Knowledge as A Bridge: Improving Cross-domain Answer Selection with External Knowledge

Yang Deng[1], **Ying Shen**[1,*], **Min Yang**[2], **Yaliang Li**[3], **Nan Du**[3], **Wei Fan**[3], **Kai Lei**[1]
[1]School of Electronics and Computer Engineering, Peking University Shenzhen Graduate School
[2]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
[3]Tencent Medical AI Lab
ydeng@pku.edu.cn, shenying@pkusz.edu.cn, min.yang@siat.ac.cn,
{yaliangli,ndu,davidwfan}@tencent.com, leik@pkusz.edu.cn

Abstract

Answer selection is an important but challenging task. Significant progress has been made in domains where a large amount of labeled training data is available. However, obtaining rich annotated data is a time-consuming and expensive process, creating a substantial barrier for applying answer selection models to a new domain which has limited labeled data. In this paper, we propose Knowledge-aware Attentive Network (KAN), a transfer learning framework for cross-domain answer selection, which uses the knowledge base as a bridge to enable knowledge transfer from the source domain to the target domains. Specifically, we design a knowledge module to integrate the knowledge-based representational learning into answer selection models. The learned knowledge-based representations are shared by source and target domains, which not only leverages large amounts of cross-domain data, but also benefits from a regularization effect that leads to more general representations to help tasks in new domains. To verify the effectiveness of our model, we use SQuAD-T dataset as the source domain and three other datasets (i.e., Yahoo QA, TREC QA and InsuranceQA) as the target domains. The experimental results demonstrate that KAN has remarkable applicability and generality, and consistently outperforms the strong competitors by a noticeable margin for cross-domain answer selection.

1 Introduction

Answer selection, which is a key component of question answering (QA), has attracted increasing attention recently due to its broad applications in natural language processing and information retrieval, such as factoid question answering (Wang et al., 2007) and community-based question answering (Tay et al., 2017). Given a question, answer selection aims to pick out the most relevant answer from a set of candidates. In the literature, answer selection has been extensively studied in the last decade (Severyn and Moschitti, 2015; Wang and Nyberg, 2015; Tan et al., 2016; dos Santos et al., 2016).

Despite the effectiveness of previous studies, answer selection remains a challenge in real-world applications for two reasons. (1) The background information and knowledge beyond the context, which play crucial roles in human text comprehension, have received little attention in recent work for answer selection. (2) Impressive answer selection performances were achieved in domains where a large amount of labeled data is available. However, such fruitful results are subject to an assumption that the test data should be drawn from the same distribution as the training data. Previous studies struggle to cope with answer selection across different data domains. For example, the model trained on SQuAD-T dataset that consists of open-domain factoid questions, is difficult to generalize to the InsuranceQA dataset that consists of non-factoid questions in the insurance domain. In a new domain where its own labeled data is in short supply, obtaining more labels is usually labor-intensive and time-consuming.

Knowledge base (KB), such as YAGO (Weikum et al., 2007), Freebase (Bollacker et al., 2008), provides rich information of relations between entities. It has been widely studied and applied in many tasks (Yang and Mitchell, 2017; Liu et al., 2017). However, its applicability to answer selection has yet to be

* Corresponding Author

Proceedings of the 27th International Conference on Computational Linguistics, pages 3295–3305
Santa Fe, New Mexico, USA, August 20-26, 2018.

Question	Who played ***Dumbledore*** in ***Harry Potter***?
Positive Answer	After Harris' death , ***Michael Gambon*** portrayed ***Dumbledore*** for all of the remaining films.
Negative Answer	Professor Albus Percival Wulfric Brian ***Dumbledore*** is a major character and protagonist of ***J. K. Rowling***'s ***Harry Potter*** series.

Table 1: Example of QA Candidate Pairs.

well-studied. Considering the example in Table 1, existing context-based models may assign a higher score to the negative answer than the positive answer, since the negative answer is more similar to the given question at word level. However, with the background knowledge, we can correctly identify the positive answer based on the relative facts contained in the knowledge base (KB) such as (*Dumbledore*, *played_by*, *Michael Gambon*), (*Michael Gambon*, *cast_in*, *Harry Potter*). Furthermore, QA datasets in different domains or types might differ from syntactic and lexical features, but relations of knowledge in sentences are coherent in the same knowledge base.

Inspired by recent work on transfer learning (TL) and domain adaptation, in this paper, we study how we can leverage labeled data of source domain and external knowledge in knowledge base to help the answer selection in the new target domain which has only limited labeled data. Although transfer learning was employed in many applications (Mou et al., 2016; Li et al., 2017), its application in answer selection is still a relatively new territory and under-explored.

Our idea of leveraging external knowledge as a bridge between source and target domains is motivated by the observation that the data in different domains shares certain common background knowledge which can possibly be transferred from the source domain to the target domain. Thus, we proposed Knowledge-aware Attentive Network (KAN) for transfer learning on answer selection task. In particular, we design a context-guided attentive convolutional neural network, which incorporates knowledge embeddings into sentence representations, to strengthen the representation learning of documents. The training of the transfer learning is performed in two steps: First, the proposed model is trained with the labeled data from source domain and the external knowledge from knowledge base, called pre-training procedure. We expect that the pre-training learns the knowledge-based representation, which enables domain-independent knowledge to be transferred across domains. In addition, pre-training also gives a good initialization of the model parameters, and therefore training at the latter stage gives a good generalization performance even if the size of the target domain dataset is limited. Second, we fine-tune the model on a target domain dataset which has limited labeled data, with the hope that one can safeguard the performance of answer selection in the target domain by leveraging the shared knowledge learned from knowledge base.

To verify the effectiveness of our model for cross-domain answer selection, we use SQuAD-T data as the source domain and three other datasets (i.e., Yahoo QA, TREC QA and InsuranceQA) as the target domains. The experimental results show that our model consistently outperforms previous methods.

2 Related Work

Answer Selection Recent years have witnessed great successes of applying different neural networks, e.g., convolutional neural network (CNN) (Severyn and Moschitti, 2015) and recurrent models like the long short-term memory (LSTM) (Wang and Nyberg, 2015), into Answer Selection. The key idea behind deep neural networks is to encode the input sentences as vector representations. Based on the representations, an output layer is utilized to provide the matching score of two texts. Instead of learning the representations of the question and the answer separately, some recent studies exploit attention mechanisms to learn the interaction information between questions and answers, which can better focus on relevant parts of the input (Tan et al., 2016; dos Santos et al., 2016; Chen et al., 2017). However, these methods are subject to the amount of labeled data and the limited information provided by contexts. Thus, we attempt to apply transfer learning and knowledge base to address these issues.

Transfer Learning Transfer learning aims to transfer knowledge from the source data to the target data in different domains, tasks, or distributions (Pan and Yang, 2010). Most recent studies in transfer learning for natural language processing employ deep neural networks to learn the shared feature representation between two different datasets (Mou et al., 2016; Li et al., 2017). However, it was not until recent years that the application of transfer learning on QA received extensive attention. Min et al. (2017) and Wiese et al. (2017) both employ supervised transfer learning techniques to pre-train a model from a large-scale dataset. Yuan et al. (2018) and Yu et al. (2018) study unsupervised transfer learning under the circumstance that there is only a little labeled target data or some unlabeled target data. In this paper, to make the shared representation more knowledge-aware, we go deeper into the first kind of researches and incorporate external knowledge into transfer learning framework in our method.

Knowledge Base Application As seen in many other tasks, it is a trend to leverage external knowledge from KBs to enrich the representational learning of deep learning models. Several efforts have been made on integrating knowledge embeddings trained by knowledge embedding methods (Bordes et al., 2013) to learn a knowledge-aware sentence representation on machine reading (Yang and Mitchell, 2017), entity typing (Xin et al., 2018) and relation extraction (Han et al., 2018). In this paper, we aim to develop a more general framework for current models to learn knowledge-aware sentence representations on answer selection task.

Inspired by these researches, we propose a transfer learning method for answer selection task, which leverages external knowledge from knowledge base to enrich the representational learning of QA sentences as well as bridge cross-domain QA datasets.

3 Model

In this section, we present the general framework of our transfer learning method on answer selection, which leverages external knowledge from knowledge base as a bridge connecting cross-domain QA datasets. Given a question q, our model aims to rank a set of candidate answers $A = \{a_1, \ldots, a_n\}$.

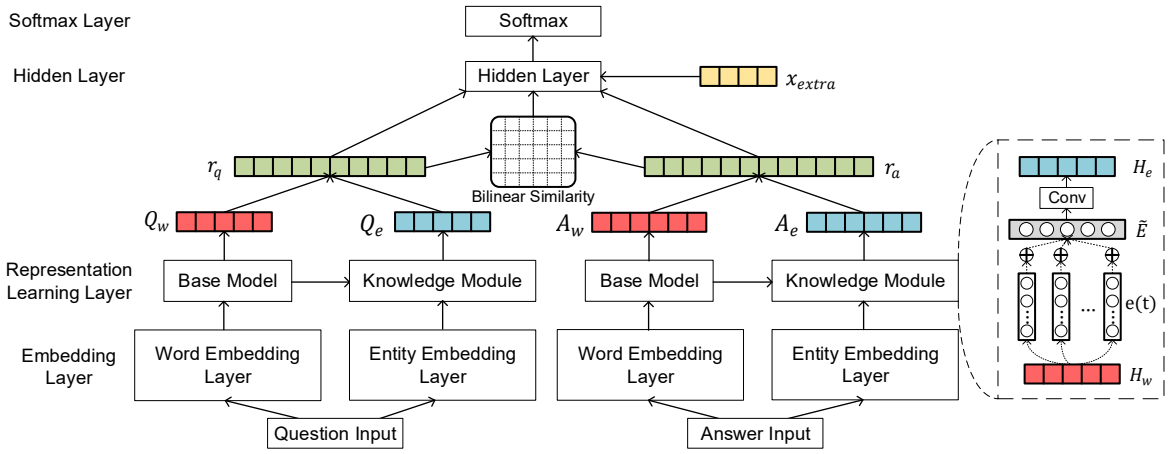

Figure 1: Knowledge-aware Attentive Network for Cross-domain Answer Selection. Blue, red and green matrices denote knowledge-based representations, context-based representations and final knowledge-aware sentence representations, respectively.

As is illustrated in Figure 1, the overall architecture of KAN contains two main components: *Base Model* and *Knowledge Module*. In base model, we employ a pair of deep neural networks to learn the initial context-based representations of questions and answers, separately (Section 3.1). In knowledge module, a context-guided attentive CNN is designed to learn the knowledge-based sentence representation from entities in the sentence (Section 3.2). Afterwards, for both question and answer sentences, there are two different sentence-level representation vectors. Q_w and A_w are learned from base model, while

Q_e and A_e are derived from knowledge module. We obtain the final knowledge-aware attentive sentence representations of question q (i.e., $r_q = [Q_w : Q_e]$) and answer a (i.e., $r_a = [A_w : A_e]$), where [:] is the concatenation operation. There is a fully connected hidden layer before the final binary classification to join all the features (Section 3.3). Finally, a transfer learning method is proposed to transfer the shared knowledge to bridge two different datasets (Section 3.4).

3.1 Base Models

Given a question q and a set of candidate answers $A = \{a_1, \ldots, a_n\}$, we first transform them into vector representations with an embedding layer, and then input these embedding vectors into the base model.

Among numerous deep learning models proposed for answer selection task, we adopt several popular and typical models as our base model to demonstrate the strong applicability and generality of our transfer learning method. The selected base models include: (1) Bi-LSTM, a bidirectional Long-Short Term Memory (Bi-LSTM) network to generate sentence representation (Wang and Nyberg, 2015); (2) Att-LSTM, an Attentive LSTM model with a simple but effective attention mechanism for the purpose of improving the semantic representations for the answers based on the questions (Tan et al., 2016); (3) AP-LSTM, an AP-LSTM model with attentive pooling, a two-way attention mechanism that information from the question and the answer can directly influence the computation of each others representations (dos Santos et al., 2016); and (4) Conv-RNN, a hybrid framework of attention-based Convolutional Recurrent Neural Network (Conv-RNN) (dos Santos et al., 2016) with a similar attention mechanism as Att-LSTM but in the input of the RNN. Note that, unlike the original paper of Conv-RNN, we employ Bi-LSTM as the RNN model instead of the Bi-GRU. For the details of the models, please refer to the original papers.

As for a question vector q and an answer vector a, we obtain the initial context-based sentence representation H_w for the question and the answer from the base model, namely Q_w and A_w.

3.2 Knowledge Module

Knowledge module is responsible for transferring external knowledge from KB into the knowledge-based sentence representations. We first employ n-gram matching to detect all the entity mentions in the sentence, and then retrieve a set of top-K entity candidates from KB for each entity mention. However, the ambiguity issue of the entity is still remained to be tackled, e.g., Santiago can refer to a city or a person. Thus, a context-guided attention mechanism is designed to perform a soft entity linking and learn the knowledge representation for each entity mention simultaneously. We present candidate entities that related to the t-th word in the sentence as $e(t) = \{e_{t_1}, e_{t_2}, \ldots, e_{t_K}\} \in \mathbb{R}^{K \times d_e}$, where d_e is the dimension of the entity embedding in KB. Then, the context-guided knowledge embedding for t-th word is given by

$$m(t) = \tanh\left(W_{em} e(t) + W_{hm} H_w\right), \tag{1}$$

$$\alpha_{t_i} = \frac{exp\left(w_m^T m_{t_i}\right)}{\sum_{m_{t_j} \in m(t)} exp\left(w_m^T m_{t_j}\right)}, \tag{2}$$

$$\widetilde{e_t} = \sum_{e_{t_i} \in e(t)} \alpha_{t_i} e_{t_i}, \tag{3}$$

where W_{em}, W_{hm} and w_m are parameter matrices to be learned. $m(t)$ is a context-guided knowledge vectors, and α_{t_i} denotes the context-guided attention weight that is applied over each candidate entity embedding e_{t_i}. The contextual sentence representations H_w of questions and answers are learned by the base model, while the embeddings of entities in KB are learned by TransE (Bordes et al., 2013).

This procedure produces a context-guided representation for each entity mention in the sentence. Then we apply an CNN layer to learn a higher level knowledge-based sentence representation. The input of the CNN layer are the attentive knowledge embeddings $\widetilde{E} = \{\widetilde{e_1}, \widetilde{e_2}, \ldots, \widetilde{e_L}\} \in \mathbb{R}^{L \times d_e}$.

In the convolution layer, a filter of size n slides over the input embedding matrix to capture the local n-gram information, which is useful to extract the entity features since an entity is likely to be a phrase. Each move computes a hidden layer vector as

$$x_i = [\widetilde{e}_{i-\frac{n-1}{2}}, \ldots, \widetilde{e}_i, \ldots, \widetilde{e}_{i+\frac{n-1}{2}}], \tag{4}$$

$$h_i = \tanh\left(W x_i + b\right), \tag{5}$$

where W and b are the convolution kernel and the bias vector to be learned. Then we employ max-pooling over the hidden layer vectors $h_1, \ldots, h_n$ to generate the final output vector y:

$$y_j = max\left\{h_{1j}, \ldots, h_{nj}\right\}, \tag{6}$$

where y_j and h_{ij} are the j-th value of the output vector y and the hidden vector h_i.

Due to the uncertainty of the length of entities, we exploit several filters of various sizes to obtain n-gram features $\left\{y^{(1)}, y^{(2)}, \ldots, y^{(n)}\right\}$, where $y^{(i)}$ denotes the output vector obtained by the i-th filter. We pass these output vectors through a fully-connected layer to get the final knowledge-based sentence embedding $H_e \in \mathbb{R}^{L \times d_f}$, where d_f is the total filter sizes of CNN and L is the length of the sentence. As for the question q and the answer a, we generate their knowledge-based sentence representations Q_e and A_e as:

$$Q_e = [y_q^{(1)}, y_q^{(2)}, \ldots, y_q^{(n)}]; \quad A_e = [y_a^{(1)}, y_a^{(2)}, \ldots, y_a^{(n)}]. \tag{7}$$

3.3 Training

The initial context-based representations and the knowledge-based representations are concatenated to form the final sentence representations of question q and answer a:

$$r_q = [Q_w : Q_e]; \quad r_a = [A_w : A_e]. \tag{8}$$

Following the ideas in Severyn and Moschitti (2015) and Tay et al. (2017), we incorporate the same additional features into our overall architecture. First, we compute the bilinear similarity score between final question and answer vectors, $s\left(r_q, r_a\right) = r_q^T W r_a$, where $W \in \mathbb{R}^{L \times L}$ is a similarity matrix to be learned. Besides, the same word overlap features $x_{extra} \in \mathbb{R}^4$ as Severyn and Moschitti (2015) and Tay et al. (2017) are also integrated into our model. Thus, the inputs of the hidden layer is a vector $x = [r_q, s(r_q, r_a), r_a, x_{extra}]$, and its output then go through a softmax layer for binary classification:

$$y(q, a) = \text{softmax}\left(W_s x + b_s\right). \tag{9}$$

where $W_s \in \mathbb{R}^{d_x \times 2}$ and $b_s \in \mathbb{R}^2$ are the parameters in the hidden layer. The overall end-to-end model is trained to minimize the cross-entropy loss function:

$$L = -\sum_{i=1}^{N} \left[y_i \log p_i + (1 - y_i) \log\left(1 - p_i\right)\right] + \lambda \|\theta\|_2^2, \tag{10}$$

where p is the output of the softmax layer. θ contains all the parameters of the network and $\lambda\|\theta\|_2^2$ is the L2 regularization.

3.4 Transfer Learning Method

Our goal is to leverage external knowledge from an open-domain knowledge base to bridge cross-domain answer selection data. The transfer learning method between different datasets is divided into two steps: we first initialize the the parameters of model pre-trained on the source dataset, then we further fine-tune on the target dataset. A straightforward way is to fine-tune all the parameters pre-trained by the source data on the target training dataset. Another fashion is to fine-tune a certain part of parameters and keep the rest of parameters fixed during fine-tuning. In our overall model, we present three ways to

fine-tune: (1) Find-tune the entire model; (2) Only find-tune all the weights in the knowledge module. This way aims at sharing the context-based representational learning module learned from the source data and training a better knowledge-based representational learning module with the target training set; (3) Only find-tune all the weights in the base model. On the contrary, this way considers knowledge-based representational learning module a coherent part for transfer, but the context-based representational learning module need to be fine-tuned. Note that for training the model, we followed the same procedure as in Yuan et al. (2018), where pre-trained word embeddings are not updated during training. In this work, so are pre-trained knowledge embeddings.

4 Experiments

4.1 Datasets and Metrics

In the paper, we adopt four widely-used QA datasets for evaluation, including two factoid QA datasets, SQuAD-T and TREC QA, and two community-based QA datasets, InsuranceQA and Yahoo QA. We adopt SQuAD-T as our source dataset for transfer learning due to its high quality and large quantity, while the other datasets are used as target datasets.

- **TREC QA**, collected from TREC QA track 8-13 data (Wang et al., 2007), is a benchmark for open-domain factoid question answering. Most questions are short and factoid-based, and answers are usually consisting trivia information. Following previous works (Tay et al., 2018; Rao et al., 2016), we use Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR) as evaluation metrics.

- **InsuranceQA**, proposed by (Feng et al., 2015), contains community-based question and answer pairs from the insurance domain and is split into a training set, a validation set, and two test sets. For the development and test sets, the InsuranceQA also includes an answer pool of 500 candidate answers for each question. This answer pool was produced by including the correct answer and randomly selected candidates from the complete set of unique answers. The top-1 accuracy of the answer selection is reported.

- **Yahoo QA**, introduced in Tay et al. (2017), is a cleaned version of original Yahoo QA dataset [1], which is an open-domain CQA dataset collected from Yahoo Answers. In their setting, questions and answers that are not in the range of 5 - 50 tokens are filtered. Additionally, 4 negative samples are generated for each question by sampling from the top 1000 hits. Note that almost all the cases in this dataset are non-factoid questions. For this dataset, we use the same metrics as (Tay et al., 2017; Tay et al., 2018), including Precision@1 and MRR.

- **SQuAD-T**, introduced in Min et al. (2017), is a modification of SQuAD dataset (Rajpurkar et al., 2016) for answer selection task. SQuAD is a recent open-domain machine reading QA dataset, in which each case is a pair of context paragraph from Wikipedia and a question created manually, and the answer is a span in the context. The task of SQuAD-T is to classify whether each sentence contains the answer, since its contained context paragraphs are split into sentences.

The statistics of all datasets, i.e., training sets, development sets and testing sets, are given in Table 2.

Dataset	#Question (train/dev/test)	#QA Pairs (train/dev/test)	Question Example
TREC QA	1229/82/100	53417/1148/1517	What was the first Gilbert and Sullivan opera?
InsuranceQA	12887/1000/1800x2	37.1K/500K/900Kx2	Why have renter insurance?
Yahoo QA	50.1K/6.2K/6.2K	253K/31.7K/31.7K	How to maximize returns on your investments?
SQuAD-T	87.1K/10.5K/-	708K/53.6K/-	What sits on top of the main building at Notre Dame?

Table 2: Summary statistics of datasets.

[1] http://webscope.sandbox.yahoo.com/catalog.php?datatype=l&did=10

4.2 Experimental Settings

Pre-trained GloVE embeddings[2] of 300 dimensions are adopted as word embeddings. We use a subset of Freebase (FB5M[3]) as our KB, which includes 4,904,397 entities, 7,523 relations, and 22,441,880 facts.

For the base models, we followed exactly the same parameter settings as those in their original papers. For the knowledge module, the width of the convolution filters is set to be 2 and 3, and the number of convolutional feature maps and the attention sizes are set to be 200. For the rest of our models, the final hidden layer size is set to 200 and all other parameters are randomly initialized from [-0.1, 0.1]. The model parameters are regularized with an L2 regularization strength of 0.0001. The learning rate and the dropout rate are set to 0.0005 and 0.5 respectively. The maximum length of sentence is set to be 40. We train all the models in batches with size of 64.

4.3 Results and Analysis

4.3.1 Comparisons Between TL Methods

To evaluate the proposed transfer learning method, we conduct the following comparison experiment with the same base model AP-LSTM (dos Santos et al., 2016), i.e., KAN(AP-LSTM). We first pre-train the models on the source dataset, and then fine-tune the models on target datasets. We also report the ablation tests in terms of discarding the knowledge module (i.e., AP-LSTM) and exploiting several different TL methods as follows:

- **Tgt-Only** is the baseline trained in the training set of target data.
- **Src-Only** is another baseline trained in the training set of source data.
- **Mixed** is to simply mix the training data from both target and source dataset to train the model.
- **Fine-Tune** is a widely used TL method, where we first train a model on the source data, and then use the learned parameters to initialize the model parameters for training another model on the target data. As is presented in Section 3.4, we employ three kinds of fine-tuning methods. Fine-Tune (all) means that we fine-tune all the parameters on the target data. Find-Tune (1) indicates that we fix all the weights of the base model and fine-tune parameters of the rest part of the overall model, while we fix all the weights of the knowledge module in Find-Tune (2).

Model	TL Method	Yahoo QA		TREC QA		InsuranceQA		
		P@1	MRR	MAP	MRR	DEV	TEST1	TEST2
AP-LSTM	(a) Tgt-Only	60.8	76.1	75.9	81.0	69.2	69.8	67.0
	(b) Src-Only	14.5	36.2	74.9	78.1	61.1	62.1	58.8
	(c) Mixed	56.8	73.1	75.1	79.5	69.1	70.2	67.3
	(d) Fine-Tune (all)	72.4	81.9	76.9	82.4	72.2	73.3	70.4
KAN(AP-LSTM)	(e) Tgt-Only	67.2(+6.4)	80.3(+4.2)	78.1(+2.2)	83.1(+2.1)	71.3(+2.1)	71.5(+1.7)	68.8(+1.8)
	(f) Src-Only	15.9(+1.4)	37.4(+1.2)	76.7(+1.8)	81.8(+3.7)	58.1(-3.0)	59.4(-2.7)	56.1(-2.7)
	(g) Mixed	62.4(+5.6)	76.8(+3.7)	76.7(+1.6)	81.6(+2.1)	70.2(+1.1)	71.0(+0.8)	68.5(+1.2)
	(h) Fine-Tune (all)	74.2(+1.8)	83.4(+1.5)	78.8(+1.9)	84.1(+1.7)	**74.6**(+2.4)	75.3(+2.0)	72.1(+1.7)
	(i) Fine-Tune (1)	73.0	82.6	78.7	84.1	74.4	**75.6**	**72.6**
	(j) Fine-Tune (2)	**74.4**	**84.0**	**79.7**	**85.0**	74.4	75.2	72.5
Rank 1		60.1	75.5	77.1	83.8	71.7	71.4	68.3
Rank 2		57.3	73.6	78.0	83.4	68.4	71.7	66.4
Rank 3		55.7	73.5	75.0	81.5	70.0	70.1	62.8

Table 3: Comparisons Between TL Methods. The number in the parenthesis indicates the accuracy change over the same TL method in the base model.

Table 3 reports the experimental results of our transfer learning method on Yahoo QA, InsuranceQA and TREC QA and the performance of previous models that achieve the state of the art. To compare the previous competing models, we adopt the best three results on each dataset reported in the literature, which are presented as Rank 1,2,3 in Table 3. Both Yahoo QA and TREC QA results are reported from

[2]http://nlp.stanford.edu/data/glove.6B.zip

[3]https://research.facebook.com/researchers/1543934539189348

Tay et al. (2018), including Tay et al. (2018), Bradbury et al. (2016), Rao et al. (2016) and Tay et al. (2017). For InsuranceQA, they are from Wang et al. (2017), dos Santos et al. (2016), and Wang et al. (2016). There are multiple interesting observations from Table 3 as follows:

(1) Our transfer learning method (row (h,i,j)) outperforms the state-of-the-art results on Yahoo QA, TREC QA and InsuranceQA datasets by about 10%, 2% and 4% respectively.

(2) The results of KAN(AP-LSTM) (row (e-h)) significantly outperform AP-LSTM (row (a-d)), which demonstrates the effectiveness of incorporating external knowledge into the base model. Among these results, Src-Only on InsuranceQA is an exception since the target dataset is completely different from the source dataset in domain, which interferes the training of the knowledge module with the source dataset and brings a negative effect on the performance.

(3) For row (e), (f) and (g), we observe that Src-Only perform much worse than Tgt-Only, which indicates that the source QA dataset is quite different from all the target QA datasets. The greater the difference between the experimental results, the greater the difference between source and target datasets. Thus, we notice that Yahoo QA is the most different from SQuAD-T, and InsuranceQA is also very different from SQuAD-T, while TREC QA is the most similar one. Besides, the performance of Mixed is also worse than Tgt-Only, which implies that simply mixing the training data from two different datasets may lead to the overfitting of the source data.

(4) For row (h), (i) and (j), it can be observed that the Fine-Tune methods outperform the Tgt-Only method (row (e)), demonstrating that pre-training the model parameters on a source dataset is better than randomly initializing them. In specific, fine-tuning all the parameters of the model (row (h)) cannot always achieve the best result, because it may cause overfitting to fine-tune the entire model. For Yahoo QA and TREC QA, only fine-tuning the parameters of the base model (row (j)) achieves the best result. We infer that it is due to the shared knowledge from knowledge base is coherent between the source and target datasets. Conversely, InsuranceQA achieves its best performance with fine-tuning only the weights of the knowledge module (row (i)). Results on InsuranceQA indicate that one should pay attention to fine-tune the knowledge module rather than the base model when domain knowledge of the target dataset is quite different from the source dataset as InsuranceQA and SQuAD-T.

4.3.2 Applicability and Generality of our TL Method

To demonstrate the applicability and generality of our transfer learning method, besides AP-LSTM, we implement the overall transfer learning framework with other three base models, including Conv-RNN (Wang et al., 2017), Bi-LSTM (Wang and Nyberg, 2015), and Att-LSTM (Tan et al., 2016). The experimental results on TREC QA are summarized in Table 4.

From Table 4, we observe a similar result as Section 4.3.1. For these three base models, it makes a significant performance boost to incorporate external knowledge into the overall architecture. Besides, compared with Tgt-Only, Fine-Tune methods perform much better, which demonstrates that our transfer learning framework works on all the given models.

TL	Knowledge	Conv-RNN		Bi-LSTM		Att-LSTM	
Method	Module	MAP	MRR	MAP	MRR	MAP	MRR
Tgt-Only	w/o	77.1	82.4	75.0	80.4	73.5	79.2
	w/	78.0	83.1	77.4	82.5	75.9	80.1
Src-Only	w/o	74.9	78.7	74.1	79.5	72.1	76.9
	w/	76.3	80.2	75.6	81.3	74.8	78.6
Mixed	w/o	76.4	81.0	75.2	80.6	73.3	79.3
	w/	77.6	82.7	77.0	82.4	75.5	80.0
Fine-Tune (all)	w/o	78.3	83.2	76.8	82.3	75.3	79.8
	w/	79.4	**84.8**	78.2	83.6	77.1	81.3
Fine-Tune (1)	w/	78.8	84.0	78.4	83.1	77.7	81.6
Fine-Tune (2)	w/	**79.7**	**84.8**	**78.5**	**83.7**	**77.9**	**81.9**

Table 4: Experiment with Different Base Models.

4.3.3 Size of Target Dataset for Fine-tuning

We conduct experiments to study the relationship between the performance and the amount of training data from the target dataset for fine-tuning the model. We first pre-train the models on SQuAD-T, then vary the training data size of the target dataset, i.e., TREC QA and InsuranceQA, for fine-tuning. Note that we employ the AP-LSTM as the base model (i.e., KAN(AP-LSTM)) and fix all the parameters of the knowledge module during fine-tuning (i.e., Fine-Tune (2)).

The experimental result is summarized in Table 5. In general, fine-tuning with more target data actually improves the overall results. We observe that the performance increase from using 0% to 20% of target training data is substantially larger than latter increases. This result shows that we can achieve a competitive result with a small amount of target labeled data, by using our transfer learning method. Besides, it is obvious to notice that when increasing the number of target training data, the improvement on InsuranceQA is much more significant than that on TREC QA. As is presented in the analysis in Section 4.3.1, the InsuranceQA dataset is much different from the source dataset than the TREC QA dataset, not only in types but also in domains. Thus, the experiment result demonstrates the strong applicability in transferring shared knowledge between diverse datasets, even with limited labeled target data.

Percentage of target	TREC QA		InsuranceQA		
data for fine-tuning	MAP	MRR	DEV	TEST1	TEST2
0%	76.7	81.8	58.1	59.4	56.1
20%	78.1 (1.4)	83.4 (1.7)	72.0 (13.9)	72.8 (13.4)	69.5 (13.4)
40%	78.9 (0.8)	83.7 (0.3)	73.2 (1.2)	74.0 (1.2)	71.1 (1.6)
60%	79.1 (0.2)	84.2 (0.5)	73.6 (0.4)	74.5 (0.5)	71.6 (0.5)
80%	79.3 (0.2)	84.7 (0.5)	74.1 (0.5)	74.9 (0.4)	72.1 (0.6)
100%	79.7 (0.4)	85.0 (0.3)	74.4 (0.3)	75.2 (0.3)	72.5 (0.4)

Table 5: Results of varying sizes of the target datasets used for fine-tuning. The number in the parenthesis indicates the accuracy increases over the previous row.

4.3.4 Completeness of Knowledge Base

To observe the effect of the completeness of KB on the performance, we report the results on TREC QA and InsuranceQA with the incomplete knowledge base that randomly drops 20%-80% knowledge. Note that as the experimental settings in Section 4.3.3, we also employ the AP-LSTM as the base model (i.e., KAN(AP-LSTM)) and fix all the parameters of the knowledge module during fine-tuning (i.e., Fine-Tune (2)). Figure 2 shows that our model is robust and achieves excellent performance on the KB with different completeness. Besides, the more complete the knowledge base we leverage, the better the overall performance, which demonstrates the effectiveness of integrating external knowledge into the proposed method.

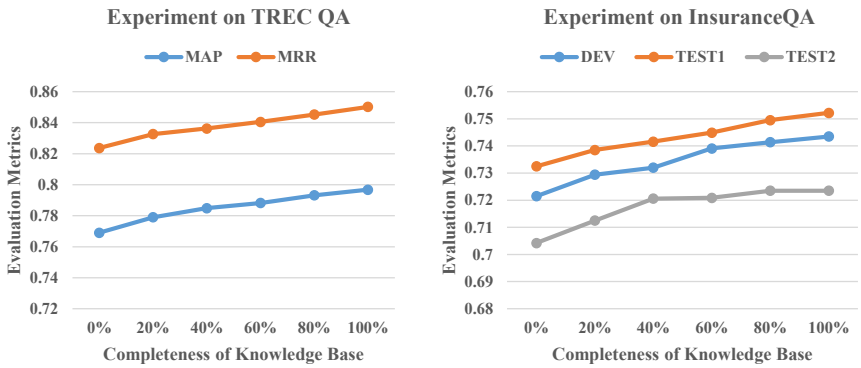

Figure 2: Effect of KB Completeness

5 Conclusion

In this paper, we propose knowledge-aware attentive network (KAN), a novel and general transfer learning framework, for cross-domain answer selection. We incorporate external knowledge from knowledge base into deep learning models to enrich the sentence representational learning and aid in transferring more valuable information between cross-domain datasets. Experimental results on three benchmark datasets demonstrate the superiority of our proposed method on answer selection task. We also conduct experiments to show the applicability and generality of our method and show that a resource-poor dataset can benefit from not only the scale of a resource-rich dataset but also the shared knowledge learned from knowledge base.

Acknowledgements

This work was financially supported by the National Natural Science Foundation of China (No.61602013), the Shenzhen Science and Technology Innovation Committee (Grant No. JCYJ20170412150946024 and JCYJ20170818091546869), and Huawei Innovation Research Program (YBN2017125201).

References

[Bollacker et al.2008] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD*, pages 1247–1250. ACM.

[Bordes et al.2013] Antoine Bordes, Nicolas Usunier, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795.

[Bradbury et al.2016] James Bradbury, Stephen Merity, Caiming Xiong, and Richard Socher. 2016. Quasi-recurrent neural networks.

[Chen et al.2017] Qin Chen, Qinmin Hu, Jimmy Xiangji Huang, Liang He, and Weijie An. 2017. Enhancing recurrent neural networks with positional attention for question answering. In *SIGIR*, pages 993–996. ACM.

[dos Santos et al.2016] Cıcero Nogueira dos Santos, Ming Tan, Bing Xiang, and Bowen Zhou. 2016. Attentive pooling networks. *CoRR, abs/1602.03609*.

[Feng et al.2015] Minwei Feng, Bing Xiang, Michael R. Glass, Lidan Wang, and Bowen Zhou. 2015. Applying deep learning to answer selection: A study and an open task. pages 813–820.

[Han et al.2018] Xu Han, Zhiyuan Liu, and Maosong Sun. 2018. Neural knowledge acquisition via mutual attention between knowledge graph and text. In *AAAI*.

[Li et al.2017] Zheng Li, Yu Zhang, Ying Wei, Yuxiang Wu, and Qiang Yang. 2017. End-to-end adversarial memory network for cross-domain sentiment classification. In *Twenty-Sixth International Joint Conference on Artificial Intelligence*, pages 2237–2243.

[Liu et al.2017] Shulin Liu, Yubo Chen, Kang Liu, and Jun Zhao. 2017. Exploiting argument information to improve event detection via supervised attention mechanisms. In *Meeting of the Association for Computational Linguistics*, pages 1789–1798.

[Min et al.2017] Sewon Min, Minjoon Seo, and Hannaneh Hajishirzi. 2017. Question answering through transfer learning from large fine-grained supervision data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 510–517.

[Mou et al.2016] Lili Mou, Zhao Meng, Rui Yan, Ge Li, Yan Xu, Lu Zhang, and Zhi Jin. 2016. How transferable are neural networks in nlp applications? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 479–489.

[Pan and Yang2010] Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Transactions on Knowledge & Data Engineering*, 22(10):1345–1359.

[Rajpurkar et al.2016] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

[Rao et al.2016] Jinfeng Rao, Hua He, and Jimmy Lin. 2016. Noise-contrastive estimation for answer selection with deep neural networks. In *CIKM*, pages 1913–1916. ACM.

[Severyn and Moschitti2015] Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to rank short text pairs with convolutional deep neural networks. In *SIGIR*, pages 373–382.

[Tan et al.2016] Ming Tan, Cicero Dos Santos, Bing Xiang, and Bowen Zhou. 2016. Improved representation learning for question answer matching. In *Meeting of the Association for Computational Linguistics*, pages 464–473.

[Tay et al.2017] Yi Tay, Minh C Phan, Luu Anh Tuan, and Siu Cheung Hui. 2017. Learning to rank question answer pairs with holographic dual lstm architecture. In *SIGIR*.

[Tay et al.2018] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Cross temporal recurrent networks for ranking question answer pairs. In *AAAI*.

[Wang and Nyberg2015] Di Wang and Eric Nyberg. 2015. A long short-term memory model for answer sentence selection in question answering. In *Meeting of the Association for Computational Linguistics*, pages 707–712.

[Wang et al.2007] Mengqiu Wang, Noah A. Smith, and Teruko Mitamura. 2007. What is the jeopardy model? a quasi-synchronous grammar for qa. In *EMNLP*, pages 22–32.

[Wang et al.2016] Bingning Wang, Kang Liu, and Jun Zhao. 2016. Inner attention based recurrent neural networks for answer selection. In *Meeting of the Association for Computational Linguistics*, pages 1288–1297.

[Wang et al.2017] Chenglong Wang, Feijun Jiang, and Hongxia Yang. 2017. A hybrid framework for text modeling with convolutional rnn. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2061–2069. ACM.

[Weikum et al.2007] Gerhard Weikum, Gerhard Weikum, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *International Conference on World Wide Web*, pages 697–706.

[Wiese et al.2017] Georg Wiese, Dirk Weissenborn, and Mariana Neves. 2017. Neural domain adaptation for biomedical question answering. In *Conference on Computational Natural Language Learning*, pages 281–289.

[Xin et al.2018] Ji Xin, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2018. Improving neural fine-grained entity typing with knowledge attention. In *AAAI*.

[Yang and Mitchell2017] Bishan Yang and Tom Mitchell. 2017. Leveraging knowledge bases in lstms for improving machine reading. In *Meeting of the Association for Computational Linguistics*, volume 1, pages 1436–1446.

[Yu et al.2018] Jianfei Yu, Minghui Qiu, Jing Jiang, Jun Huang, Shuangyong Song, Wei Chu, and Haiqing Chen. 2018. Modelling domain relationships for transfer learning on retrieval-based question answering systems in e-commerce.

[Yuan et al.2018] Chung Yuan, Hung Yi Lee, and James Glass. 2018. Supervised and unsupervised transfer learning for question answering. In *NAACL-HLT*.