Interspeech 2018
2-6 September 2018, Hyderabad

Follow-up Question Generation using Pattern-based Seq2seq with a Small Corpus for Interview Coaching

Ming-Hsiang Su[1], *Chung-Hsien Wu*[1], *Kun-Yi Huang*[1], *Qian-Bei Hong*[2], and *Huai-Hung Huang*[1]

[1] Department of Computer Science and Information Engineering,
National Cheng Kung University, Taiwan
[2] PhD Program for Multimedia Systems and Intelligent Computing,
National Cheng Kung University and Academia Sinica, Taiwan

{huntfox.su, chunghsienwu, iamkyh77, qbhong75}@gmail.com, coditol@hotmail.com

Abstract

Interview is a vital part of recruitment process and is especially challenging for the beginners. In an interactive and natural interview, the interviewers would ask follow-up questions or request further elaborations when they are not satisfied with the interviewee's initial response. In this study, as only a small interview corpus is available, a pattern-based sequence to sequence (Seq2seq) model is adopted for follow-up question generation. First, word clustering is employed to automatically transform the question/answer sentences into sentence patterns, in which each sentence pattern is composed of word classes, to decrease the complexity of the sentence structures. Next, the convolutional neural tensor network (CNTN) is used to select a target sentence in an interviewee's answer turn for follow-up question generation. In order to generate the follow-up question pattern, the selected target sentence pattern is fed to a Seq2seq model to obtain the corresponding follow-up question pattern. Then the word class positions in the generated follow-up question sentence pattern is filled in with the words using a word class table obtained from the training corpus. Finally, the *n*-gram language model is used to rank the candidate follow-up questions and choose the most suitable one as the response to the interviewee. This study collected 3390 follow-up question and answer sentence pairs for training and evaluation. Five-fold cross validation was employed and the experimental results show that the proposed method outperformed the traditional word-based method, and achieved a more favorable performance based on a statistical significance test.

Index Terms: Interview system, follow-up question, convolutional neural tensor network, sequence to sequence

1. Introduction

In recent years, spoken dialogue systems (SDS) have been popular with the people who need some extra help, and have been extensively developed in a variety of areas, such as ticket booking, hotel reservations, interview coaching [1-4], etc. Among these applications, the interview coaching system attempts to simulate an interviewer to provide the mock interview to the users. In order to increase opportunities for people to practice interview skills, such as admission interview and job interview, many researchers engaged in the design and development of interview training systems [5-7]. Regarding the coaching systems with a fixed scenario, MACH [5] analyzed a user's nonverbal behaviors, such as facial expressions, voice going up or down, head movements, smiling, and eye contact. At the end of a dialog flow, the system provided a summary feedback, indicating which nonverbal behaviors need to be improved. LISSA [6] helped people practice their conversational skills by having short conversations with a human like virtual agent and receiving real-time feedback on their nonverbal behavior. TARDIS [7] built a scenario-based serious-game simulation platform and aimed to improve the social skills of young people, with a focus on emotional computing.

Although all of these coaching systems were used to improve people's conversational skill, few of them considered the semantic representation of the interviewee's responses and question generation based on the responses, and most of the interview process was pre-defined. It is important that a conversational interview coaching system should consider the semantic representation of the interviewee's responses, and automatically generate ordinary and follow-up questions for the interview to proceed smoothly. Preferably, if the system can understand interviewee's response and ask the follow-up questions accordingly, interviewee can practice their interview skills more realistically and effectively. Therefore, this study focuses on how to encode the interviewee's response into a semantic representation and how to generate the follow-up questions based on the interviewee's response, as shown in Figure 1. The main contributions of this study are summarized as follows. First, this study uses CNTN-based sentence selection model to select the most appropriate sentence of interviewee's response as the target sentence. Second, based on the selected target sentence, this study generates appropriate follow-up questions to make the coaching system more lively and realistically based on a small interview corpus. As there are only a small interview corpus available, this study adopts sentence patterns to represent the sentences with similar sentence structures. Third, this study uses seq2seq-based model to generate a follow-up question pattern, and then convert the follow-up question pattern into the final follow-up question based on word filling. Finally, the *n*-gram language model is used to rank the generated candidate follow-up questions to choose the most suitable one as the response to the interviewee.

2. MHMC-IV Database Collection

In order to construct an interview coaching system, we invited forty participants to collect the interview database. The question types and topics for the interviews were related to the entrance admission of graduate students. During database collection,

10.21437/Interspeech.2018-1007

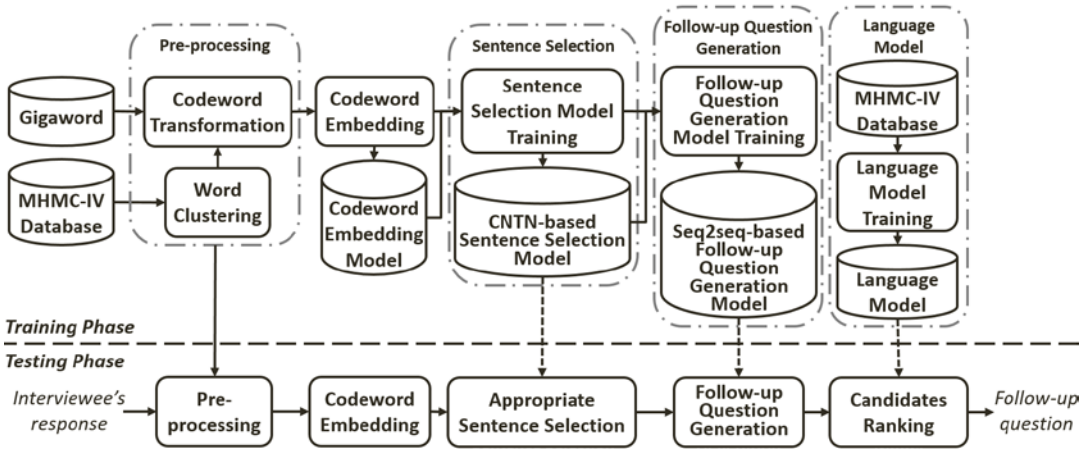

Figure 1: *The block diagram of the proposed follow-up question generation system.*

every two participants, one serving as the interviewer and the other as the interviewee, had the freedom to complete the interview without using predesigned questions. The interviewee was assigned a random identity background to simulate the real situation. In the collected interview corpus, there were two different questions, namely, ordinary questions and follow-up questions. Ordinary questions were the questions not related to the previous question or interviewee's previous response, while the follow-up questions were asked based on the interviewee's previous response to elaborate the initial response. Finally, 260 dialogs with 1754 ordinary questions and 3390 follow-up questions were collected to form the MHMC-IV database, as shown in Table 1. For data analysis, this study divides the follow-up questions asked by the interviewer into 16 types [8], and the follow-up question distribution of the 16 question types is shown in Figure 2.

3. System Framework

3.1. Sentence Pattern Generation and Word Embedding

The traditional way to generate interview questions is to use a set of well-defined sample questions. Because the interviewee's response is volatile for the same question, it is difficult to define all the sample follow-up questions in the database collected for the interview coaching system. Accordingly, this study adopts the word clustering method for automatic sentence pattern generation. For word clustering, we first use Jieba (the Chinese word segmentation tool) [9] for word segmentation of all the sentences in the MHMC-IV database. A total of 7,132 words were obtained from the MHMC-IV database.

For word clustering, the relations between contextual words are considered for word similarity estimation. We calculate the word similarity score as follows.

$$S(w_1, w_2) = \frac{F_p \times \alpha + F_c \times \beta + F_s \times \gamma}{\alpha + \beta + \gamma} \quad (1)$$

where w_1 and w_2 are the words in MHMC-IV database; α, β, and γ are the weights for similarity combination from different similarity measures, and the value of the weight lies between 0 to 1; F_p, F_c and F_s are the preceding word score, center word score, and succeeding word score, respectively. The equation used to calculate the three scores F_p, F_c, and F_s is depicted as follows. The three scores are estimated using the same equation.

Table 1: *Details of the MHMC-IV database.*

	Total
Number of turns	5144
Average number of turns	19.7
Average number of ordinary/follow-up turns	6.7/13.0
Average number of sentences in each answer	3.84
Interview time (minutes) per interview	20

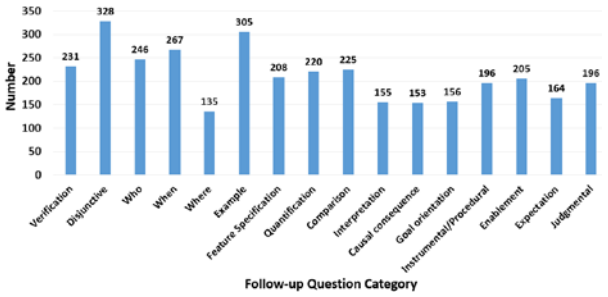

Figure 2: *Follow-up question distribution.*

$$F_p = \frac{\sum_{i=1}^{I}\sum_{j=1}^{J} Sim(p_{1,i}, p_{2,j})}{I \times J} \quad (2)$$

where $p_{1,i}$ is the i^{th} preceding word with respect to w_1; $p_{2,j}$ is the j^{th} preceding word with respect to w_2; I is the number of preceding words of w_1; J is the number of preceding words of w_2. The word similarity between $p_{1,i}$ and $p_{2,j}$ represented as $Sim(p_{1,i}, p_{2,j})$ is calculated by Eq. 3.

$$Sim(p_{1,i}, p_{2,j}) = \frac{d}{d + L} \quad (3)$$

where d is the depth from the root node to parent node of $p_{1,i}$ and $p_{2,j}$ in *E*-HowNet [10], and L is the shortest path between $p_{1,i}$ and $p_{2,j}$ in *E*-HowNet. Thus, we can construct a word similarity matrix. Finally, we use the affinity propagation-clustering algorithm [11] to cluster the words automatically

without predefining the number of clusters. At last, the values of α, β and γ are empirically chosen as 0.05, 0.9, 0.05, respectively.

Based on word clustering, we then generate a word class table. It contains 1140 word classes and 7,132 words. After we have the word class table, we transfer all of sentences in MHMC-IV database into sentence patterns by looking up the word class table. Therefore, we can generate sentence patterns without predefining sentence templates or rules.

Then we combine Chinese Gigaword database [12] and MHMC-IV database to train a GloVe-based [13] word embedding model. The Chinese Gigaword contains about 1.12 billion Chinese characters, including 735 million characters from Taiwan's Central News Agency, and 380 million characters from China's Xinhua News Agency [12]. GloVe is an unsupervised learning algorithm for obtaining vector representations of the words. The GloVe model is trained on the non-zero entries of a global word-word co-occurrence matrix, which tabulates how frequently words co-occur with one another in a given corpus [13]. Finally, each word class is represented as a 300-dimensional word embedding vector in this study.

3.2. CNTN-based Sentence Selection Model

When the interviewee's response contains many sentences, it is a challenge to find the most appropriate one as the target sentence for follow-up question generation. This study uses the CNTN-based [14] sentence selection model to slove the problem. The CNTN is composed of a convolutional neural network (CNN) [15] and a neural tensor network (NTN) [16], as shown in Figure 3. The CNN is used to encode the sentences of the question and the response, and the NTN is used to learn the relationship between the question and the response sentences. Given a sentence s, we use GloVe algorithm [13] to obtain the word embedding vector $\mathbf{w}_i \in \mathbb{R}^{n_w}$ for each word w in sentence s. Then we take the word vector $\mathbf{w}_i$ to obtain the input matrix $\mathbf{s} \in \mathcal{R}^{n_w \times l_s}$, where l_s denotes the sentence length. Next, a convolutional layer is obtained by convolving a matrix of weights $\mathbf{m} \in \mathcal{R}^{n \times m}$ with the matrix of activations at the layer below, where m is the filter width. Given a value k and a row vector $\mathbf{p} \in \mathcal{R}^p$, we use k-max pooling to select the subsequence $\mathbf{p}^p_{max}$ of the k highest values of $\mathbf{p}$. The k-max pooling operation makes it possible to pool the k most active features in $\mathbf{p}$. The final output of CNN is a vector $\mathbf{v}_s \in \mathcal{R}^{n_s}$, which represents the embedding of the input sentence s. Given a sentence of interviewee's response q and a sequence r where r is formed by q and interviewee's response in sequence, we can model $\mathbf{v}_q$ and $\mathbf{v}_r$ by using CNN. Then the tensor layer calculates the relevance score of a question-response pair by Eq. 4.

$$s(q,r) = \mathbf{u}^T f\left(\mathbf{v}_q^T \mathbf{M}^{[1:a]} \mathbf{v}_r + \mathbf{V} \begin{bmatrix} \mathbf{v}_q \\ \mathbf{v}_r \end{bmatrix} + \mathbf{b} \right) \quad (4)$$

where f is a standard nonlinearity applied element-wise, $\mathbf{V} \in \mathcal{R}^{a \times 2n_s}$, $\mathbf{b} \in \mathcal{R}^a$, $\mathbf{u} \in \mathcal{R}^a$, $\mathbf{M}^{[1:a]} \in \mathcal{R}^{n_s \times n_s \times a}$ is a tensor and the bilinear tensor product $\mathbf{V}_q^T \mathbf{M}^{[1:a]} \mathbf{v}_r$ results in a vector $h \in \mathcal{R}^a$, where each entry is computed by one slice $i = 1, \ldots, a$ of the tensor $h_i = \mathbf{V}_q^T \mathbf{M}^i \mathbf{v}_r$.

In this study, we select the sentence with the highest relevance score from the interviewee's response as the target sentence for follow-up question generation by using the CNTN-based target sentence selection model. The selected sentence is then used for generating follow-up question pattern.

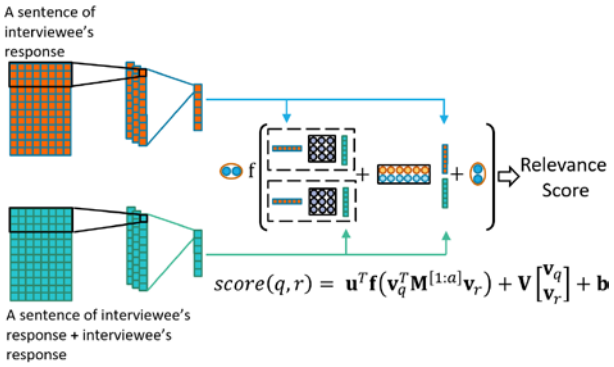

Figure 3: *CNTN-based Target Sentence Selection Model.*

3.3. Follow-up Question Pattern Generation

The LSTM-based seq2seq model [17] used to generate a sequence with respect to the input sequence is an unsupervised machine learning method. In this study, we use the LSTM-based seq2seq model to learn the relationship between the selected target sentence pattern and the follow-up question pattern. The seq2seq model consists of two LSTMs: an encoder and a decoder. The encoder encodes the input sequence into a context vector. The decoder cell initializes the value of the first hidden vector with the context vector. In the seq2seq model, an encoder transforms the selected sentence, $\mathbf{x} = (x_1, \ldots, x_T)$, into a vector c. The hidden output and encoder output are calculated by Eq. 5 and Eq. 6.

$$h_t^e = f(x_t, h_{t-1}^e) \quad (5)$$

$$c = q(\{h_1^e, \ldots, h_T^e\}) \quad (6)$$

where $h_t^e \in \mathbb{R}^n$ is a hidden state of the encoder at time t, and f and q are nonlinear functions. The decoder is often trained to predict the next word y_t, given the context c and all the previously predicted words $\{y_1, \ldots, y_{t'-1}\}$.

$$p(\mathbf{y}) = \prod_{t=1}^{T} p(y_t | \{y_1, \ldots, y_{t-1}\}, c) \quad (7)$$

where $\mathbf{y} = (y_1, \ldots, y_T)$. With a decoder, each conditional probability is modeled as

$$p(y_t | \{y_1, \ldots, y_{t-1}\}, c) = g(y_{t-1}, h_t^d, c) \quad (8)$$

where g is a nonlinear function, and h_t^d is the hidden state of the decoder at time t.

In this study, we use the seq2seq with attention model which consists of a bidirectional LSTM as an encoder and a decoder to generate follow-up question pattern, as shown in Figure 4. In Figure 4, c_i is the context vector, α_{ij} is the weight of each annotation h_t^e, and e_{ij} is an alignment model to estimate the matching scores between the input around position j and the output at position i.

3.4. Candidates Ranking

After generating the follow-up question pattern, we fill the related words into the word class positions in the follow-up

question sentence pattern according to the constructed word class table. As there are many candidate questions after word filling, this study uses the *n*-gram SriLM toolkit to choose the best question as the interviewer's question. SriLM is a statistical language model (LM) which calculates co-occurrence probability between words and finds the best sentence according to co-occurrence probability, as shown in Figure 5.

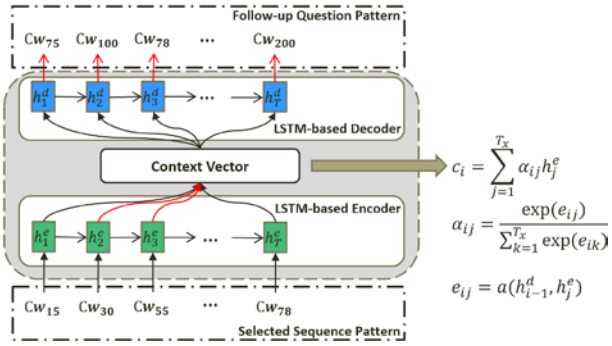

Figure 4: *LSTM-based Sequence-to-sequence with Attention.*

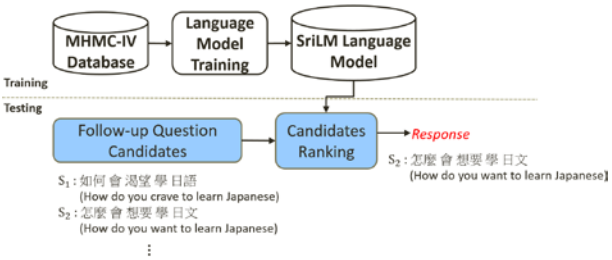

Figure 5: *Illustration of candidate ranking based on SriLM language model.*

4. Experimental Results

4.1. Effect of Hidden Node Number and Tensor Dimension for Sentence Selection

In this experiment, five participants were asked to annotate the relevance type (positive or negative) between each sentence in the interviewee's response and the entire interviewee's response. We selected the data annotated with the same relevance type by more than three participants. Finally, the total numbers of the sentences for relevance and irrelevance were 2,817 and 2,346, respectively. We evaluated the sentence selection accuracy for using CNTN and traditional TF-IDF methods based on the five-fold cross validation method.

The experimental results show that the accuracy of the method using TF-IDF was 80.8%, and the accuracy of the proposed CNTN achieved **88.0%** when the tensor dimension was 5, and the number of CNN filters was 16. We used these experimental results for the subsequent experiments.

4.2. Pattern-based vs. Word-based methods in response generation

In this experiment, the BLEU (bilingual evaluation understudy) score [18] was adopted to evaluate the performance of the traditional method and the proposed pattern-based method with language model. The BLEU score is a measure for evaluating the quality of text that has been machine-translated from one natural language to another. We also compared the proposed pattern-based method with the template matching method and the beam search method. In the template matching method, we calculated the BLEU score to find a suitable template out of the 114 templates obtained from the MHMC-IV corpus. We then used the selected templates for word filling and candidate ranking based on an *n*-gram language model. The beam search is a heuristic search algorithm that explores a graph by expanding the most promising node in a limited set.

Table 2 shows the BLEU score of the traditional method and the proposed methods. The best accuracy was obtained using the proposed method compared to the template matching method and the beam search method. The results show that the proposed method as an objective measure of question generation was better than the result obtained using the traditional word-based method with the small MHMC-IV database.

Table 2: *Experimental results of the word-based and the proposed methods.*

Method	BLEU
Word-based Baseline Method	0.133
Pattern-based + LM	0.260
Pattern-based + LM + Beam Search	0.263
Pattern-based + Template Matching + LM + Beam Search	**0.316**

5. Conclusions

This work presents an approach to follow-up question generation using the integration of CNTN, seq2seq model and *n*-gram language model. First, word clustering is employed to automatically transform the question/answer sentences into sentence patterns. Next, the CNTN model is used to select a target sentence in an interviewee's answer turn. The selected target sentence pattern is fed to a seq2seq model to obtain the corresponding follow-up question pattern. Then the generated follow-up question sentence pattern is filled with the words using a word class table to obtain the candidate follow-up questions. Finally, the *n*-gram language model is used to rank the candidate follow-up questions and choose the most suitable one as the response to the interviewee. For evaluation, five-fold cross validation was used. The experimental results show that the difference between TF-IDF method and the proposed method in terms of sentence selection was very significant. In terms of objective measures, the proposed method achieved a better BLEU score compared to some traditional methods. The generated follow-up questions using the proposed method is more informative and has greater variety than the traditional methods.

There are several issues needed to be further explored in the future. First, this study calculates the word similarity for word clustering through looking up the *E*-HowNet. However, some words are not included in *E*-HowNet, which requires word clustering manually. We hope to design an automatic word clustering method in the future. This will save the time and result in higher word clustering accuracy. Second, this study uses the template matching method to ensure the semantics of the response but lack the variability. Therefore, determining how to generate more informative, relevant, and lively follow-up questions is an issue that still requires effort.

6. References

[1] Y. N. Chen, A. Celikyilmaz, and D. Hakkani-Tür, "Deep Learning for Dialogue Systems," in *the 55th Annual Meeting of the Association for Computational Linguistics*, *July 30 - August 4*, *Vancouver*, *Conada*, *Proceedings*, 2017, pp. 8-14.

[2] M. H. Su, K. Y. Huang, T. H. Yang, K. J. Lai, and C. H. Wu, "Dialog State Tracking and Action Selection Using Deep Learning Mechanism for Interview Coaching," in *the International Conference on Asian Language Processing* (*IALP*), *November 21-23*, *Tainan*, *Taiwan*, *Proceedings*, 2016, pp. 6-9.

[3] M. H. Su, C. H. Wu, K. Y. Huang, T. H. Yang, and T. C. Huang, "Dialog State Tracking for Interview Coaching Using Two-Level LSTM," in *the 10th International Symposium on Chinese Spoken Language Processing* (*ISCSLP*), *October 17-20*, *Tianjin*, *China*, *Proceedings*, 2016.

[4] C. H. Wu, M. H. Su, and W. B. Liang, "Miscommunication Handling in Spoken Dialog Systems Based on Error-aware Dialog State Detection," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 9, 2017.

[5] M. E. Hoque, M. Courgeon, J. C. Martin, B. Mutlu and R. W. Picard, "Mach: My Automated Conversation Coach," in *the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, *September 08-12*, *Zurich*, *Switzerland*, *Proceedings*, 2013, pp. 697-706.

[6] M. R. Ali, D. Crasta, L. Jin, A. Baretto, J. Pachter, R. D. Rogge and M. E. Hoque, "LISSA - Live Interactive Social Skill Assistance," in *the International Conference on Affective Computing and Intelligent Interaction* (*ACII*), *September 21-24*, *Xi'an*, *China*, *Proceedings*, 2015, pp. 173-179.

[7] H. Jones and N. Sabouret, "TARDIS-A Simulation Platform with An Affective Virtual Recruiter for Job Interviews," in *Intelligent Digital Games for Empowerment and Inclusion* (*IDGEI*), *May 14-17*, *Chania*, *Crete*, *Greece*, *Proceedings*, 2013.

[8] V. Rus, and C. G. Arthur, "The Question Generation Shared Task and Evaluation Challenge," in *the University of Memphis. National Science Foundation*, *Proceedings*, 2009.

[9] Z. Zhang, and P. Zweigenbaum, "zNLP: Identifying Parallel Sentences in Chinese-English Comparable Corpora," in *the 10th Workshop on Building and Using Comparable Corpora*, *August 3*, *Vancouver, Canada*, *Proceedings*, 2017, pp. 51-55.

[10] K. J. Chen, S. L. Huang, Y. Y. Shih, and Y. J. Chen, "Extended-HowNet: A representational framework for concepts," in *OntoLex 2005-Ontologies and Lexical Resources*, *October 15*, *Jeju Island*, *Koera*, *Proceedings*, 2005.

[11] B. J. Frey, and D. Dueck, "Clustering by Passing Messages Between Data Points," *Science*, vol. 315, no. 5814, pp. 972-976, 2007.

[12] J. F. Hong, and C. R. Huang, "Using Chinese Gigaword Corpus and Chinese Word Sketch in Linguistic Research," in *the 20th Pacific Asia Conference on Language, Information and Computation* (*PACLIC*), *November 1-3*, *Wuhan*, *China*, *Proceedings*, 2006, pp. 183-190.

[13] J. Pennington, R. Socher, and C. Manning, C. "Glove: Global Vectors for Word Representation," in *the conference on empirical methods in natural language processing* (*EMNLP*), *October 25-29*, *Doha*, *Qatar*, *Proceedings*, 2014, pp. 1532-1543.

[14] X. Qiu, and X. Huang, "Convolutional Neural Tensor Network Architecture for Community-Based Question Answering," in *the Twenty-Fourth International Joint Conference on Artificial Intelligence* (*IJCAI*), *July 25-31*, *Buenos Aires*, *Argentina*, *Proceedings*, 2015, pp. 1305-1311.

[15] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," *International Journal of Computer Vision*, vol. 116, no. 1, pp. 1-20, 2016.

[16] R. Socher, D. Chen, C. D. Manning, and A. Ng, "Reasoning with neural tensor networks for knowledge base completion," in *Advances in Neural Information Processing Systems*, *December 5-10*, *Lake Tahoe*, *Nevada*, *USA*, *Proceedings*, 2013, pp. 926-934.

[17] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," in *6th International Conference on Learning Representations* (*ICLR*), *May 7-9*, *San Diego*, *CA*, *Proceedings*, 2014.

[18] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu, " BLEU: a method for automatic evaluation of machine translation," in *the 40th annual meeting on association for computational linguistics*, *July 7-12*, *Stroudsburg*, *PA*, *USA*, *Proceedings*, 2002, pp. 311-318.