Journal of Computers Vol. 32 No. 2, 2021, pp. 99-107
doi:10.3966/199115992021043202009

A Hybrid Deep Architecture for Improving Academic Evaluation Capacity in Smart Campus System

Ling Wang[1,2*], Guangjie Han[3]

[1] School of Marxism, Hohai University, Nanjing 210098, China
lingwang@hhu.edu.cn

[2] Dean's office, Hohai University, Nanjing 210098, China
lingwang@hhu.edu.cn

[3] Department of Information and Communication Systems, Hohai University, Changzhou 213022, China
hanguangjie@gmail.com

Received 21 December 2020; Revised 20 January 2021; Accepted 21 January 2021

Abstract. With the fast growing of education informatics, academic evaluation is important for university study life. It bring a series of research questions, including the research about E-learning behavior which is one of hot issues in it. Meanwhile, the modern education attaches importance to Individualized cultivation. More subjective opinions are collected from the students. Thus, automatic inferring academic behavior is gradually becoming the key of perfecting academic evaluation system (AES). In the paper, we collect the academic evaluation information from the online platform of Hohai University. According to the overall opinions, we give the labels to the detail textual comments. A two-stage network is designed and implemented for subjective text analysis and screening the final answers. The former stage is based on bidirectional long and short term memory (LSTM) networks to output a soft label for each sub-sentence. According to the total number of the questions, we locate their answers, and product the inputs of deep forests by cascading their predicted soft labels. Based on cascade forest structure, multi-level forests are reweighted by the forests' contributions. A level-wise growing strategy is used to control the cascade level of the entire structure. Experiment results demonstrate our work are competent for a kernel approach of AES. Meanwhile, we capture many problems in the teaching and learning processes, which are easy to be ignored in the conventional questions and answers (QA) step, and offer some constructive suggestions for the future of smart campus systems.

Keywords: Text analysis, Hierarchical attention mechanism, Ensemble learning, LSTM, AES

1 Introduction

Academic evaluation is the baton to stimulate and promote students' learning and effectively test students' learning effectiveness [1]. The development of information technology and the deep integration of big data, internet, and higher education have deepened the reform of higher education, shaped the new form of education, and promoted academic evaluation mechanism transformation. To explore college students' academic evaluation mechanism under the information technology environment has become a significant development trend of higher education evaluation in the new era. Therefore, making a timely, effective and comprehensive evaluation of college students' studies and using the evaluation results to improve teaching and learning is to stimulate students' interest and motivation in learning, promote students' ability improvement and guarantee the quality of talent cultivation. Since 2015, Hohai

[*] Corresponding Author

University was embedding the academic evaluation module into the smart campus public data system for all its students and teachers. The architecture of the system is illustrated in Fig. 1.

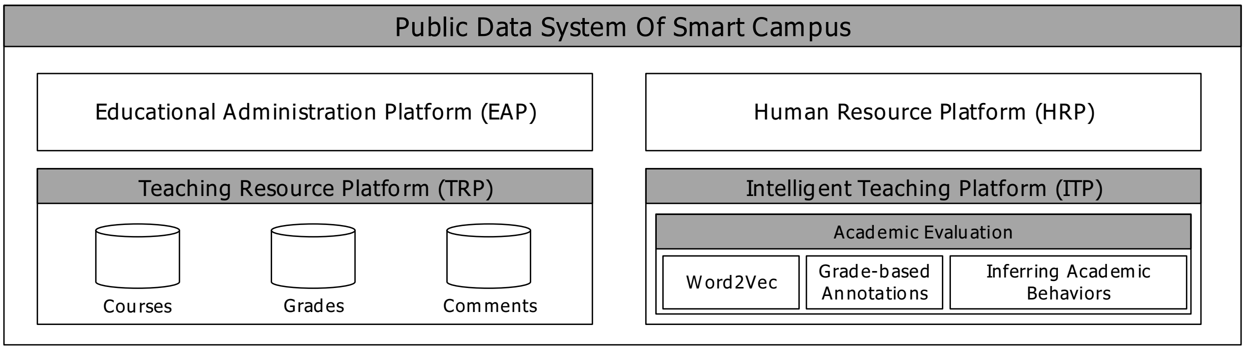

Public Data System Of Smart Campus
Educational Administration Platform (EAP)
Human Resource Platform (HRP)
Teaching Resource Platform (TRP)
Courses
Grades
Comments
Intelligent Teaching Platform (ITP)
Academic Evaluation
Word2Vec
Grade-based Annotations
Inferring Academic Behaviors

Fig. 1. The overall of system architecture

This architecture module covers educational administration platform, human resource platform, teaching resource platform, and intelligent teaching platform. The academic evaluation module is implanted in the intelligent teaching platform, aiming to monitor and analyze academic behaviors. For each academic behavior, Albert [2] considers dividing the evaluation into eight essential components: purpose, content, object, strategies, subject, environmental factors, data processing, and criteria [2]. For better performance of component recognition, most of the solutions for similar problems are based on text analysis, including sentimental analysis, context structural analysis, and so on. The existing state-of-the-art (SOTA) methods are always pouring their attention on some deep network architecture [3] for automatically extracting the high-level semantic information, which is more representative and discriminative. Data-driven architectures can partly overcome the difficulties of generalization. A classic deep Convolutional Neural Networks (CNN)-based architecture contains more than one layer-to-layer processing units, consisting of convolutional operators, pooling layers, and activators. Their outputs are connected and flatted with the fully-connected layers and activated with a Sigmoid or Softmax function for binary or multiple classification tasks, respectively. The contextual information for text academic evaluation is always based on a transformer between words and vectors.

Meanwhile, a summary annotation is used to give a referenced label for the entire textual segment. For inferring the behaviors, the word vectors are fulfilled into the NN-based architecture's inputs with a grammar model to output a predicted label. The grammar-based architecture [4] formats some rules for the organization patterns of the input sentences. Some dummy values of the absent components are inserted into the rule-based constructions to form the fix-sized inputs. The positive samples with few practical components are transformed into the over-sparse representations, perhaps leading to a higher false positive rate.

Recurrent Neural Networks (RNNs) are another classic network architecture. A chain of neurons encodes the word vectors into the matrices of weighted hidden states for the output layer. The RNN units are unfolded and trained by a back-propagation to update a tri-tuple parameter set iteratively for the input sequence of words. In contrast to CNN-based schemes, the RNN-based architecture can boost performance in sequential context detection tasks, such as machine translation and reading comprehension. However, for some long-term inferring tasks, the RNN-based cannot be competent due to the vanishing gradient problem in a long chain of RNN units. Long short term memory (LSTM) cell, a variant of the RNN unit, is introduced to endow the units with memories. A gate function is used to filter the previous message and combine with the current inputs to update the massage. LSTM can avoid the emergence of the long-chain effect but lack for parallel [5]. For long and complicated sentences, each LSTM cell owns a deep network and more fully-connected layers, which rocket the computational cost. Though part of the advanced methods is appended attention mechanism to reweight the hierarchical contexts, the performance has been improved slowly due to the ill-posed mechanism selection.

Academic evaluation system (AES) is a complex NLP problem. The scoring and subjective evaluation are synchronously finished in each academic evaluation. Thus, compared with machine reading comprehension (MRC) [6], the entire works do not need marking personal items. The main flow of AES contains two tasks: word representation, semantic understanding [7].

Word representation is essential for text analysis, aiming to transform the input sentence's words into vectors. Before the emergence of the grammar-based methods, conventional methods are always selecting the keyword and label its neighborhood with 0-1 to fulfill a sparse matrix [8]. Words, phrases, and sub-sentences are organized as the centers of the sentence, but if the keywords are wrong for the nodes of the phase tree, the performance of the detection tasks heavily depends on the keywords' locations. Dhingra et al. [9] found that the small-scale replacement of the elements will misguide the readers' understanding, in which some verbs need multi-layer embedding. Peters et al. [10] proposed ELMo, a downstream model to boost the fine-tuning for the contextual information effectively. A hierarchical pre-trained language model packages character-level and word-level context for word embedding. Devlin et al. [11] design BERT to pre-train a deep bidirectional transformer for understanding contextual information.

Semantic analysis can seem like a classification framework for the embedded vectors. The entire framework is roughly divided into two parts, including semantic segmentation and text sentiment analysis. Almost all the mainstream methods are based on Neural Network, aiming to compute the similarity between contextual information and query information. Wang et al. [12] use a multi-perspective window to collect the input sentence's critical context as the answers and sort them based on similar scores. Their further study update GRU (Gated Recurrent Units) with an attention mechanism [13] to emphasize some particular questions' answers. Yu et al. [14] give a multi-stream reasoner to infer the more accurate positions for the varying questions' answers.

In this paper, we first use an LSTM network with an attention mechanism to locate the questions' answers. We set some vital questions to collect the comments for teaching resources and learning predicament. All the answers are listed in a vector to be orderly fed into the cascade forest. For a multi-perspective observation, we use a multi-grained scanner to manufacture the varying-sized inputs. Each level of the cascade forests parallels four random forests and uses a level-wise growing mode to determine the next cascade's need.

The rest of this paper is organized as follows: Sec. 2 introduces materials and methodology, containing the data pre-processing and the proposed framework. Sec. 3 reports experimental results from overall to local. Sec. 4 is used to analyze the problems covered by the proposed framework's results and give some constructive suggestions. Sec. 5 concludes our works and delineates the future works.

2 Materials and Methodology

2.1 Data Collection and Annotation

The development of internet technology causes the transmission and acquisition of information very convenient and changes the learning model of college students, making college students learning new characteristics. In this study, 1200 college students from Hohai University were randomly selected by the questionnaire survey, and 1156 valid questionnaires were collected. The results show that on average, 22.29% of college students surf the internet for 1-2 hours, 41.1% for 3-4 hours, 24.5% for 5-6 hours, and 12.11% for more than 6 hours every day. It can be seen from the survey that college students spend much time on the internet, 41.1% of them spend 3-4 hours on the internet, and 12.11% of them spend more than 6 hours a day on average. We collect 24500 comments on academic evaluation. According to the 5-class radios, we annotate the 24500 comments with degrees, including excellent, good, general, bad, and worse. We select excellent and good groups into the positive subset and bad and worse groups into the negative subset. The total number of comments is 19532, and the ratio of the positive/negative cases is around 1:1.2 (8878:10654). The training and testing sets occupy 90% and 10% of the entire dataset, respectively.

2.2 Data Augmentation

Considering the replacement of the contextual information, we employ the corpus of NLPCC2013, which contains the extracted 13049 sub-sentences from the 4000 posts. They are merged with the 19352 comments with clear attitudes. After the data augmentation, we obtains 35520 sub-sentence from the merged dataset of 23352 comments.

2.3 The Proposed Framework

Word Embedding: In this section, we follow the ELMo-based word embedding approach. We use three-layer nested structure to embed the contextualized, word and character information. As a basic encoder, a word2vec model is first use to represent the words of the sentence with a skip-gram model, which is shown in Fig. 2.

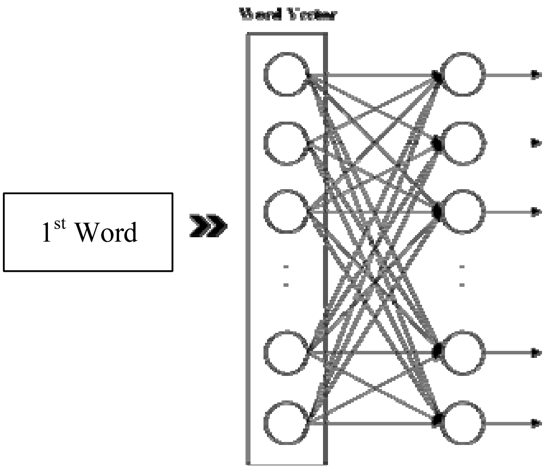


Fig. 2. A word2vec model with a skip-gram model

Contextualized Embedding: A pre-trained Bi-LSTM model is working between the contextualized embedding layer and the word embedding layer. Concretely, each LSTM cell is composed of input gate, cellular state, current cellular state, hidden state, memory gate, forgotten gate, and output gate. A chain of LSTM cells is shown in Fig. 3.

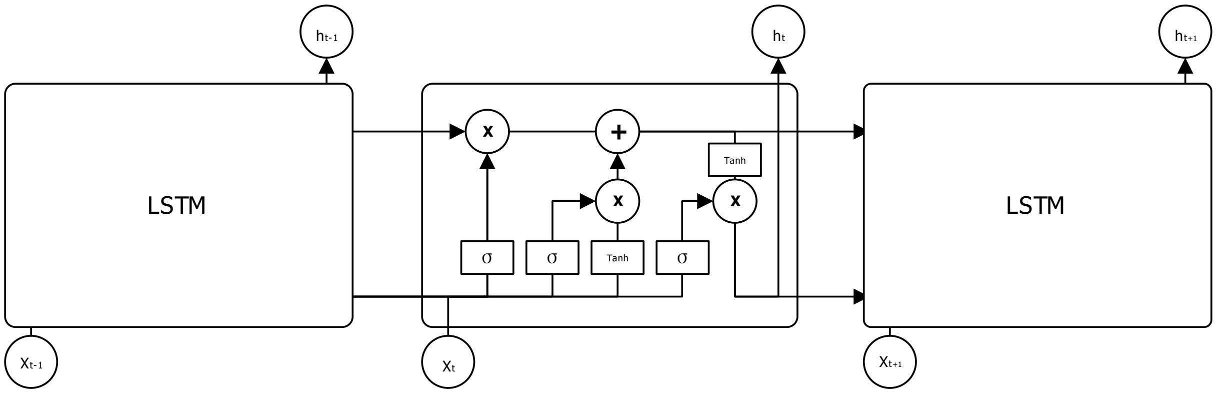


Fig. 3. A chain of LSTM cells

Seem from Fig. 3, it covers four computational process, consists of computing forgotten gate f_t, computing memory gate i_t, updating current cellular state C_t, and computing output gate o_t and hidden state h_t. The process can be formulated as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{1}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\tilde{C}_t = \tan h(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{3}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{4}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad \textbf{(5)}$$

$$h_t = o_t * \tan h(C_t) \quad \textbf{(6)}$$

Character Embedding: For a given sentence of N words, we can build a bidirectional propagation path. Each word has two hidden states $\vec{h}_n^t$ and $\overleftarrow{h}_n^t$ by encoding with the forward and backward LSTM layers, respectively. The context-dependent representation of the *n*-th word can be written by

$$R_n = \{x_n, h_n\} = \{x_n, \vec{h}_n^t, \overleftarrow{h}_n^t\} \quad \textbf{(7)}$$

where x_n is the *n*-th word vector, and $h_n^t = \{\vec{h}_n^t, \overleftarrow{h}_n^t\}$ is a couple of its hidden states at the *t*-layer of BiLSTM. All the hidden state of T-layer can be linearly combined into a vector to weigh the contextualized information.

$$x_n^e = r_n \sum_{t=1}^{T} S_l(w_n) \cdot h_n^t \quad \textbf{(8)}$$

where $r_n \in R_n$, and S_l is a softmax function. We used a CNN-based encoder to embed each word into a 64-dimension vector. The vectors are combined with the others, and socket into a 350-dimension vector, due to the most length of the sub-sentences are less than 250. Tailing with a chain of 350 LSTM cells, the length of the hidden neurons with ReLU activators is 512. After the above process, the input context is encoded into a contextualized vector, and it can compute a similarity matrix S of $C \times Q$ to choose a query vector, in which its element S_{cq} is a cosine distance between the *c*-th contextualized vector and the *q*-th query vector.

Hierarchical Attention Mechanism: Integrating with the outputs of charter embedding, we obtain a tri-tuple vector $\{x_n^c, x_n^W, x_n^e\}$ to update the representation of the sub-sentences. Meanwhile, we consider to introduce an attentive mechanism for reweighting the elements of the vectors, which can be formulated by

$$a_t = \frac{\exp(S_t)}{\Sigma_t \exp(S_t)} \quad \textbf{(9)}$$

$$o_t = h_t \cdot a_t \quad \textbf{(10)}$$

The reweighted vectors are fed into a chain of 50-dimension LSTM cells to output the final evaluation labels. We minimize the sum of the negative log probability for the K answer covered by the input sentence, which can be computed by

$$L = -\frac{1}{N}\sum_{n=1}^{N}\left[\sum_{k}\log(p_k^b) + \sum_{k}\log(p_k^e)\right] \quad \textbf{(11)}$$

where p_t^b and p_t^e are the starting and ending words of each predicted answer. For the predicted K answers, we can cascade them to generate a newborn vector for the next deep decision framework.

Deep Forests: We follow the scanning mode of the original deep forests [15]. We use a multi-scale sliding windows to generate the varying-sized outputs for cascade forests. For each cascade, the inputs of cascade forests are ending with a 4-bit decision vector. It can be weighted with the contributions of forests [16], which are computed by the standard deviations of the top-5 features for each forest.

3 Experiments

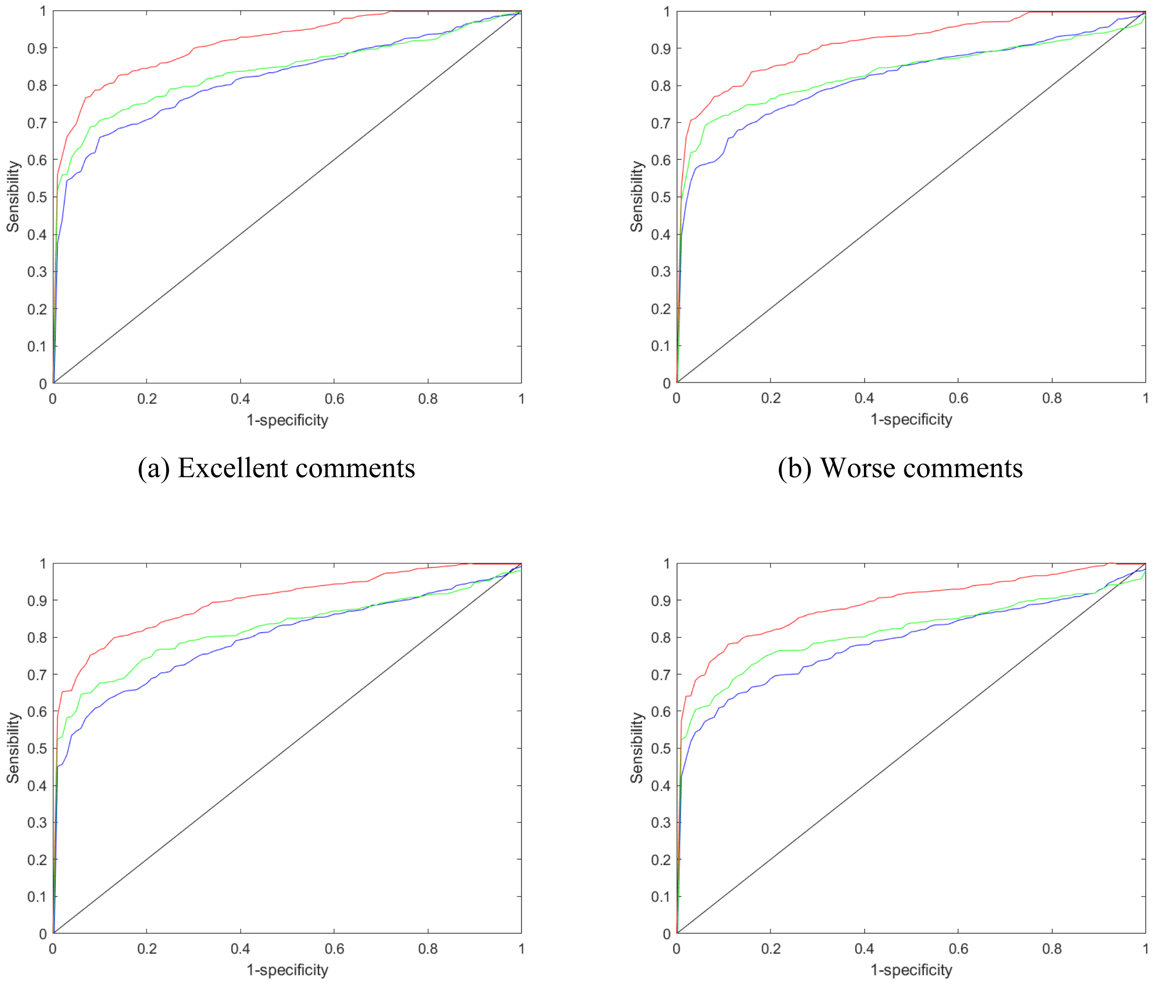
3.1 Evaluation Metrics

In our experiments, we employ CPM and AUC to evaluate the performance of the proposed framework. AUC (Area under Receiver Operating Characteristic (ROC) Curve) is another evaluation metric to observe the relation between the true positive rate and the false positive rate. Competition Performance

Metric (CPM) [17] is an average value of the seven operating points at Free Receiver Operating Characteristic (FROC) curve. Compared with ROC, FROC-based CPM focus on the speed of regressing to 1, which are used to prove the robustness of the proposed framework.

3.2 Results

We report the experimental results from the global and local perspectives. Overall, we shows a detection results of a binary classification task to identify the input sentences belong to the positive/negative groups. For a fair comparison, all the experiments are running on the server with a Nvidia GTX 3090ti GPU (Graphic Processing Unit) of 24G available memory. We select 2 classic schemes of text analysis, including LSTM+XGboost and AT-GRU (Attentive GRU). AT-GRU is a variant of attentive LSTM. It need less parameter tuning to reduce the risk of overfitting. XGboost is an open source project based on GBDTs (Gradient Boosted Decision Trees) to build a decision framework, which is more similar with the deep forests (DFs). The proposed framework is based on deep forests with hierarchical attentive LSTM model. Without the general group, we give the AUCs of the other four groups. The ROCs and the AUCs are shown in Fig. 4 and Table 1, respectively.



(a) Excellent comments

(b) Worse comments

(c) Good comments

(d) Bad comments

Fig. 4. The ROCs of the different schemes, including LSTM+XGboost (Blue group), AT-GRU (Green group) and ours (Red group)

Table 1. The AUCs of the different schemes

Schemes	LSTM+XGboost	AT-GRU	HALSTM+DFs (Ours)
Excellent	0.8179	0.8328	0.9105
Good	0.7865	0.8119	0.8881
Bad	0.7960	0.8201	0.8961
Worse	0.8162	0.8357	0.9124

Seen from Fig. 4, we find the curves of the excellent and worse comments have higher speed closing to 1 than those of the good and bad comments. The performance of the proposed framework is prior to the other two methods. The LSTM+XGboost schemes give a worse AUC in this comparison. Because LSTM cells without attention mechanism could be not used to analyze the long chain of the words effectively. Moreover, in the application of XGboost, we need first pre-sort the features to select the discriminative ones from the entire feature set. But not all the pre-sort processing is necessary and the classification results are depend on the outputs of LSTM cells. Thus, it give the lowest AUC in this comparison. Table 1 shows the detail values of the AUCs.

Locally, we further observe the FROC curves of different schemes. Table 2 shows a quantitative comparison of CPM score and the seven operating points. To simply the experiment, we select the attentive GRU scheme as the compared scheme. In Table 2, we find the sensibilities of the proposed framework are apparently higher than those of AT-GRU. Especially at 1/8, 1/4 and 1/2 FP/p (False positives/sentence), HALSTM give the higher sensibilities of 0.814, 0.854 and 0.911. These demonstrate the proposed method has better detection performance than the compared method.

Table 2. CPM scores of text sentiment analysis among the SOTA methods and ours

Schemes	AT-GRU	HALSTM+DFs (Ours)
1/8 FP/s	0.782	0.814
1/4 FP/s	0.819	0.854
1/2 FP/s	0.902	0.911
1 FP/s	0.933	0.942
2 FP/s	0.945	0.956
4 FP/s	0.948	0.972
8 FP/s	0.951	0.981
CPM	0.897	0.919

4 Application

4.1 Problems

It is a systematic project to establish the academic evaluation system that conforms to the law of major and promotes students' development. The current academic evaluation mechanism tends to emphasize the result rather than the process. There is a fuzzy correlation between the course examination sites and the achievement degree of students' development in the academic evaluation. The main body of the evaluation is relatively single, and the third party is lack intervention. Besides, academic evaluation relies too much on written tests, and diversified evaluation forms have not been formed. Teachers' knowledge and awareness of academic evaluation are still insufficient.

4.2 Solutions

Strengthening teachers' academic evaluation knowledge and change the concept of teachers' participation in academic evaluation. The training of teachers' academic evaluation includes the content of the academic evaluation, curriculum teaching reform, and information literacy, strengthening the training of teachers' academic evaluation policies and content, guiding and promoting teachers to carry out curriculum teaching reform, improving teachers' information literacy and improving teachers' ability to use information technology to carry out academic evaluation reform.

Establishing an academic support system for college students to provide academic guidance and help for college students. Under the Internet technology environment, teaching and learning have changed, and the ways of obtaining information are diversified. How to adapt to learning in the new era, strengthen the self-monitoring of network learning, and improve information selection and application are the problems that colleges and universities need to solve. Besides, according to the development tasks of different grades, training objectives of different majors, and personalized differences of students, colleges and universities need to rely on college students' academic institutions to develop personalized academic programs to support the study and development of college students.

We will improve the academic evaluation system and reform examination management. Innovate the form of assessment, the implementation of written test, oral test, computer, report, paper, and other assessment forms. Strengthen the construction of the question bank, take the improvement and development of students' ability as the goal, and optimize academic evaluation. A third-party evaluation is introduced to evaluate the contents, methods, and effectiveness of academic evaluation to promote scientific nature.

5 Conclusions

This paper proposes a two-stage framework with BiLSTMs (Bidirectional Long and Short Term Memory) Networks and DFs (Deep Forests) for seeking the answers from the input sentences and collecting the answer vectors to exactly locate the positions of the answers with random forests (RFs)-based deep decision framework. Considering the characteristics of the Chinese comments, we collect NLPCC2013 corpus, and merge with self-collected dataset. By using a divide and conquer methodology, the pre-trained BiLSTM networks output the answers and infer their more exact locations, and then cascades all the answers into an available vector for a final decision. Both in the comparisons of AUC values and CPM scores, the proposed framework gains the desired results, which have achieved 0.919 and 0.902. In our future work, we are going to introduce LightGBM to replace DFs, which update level-wise cascade to a novel leaf-wise cascade. It can boost the computation speed with minor performance loss. Meanwhile, we trackback some key problems in teaching and studying processes to guide the future development of smart campus systems.

Acknowledgements

This work was supported by the higher education science research project of Hohai University "Research and Practice on the Academic Evaluation Mechanism of Innovative Talents' Diversification from the Perspective of First-class Undergraduate Education" (NO. 2018B51614)

References

[1] W. Geng, Research on Academic Evaluation of College Students Based on Big Data, in: Proc. 2019 34rd Youth Academic Annual Conference of Chinese Association of Automation (YAC), 2019.

[2] L. Pombo, V. Carlos, M. J. Loureiro, EDUlabs for the integration of technologies in Basic Education–monitoring the AGIRE project, International Journal of Research in Education and Science 2(1)(2016) 16-29.

[3] Y. Lecun, Y. Bengio, G. Hinton, Deep learning, Nature 521(7553)(2015) 436-444.

[4] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Proc. Adv. Neural Inf. Process. Syst., 2013.

[5] W. Wang, N. Yang, F. Wei, B. Chang, M. Zhou, Gated self-matching networks for reading comprehension and question answering, in: Proc. 55th Annu. Meeting Assoc. Comput. Linguistics, 2017.

[6] K.M. Hermann, T. Kočiský, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, P. Blunsom, Teaching machines to read and comprehend, in: Proc. Adv. Neural Inf. Process. Syst., 2015.

[7] A simple and efficient neural architecture for question answering. <https://arxiv.org/abs/1703.04816/>, 2017.

[8] Reinforced mnemonic reader for machine reading comprehension. <https://arxiv.org/abs/1705.02798/>, 2017.

[9] D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by back-propagating errors, Nature 323(6088)(1986) 533.

[10] Deep contextualized word representations. <https://arxiv.org/abs/1802.05365/>, 2018.

[11] Pretraining of deep bidirectional transformers for language understanding. <https://arxiv.org/abs/1810.04805/>, 2018.

[12] Multi-perspective context matching for machine comprehension. <https://arxiv.org/abs/1612.04211/>, 2016.

[13] Bidirectional attention flow for machine comprehension. <https://arxiv.org/abs/1611.01603/>, 2016.

[14] Combining local convolution with global self-attention for reading comprehension. <https://arxiv.org/abs/1804.09541/>, 2018.

[15] Z.-H. Zhou, J. Feng, Deep forest: towards an alternative to deep neural networks, in: Proc. Proceedings of the 26th International Joint Conference on Artificial Intelligence, 2017.

[16] Y. Guo, S. Liu, Z. Li, X. Shang, Towards the classification of cancer subtypes by using cascade deep forest model in gene expression data, IEEE International Conference on Bioinformatics and Biomedicine (2017) 1664-1669, doi: 10.1109/BIBM.2017.8217909.

[17] M. Niemeijer, M. Loog, M.D. Abramoff, M.A. Viergever, M. Prokop, B. van Ginneken, On combining computer-aided detection systems, IEEE Transactions on Medical Imaging 30(2)(2010) 215-223.