Agricultural Question Classification Based on CNN of Cascade Word Vectors

Lei Chen[1(✉)], Jin Gao[1,2], Yuan Yuan[1], and Li Wan[1]

[1] Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei, China
chenlei@iim.ac.cn
[2] University of Science and Technology of China, Hefei, China

Abstract. Compared with traditional search engines, the query method of QA system is more intelligent and applicable in non-professional scenes, e.g., agricultural information retrieval. Question classification is an important issue in QA system. Since the particularities of agricultural questions, such as words sparsity, many technical terms, and so on, some existing methods are difficult to achieve the desired result in the agricultural question classification task. Hence, it is necessary to investigate how to extract as many useful information as possible from short agricultural questions to improve the efficiency of agricultural question classification. In order to solve this problem, the paper explores effective semantic representation of agricultural question sentences and proposes a method for agricultural question classification based on CNN of cascade word vectors. Different combinations of questions, answers, and synonym information are used to learn different cascade word vectors, which are taken as the input of CNN to construct the model of question classification. The experimental results show that our method can achieve better result in the agricultural question classification task.

Keywords: Agricultural question classification
Cascade word vector · Semantic representation · CNN
Question and answering system

1 Introduction

Different from traditional search engines which use keywords as input and return some candidate answers list, question and answering (QA) system takes users' natural language questions as input and returns accurate answers [15] and has become a hot topic in the field of natural language processing [2,12]. Since the more intelligent query method, the query method of QA system is more applicable for non-professional users. For example, it is very suitable for the actual situation of agricultural information retrieval.

As an important part of QA system, question classification can reduce the space of candidate answers and formulate corresponding strategies for answer extraction, so as to improve the efficiency of whole QA system. Particularly in

J.-H. Lai et al. (Eds.): PRCV 2018, LNCS 11257, pp. 110–121, 2018.
https://doi.org/10.1007/978-3-030-03335-4_10

some professional fields, QA system allows users to ask questions in the specified field. So knowing the type of given question is very helpful for finding the answer of the corresponding type. Moldovan, et al. [14] discussed the influence of each module on the performance of QA system, showing that whether in open field or in professional field, question classification has an important influence on the performance of whole system. Traditional question classification methods include: experience rules based, statistics based and other machine learning models such as SVM, maximum entropy and so on [9,10]. In recent years, question classification based on deep learning has been widely studied [5,6,8]. And with the research of word vector representation, text can be effectively represented in low dimensional continuous form. Different types of word vectors for sentence classification have also been studied [7,11]. Most of the above works are focused on open field. They did not take into account the particularities of the agricultural field, such as words sparsity, many technical terms, and so on. So some existing methods are difficult to achieve the desired result in the agricultural question classification task. At present, there are some researches on information classification for agriculture [4,17,19], which depend on the agricultural ontology library or large-scale corpus. The information representation of agricultural question itself still needs further study.

The paper proposes a method which effectively obtains the semantic representation of agricultural questions based on CNN of cascade word vectors to implement classification task. First, the synonym dictionary is adopted to expand the features of agricultural questions and meanwhile the answer information is used to assist the procedure of question vectors learning. And then the cascade word vectors trained by integrating synonym information with answer information are taken as the input of CNN to construct the model of question classification.

2 Question Classification

2.1 Method Overview

The proposed agricultural question classification method consists of the cascade word vector learning module and the CNN classifier training module, described respectively as follows:

- Cascade word vector learning module: Using different combinations of questions, corresponding answers, and synonym information to learn word vectors, and cascading them to obtain different cascade word vectors.
- CNN classifier training module: Taking the cascade word vectors as the input of CNN model and using the function *softmax* to implement multi classification of agricultural questions in the output layer.

2.2 Cascade Word Vector

As the input of CNN model, the quality of word vectors directly affects the result of the final classification model. Therefore, how to learn the higher quality word

vectors is very important. To achieve this goal, the paper introduces a neural network language model and proposes a cascade word vector learning model which integrates synonyms with answer information. The procedure includes the following steps:

1. Using the information of synonyms to extend the feature of question sentence and taking the expanded question, which includes synonym information, as the input of the neural network language model *word2vec* [13] to obtain the word vector of question.
2. Using the answer information to assist the process of question vectors learning, which trains the questions and answers together to learn the word vectors with more semantic information.
3. Cascading the above two word vectors to obtain the cascade word vectors.

Concretely, in the first step, the feature dimension maybe lost in the training process because the training data is limited. The classifier may ignore the synonymous expression of some missing features, and then affect the final classification result. Therefore, this paper uses synonyms information to expand the features of agricultural questions, which can alleviate the sparsity of question features, enhance the semantic representation ability of questions, and make up for the deficiency of insufficient information in agricultural questions. The synonym dictionary called HIT IR-lab Tongyici Cilin (Extended) [3] is taken as the semantic resources in this paper. In this paper, we use the fifth level synonym information of the dictionary, where the meanings of words are depicted most meticulously, to expand the features of questions. Given an agricultural question, after conducting Chinese word segmentation, the main words including nouns and verbs are concerned and extracted. And then the synonyms of each main word are searched in the synonym dictionary HIT IR-lab Tongyici Cilin (Extended). If the word appears in the dictionary, its synonym is taken out and it is directly affixing to the given question as a new feature. We can see that the extended feature of question can contain the information of all the synonyms of main words. The specific process of question feature extension is shown in the following Algorithm 1.

In the second step, the agricultural questions are usually short with less information but more terminologies, which may cause many difficulties to the classification tasks. Inspired by the work of Zhang et al. [18], the distributed representation of words will be learned by using the common context of questions and answers, which can make full use of the implied semantic information in the answer to enhance the representation ability of question vectors.

In the third step, we use many different combinations of various information, including questions, synonyms, and answers, to train a number of word vectors with different expressions, so as to investigate the effect of different training information on the quality of word vectors. Comparisons between different word vector models will be detailed in subsequent experimental section.

Given a question with n words, denoted by $q = \{w_1, w_2, \ldots, w_n\}$, when its feature extended question is taken as the input of *word2vec*, the ith word w_i will

Algorithm 1. Question feature extension based on synonym

Input: A question q; A synonym dictionary D_s;
Output: A feature extended question q';
1: Conduct word segmentation of the given question q to obtain the word sequence $W_q = \{w_1, w_2, ..., w_n\}$;
2: Extract main words from W_q, denoted by $W_m = \{w_i, ..., w_{i+k}\}$;
3: Initialize the synonym sequence $W_s = \phi$;
4: **for** each $w \in W_m$ **do**
5: if D_s has the synonym of w, put the synonym into W_s;
6: **end for**
7: $q' = W_q + W_s$;
8: **return** q'.

be expressed by the vector x_{ia} of k_{ia} dimension after training. If the parallel corpus including question and answer pairs is taken as the input of *word2vec*, the word w_i will be expressed by the vector x_{ib} of k_{ib} dimension after training. Then the cascade word vector x_i can be obtained, denoted by $x_i = [x_{ai}\ x_{bi}]$, where the dimension of x_i is k_i and $k_i = k_{ia} + k_{ib}$. The cascade word vector x_i simultaneously integrates two kinds of feature information, namely synonyms and answers, so as to enhance the expression ability of question word vectors.

2.3 CNN Model

The structure of CNN model is shown in Fig. 1. The cascade word vectors are taken as the input of CNN to transform features from word granularity to sentence level. The convolution layer uses multi-granularity convolution kernel to further mining question features. The pooling layer extracts the features again and combines them to get global features. The mapping of different types of features is implemented at the full connection layer to get the final results.

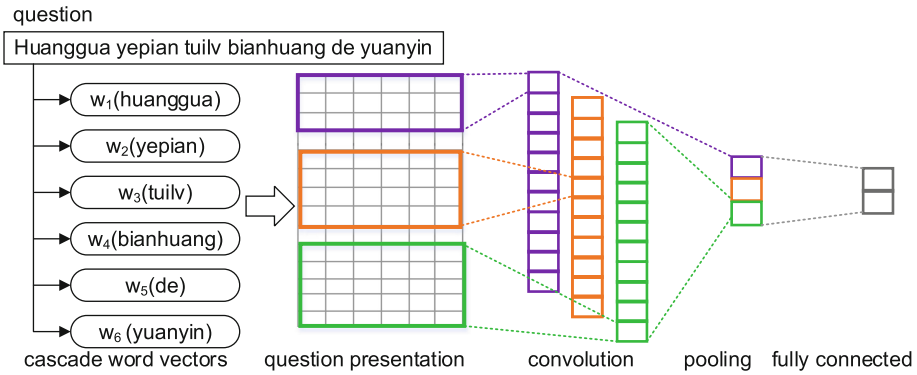

Fig. 1. The structure of CNN model

In the question presentation layer, the cascade word vectors of all words are stacked vertically to get two dimensional question feature data, that is, the feature conversion from word granularity to sentence level is conducted. Given a question $q = \{w_1, w_2, \ldots, w_n\}$, the cascade word vector of word w_i is x_i of k_i dimension and the matrix representation of question q is defined as follows:

$$x_{1:n} = x_1 \oplus x_2 \oplus ... \oplus x_n = \begin{bmatrix} x_{1a} & x_{1b} \\ x_{2a} & x_{2b} \\ ... & ... \\ x_{na} & x_{nb} \end{bmatrix}, \tag{1}$$

where $\oplus$ is the concatenation operator and $x_{i:m}$ is the feature matrix which consists of $x_i, x_{i+1}, \ldots, x_{i+m}$. The question presentation layer is a two-dimensional feature matrix of $n \times k$ if the maximum length of the input question is n.

In the feature convolution layer, in order to fully extract more implicit semantic features of questions, multi granularity convolution kernels are adopted to capture more contextual semantic information and extract multiple local features in the question as far as possible. Generally, the convolution kernel of $h \times k$ dimension is chosen to slide between the adjacent words of input question sentence matrix to get convolution features, where k is the dimension of the word vector and h is the size of the convolution window, namely the number of words slid across the convolution kernel, which can be changed to design the convolution kernel structure of different sizes in experiments. The h words $x_{i:i+h-1}$ in the window are performed the convolution operation to generate the following new feature output c_i:

$$c_i = f(m \cdot x_{i:i+h-1} + b) \tag{2}$$

where $m \in R^{hk}$ is a convolution kernel matrix, $b \in R$ is a bias term and f is a nonlinear activation function, using $ReLu : f(x) = max(0, x)$ in this paper to accelerate the convergence speed of training. After performing the convolution operation, the input question will be mapped into the following feature vector:

$$c = [c_1, c_2, \ldots, c_{n-h+1}],\ c \in R^{n-h+1}. \tag{3}$$

In the pooling layer, we use the maximum pooling operation to process the output feature vectors of the convolution layer to obtain the questions expression of fixed dimension, which chooses the feature with the largest value in feature vectors as the optimal feature output.

$$c^* = \max(c) = \max(c_1, c_2, \ldots, c_{n-h+1}) \tag{4}$$

Subsequently, the whole semantic representation of the question can be obtained by connecting each optimal feature extracted from the pooling layer, where c_i^* denotes the optimal feature obtained from the ith convolution kernel.

$$C = [c_1^*, c_2^*, \ldots, c_n^*] \tag{5}$$

The pooling layer implements the transformation from local features to global features. Meanwhile, both the feature matrix of the question and the number of parameters of the final classification are reduced.

The definition of the fully connected output layer is as follows:

$$y = f(w \cdot C + b), \tag{6}$$

where f is the activation function, w and b are the corresponding weight parameter and bias term when the output of the fully connected layer is y. After using the function *softmax* to normalize, the probability of question q belonging to category t can be obtained:

$$P(t|q,\theta) = \frac{\exp(y_t)}{\sum_{i=1}^{m} \exp(y_i)}, \tag{7}$$

where m is the number of output categories and θ is the set of network parameters. Then the classification labels for final question prediction can be defined:

$$\hat{y} = \arg\max_{t} P(t|q,\theta). \tag{8}$$

The objective function of network training is as follows:

$$J(\theta) = -\frac{1}{m} \sum_{t=1}^{m} l_t \cdot \log(P(y_t)), \tag{9}$$

where l_t is the category label of training samples, y_t is the real label of the question q and $P(y_t)$ is the estimation probability of each category when using the *softmax* function to classify.

In addition, when training samples are few, the over-fitting phenomenon is easily to occur in the network model training process. In order to prevent this problem, we use L_2 regularization to constrain the parameters of CNN model. And the Dropout strategy [16] is introduced in the training process of the fully connected layer.

In summary, the following Algorithm 2 describes the overall flow of the proposed method.

3 Experiments

3.1 Experimental Data

Different from some open fields, this paper concerns on the research of agricultural question classification. Since there is no public data set in the field of agriculture at present, the agricultural question and answering corpus adopted in this paper is mined from Internet, including the following agricultural website:

Algorithm 2. Question classification based on CNN of cascade word vectors

Input: A set of questions $Q = \{q_1, q_2, ..., q_n\}$; A set of word vectors V_1 containing information of questions and synonyms; A set of word vectors V_2 containing information of questions and answers; A test set of questions T; A corresponding set of real categories L;

Output: The classification accuracy Acc of the test set of questions T;

Conduct word segmentation of all questions in Q to obtain the word sequence $q = \{w_1, w_2, ..., w_k\}$;
for each $q \in Q$ **do**
    for each $w \in q$ **do**
        Find the word vector A or B corresponding to the word w in V_1 or V_2;
        Conduct cascading operation between A and B;
    end for
    Connect the cascade vector of each word in q to get the question vector v_q;
end for
Divide Q into training set S_1 and validation set S_2;
Train a CNN model M_1 by using S_1;
Conduct category prediction of S_2 by using M_1;
Perform iterative of the above two steps n times, and select the model with the highest accuracy of validation as the optimal classification model M^*;
Use M^* to do the category prediction of T to obtain the category prediction set L^* of test samples;
Compare L^* with L to get the number of correctly classified samples m;
return $Acc = m/|L|$.

Nongye Wenwen[1], Chinese planting technology website[2], Planting Q&A[3]. After noise cleaning and other preprocessing for the acquired data, our experimental data includes five categories: vegetable planting, fruit tree planting, flower planting, edible fungi planting and field crop planting. Each category contains 2,000 questions, where 15% (300) is used as test set and in the remaining 85%, the validation set and training set are randomly selected 10% (170) and 90% (1530) respectively. Besides, the opensource Chinese word segmentation tool jieba [1] is used to conduct word segmentation and POS tagging of questions.

3.2 Parameter Discussion

In the step of word vector training, we use the neural network language model *word2vec* to learn the distributed representation of words. Words that appear less than 3 times in corpus will be abandoned. The words that do not appear in the *word2vec* vocabulary are initialized by using the values between −1 and 1. The specific parameter settings of this model are given as follows:

– Word vector dimension: 64, 128, 192, 256;

[1] http://wenwen.yl01.com/index.html.
[2] http://zz.ag365.com/.
[3] http://www.my478.com/.

– Selected algorithm: Skip-gram;
– Context window: 5;
– Sampling threshold: 1e−4;
– The number of iterations: 30.

We discuss the specific values of word vector dimensions to explore the influence of the original word vector qualities on classification results. Figure 2 shows the results of verification set accuracy of different word vector dimensions. The accuracy of the validation set is increasing when the number of epochs in the network training increases, especially from 20 to 100. However, excessive epoch may lead to over fitting problem. Therefore, in this paper, the epoch value is set to 100 in all experiments.

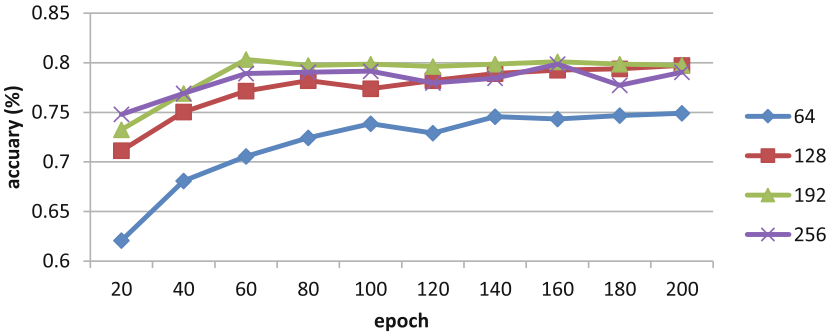

Fig. 2. Verification set accuracy of different word vector dimensions at different epochs

Table 1 gives the test set accuracy of different word vector dimensions at epoch 100. When the word vector dimension increases from 64 to 192, the accuracy of the model increases obviously. However, when the vector dimension continues to increase to 256, the accuracy rate decreases slightly. The analysis is that although the increase of the word vector dimension can contain more word statistics and semantic information, when a certain dimension is reached, more dimensions can not only improve the semantic information of word vectors, but also increase the training complexity of the whole model. Hence, the word vector dimension is set to 192 in the experiments of this paper.

Table 1. Test set accuracy of different word vector dimensions at epoch 100

Word vector dimension	64	128	192	256
Test set accuracy	75.07%	80.66%	83.43%	82.86%

In the step of question classification, we trains the CNN model with Adam update rules. Random gradient descent is performed for each batch of data. And

the parameters of network are updated and optimized by back propagation in each round of training iteration. The specific parameter settings of this CNN model are given as follows:

- Convolution kernel size: 3 × 192, 4 × 192, 5 × 192;
- Learning rate: 1e−3;
- Batch size: 64;
- Regularization coefficient L_2: 3;
- Dropout probability: 1, 0.75, 0.5, 0.25;
- Epoch: 100.

The results of model training with different dropout values are shown in the following Fig. 3. We can see that when the dropout value is 0.75, the accuracy of validation set is the highest. Therefore, the dropout is set to 0.75 in the experiments and the accuracy of test set achieves 82.86% at this dropout value.

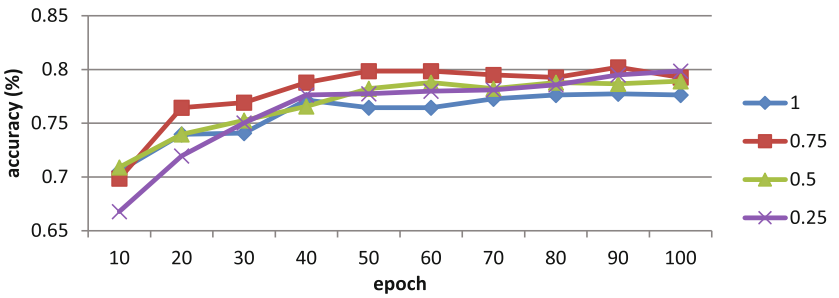

Fig. 3. Verification set accuracy of different dropout values at different epochs

3.3 Contrast Experiments

Different combinations of distributed representation of word vectors are used as the inputs of CNN model with the same structure to conduct several contrast experiments, detailed as follows:

1. CNN+Rand: Using Gauss distribution to randomly initialize all words in a question and transforming each question into a two dimensional feature matrix $n \times k$ as the input of CNN model, where n is the max size of question and k is the dimension of word vector;
2. CNN+Q: Training the word vectors by using the question set only;
3. CNN+Q_A: Training the word vectors by using the question set which is extended the feature by using the information of synonyms;
4. CNN+Q_B: Training the word vectors by using the parallel connection of the question set and the corresponding answer set;
5. CNN+Q_A_B: Training the word vectors by using the parallel connection of the extended question set with the information of synonyms and the corresponding answer set;

6. CNN+Q_A+Q_B: Using the cascade word vectors of Q_A and Q_B as the input of CNN model;
7. CNN+Q_A_B+Q_A_B: Using the cascade word vectors of Q_A_B and itself as the input of CNN model.

The experimental environment of this paper is given as follows: operating system Ubuntu 16.04, 32 GB RAM, CPU Intel Xeon E5-2687W v2 3.4 GHz, deep learning framework Tensorflow 1.4.0, programming language Python 3.5.

3.4 Experimental Results

Figure 4 shows the experimental results of each contrast model on the task of agricultural question classification.

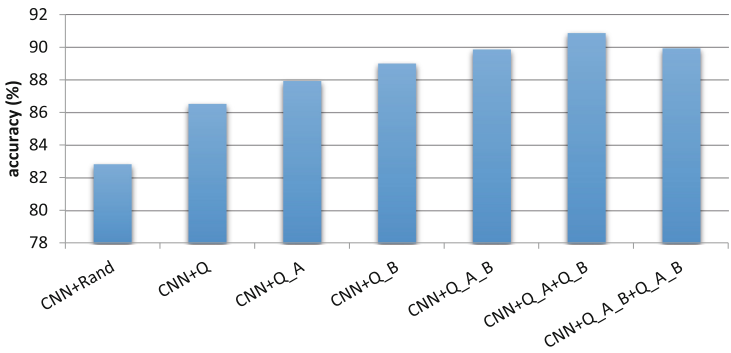

Fig. 4. Experimental results of agricultural question classification

We can see that the proposed CNN model of cascade word vectors, namely (CNN+Q_A+Q_B), can achieve the better result than other contrast methods. More concretely, comparing CNN+Rand with CNN+Q, the learned word vector can capture the contextual semantic information of sentences, and significantly improve the performance of question classification. Comparing CNN+Q with CNN+Q_A, the information of synonyms can extend the features of questions to make up for the sparsity problem, so as to capture more abundant and accurate semantic information in the process of training word vectors. Comparing CNN+Q with CNN+Q_B, using the answer information to assist the learning of question word vectors can also enhance the expression ability of word vectors. Comparing CNN+Q_A+Q_B with CNN+Q_A_B and CNN+Q_A_B+Q_A_B, although the information of synonyms and answers are also incorporated into the training of word vectors, the simple integration of multiple information may cause some information overlay and redundancy. Therefore, the result of using cascade word vectors is better than that of joint training.

4 Conclusion

This paper proposes the CNN model of cascade word vectors to deal with the question classification issue in agricultural filed. The main contributions of this work are given as follows:

- Synonymous information is introduced to expand the features of agricultural questions, so as to alleviate the problem of word sparsity, caused by many professional vocabularies and sparse distribution of words, in agricultural questions.
- The paper proposes the CNN model of cascade word vectors and discusses the influence of different information combinations on the performance of word vectors, showing that the proposed cascaded word vectors can simultaneously express more semantic features of different information.
- The parameter selection of CNN model is discussed through experiments and some contrast experiments of different inputs are carried out, showing that the proposed CNN model of cascade word vectors is efficient in the task of agricultural question classification.

The work in this paper is still preliminary. In next work, the semantics of agricultural terms need to be better exploited and applied. And comparisons with other open set methods will also be considered.

Acknowledgments. The authors would like to thank the anonymous reviewers for their helpful reviews. The work is supported by National Natural Science Foundation of China (Grant No. 31771677) and National Natural Science Foundation of Anhui (Grant No. 1608085QF127).

References

1. Chinese word segmentation component of Python: Jieba. http://www.oss.io/os/fxsjy/jieba
2. Das, R., Zaheer, M., Reddy, S., McCallum, A.: Question answering on knowledge bases and text using universal schema and memory networks. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Short Papers, vol. 2, pp. 358–365 (2017)
3. HIT-SCIR: HIT IR-lab Tongyici Cilin (Extended). https://www.ltp-cloud.com/
4. Hu, D.: The research of question analysis based on ontology and architecture design for question answering system in agriculture. Ph.D. thesis. Chinese Academy of Agricultural Sciences (2013). (in Chinese)
5. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, Long Papers, vol. 1, pp. 655–665 (2014)
6. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, A Meeting of SIGDAT, a Special Interest Group of the ACL, pp. 1746–1751 (2014)

7. Komninos, A., Manandhar, S.: Dependency based embeddings for sentence classification tasks. In: The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016, pp. 1490–1500 (2016)
8. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, pp. 2267–2273 (2015)
9. Le-Hong, P., Phan, X.-H., Nguyen, T.-D.: Using dependency analysis to improve question classification. In: Nguyen, V.-H., Le, A.-C., Huynh, V.-N. (eds.) Knowledge and Systems Engineering. AISC, vol. 326, pp. 653–665. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-11680-8_52
10. Liu, L., Yu, Z., Guo, J., Mao, C., Hong, X.: Chinese question classification based on question property kernel. Int. J. Mach. Learn. Cybern. **5**(5), 713–720 (2014)
11. Ma, M., Huang, L., Xiang, B., Zhou, B.: Group sparse CNNs for question classification with answer sets. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Short Papers, vol. 2, pp. 335–340 (2017)
12. Mao, X., Li, X.: A survey on question and answering system. J. Front. Comput. Sci. Technol. **6**(3), 193–207 (2012). (in Chinese)
13. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR abs/1301.3781 (2013)
14. Moldovan, D.I., Pasca, M., Harabagiu, S.M., Surdeanu, M.: Performance issues and error analysis in an open-domain question answering system. ACM Trans. Inf. Syst. **21**(2), 133–154 (2003)
15. Song, H., Ren, Z., Liang, S., Li, P., Ma, J., de Rijke, M.: Summarizing answers in non-factoid community question-answering. In: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM 2017, pp. 405–414 (2017)
16. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
17. Wei, Z., Meng, F., Guo, J.: Design and implementation of agricultural information search engine classifier. J. Agric. Mech. Res. **2014**(3), 186–189 (2014). (in Chinese)
18. Zhang, D., Li, S., Wang, J.: Semi-supervised question classification with jointly learning question and answer representation. J. Chin. Inf. Process. **31**(1), 1–7 (2017). (in Chinese)
19. Zhang, X.: Research on agricultural information classification method based on deep learning. Master's thesis. Northwest A & F University (2017). (in Chinese)