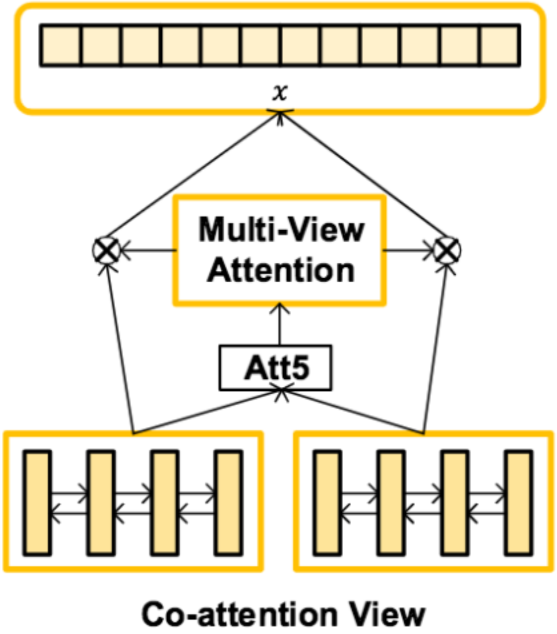
$\alpha_q^{(3)}$ & $\alpha_a^{(3)}$: attention weights from semantic view

$\alpha_q^{(4)}$ & $\alpha_a^{(4)}$: attention weights from knowledge semantic view

Methodology

- Co-attention View

M_{co} =tanh $(S_q^T\ Us\ S_a)$; S_q & S_a : final question and answer representations



Co-attention View

$$\alpha_q^{(5)} = \text{softmax}(\text{Max}(M_{co})),$$

$$\alpha_a^{(5)} = \text{softmax}(\text{Max}({M_{co}}^{T})),$$

$U_W \in R^{ds \times ds}$: attention parameter matrix to be learned

ds : dimension of final QA representation

$\alpha_q^{(5)}$ & $\alpha_a^{(5)}$: co-attention weights for question & answer

Methodology

- Multi-View Attentive Representation

$$\alpha_q = softmax \sum_{i=1}^{5} \lambda_q^{(i)} \alpha_q^{(i)} ; \alpha_a = softmax \sum_{i=1}^{5} \lambda_a^{(i)} \alpha_a^{(i)} ;$$

$\lambda_q^{(i)}$ & $\lambda_a^{(i)}$: hyper parameters that determine the weights of the five kinds of attentions.

- Final attentive question answer representation:

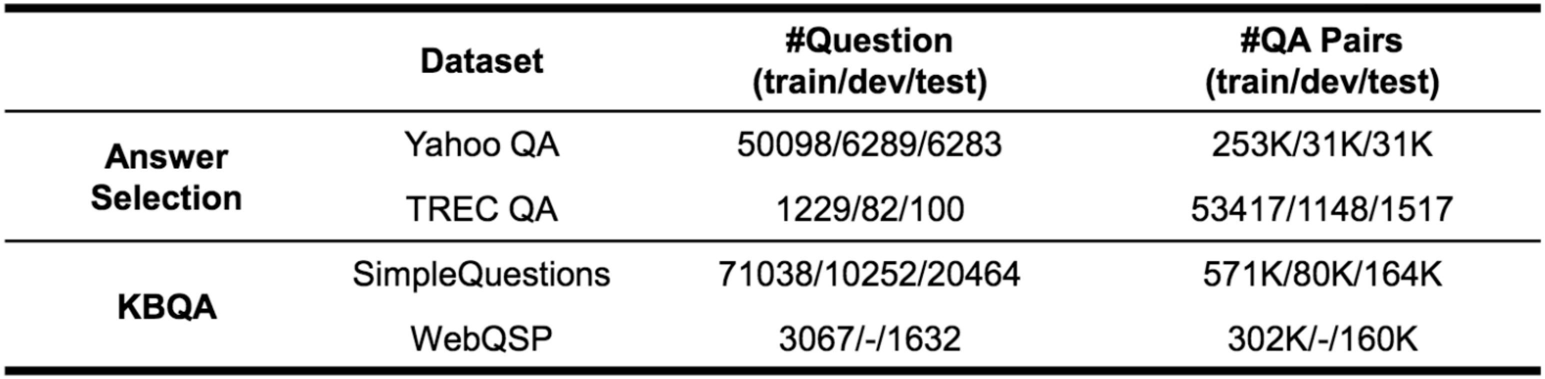
$$s_q = S_q \alpha_q; \quad s_a = S_a \alpha_a;$$

Outline

- Introduction
  - Answer Selection & Knowledge Base Question Answering
  - Multi-task Learning
- Methodology
  - Problem Definition
  - Multi-Task Learning for Question Answering
  - Multi-View Attention Scheme
- **Experiment**
  - Multi-Task Learning Results
  - Ablation Analysis of Multi-View Attention
  - Case Study of Multi-View Attention
- **Summary**

Experiment

- **Dataset**



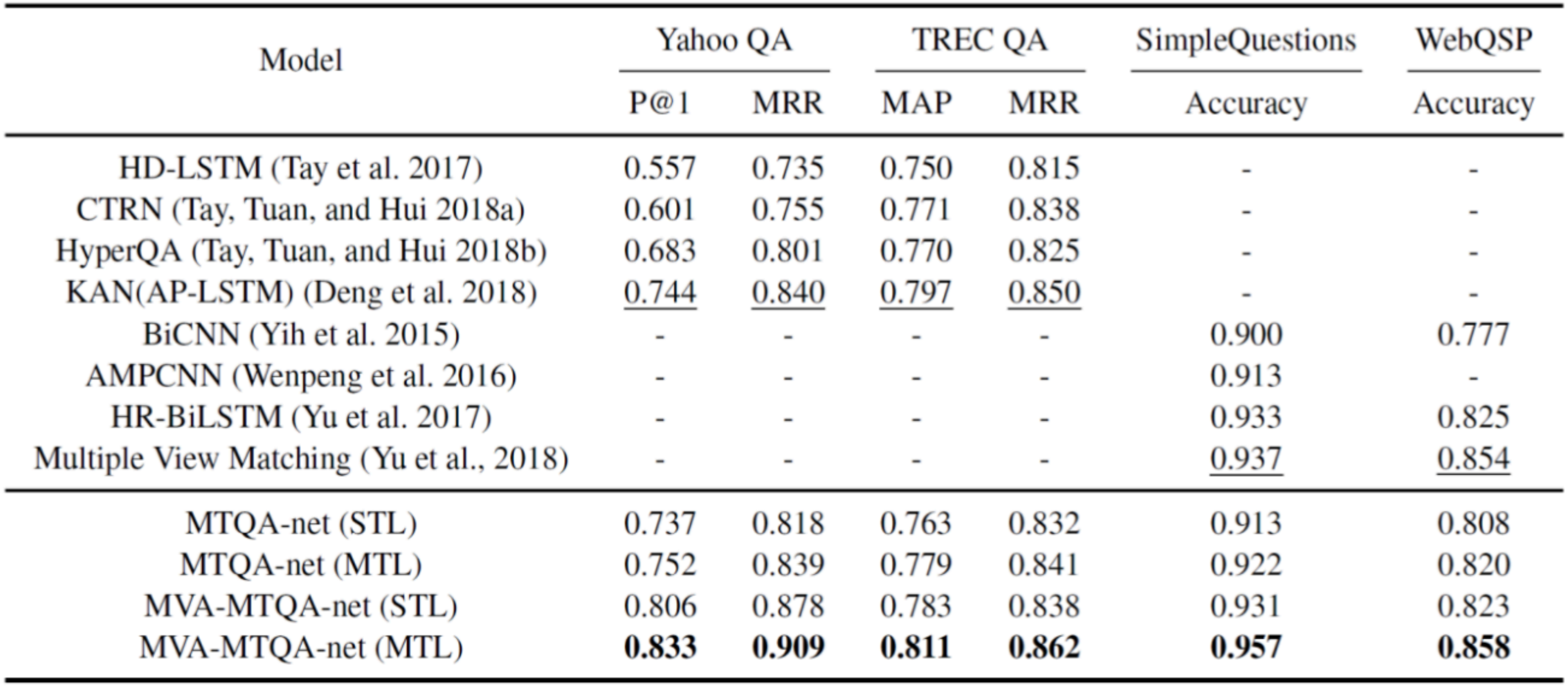
	Dataset	#Question (train/dev/test)	#QA Pairs (train/dev/test)
Answer Selection	Yahoo QA	50098/6289/6283	253K/31K/31K
	TREC QA	1229/82/100	53417/1148/1517
KBQA	SimpleQuestions	71038/10252/20464	571K/80K/164K
	WebQSP	3067/-/1632	302K/-/160K

- Experiment setting:

- Word embeddings: 300 dimension pre-trained Glove
- Knowledge embedding: TransE used to generate knowledge embeddings
- LSTM hidden layer size: 200
- Convolutional filter width: 2 and 3
- Features maps: 100
- Learning rate: 0.0005, dropout rate: 0.5
- Batch size: 128

Experiment

- Multi-task Learning Results

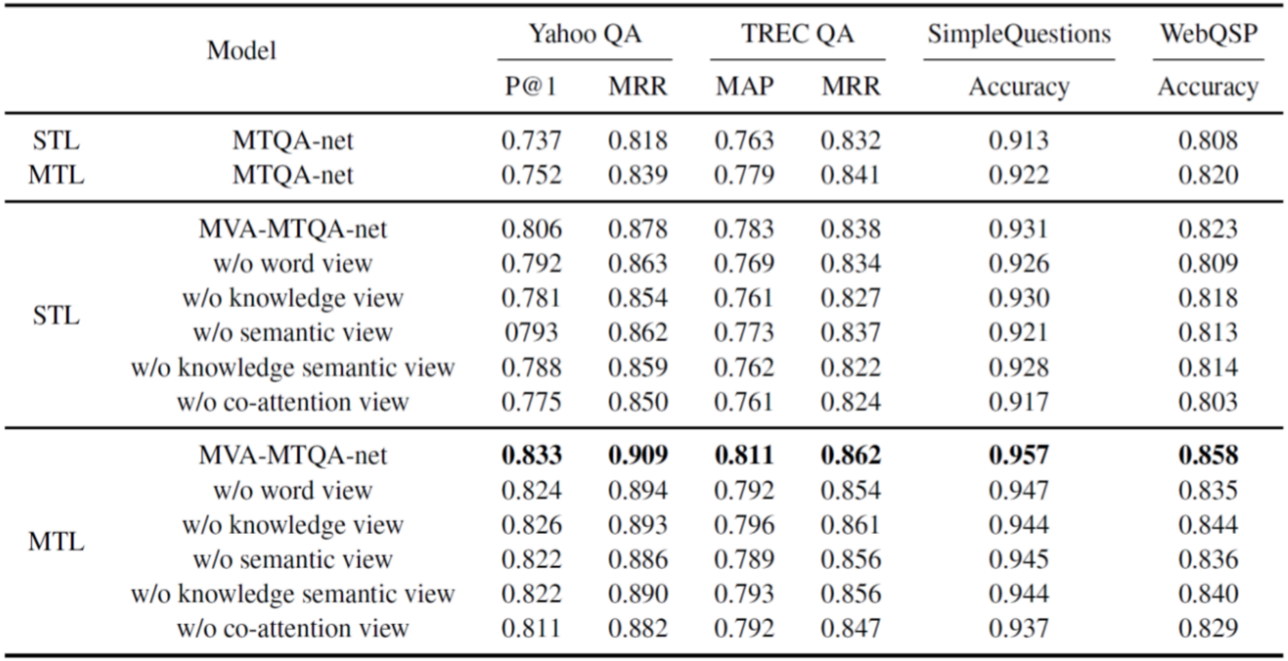


Model	Yahoo QA		TREC QA		SimpleQuestions	WebQSP
	P@1	MRR	MAP	MRR	Accuracy	Accuracy
HD-LSTM (Tay et al. 2017)	0.557	0.735	0.750	0.815	-	-
CTRN (Tay, Tuan, and Hui 2018a)	0.601	0.755	0.771	0.838	-	-
HyperQA (Tay, Tuan, and Hui 2018b)	0.683	0.801	0.770	0.825	-	-
KAN(AP-LSTM) (Deng et al. 2018)	0.744	0.840	0.797	0.850	-	-
BiCNN (Yih et al. 2015)	-	-	-	-	0.900	0.777
AMPCNN (Wenpeng et al. 2016)	-	-	-	-	0.913	-
HR-BiLSTM (Yu et al. 2017)	-	-	-	-	0.933	0.825
Multiple View Matching (Yu et al., 2018)	-	-	-	-	0.937	0.854
MTQA-net (STL)	0.737	0.818	0.763	0.832	0.913	0.808
MTQA-net (MTL)	0.752	0.839	0.779	0.841	0.922	0.820
MVA-MTQA-net (STL)	0.806	0.878	0.783	0.838	0.931	0.823
MVA-MTQA-net (MTL)	**0.833**	**0.909**	**0.811**	**0.862**	**0.957**	**0.858**

- MVA-MTQA-net(MTL) outperforms state of art results by a noticeable margin on all datasets
- In both MVA-MTQA-net and its basic model (MTQA-net), multi-task learning(MTL) methods can significantly improve the performance of all four datasets compared with single-task learning (STL)

Experiment

- Ablation Analysis of Multi-View Attention

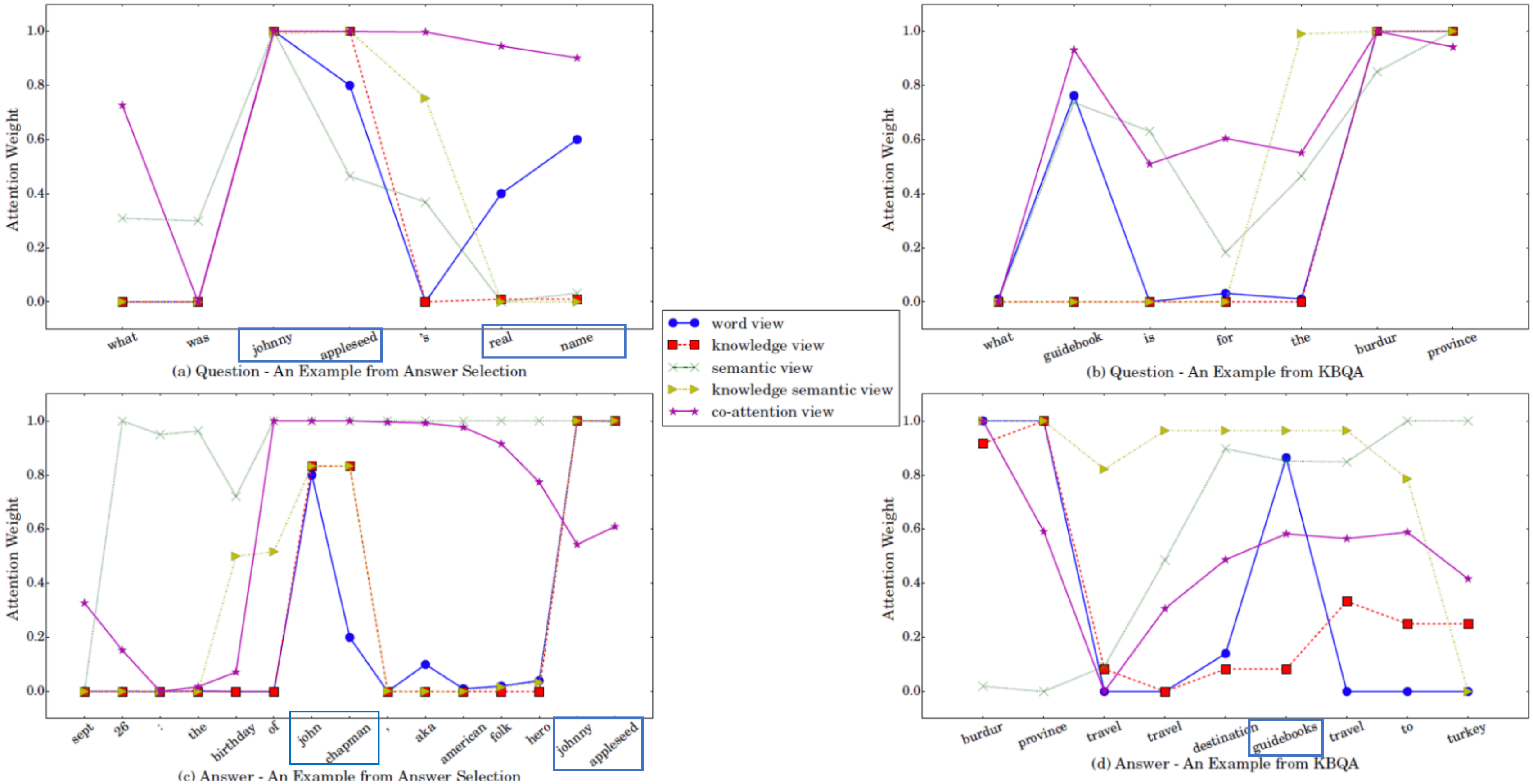


	Model	Yahoo QA		TREC QA		SimpleQuestions	WebQSP
		P@1	MRR	MAP	MRR	Accuracy	Accuracy
STL	MTQA-net	0.737	0.818	0.763	0.832	0.913	0.808
MTL	MTQA-net	0.752	0.839	0.779	0.841	0.922	0.820
STL	MVA-MTQA-net	0.806	0.878	0.783	0.838	0.931	0.823
	w/o word view	0.792	0.863	0.769	0.834	0.926	0.809
	w/o knowledge view	0.781	0.854	0.761	0.827	0.930	0.818
	w/o semantic view	0793	0.862	0.773	0.837	0.921	0.813
	w/o knowledge semantic view	0.788	0.859	0.762	0.822	0.928	0.814
	w/o co-attention view	0.775	0.850	0.761	0.824	0.917	0.803
MTL	MVA-MTQA-net	**0.833**	**0.909**	**0.811**	**0.862**	**0.957**	**0.858**
	w/o word view	0.824	0.894	0.792	0.854	0.947	0.835
	w/o knowledge view	0.826	0.893	0.796	0.861	0.944	0.844
	w/o semantic view	0.822	0.886	0.789	0.856	0.945	0.836
	w/o knowledge semantic view	0.822	0.890	0.793	0.856	0.944	0.840
	w/o co-attention view	0.811	0.882	0.792	0.847	0.937	0.829

- All kinds of view contribute more or less performance boost to the model.
- Co-attention view attention makes the most contribution to the improvement
- For STL , knowledge and knowledge semantic view attentions are more distinguishable than word view and semantic view in two answer selection tasks, while the word view and semantic attentions contribute more in the KBQA tasks.
- For MTL, we observe that each view of attention makes similar contribution to the Improvement in four tasks.

Experiment

- Case Study of Multi-View attention



(a) Question - An Example from Answer Selection

(b) Question - An Example from KBQA

(c) Answer - An Example from Answer Selection

(d) Answer - An Example from KBQA

Summary

- We explore multi-task learning approaches for answer selection and knowledge base question answering.
- We propose a novel multi-task learning scheme that leverages multi-view attention mechanism to bridge different asks.
- Experimental results show that multi-task learning of answer selection and KBQA outperforms state-of-the-art single-task learning methods