2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology

A Novel Hierarchical Convolutional Neural Network for Question Answering over Paragraphs

Suncong Zheng, Hongyun Bao, Jun Zhao, Jie Zhang, Zhenyu Qi, Hongwei Hao
Institute of Automation, Chinese Academy of Sciences, 100190, Beijing, P.R. China
{suncong.zheng, hongyun.bao, jie.zhang, zhenyu.qi, hongwei.hao}@ia.ac.cn, jzhao@nlpr.ia.ac.cn

***Abstract*—The question of classical Factoid Question Answering (FQA) task is always in the form of a single sentence. There also exists another kind of FQA task, whose question is a descriptive paragraph, such as quiz bowl question answering. Recently, some works try to automatically answer paragraph questions by applying machine learning methods. However, these methods neglect the correlation information between sentences in a paragraph and do not take full advantage of answer embedding information. In this paper, we propose a novel Hierarchical Convolutional Neural Network, called HCNN-E, to settle the task by considering ordinal information of sentences in paragraph and the information of answer embeddings. The experimental results on two public datasets demonstrate the effectiveness of proposed method, and the proposed method can achieve approximately $10\% - 20\%$ improvements, when comparing with the baselines.**

***Keywords*—Text Mining, Convolutional Neural Network, Question Answering, Information Retrieval.**

I. Introduction

Factoid Question Answering (FQA) is the task of extracting answers, which are always semantic entities, when given a natural language questions. The most common question of FQA is in the form of a single sentence. For example: "which city is the capital of America?". This kind of FQA is always accompanied by a knowledge base [1], [2] or a list of relevant documents [3], which can provide clues for extracting the answer entity in the knowledge base or documents. There exists another form of FQA, whose question is a descriptive paragraph such as: quiz bowl question answering [4], [5]. Different from classical FQA, quiz bowl question is consist of some sentences that describe the answer and the answer is extracted from a given answer entity set. Existing works on FQA are mainly focus on the questions that are typically a single sentence and few works try to answer the question which is a paragraph. Therefore, the task in this paper is to settle the problem of paragraph question answering: automatically matching descriptive paragraph to its answer entity from a given entity set.

Table I is an example of quiz bowl question that describes the city of "Washington D.C.". It shows that the descriptive content may not contain the entity it discusses. Instead, it contains many related words to describe the target entity from different perspectives. Thus, in order to solve this problem, we start with two aspects: one is to capture the topic the given paragraph discusses and then to enhance the semantic relationship between entity (answer) and the paragraph.

TABLE I. An example of quiz bowl question about the "Washington D.C.".

> **Description:** *It is a city named in honor of a great man. The signing of the Residence Act on July 16, 1790 approved the creation of a capital district located along Potomac River on the country's East Coast. The U.S. Constitution provided for a federal district under the exclusive jurisdiction of the Congress and the District is therefore not a part of any U.S. state. The states of Maryland and Virginia each donated land to form the federal district...*
>
> **Answer:** *"Washington D.C."*

Classical methods for capturing the topic of a paragraph are mainly based on topic models such as LDA [6] and PLSA [7]. Topic models implicitly capture the word co-occurrence patterns in document-level, thus suffer from the severe data sparsity in short texts and lost the information of word order. Fortunately, in recent years, many neural network methods can cover the above shortage based on word embedding [8], [9]. However, when processing the paragraph, they always average (sum) the sentences' representations in a paragraph [8], [4] or treat the paragraph as a long sentence [9], [10], which do not conform to the objective facts. In fact, a paragraph is composed of sentences in logical order, and the ordinal information of sentences in a paragraph is very important for paragraph semantic representation. For example, if we change the order of descriptive sentences in Table I, the descriptive content of "Washington D.C." seems to discuss the *"Residence Act"* other than the city, showed as follows:

> *"The signing of the Residence Act ... It is a city named in honor of...".*

Capturing the topic of descriptive paragraph is not enough to find the right answer. Thus we need to further enhance semantic correlation between entity (answer) and the paragraph. Word embeddings have been shown to preserve the semantic relationship between words [11], [12]. And as Table I shows that the paragraph may not contain the entity "*Washington D.C.*" directly, it contains many related phrases such as "*Residence Act*","*U.S.*" and "*city*" etc. Hence, entity embeddings are also useful information to enhance the semantic representation of descriptive paragraph when matching the entity answer to its descriptive text.

Convolutional neural networks (CNN) [13], [10], [14] have been great successful in sentence embedding composition. It is able to preserve ordinal information between words and extract the keywords information in a sentence. Building from above insights, we propose a novel Hierarchical Convolutional

978-1-4673-9618-9/15 $31.00 © 2015 IEEE
DOI 10.1109/WI-IAT.2015.20

Neural Network, called HCNN-E, which provides a unified way to consider the ordinal information and entity embedding to tackle the problem of answering quiz bowl question automatically. Firstly, HCNN-E combines word embeddings into sentence embeddings based on CNN; Secondly, we further combine the sentence embeddings into paragraph embedding by considering the ordinal information of sentences, which is better for understanding the topic of paragraph. At last, we adopt a pair-wise manner to enhance the semantic relationship between entity (answer) embeddings and the paragraph embeddings.

Summary of main contributions: 1). We proposed a novel Hierarchical Convolutional Neural Network (HCNN-E) to extract the entity answer, when given a descriptive paragraph question and an entities set. 2). Compared with classical CNNs, HCNN-E is able to effectively represent the semantic meaning of paragraph-level by considering the ordinal information of sentences and incorporating the information of entity embeddings. 3). The experimental results on two public datasets demonstrate the effectiveness of proposed methods.

II. Related Work

In this paper, the problem is related to the works of quiz bowl question-answering and entity search. And the method we propose is related to the neural-network-based models on sentence/paragraph semantic composition.

Quiz bowl question answering Quiz bowl question answering is a popular quiz game played by students throughout the United States, Canada, and the United Kingdom. The input of quiz bowl question is a textual paragraph and the goal is to identify the answer this paragraph discuss. In recent years, some works try to answer quiz bowl questions by applying machine learning methods. For example: Boyd [5] propose a Naive-bayes model to identify the answers based on manually defined string matching rules and bag of words representations. Iyyer [4] applies Dependency Tree Recursive Neural Networks (DT-RNN) to compose the semantic representation of sentence based on word embeddings, then maps sentence representation to its according answer.

Entity search Entity search is the most common search type on the web, and it has been the main task of TREC Entity track from 2009 to 2011 [15]. The task of entity search is to retrieve relevant entities from a semantic data set about entities, when given a structured query [16] or a keyword query [17]. Different from entity search, the given text, in this paper, is a descriptive paragraph not structured or keyword marked query. Hence, we still need to understand the topic of the given paragraph and find out the clues of keywords.

Sentence/paragraph semantic composition The semantic composition of sentence and paragraph is the core task, when linking paragraph text to an entity. Classical methods for representing the sematic of sentence or short text are based on bag-of-words or topic models [6], [7], which suffer from the severe data sparsity and lost the information of word order. Recently, deep learning methods have been successfully applied to many NLP tasks and many works try to learn sentence/paragraph semantic representations based on word embeddings. The most common neural-network-based models are Recurrent/Recursive Neural Networks (RNN) and Convolutional Neural Network (CNN).

Recurrent Neural Networks deal successfully with time-series data and they were also applied on NLP [18], [19] by modeling a sentence as tokens processed sequentially. These models generally consider no linguistic structure aside from word order. Recursive neural models, by contrast, are structured by syntactic tree structure. When compositing sentence's representation based on word enbedddings, Recursive neural models should firstly determine the tree structure. The works of [4], [20], [21] introduce Dependency Tree RNN models for sentences' semantic composition. The tree structure is determined by semantic parser. The works of [22], [23] apply a kind of auto-encode structure which could be good for phase composition.

When compositing sentence's representation, the RNN models may come across the problem of bias [24]. To tackle the bias problem, the Convolutional Neural Network (CNN) can fairly determine discriminative phrases in a text and may better capture the semantic of texts compared to recursive or recurrent neural networks. Hence, the method we proposed in this work is based on the CNN model, which have been applied successfully in image [25], speech [26]. In recent years, CNN has also shown the effectiveness for sentence representation [13], [10], [14]. However, the works on modeling of paragraph based on CNN are rare. When processing paragraph text, existing works always loss the ordinal information of sentences in paragraph. Therefore, we propose HCNN-E to model paragraph embeddings by considering ordinal information of sentences and we also utilize the information of entity embeddings to enhance the sematic representation of paragraph.

III. The Model of HCNN-E

The task of matching paragraph question to answer entity, in this paper, can be described as: given a descriptive paragraph $D \in \mathbb{D}$ and a set of answer entities E, then to compute:

$$e_D = \operatorname*{argmax}_{e \in E} P(e|D, \Theta). \tag{1}$$

Where Θ is the parameters of model to be learned, $\mathbb{D}$ is the dataset of paragraphs, and e_D is the entity corresponds to the descriptive paragraph D.

We propose HCNN-E to achieve the goal based on two observations. First, a paragraph is composed of sentences in logical order, and the ordinal information of sentences in paragraph is very important for paragraph semantic representation. Second, word embeddings have been shown to preserve the semantic relationship between words [11], [12]. Hence, entity embeddings are very useful when linking entity and its descriptive text. In the following sections, we firstly present the architecture of our model shown in Figure 1 and detail each components of the model. After that, we introduce objective function and parameters inference.

A. The Architecture of HCNN-E

The architecture of HCNN-E is illustrated in Figure 1. The model contains: (1) word embedding layer which initialized by running word2vec [27]; (2) sentence embedding layer that is composed of word embeddings by applying convolutional neural networks; (3) paragraph embedding layer which is composed of sentence embeddings and reinforced by entity embeddings; and (4) entity embedding layer which can enhance the semantic representation of paragraph. In what follows, we describe these components in detail.

1) Word embedding layer: In this paper, we initialize the word embeddings with word2vec [27] and the dataset for initializing word embeddings also contains the training text in our experiment. Out-of-vocabulary words from the test set are initialized randomly. The model we use to train word embedding is the hierarchical skipgram model setting with a window size of five words, and the dimension of word embedding is d. Hence, we define $X \in \mathbb{R}^{|V| \times d}$ as the set of word embeddings and the size of vocabulary is $|V|$.

2) Modeling sentence embedding: Let $x_i \in \mathbb{R}^d$ be the d-dimensional word vector corresponding to the *i*-th word in the sentence. Hence, a sentence of length n is represented by a matrix: $s = (x_1; x_2; ...; x_n)$. Accordingly, $s_{i:i+h_s-1} = (x_i; x_{i+1}; ...; x_{i+h_s-1})$ refer to h_s continuous words, which begins with the *i*-th word in sentence *s*. Given a filter $W^{(1)} \in \mathbb{R}^{h_s}$, we produce feature representation of h_s continuous words in sentence s by:

$$s^{(i)} = f(W^{(1)} \cdot s_{i:i+h_s-1} + b^{(1)}), \tag{2}$$

where $b^{(1)} \in \mathbb{R}^d$ is a bias term, f is an activation function and $s^{(i)} \in \mathbb{R}^d$ refers to sentence's features produced by words: $\{x_i, x_{i+1} ... x_{i+h_s-1}\}$. Hence, the all convolutional features of sentence are represented as: $s = (s^{(1)}, s^{(2)}, ... s^{(n-h_s+1)})$. Then, we apply average pooling operation [28], [29] over the convolutional features and take the mean value as the latent features of the sentence. Namely:

$$\bar{s} = \frac{1}{n - h_s + 1} \sum_{i=1}^{n-h_s+1} s^{(i)}. \tag{3}$$

A filter $W_i^{(1)}$ generate a d-dimensional feature vector $\bar{s}_i$. If we use k filters, the feature vector of sentence is represent as $s = (\bar{s}_1, \bar{s}_2, ... \bar{s}_k)$. Therefore, the dimension of sentence embedding is $d \cdot k$, where k is the number of filters we apply.

There are two "channels" after sentence embedding layer, one is to the labels of entities, the other is to produce paragraph embedding. The channel of entity labels is to reinforce the semantic representation of sentence. It is a soft-max layer [30] with dropout [31], which is defined as:

$$ys = W^{(2)} \cdot (s \circ r) + b^{(2)}, \tag{4}$$

$$ps_i = \frac{exp(ys_i)}{\sum\limits_{j=1}^{m} exp(ys_j)}, \tag{5}$$

where $W^{(2)} \in \mathbb{R}^{m \times d \cdot k}$ is the weights between sentence embedding layer and the layer of entity labels, m is the total number of entities, symbol $\circ$ denotes the element-wise multiplication operator and $r \in \mathbb{R}^{d \cdot k}$ is a binary mask vector drawn from Bernoulli with probability ρ. Dropout guards against overfitting, which makes the model more robust. In Formula 5, ps_i means the probability that the sentence describes entity *i*. The other channel of paragraph will be detailed in section III-A3.

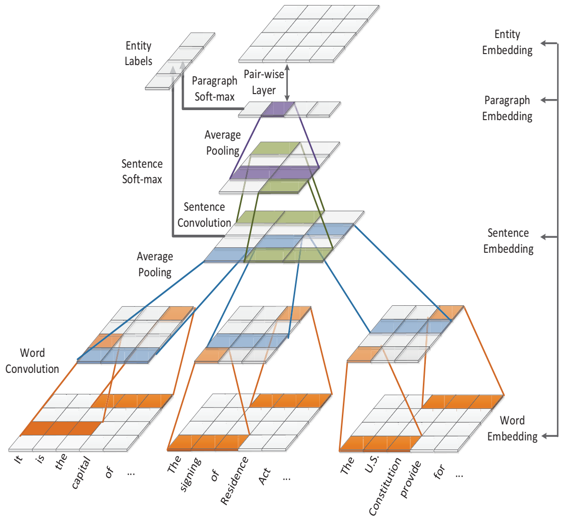

Fig. 1. The Architecture of HCNN-E

3) Modeling paragraph embedding and entity embedding: After the sentence-level embedding is produced by the average pooling operation, we further combine the sentence embeddings into paragraph embedding by considering the ordinal information of sentences and the information of entity embeddings.

In order to consider the ordinal information of sentences, we still adopt the operation of sliding window based feature extraction. It is designed to capture the sentence-n-gram contextual features. Let $s_i \in \mathbb{R}^{d \cdot k}$ be the $d \cdot k$ dimensional sentence vector corresponding to the *i*-th sentence in the paragraph. Likewise, a paragraph text of length *l* is represented by a matrix $t = (s_1; s_2; ...; s_l)$ and $t_{i:i+h_t-1} = (s_i; s_{i+1}; ...; s_{i+h_t-1})$ refer to h_t continuous sentences in paragraph *t*. We apply the operation of convolution to produce a feature representation of h_t continuous sentences in paragraph t as follows:

$$t^{(i)} = f(W^{(3)} \cdot t_{i:i+h_t-1} + b^{(3)}), \tag{6}$$

where $W^{(3)} \in \mathbb{R}^{h_t}$ is the filter applied in sentences embedding. Hence, $t^{(i)}$ capture the contextual information from sentences i to $i + h_t - 1$. In order to fix the length of paragraph embedding and composite sliding information, we also apply average pooling operation over $t^{(i)}$, which defined as

$$\bar{t} = \frac{1}{l - h_t + 1} \sum_{i=1}^{l-h_t+1} t^{(i)}. \tag{7}$$

For the sake of further enhancing semantic representation of paragraph, we add an pair-wise layer after layer $\bar{t}$ by taking advantage of entity embeddings. The parewise layer is defined as:

$$z = W^{(4)} \cdot \bar{t}, \tag{8}$$

$$sim(z, E) = \{z \cdot e_1, z \cdot e_2, ... z \cdot e_m\}, \tag{9}$$

where $W^{(4)} \in \mathbb{R}^{d \times d \cdot k}$ is the weights between layer $\bar{t}$ and entity embeddings layer. $E = \{e_1, e_2, ... e_m\}$ is the entity embedding set, which are also initialized by using word2vec, the same as word embeddings. The operation of $z \cdot e$ refers the similarity between paragraph embedding z and entity embedding *e*. We expect to learn a high similarity between paragraph embedding and its related entity, and a low similarity between irrelevant paragraph and entity. Hence, the objective function for parewise layer is contrastive max-margin which will be detailed in section III-B.

The pair-wise layer is to enhance paragraph and entity embeddings. The output layer of paragraph is also soft-max, the same as sentence level, which is represented as:

$$y = W^{(5)} \cdot (z \circ r) + b^{(5)}, \tag{10}$$

$$p_i = \frac{exp(y_i)}{\sum_{j=1}^{m} exp(y_j)} \ . \tag{11}$$

B. Objective Function and Training

Different from classical CNNs, the objective function of HCNN-E consists of two level: objective function of sentence-level named as L_s and paragraph-level objective function L_p. Meanwhile, the objective function of paragraph-level contains two parts: one is the contrastive max-margin objective function L_{p1} on parewise layer, and the other is cross entropy objective function L_{p2} on soft-max layer of paragraph.

Sentence-level objective function L_s. The goal in this paper is focus on paragraph semantic composition, which is based on sentence embeddings. Hence, we also need to reinforce the semantic embeddings of sentences. The sentences in a paragraph always share the same topic of the paragraph. Thus we treat paragraph's label as the sentences'. We add a soft-max layer after layer of sentence embedding shown in Formula 4 and 5. The objective function of sentence-level is to minimise the cross entropy errors of the predicted entity label distribution and the actual distribution, which defined as:

$$L_s = -\sum_{i=1}^{|\mathbb{D}_s|} -log(P(ps_r^{(i)}|s^{(i)}, W^{(2)}, b^2)), \tag{12}$$

where $|\mathbb{D}_s|$ is the sentences number in dataset set $\mathbb{D}$ and $ps_r^{(i)}$ is the correct class of sentence *i*.

Objective function of paragraph-level L_p. After paragraph embedding layer, there also exists two "channels", one is parewise layer, the other is soft-max layer of paragraph.

In order to enhance semantic representation of paragraph and entities, we use a contrastive max-margin objective function to learn a high similarity between paragraph embedding and its related entity, and a low similarity between irrelevant paragraph and entity. The objective function in parewise layer is:

$$\begin{aligned} L_{p1} = \sum_{i=1}^{|\mathbb{D}|} \sum_{e_j \in E(e_j \neq e_r)} & max(0, \\ & 1 - sim(z^{(i)}, e_r^{(i)}) + sim(z^{(i)}, e_j^{(i)})), \end{aligned} \tag{13}$$

where e_r is the relevant entity of given paragraph *z*.

The output of the network is the probability distribution of paragraph over entity set, which is a soft-max layer. And the objective function of the output layer is also the cross entropy errors:

$$L_{p2} = -\sum_{i=1}^{|\mathbb{D}|} -log(P(p_r^{(i)}|z^{(i)}, W^{(4)}, b^4)), \tag{14}$$

where $p_r^{(i)}$ is the correct class of paragraph *i*.

Therefore, objective function of paragraph level can be represented as:

$$L_p = \alpha \cdot L_{p1} + (1 - \alpha) \cdot L_{p2}, \tag{15}$$

where α is a weighting factor to balance the influence of L_{p1} and L_{p2}, and it is also a parameter of the model which need to be trained.

Accordingly, the final objective function of the model is:

$$L = L_s + L_p. \tag{16}$$

Training The parameters of the model to be trained are concluded as: $\Theta = \{X, W_i^{(1)}, b_i^{(1)}, W^{(2)}, b^{(2)}, \alpha, W^{(3)}, b^{(3)}, W^{(4)}, b^{(5)}, W^{(5)}, E\}$. The training target of the network is to minimise the objective function L with respect to parameters Θ. We use stochastic gradient descent [32] to optimize the training target. Gradients are backpropagated only through the unmasked units in these layers with dropout. Besides, the learned weight parameters in these dropout layer need to be scaled by ρ such that $W^{(2)} = \rho \cdot W^{(2)}$ and $W^{(5)} = \rho \cdot W^{(5)}$. The hyper-parameter ρ is dropout rate which has been described in section III-A2.

C. Inference

At test time, when given a paragraph, we firstly compute the sentence embeddings s based on Formula 2 and 3. Secondly, we apply the operation of Formula 6, 7 and 8 to get paragraph embedding z based on the sentence embeddings. At last, we compute the paragraph's probability distribution over entity set based on Formula 17 and 11, and choose the entity with maximum probability as the predict result. Similar with Formula 10, the Formula 17 is defined as follows without dropout:

$$y = W^{(5)} \cdot z + b^{(5)}. \tag{17}$$

IV. Experimental Setup

In this paper, the task is to identify the entity when given a descriptive paragraph and an entities set. To demonstrate the effectiveness of our proposed method, we compare the performance of HCNN-E against multiple strong baselines on two public datasets.

A. Datasets

All methods are performed on two public datasets,[1] created by iyyer [4] based on quiz bowl resources. Specially, the datasets, used in this paper, are the released ones, which are little different from the datasets reported in [4]. One of the datasets is about the historical knowledge, the other is about the knowledge of literature. The descriptive paragraph of the datasets consist of four to eight sentences and each paragraph

[1] http://cs.umd.edu/ miyyer/qblearn/

corresponds to only one entity. Each sentence in the description data is guaranteed to contain clues that identify its entity. We do not remove any stop words or symbols in the text and the sizes of train/test/dev are also the same as it was released. The statistics of these datasets are shown in Table II.

TABLE II. Statistics of the datasets provided by Iyyer. Train/Test/Dev: the size of train/dev/test set. $|E|$: the size of entity set.

Dataset	Type	Train/Test/Dev	$\lvert E \rvert$
History	sentence paragraph	6770/897/472 1422/192/90	409
Literature	sentence paragraph	8502/1415/650 1794/310/143	445

B. Baselines

The baselines are widely used on these two datasets. They can be roughly divided into two categories: models based on bag of words, methods based on word embeddings.

BoW-LR: The baseline mainly use logistic regression (LR) algorithms with unigram and bigrams as features. This simple discriminative model is an improvement over the generative quiz bowl answering model of [5].

BoW-DT-LR: Different from the baseline of BoW-LR, BoW-DT-LR considers not only the features of unigram and bigrams but also dependency relation. Besides, the method is also treated as an effective baseline in [4].

BTW-LR: Biterm topic model (BTM) [33] is a novel probabilistic topic model for short texts. The quiz bowl question is a short text, so BTM is more suitable than LDA for the problem. We use BTM to extract question features, then apply logistic regression (LR) algorithms to find the right answer.

Average-Embedding-LR: This baseline uses the weighted average of the word embeddings and subsequently applies a logistic regression (LR) algorithms. The weight of each word is its tf-idf value. [24][34] also used this strategy as an baseline for document classification.

DT-RNN: Iyyer et al. [4] use a Dependency-Tree RNN model to compose sentence semantic representation based on word embeddings, then average the representations of sentences as the paragraph's representation.

CNN-1: We also select a convolutional neural network [13] for comparison. CNN-1 improves upon the state of the art on 4 out of 7 tasks, which include sentiment analysis and question classification.

C. Metric

The task in this paper is to identify the entity when given a descriptive paragraph, and the paragraph corresponds to only one unique entity. Hence, we use accuracy as our metric to evaluate the performances of the methods. We run 50 times for each model and get the average accuracy as the final results. Besides, we also compute the standard deviation of the results to reflect the stability of models.

TABLE III. Hyper parameters of our models. ρ is dropout rate, h_s is the size of words' window, h_t is the size of sentences' window, d is the size of word embedding and k is the filter number used in word convolution layer.

Dataset	ρ	h_s	h_t	d	k
History	0.5	3	6	100	1
Literature	0.5	3	8	100	1

D. Hyper Parameters Setting

The hyper parameters used in this experiments are summarized in Table III.

Both history and literature share the same hyper parameters except h_t. Because the datasets have the property that sentences in a paragraph are connected very tightly and the experimental results show that when the size of h_t equals the length of paragraph, the predict result is best. Different paragraphs have different length, the best windows size is always the maximum length of paragraphs in the dataset. Therefore, the value of h_t can exceed the average length of paragraphs in datasets and the length of paragraph are different in these two datasets, so it is not suitable to share same hyper parameter of h_t for these two datasets.

V. Results and Discussions

A. Comparison with Baselines

In this section, we compare HCNN-E with baseline methods, which are widely applied on these two datasets. Table IV summarizes the results of various models over the two datasets.

TABLE IV. The accuracy of various models on two public datasets. We run 50 times for each model and get the average accuracy as the final results. The top part is the baselines of bag-of-words models. The middle part is the baselines of embedding-based models. The bottom part is the result of our model.

Methods	History(%)	Literature(%)
BoW-LR	65.10 ± 0.01	61.17 ± 0.05
BoW-DT-LR	68.23 ± 0.03	63.43 ± 0.07
BTW-LR	59.22 ± 0.03	57.30 ± 0.02
Average-Embedding-LR	28.64 ± 0.01	46.92 ± 0.02
DT-RNN	59.38 ± 1.01	54.69 ± 1.04
CNN-1	77.01 ± 3.92	74.50 ± 10.3
HCNN-E	$\mathbf{89.41 \pm 1.05}$	$\mathbf{91.26 \pm 0.50}$

Comparison with BoW-based methods When comparing with BoW-based methods, HCNN-E achieves a $+20\%$ improvement over these BoW-based models. Since the descriptive paragraphs may not contain the entities they mention and it requires inference and consideration of background knowledge to analyze the paragraphs. Besides, BoW-based models surfer from the data sparsity problem. Therefore, HCNN-E can perform better than BoWs.

Comparison with embedding based approaches The Average-Embedding-LR method is heavily dependent on the quality of word embeddings and it also surfers from the data sparsity problem, so this simple manner of averaging embeddings achieves pool results.

We also compare HCNN-E with well-used deep learning models based on embeddings: RNN and CNN. The experimental results also show that the proposed method outperforms these neural network approaches. The DT-RNN [4] applies the parser tool to construct a tree-structure of sentence which can affect the performance. Besides, it divides the task into two independent steps which may also hurt the final result. The CNN-1 [13] model just treats the task as classification problem, and it neglects the information of entity embeddings. Expecially, both methods do not consider the ordinal information of sentences when composite paragraph embedding.

B. Analysis

In order to analyze the impact factors of entity embeddings and sentences' ordinal information, which are considered in HCNN-E. We compare HCNN-E with two variations: CNN-2 and HCNN and the performance of the model variations are shown in Table V.

CNN-2 is the convolutional neural network, used in this paper, to produce sentence's embedding. Different from CNN-1 [13], CNN-2 applies one-dimensional convolution and average pooling. When compared with HCNN-E, CNN-2 does not consider the information of sentence order and entity embeddings.

HCNN is a hierarchical convolutional neural network to produce paragraph's embedding, which considers the information of sentences' order. But it does not take advantage of entities' embedding information, when comparing with HCNN-E.

TABLE V. THE ACCURACY OF MODEL VARIATIONS ON TWO PUBLIC DATASETS.

Methods	History(%)	Literature(%)
CNN-2	78.17 ± 3.92	75.61 ± 5.22
HCNN	83.35 ± 0.81	89.41 ± 0.49
HCNN-E	$\mathbf{89.41 \pm 1.05}$	$\mathbf{91.26 \pm 0.50}$

The effect of sentences' ordinal information When comparing CNN-2 with HCNN, we observe that the model considering sentences' ordinal information achieves much improvement against this without considering the information. When testing on the dataset of literature, HCNN is about $+10\%$ higher than CNN-2. The improvement is also large when testing on the dataset of history by considering ordinal information. The results prove the effectiveness of sentences' ordinal information.

The effect of entity embeddings In order to illustrate the effect of entity embeddings, we compare HCNN-E with HCNN without considering entity embeddings. The results show that HCNN-E can achieve a $+6\%$ improvement compared with HCNN on the dataset of history and a $+2\%$ on the dataset of literature. The results indicate the importance of entity embeddings.

Besides, the standard deviation of HCNN and HCNN-E are smaller than CNN-2, which just averages the representations of sentences in the paragraph. It also shows the stabilize of hierarchical convolutional neural network when composing paragraph's embedding.

C. Parameter Sensitivity

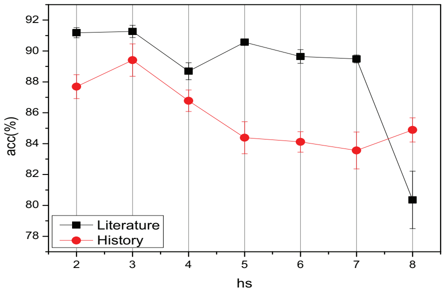

Fig. 2. The effect of words window size. We change the words window size h_s of HCNN-E from 2 to 8, and the curves of accuracy almost share the same shape on these two datasets.

Convolution based models use a fixed window of words as contextual information. The performance of these models are affected by the size of words' window. In order to analyze the influence of contextual information, we consider words' window sizes h_s from 2 to 8. The results on two datasets are shown in Figure 2. The accuracy firstly improves with the increasing of the window size, then achieves the peak when size is 3, after that it begins to decrease. There are many phases in the paragraphes, and the average size of phrases is between 2 and 3. Hence, when the window size is 2 or 3, the model can achieve a higher accuracy. When the windows size is larger than phrase's size, it can break the structure of phrase, which is bad for the final results. Therefore, the curve of accuracy become to come down after reaching the peak value.

VI. Conclusion

In this paper, we focus on the task of quiz bowl question answering, which can be seen as linking textual description to answer entity automatically. In order to settle the problem, we proposed a model, called HCNN-E, which takes ordinal information of sentences in paragraph and information of entity embeddings into consideration. We compare the model with multiple approaches on two public datasets, and the experimental results show the stability and effectiveness of our propose method. We also analyze the influence of entity embeddings and sentences' ordinal information. The results show that these factors considered in the model are effective for the task.

In this paper, the datasets we used are released by Iyyer [4], which are little different from the datasets reported in [4]. Although it can meet the requirement of quiz bowl question answering, it has the shortage of scale-limited. Besides, exist methods for quiz bowl question answering can not well solve the problem that the answer entity is not appear in train dataset. In the future, we plan to experiment in larger datasets and we also expect to design a algorithm to deal with the problem of unknown answer entity.

Acknowledgments

This work is supported by the National High Technology Research and Development Program of China (863 Program) (Grant No. 2015AA015402) and National Natural Science Foundation (Grant No. 71402178). We also thank Iyyer for providing the experimental datasets.

References

[1] X. Yao and B. Van Durme, "Information extraction over structured data: Question answering with freebase," in *Proceedings of ACL*, 2014.

[2] J. Bao, N. Duan, M. Zhou, and T. Zhao, "Knowledge-based question answering as machine translation," *Cell*, vol. 2, p. 6, 2014.

[3] S. K. Dwivedi and V. Singh, "Research and reviews in question answering system," *Procedia Technology*, vol. 10, pp. 417–424, 2013.

[4] M. Iyyer, J. Boyd-Graber, L. Claudino, R. Socher, and H. D. III, "A neural network for factoid question answering over paragraphs," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 633–644.

[5] J. Boyd-Graber, B. Satinoff, H. He, and H. Daumé III, "Besting the quiz master: crowdsourcing incremental classification games," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, 2012, pp. 1290–1301.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[7] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999, pp. 50–57.

[8] K. M. Hermann and P. Blunsom, "Multilingual models for compositional distributed semantics," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 58–68. [Online]. Available: http://www.aclweb.org/anthology/P/P14/P14-1006

[9] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," *arXiv preprint arXiv:1405.4053*, 2014.

[10] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 655–665. [Online]. Available: http://www.aclweb.org/anthology/P/P14/P14-1062

[11] R. Fu, J. Guo, B. Qin, W. Che, H. Wang, and T. Liu, "Learning semantic hierarchies via word embeddings," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 1199–1209. [Online]. Available: http://www.aclweb.org/anthology/P/P14/P14-1113

[12] T. Mikolov, W.-t. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations." in *HLT-NAACL*. Citeseer, 2013, pp. 746–751.

[13] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[14] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *The Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.

[15] K. Balog, P. Serdyukov, and A. P. d. Vries, "Overview of the trec 2010 entity track," DTIC Document, Tech. Rep., 2010.

[16] Y. Fang, L. Si, N. Somasundaram, S. Al-ansari, and Z. Y. Y. Xian, "Purdue at trec 2010 entity track: a probabilistic framework for matching types between candidate and target entities," in *In Proceedings of the Nineteenth Text REtrieval Conference (TREC 2010*. Citeseer, 2010.

[17] X. Liu and H. Fang, "A study of entity search in semantic search workshop," in *Proc. of the 3rd Intl. Semantic Search Workshop*, 2010.

[18] T. Mikolov, S. Kombrink, L. Burget, J. H. Cernocky, and S. Khudanpur, "Extensions of recurrent neural network language model," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 5528–5531.

[19] N. Kalchbrenner and P. Blunsom, "Recurrent continuous translation models." in *EMNLP*, 2013, pp. 1700–1709.

[20] R. Socher, Q. Le, C. Manning, and A. Ng, "Grounded compositional semantics for finding and describing images with sentences," in *NIPS Deep Learning Workshop*, 2013.

[21] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Citeseer, 2013, pp. 1631–1642.

[22] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-supervised recursive autoencoders for predicting sentiment distributions," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2011, pp. 151–161.

[23] K. M. Hermann and P. Blunsom, "The role of syntax in vector space models of compositional semantics." in *ACL (1)*, 2013, pp. 894–904.

[24] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[26] L. Deng, O. Abdel-Hamid, and D. Yu, "A deep convolutional neural network using heterogeneous pooling for trading acoustic invariance with phonetic confusion," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6669–6673.

[27] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[28] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[29] Y.-L. Boureau, J. Ponce, and Y. LeCun, "A theoretical analysis of feature pooling in visual recognition," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 111–118.

[30] K. Duan, S. S. Keerthi, W. Chu, S. K. Shevade, and A. N. Poo, "Multi-category classification by soft-max combination of binary classifiers," in *Multiple Classifier Systems*. Springer, 2003, pp. 125–134.

[31] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout," in *ICASSP*, 2013, pp. 8609–8613.

[32] L. Bottou, "Stochastic gradient learning in neural networks," *Proceedings of Neuro-Nımes*, vol. 91, no. 8, 1991.

[33] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *Proceedings of the 22nd international conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2013, pp. 1445–1456.

[34] A. Klementiev, I. Titov, and B. Bhattarai, "Inducing crosslingual distributed representations of words," 2012.