Attention-Based Hybrid Model for Automatic Short Answer Scoring

Hui Qi[1,2], Yue Wang[1,2], Jinyu Dai[1,2], Jinqing Li[1,2], and Xiaoqiang Di[1,2](✉)

[1] School of Computer Science and Technology,
Changchun University of Science and Technology, Changchun, China
dixiaoqiang@cust.edu.cn

[2] Jilin Province Key Laboratory of Network and Information Security,
Changchun, China

Abstract. Neural network models have played an important role in text applications, such as document summaries and automatic short answer questions. In previous existing works, questions and answers are together used as input in recurrent neural networks (RNN) and convolutional neural networks (CNN), then output corresponding scores. This paper presents a method for measuring the score for short answer questions and answers. This paper makes scoring by establishing a hierarchical word-sentence model to represent questions and answers and using the attention mechanism to automatically determine the relative weight of questions and answers. Firstly, the model combines CNN and Bidirectional Long Short-Term Memory Networks (BLSTM) to extract the semantic features of questions and answers. Secondly, it captures the representation vector of relevant questions and answers from the sentence-level features. Finally, all feature vectors are concatenated and input to the output layer to obtain the corresponding score. Experiment results show that the model in this paper is better than multiple baselines.

Keywords: Attention-based hybrid model · Automatic short answer scoring · BLSTM · CNN

1 Introduction

Automatic Short Answer Scoring (ASAS) refers to the scoring of answers to short answers without human intervention. The process is mainly to judge the similarity between the answer and the standard answer in terms of words and semantics. In most cases, the answer and the standard answer are not necessarily identical. Answers with similar meanings are acceptable. In addition to the need to have strong professionalism, the reviewer needs to be patiently thinking, and the number of short answer questions is significantly higher than objective questions. Therefore, there are two disadvantages of the manual scoring methods for short answer questions in the scoring process, such as the difficulty of ensuring fairness, the speed and lower efficiency of scoring. As for the automatic scoring

H. Song et al. (Eds.): SIMUtools 2019, LNICST 295, pp. 385–394, 2019.
https://doi.org/10.1007/978-3-030-32216-8_37

of short answer questions, it's the key issue to effectively utilize the information in the answer and the reference answer. It also serves as a research point in natural language processing, and has a significance for automatic evaluation of short answer scoring.

Therefore, the establishment of a complete automatic scoring model is the key point to the automatic evaluation of short-answer questions. Traditional scoring models use sparse features, such as word bags, part-of-speech tags, grammatical complexity metrics, and essay lengths, but these features may be affected by time consumption and sparse data features. Recently, it has been proved that the results of using neural network models are better, compared to traditional manual feature statistical models. Specifically, the distributed word representation is used to input, and the neural network model is used to combine the word information to obtain a single dense vector form in the entire answer. A score is given based on the non-linear neural layer on the representation. It has been demonstrated that neural network models are more effective than statistical models in different fields without manual features.

Deep learning gradually evolves from the distribution representation of the initial computational words to the calculation of distributed representations of phrases, sentences, and texts that contain more semantic information. The most basic application of word vectors is calculating the semantic similarity of two words. Correspondingly, when obtaining the sentence vector from the model trained by the complete corpus, we can also give the semantic similarity of the two sentences. Therefore, we can use the neural network method of deep learning to represent the answer and the standard answer as a sentence vector contains rich semantic information, then scoring the similarity among vectors as the semantic similarity between the answer and the standard answer. Currently, Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) are two mainstream architectures of Deep Neural Networks (DNN), which have been widely used to handle automatic test scoring tasks. The CNN can obtain features by stacking multiple layers including a convolutional layer and a merged layer. The RNN can handle the sequential problem of propagating historical information through the chained neural network architecture, and deal with sequence problems [5, 14], such as bidirectional long short-term memory networks (BLSTM) model [16]. Combining the advantages of both RNN and CNN, this paper proposes a score based on hybrid RNN and CNN to calculate the answers of the short answer questions and applies a hierarchical attention mechanism at the word and sentence level [21].

In this study, the model uses the mixed attention network of BLSTM and CNN to capture the most important semantic information in the short answer questions. BLSTM and CNN have been proven to be very effective for simulating answer sequences and useful for learning long-term dependent data. For traditional BLSTM and CNN networks, it is important to enter each word in the sentence, which is reasonable for traditional automatic short answer scoring tasks. The main contribution of our work can be summarized as follows: (1) we explore the attention-based hybrid model to measure automatic short answer scoring; (2) we apply an attention mechanism, which can enhance the mutual relation between the aspect term and its corresponding sentences, and prevent the irrelevant words from getting more attention; (3) we carry out the experiment on the dataset by utilizing multiple methods.

The rest of this paper is organized as follows: Sect. 2 briefly introduces the work related to this study. Section 3 describes the model in detail. Experimental results are reported in Sect. 4. Section 5 concludes this paper.

2 Related Work

In the past, scholars have proposed many automatic short answer scoring methods. Project Essay Grade [13] is one of the earliest automated scoring systems that use linear regression to predict scores. Developed by the Educational Testing Service, E-Rater [2] was one of the first systems to use operational scoring in high-stakes assessments. The model utilizes many different features in scoring, the model building approach, and the final score assignment algorithm. Chen et al. [4] used a voting algorithm based on the initial scores and similarities between essays to iteratively train the system and score the essays. McNamara et al. [12] attempted to translate what we might observe in human raters within a computational algorithm by using hierarchical classification with different variables allowed to enter at each level. Fala et al. [7] developed systems that predict holistic essay scores based on features extracted from opinion expressions, topical elements, and their combinations. They also attempted to incorporate more different features into the text scoring model. Klebanov and Flor [11] showed that the higher scoring essays tend to have higher percentages of both highly associated and dis-associated pairs, and lower percentages of mildly associated pairs of words. Somasundaran et al. [15] used lexical chains, and interactions between lexical chains and explicit discourse elements, which can be harnessed for representing coherence to assess paper score.

Recently, Alikaniotis et al. [1] used the long short-term memory model (LSTM) to automatically learn paper scoring tasks, thus eliminating the need for any predefined feature templates. It uses score-specific word embeddings (SSWEs) to word representation. The last hidden state of the bidirectional LSTM is used for these representations. Taghipour and Ng [17] used the automated essay scoring LSTM model, which utilizes common word embedding and uses the average combined value of all hidden states of the LSTM layer as a paper representation. Dong and Zhang [6] obtained the final text representation by processing the text into sentences and using two layers of CNN at the sentence and text levels. Bahdanau [3] proposed a mechanism for attention in machine translation. Bahdanau applied the base concern model to machine translation, which allows the decoder to observe different parts of the source statement at each step of the output generation, rather than encoding all source statements into fixed-length vectors and explicitly finding the soft alignment of the current position and between input sources. Since then, the attention mechanism has been further used. Zhang [10] proposed a CNN based on attention pool representation sentence that uses the intermediate sentence representation generated by BLSTM as a reference to the local representation produced by the convolutional layer to obtain attention weight. Yang [18] designed Hierarchical Attention Networks (HANs) for document classification, and this document classification was applied to two levels of attention mechanisms at the word and sentence level. Yujun [20,21] proposed Hybrid Attention Networks (HANs), which combined selective attention to the vocabulary and character level. The model first applied RNN and CNN to extract the semantic features of the text. Among them, the model of this paper is most closely related to the HANs model, and the HANs model represents a sentence with a hierarchical attention mechanism.

The work of this paper is to systematically investigate the sentence-level and text-level modeling of CNN and LSTM, and notes the effectiveness of the network to automatically select more relevant n-grams and sentences for the task. Compared to the existing researches, this paper proposes to study the sentence representation based on the hybrid network HANs model of hybrid RNN and CNN. Model combined semantics and captured long-distance dependencies among words has significant advantages. In addition, this paper proposes a hierarchical attention mechanism to capture the semantic concerns in each sentence and the model helps to filter out noise that is unrelated to the overall sentiment [19].

3 An Attention-Based Hybrid Model

As shown in Fig. 1, we propose an attention-based hybrid model combining BLSTM and CNN which contains four components:

(1) Input layer: input two diverse sentences of questions and answers to this model;
(2) Embedding layer: map each word of a sentence of questions and answers into a low-dimension vector;
(3) Hybrid attention layer: produce a weight vector, and concatenate word-level features into a sentence of questions and answers feature vector by multiplying the weight vector;
(4) Output layer: calculate score by concatenating sentences of questions and answers feature vector.

Later in this section, these components will be presented in detail.

3.1 Word Embeddings

Given a sentence of questions and answers composed of N words $S = \{w_1, w_2, ..., w_N\}$, every word w_i is converted into an embedding vector e_i. A word w_i is transformed into its word embedding e_i by using the matrix-vector product:

$$e_i = W^w v^i \tag{1}$$

The embedding matrix W^w is the parameter to be learned, $W^w \in \mathbb{R}^{d^w|V|}$, where V is a fixed-sized vocabulary, and d^w is the size of word embeddings. It is a

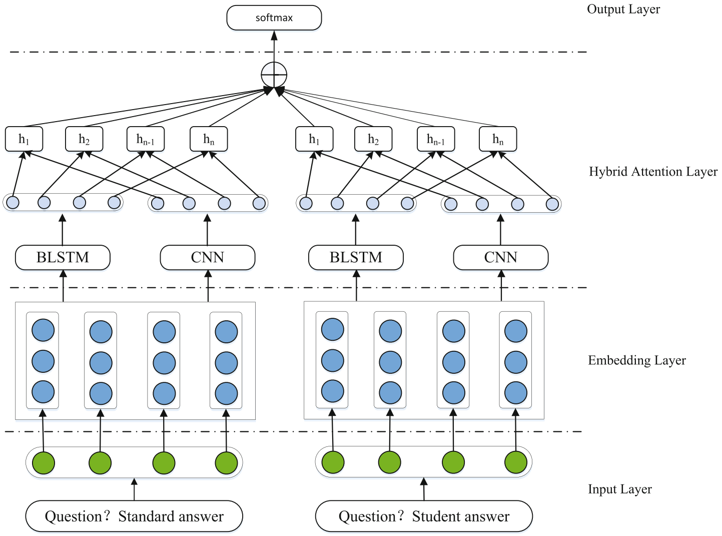


Fig. 1. The architecture of the Attention-based Hybrid model.

hyper-parameter to be set by user. The word vector v^i is a vector of size $|V|$ which has value 1 at index e_i and 0 at all other positions. Then a sentence of questions and answers is fed into the next layer as a vector $E_S = \{e_1, e_2, ..., e_n\}$.

The goal of embedding layer is to represent each word in sentences with a d-dimensional vector. The whole embedding space is used, in which the embedding is updated after each batch.

3.2 Hybrid Attention

The motivation of attention is inspired by the observation that different words should have different contributions to the final semantic representation of a sentence of questions and answers. When reading a sentence, people often pay attention to a word or several words, and these words can reflect meaning of the answer. So, we use attention mechanism focused on word-level to implement this motivation.

We only pay attention to these words whose semantic relationship have a great impact on sentence of questions and answers meaning through word-level attention mechanisms. BLSTM and CNN can extract the feature representations in word-level attention architecture. On the attention layer, the output is the concatenated representations.

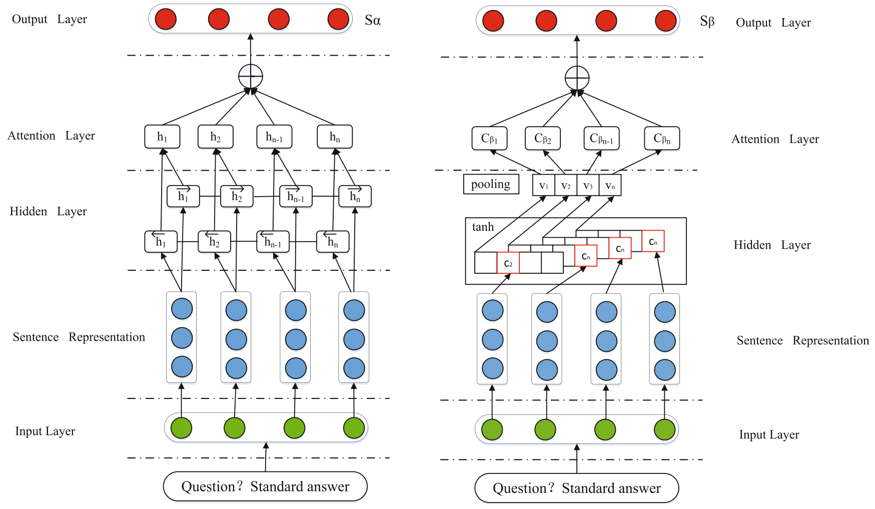
In Fig. 2(a), the BLSTM produces the output vectors $[h_1, h_2, ..., h_n]$. As Eq. (2) shows, we can use an attention-weighted sum of output vectors to generate the representation S_α of a sentence of questions and answers. The attention-weight α_i is shown in Eq. (4), where W_α is a word of weight. And Eq. (3) represents the output of hidden layer.

$$S_{\alpha} = \sum_{i=1}^{l}(\alpha_i h_i) \tag{2}$$

$$u_i = tanh(W_h h_i + b_h) \tag{3}$$

$$\alpha_i = softmax(W_{\alpha} u_i) \tag{4}$$

The attention mechanism gets the representation S_{α} for the output of forward and backward LSTM from the formulas described above. The BLSTM network proposed in [8,9] can be utilized. Past features (via forward states) and future features (via backward states) for a specific time can be used. Except for the need to unfold the hidden states efficiently, the forward and backward passes over the unfolded network through time are performed in a similar way to the forward and backward passes of conventional networks.



(a) BLSTM Attention (b) CNN Attention

Fig. 2. The architectures of BLSTM and CNN attention networks. S_{α} indicates the attentive sentence representation of forward and backward LSTM, S_{β} indicates the attentive sentence representation of CNN.

In Fig. 2(b), the convolutional layer output vectors is $[c_1, c_2, ..., c_n]$. Each element of the vector v_i is calculated by a tanh function using each convolution feature c_i in the hidden layer. And the attention weight β_i decides the information of convolution features by a softmax function. Afterwards, the pooling vector C_{β} is computed by a weighted sum of the convolutional layer output. We can compute attentive representation whose output vectors is C_{β} as follows:

$$v_i = tanh(W_c c_i + b_c) \tag{5}$$

$$\beta_i = softmax(W_\beta v_i) \tag{6}$$

$$C_\beta = \sum_{i=1}^{l}(\beta_i c_i) \tag{7}$$

Multiple local representations can be learned by using CNN. The attentive representations are produced by various CNNs through a max function, and are then fed into the model to obtain the final pooling feature vector. And Eq. (8) shows that the representation S_β of a sentence of questions and answers, where k is the length of the convolution window.

$$S_\beta = argmax(C_\beta k) \tag{8}$$

4 Experiments

Our experiments are performed on Windows platform with the memory of 16 GB and the program is written in Python. This paper evaluates the performance of our model for dataset identification. Since the dataset contains a lot of Chinese, we use Jieba to segment the data and train the word vector. The configuration, results and analysis details of the experiments are as follows.

4.1 Datasets

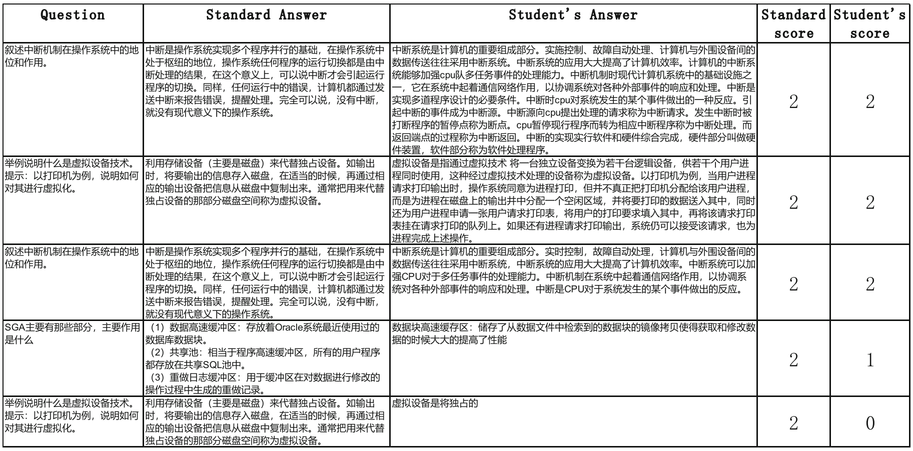
The exam question dataset contains 1669 pairs of short answer questions and answers. Each short answer question corresponds to the standard answer and the student answer, and there is a corresponding score. In the experiment, the dataset is subjected to word segmentation and the spaces and punctuation were removed. The data is randomly divided into 1335 training examples and 334 verification samples. The data in Fig. 3 is a part of the dataset.

4.2 Baselines

This article compares our model with several traditional methods for calculating answer scores as follows. This paper selected four machine learning baselines, including SVMs, CNNs, BLSTMs, and Attention-BLSTMs [9,19].

4.3 Experimental Settings

For all the experiments, we use Jieba to preprocess the dataset, including word segmentation, stop words, word vector matrix, and initialize word representation in the word embedding layer with the 300-dimensional word vectors pre-trained from the dataset. Embeddings for word that are not presented in the model are randomly initialized. In our dataset, the full score of the standard answer is 2 points. So we classify the candidate according to the possible score of the



Question	Standard Answer	Student's Answer	Standard score	Student's score
叙述中断机制在操作系统中的地位和作用。	中断是操作系统实现多个程序并行的基础，在操作系统中处于枢纽的地位，操作系统任何程序的运行切换都是由中断处理的结果，在这个意义上，可以说中断才会引起运行程序的切换。同样，任何运行中的错误，计算机都通过发送中断来报告错误，提醒处理。完全可以说，没有中断，就没有现代意义下的操作系统。	中断系统是计算机的重要组成部分。实施控制、故障自动处理、计算机与外围设备间的数据传送往往采用中断系统。中断系统的应用大大提高了计算机效率。计算机的中断系统能够加强cpu队多任务事件的处理能力。中断机制时现代计算机系统中的基础设施之一，它在系统中起着通信网络作用，以协调系统对各种外部事件的响应和处理。中断是实现多道程序设计的必要条件。中断时cpu对系统发生的某个事件做出的一种反应。引起中断的事件成为中断源。中断源向cpu提出处理的请求称为中断请求。发生中断时被打断程序的暂停点称为断点。cpu暂停现行程序而转为相应中断程序称为中断处理。而返回端点的过程称为中断返回。中断的实现实行软件和硬件综合完成，硬件部分叫做硬件装置，软件部分称为软件处理程序。	2	2
举例说明什么是虚拟设备技术。提示：以打印机为例，说明如何对其进行虚拟化。	利用存储设备（主要是磁盘）来代替独占设备。如输出时，将要输出的信息存入磁盘，在适当的时候，再通过相应的输出设备把信息从磁盘中复制出来。通常把用来代替独占设备的那部分磁盘空间称为虚拟设备。	虚拟设备是指通过虚拟技术 将一台独立设备变换为若干台逻辑设备，供若干个用户进程同时使用，这种经过虚拟技术处理的设备称为虚拟设备。以打印机为例，当用户进程请求打印输出时，操作系统同意为进程打印，但并不真正把打印机分配给该用户进程，而是为进程在磁盘上的输出井中分配一个空闲区域，并将要打印的数据送入其中，同时还为用户进程申请一张用户请求打印表，将用户的打印要求填入其中，再将该请求打印表挂在请求打印的队列上。如果还有进程请求打印输出，系统仍可以接受该请求，也为进程完成上述操作。	2	2
叙述中断机制在操作系统中的地位和作用。	中断是操作系统实现多个程序并行的基础，在操作系统中处于枢纽的地位，操作系统任何程序的运行切换都是由中断处理的结果，在这个意义上，可以说中断才会引起运行程序的切换。同样，任何运行中的错误，计算机都通过发送中断来报告错误，提醒处理。完全可以说，没有中断，就没有现代意义下的操作系统。	中断系统是计算机的重要组成部分。实时控制，故障自动处理，计算机与外围设备间的数据传送往往采用中断系统，中断系统的应用大大提高了计算机效率。中断系统可以加强CPU对于多任务事件的处理能力。中断机制在系统中起着通信网络作用，以协调系统对各种外部事件的响应和处理。中断是CPU对于系统发生的某个事件做出的反应。	2	2
SGA主要有那些部分，主要作用是什么	（1）数据高速缓冲区：存放着Oracle系统最近使用过的数据库数据块。 （2）共享池：相当于程序高速缓冲区，所有的用户程序都存放在共享SQL池中。 （3）重做日志缓冲区：用于缓冲区在对数据进行修改的操作过程中生成的重做记录。	数据块高速缓存区：储存了从数据文件中检索到的数据块的镜像拷贝使得获取和修改数据的时候大大的提高了性能	2	1
举例说明什么是虚拟设备技术。提示：以打印机为例，说明如何对其进行虚拟化。	利用存储设备（主要是磁盘）来代替独占设备。如输出时，将要输出的信息存入磁盘，在适当的时候，再通过相应的输出设备把信息从磁盘中复制出来。通常把用来代替独占设备的那部分磁盘空间称为虚拟设备。	虚拟设备是将独占的	2	0

Fig. 3. Examples of questions and answers from the dataset.

student answer, then we will divide the possible score into 3 categories: 0, 1, 2. Parameter details are listed in Table 1. During the process of training, we do not update the pre-trained word embeddings. We choose the model which works best on the train set, and then evaluate it on the validation set. This paper uses adaptive estimation for optimization. The backpropagation algorithm is used to calculate the gradient of all parameters during training.

Table 1. The experimental parameter settings.

Hidden layer size	Learning rate	Decay rate	Dropout rate	Kernel size	Batch size	Epochs
200	0.01	0.8	0.3	3	64	100

4.4 Results and Analysis

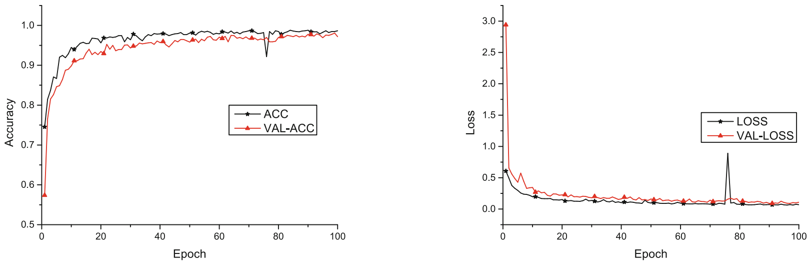
For the HANs model, the questions are entered into the model in chronological order and their parameters are the same. Table 2 compares HANs in this paper with other state-of-the-art answer score methods.

Accuracy, Precision, Recall, and F1 are used to evaluate the performance of the proposed model. Four metrics are used to evaluate the quality of each model. Our model performs very well and training takes about 30 min at a time, which indicates that the model in this paper is very effective in improving learning ability. This is because we uses a word-level mixed attention mechanism to increase the weight of meaningful words in the answer to take into account local information and summary information.

Table 2. Comparison of experimental results.

Method	Accuracy	Precision	Recall	F1
SVMs	0.7377	0.3623	0.1276	0.2332
CNNs	0.8978	0.9023	0.8892	0.8965
BLSTMs	0.9074	0.9215	0.9024	0.9081
Att-BLSTMs	0.9086	0.9190	0.9056	0.9067
This work	**0.9697**	0.9703	0.9687	0.9694

As can be seen from Table 2, we use the model to learn different classifiers based on training data, and the proposed model performs better than the other four models. Figure 4 shows the accuracy and loss of the test set and validation set for 100 periods in the HANs model. We can see that the HANs model achieves the highest prediction accuracy and the lowest loss.



(a) Accuracy

(b) Loss Function

Fig. 4. Accuracy and Loss Function of Train Set and Validation Set for 100 epoch times. Acc and Loss indicate train set accuracy and loss function, Val-Acc and Val-Loss indicate validation set accuracy and loss function.

5 Conclusion

This paper proposes a new neural network model called HANs for Automatic Short Answer Scoring. CNN is used to obtain better local information and BLSTM to focus the model on the information related to the answer, which is encouraged by the attention-based neural network model to pay attention to the words surrounding its similarity. Then, the output of CNN and BLSTM get better results. This paper tests our model HANs on the dataset and obtains an accuracy of 0.9697, which reveals that HANs is efficient and has competitive performance compared to other models.

References

1. Alikaniotis, D., Yannakoudakis, H., Rei, M.: Automatic text scoring using neural networks, pp. 715–725 (2016)
2. Attali, Y., Burstein, J.: Automated essay scoring with e-rater®; vol 2.0. J. Technol. Learn. Assess. 4(2), i–21 (2006)
3. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. Comput. Sci. (2014)
4. Chen, Y.Y., Liu, C.L., Chang, T.H., Lee, C.H.: An unsupervised automated essay scoring system. IEEE Intell. Syst. **25**(5), 61–67 (2010)
5. Chorowski, J., Bahdanau, D., Serdyuk, D., Cho, K., Bengio, Y.: Attention-based models for speech recognition. Comput. Sci. **10**(4), 429–439 (2015)
6. Dong, F., Zhang, Y.: Automatic features for essay scoring - an empirical study. In: Conference on Empirical Methods in Natural Language Processing (2016)
7. Farra, N., Somasundaran, S., Burstein, J.: Scoring persuasive essays using opinions and their targets. In: NAACL 2015 Workshop on Innovative Use of NLP for Building Educational Applications (2015)
8. Graves, A.: Generating sequences with recurrent neural networks. Comput. Sci. (2013)
9. Graves, A., Mohamed, A.R., Hinton, G.: Speech recognition with deep recurrent neural networks **38**(2003), 6645–6649 (2013)
10. Johnson, R., Zhang, T.: Effective use of word order for text categorization with convolutional neural networks. Eprint Arxiv (2014)
11. Klebanov, B.B., Flor, M.: Word association profiles and their use for automated scoring of essays. In: Proceedings of Annual Meeting of the Association for Computational Linguistics ACL, pp. 1148–1158 (2013)
12. Mcnamara, D.S., Crossley, S.A., Roscoe, R.D., Allen, L.K., Dai, J.: A hierarchical classification approach to automated essay scoring. Assessing Writ. **23**, 35–59 (2015)
13. Project Essay Grade: Project essay grade, peg (2003)
14. Pang, L., Lan, Y., Guo, J., Xu, J., Wan, S., Cheng, X.: Text matching as image recognition (2016)
15. Somasundaran, S., Burstein, J., Chodorow, M.: Lexical chaining for measuring discourse coherence quality in test-taker essays, Martin (2014)
16. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks **4**, 3104–3112 (2014)
17. Taghipour, K., Ng, H.T.: A neural approach to automated essay scoring. In: Conference on Empirical Methods in Natural Language Processing (2016)
18. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: Hierarchical attention networks for document classification. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1480–1489 (2017)
19. Zhou, X., Wan, X., Xiao, J.: Attention-based LSTM network for cross-lingual sentiment classification. In: Conference on Empirical Methods in Natural Language Processing, pp. 247–256 (2016)
20. Zhou, Y., Li, C., Xu, B., Xu, J., Cao, J.: Hierarchical Hybrid Attention Networks for Chinese Conversation Topic Classification (2017)
21. Zhou, Y., Xu, J., Cao, J., Xu, B., Li, C.: Hybrid attention networks for chinese short text classication. Computacion Y Sistemas **21**(4), 759–769 (2017)