International Journal of Computer Science and Software Engineering (IJCSSE), Volume 8, Issue 6, June 2019
ISSN (Online): 2409-4285 www.IJCSSE.org Page: 127-132

An Attention-based Deep Model for Automatic Short Answer Score

Tuanji Gong[1] and Xuaxia Yao[2]

[1, 2] School of Computer & Communication Engineering, University of Science and Technology Beijing, Beijing, China

[1]gongtuanji@foxmail.com, [2]yaoxuanxia@126.com

ABSTRACT

In traditional education scenario, scoring assignment is done by human teachers and is time-consuming and laborious. In online education scenario, it is impracticable by human scoring, so automatic scoring is applied broadly. There are two categories of scoring questions: one is short text answer with reference answer, and the other is essay scoring without reference answer. In this work we focus on the former. Existed automatic scoring methods for short text answer used handy craft features which suffered from low accuracy. In order to overcome the problem, we propose a deep learning based method with attention mechanism to automatic scoring for assignment or exam on online education scenario. The method combines pre-trained embedding word vector and RNN model with attention to learn answer vector, and then learned response answer vector and reference answer vector are fed into logistic regression model to predict response answer's score. Experimental results show that our proposed model achieves a relative 10% increase in performance compared with baseline model.

Keywords: *Automatic Scoring, Bidirectional RNN, Attention Mechanism, Word Embedding.*

1. INTRODUCTION

Assignments and exams play a key role in the learning process and are becoming popular as they can consulate knowledge and assist retention[1]. Normal assignments and exams are made up of many question types from multiple choice questions to free response. Despite multiple choice question is a common solution, short text and essay questions have greater educational benefit. Assessment for short answer and essay questions is much more difficult than multiple choice question[2]. In online education scenarios, a lot of assignments and exams are done by students in online manner every day, automatic scoring is an import and difficult task and is an active research topic.

Traditional scoring assignments and exams are the tasks of teachers. However, human scoring is time-consuming and labor-intensive, and suffers from following shortcomings:1) need long waiting time to respond to students; 2) need plenty of human cost; and 3) it is difficult to update performance of students to questions. For online education scenario, because of abovementioned drawbacks, human scoring is infeasible and costly, so an automatic scoring method is required to solve the issue.

In order to alleviate the issue, many automatic scoring methods or systems have been proposed and developed [3, 6]. Automatic answer scoring mainly pays attention on short text answer and essay questions. The difference between short answer and essay question is that the former normally has a reference answer and the latter does not. Short answer questions are widely preferred assessment type due to their effectiveness on developing cognitive skills of students and demonstrating knowledge in short text answer. In this paper we focus on short answer score. An automatic short answer scoring system is that it automatically assigns a score to an answer responded by a student through a comparison with one or more reference answers. Early automatic short answer marking systems select proper features and train classification model or regression model to predict answer's scores. Most of researches employed supervision machine learning technique for automatic short answer marking[3, 5]. Some methods used unsupervised methods to cluster responded answers[7, 8] and students' feedback[9]. These traditional methods used handcraft features and suffered from time-consuming and low accuracy.

Many deep learning based automatic scoring methods are introduced and proposed[10-14],which employed CNN,RNN, or attention mechanism[15] to learn embedding representation of essays or answers and achieved the state of the art performance. Although existing methods achieved high performance, it still remain improving space with newest technique. In the past year, pre-trained word embedding representations have an important breakthrough and have been applied successfully in various NLP tasks, such as question answering, textual entailment and sentiment analysis[16, 17]. Attention mechanism has been widely applied to many fields and attained significant improvement in many NLP tasks from sentiment classification, text summarization to automatic essay grading [10, 14].

In this work, we propose an attention-based deep model for automatic answer marking. The model employs pre-trained word embedding and uses BiRNN with long short term memory(LSTM)[18] unit and attention mechanism to learn answer vector and reference vector. The learned response answer vector and reference answer vector are fed into a linear layer to predict the score.

The rest is organized as follows. Section 2 introduces related work, and we introduce proposed model and experiment in Section 3 and 4 respectively. In Section 5 we draw a conclusion.

2. RELATED WORK

In this section we review related work for automatic short answer scoring. Much research to date has concentrated on two main subtasks of computer assisted assessment: the scoring of essays and short text answer. Pulman et al.[5] introduced an automatic short answer marking method that employs Hidden Markov Model part-of-speech method to extract feature and uses machine learning algorithm, such as decision tree and Bayesian, to predict marking. In[7], the authors explored unsupervised method and introduced a novel technique to improve performance by integrating automatic feedback from the student answers. In [8], Sumit Basu et al. proposed a novel method called Powergrading which first trained a similarity metric between student response answers and then used this metric to group response answers into clusters and sub-clusters. Suzen et al. [9] employed data mining techniques and a clustering approach that group students answers into clusters ,which each cluster is awarded the same score.

Recently deep learning technologies have achieved great success in many fields[19]. Many researchers introduced deep learning based methods to automatic scoring[10-12, 14]. Deep learning based methods normal employed word embedding representation or RNN model [12, 20] or attention-based recurrent convolutional neural network[10, 14] . In [10],the authors employed a convolutional neural network with attention pooling to learning sentence dense presentation and a RNN based LSTM with attention pooling to learning text representation, and used a sigmoid linear layer on RNN to predict essay grading. Alikaniotis et al.[11] adopted bidirectional RNN with LSTM to learn essay representations.

Tay et al.[13] presented a unified deep learning architecture which models relationships between snapshots of the hidden representations of a LSTM network as it reads and the semantic relationships between multiple snapshots as auxiliary features to predict scores.

In this work, we propose an attention-based deep model that integrate word embedding technique, attention mechanism, and bidirectional RNN to score response answer.

3. METHODOLOGY

3.1 Problem Definition

Let N be the number of students, M be the number of questions, a_i^u be the answer that the student u responded the i question. r^u is the corrected answer of question u . $\hat{s}_{ui}$ is the predicted score of answer i of user u. s_{ui} is the ground score by human teachers. The aim is to minimize the difference between predicted score and ground score. The task of automatic short scoring is regarded as a regression problem, and each score is treated as continuous values for each question. Formally, the model is trained to minimize the difference between automatic predicted scores and scores given by human teachers on a dataset:

$$\min \sum_{i=1}^{N} \sum_{j=1}^{M} (s_{ij} - s_{ij}^{'})^2 \qquad (1)$$

3.2 Architecture

In this section we introduce the architecture of the model as illustrated in Fig 1. In this architecture, the input is fed into embedding layer, BiRNN layer, and attention layer to generate embedding representation of answers. The model takes the response answer and reference answer of a question as input and outputs the score for the response answer. In training phase the model takes the corresponding score given by human teacher as extra input. The answer is segmented into sentences and these sentences are fed into sentence embedding layer to learn embedding representations. On the top of RNN component, the attention component is to learn weight of every sentence in a response answer and combines them into the response answer vector. The response answer vector subtracts reference answer vector as the final answer vector. The final answer vector is passed through a line layer with logistic regression function to predict the score ranging from 0 to 1.

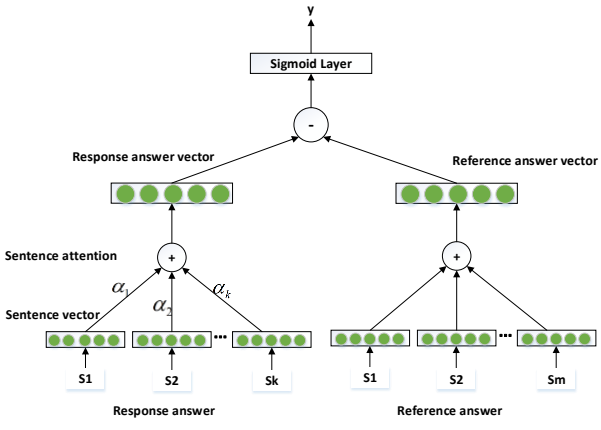

Fig. 1. Architecture of Model

3.3 Sentence Representation

In short free text answer scenario, each answer usually has more than one sentence. In this section, we introduce sentence distribute representation illustrated in Fig 2. Each sentence is tokenized by segment word tool and embedding representation of each token is obtained by lookup tables. All tokens in a sentence are concatenated into two dimensional feature. The feature is fed into BiRNN network with LSTM cell to learn a sentence dense representation. Assuming that a sentence consists of T tokens, $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T$ with $\mathbf{x}_t$ being the embedding feature representation of t-th word, the LSTM cell unit is denoted as the following equations:

$$\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_i \cdot \mathbf{x}_t + \mathbf{U}_i \cdot \mathbf{h}_{t-1} + \mathbf{b}_i) \\
\mathbf{f}_t &= \sigma(\mathbf{W}_f \cdot \mathbf{x}_t + \mathbf{U}_f \cdot \mathbf{h}_{t-1} + \mathbf{b}_f) \\
\tilde{\mathbf{c}}_t &= \tanh(\mathbf{W}_c \cdot \mathbf{x}_t + \mathbf{U}_c \cdot \mathbf{h}_{t-1} + \mathbf{b}_c) \\
\mathbf{c}_t &= \mathbf{i}_t \otimes \tilde{\mathbf{c}}_t + \mathbf{f}_t \otimes \mathbf{c}_{t-1} \\
\mathbf{o}_t &= \sigma(\mathbf{W}_o \cdot \mathbf{x}_t + \mathbf{U}_o \cdot \mathbf{h}_{t-1} + \mathbf{b}_o) \\
\mathbf{h}_t &= \mathbf{o}_t \otimes \tanh(\mathbf{c}_t)
\end{aligned} \tag{2}$$

where $\mathbf{x}_t$ and $\mathbf{o}_t$ are the input word and output word vectors at time t respectively. $\mathbf{W}_i, \mathbf{W}_f, \mathbf{W}_c, \mathbf{W}_o, \mathbf{U}_i, \mathbf{U}_f, \mathbf{U}_c$,and $\mathbf{U}_o$ are weight matrices and $\mathbf{b}_i, \mathbf{b}_f, \mathbf{b}_c$,and $\mathbf{b}_o$ are bias vectors. The symbol $\otimes$ denotes element-wise multiplication and σ is the sigmoid function.

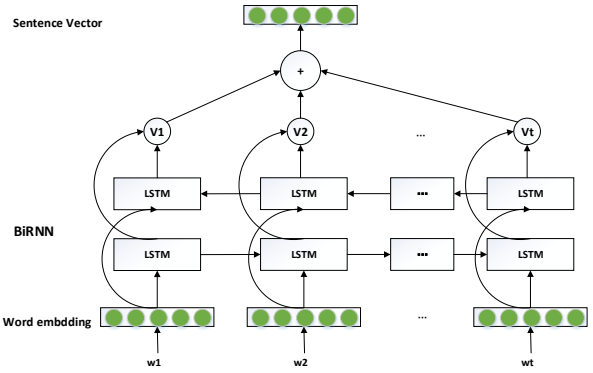

Fig. 2. Sentence embedding model using Bidirectional RNN with LSTM cell.

Following the method in fasttext[21],the embedding vector of every token in a sentence is combined by linear manner to the sentence vector.

3.4 Answer representation

Answer representation consists of two types: reference answer and response answer.

Reference answer vector. We employ weighted sentence features method which linearly combines sentence vectors in a reference answer:

$$u_r = \sum_i w_i s_i \tag{3}$$

where w_i indicates the weight of the i^{th} sentence of reference answer, and s_i is the embedding vector of i^{th} sentence.

As many questions have more than one solutions and we match the responded answer with the best match among reference answers. We pick up the maximum cosine similarity between answer vector and reference vector as the reference answer vector for

$$x_i^r = \arg\max_{x_i^k} \frac{sim(x_i^k, x_{ij})}{\sum_{k=1}^{K} sim(x_i^k, x_{ij})} \tag{4}$$

where $sim(.)$ is the similar function which we use cosine function, x_i^k denotes the k-th solution for i question, and x_{ij} denotes response answer for i question by j student.

Response answer vector

Attention mechanism is widely applied to learn latent representation and we use attention mechanism to learn the response answer representation. The attention layer is shown as follows:

$$\begin{aligned}
e_i &= a(u, v_i) \\
\alpha_i &= \frac{e_i}{\sum_{j=1}^{L} e_j} \\
c &= \sum_{i=1}^{L} \alpha_i v_i
\end{aligned} \tag{5}$$

where $u \in \mathrm{R}^{d_u}$ is the reference answer vector, v_i is the i^{th} sentence vector in the response answer. We adopt dot product $u^T v$ as the function of attention score . α_i is the attention score of the i^{th} sentence vector. c denotes the response answer vector.

3.5 Objective function

The objective function is mean square error with L2 norm regularization as follows:

$$\min_{\Theta} \frac{1}{N} \sum_{i=1}^{N} (y_i - y_i^*)^2 + \lambda \left\| \Theta \right\|^2 \tag{6}$$

$$o_i = c_i - u_i$$
$$y_i^* = \sigma(Wo_i + b) \quad (7)$$

where y_i^* and y_i denote the predicted score and ground true score for i response answer, u_i denotes the embedding representation of reference answer, c_i is the embedding representation which is computed by attention mechanism. $\sigma(.)$ is the sigmoid function. W and b are weight matrix and bias value respectively.

3.6 Training details

We use RMSprop[1] as our optimizer to train the whole model. The initial learning rate η is set to 0.001 and momentum is set to 0.9. Dropout regularization[22] is used to avoid overfitting and drop rate is 0.5. We use Tensorflow framework[23] to train and test the proposed model.

4. EXPERIMENT

4.1 Dataset

In order to evaluate the methods for short answer grading, we have created a data set of short free questions from online education platform for K12. The dataset is collected from online exercise recordings which are marked by human teachers. The scores of responded answers are normalized to [0,1] by $s_{ij}^n = s_{ij} / F_i$, where F_i denotes full mark of the i question, s_{ij} denotes the mark that human teacher score the answer of i question responded by student j, and s_{ij}^n is the normalized mark.

The dataset contains of 8,000 short answer questions and 72,4670 response answers. In those questions, there are more than 900 questions which have two or above two reference answers. The average length of response answer is 4 sentence.

4.2 Baselines

We compare our model with several baseline approaches, including traditional linear regression methods and paragraph embedding method using neural network .

Baseline 1. The text-to-text semantic similarity method is used to compute the similarity between response answer and reference answer. The first baseline model is based on latent semantic analysis(LSA) model introduced in [24]. In LSA, word co-occurrences in a corpus are captured by means of a dimensionality reduction and operated by a singular value decomposition(SVD) on the word-by-answer matrix T representing the corpus. The cosine distance between response answer vector and reference answer vector is regarded as the predicted scoring.

Baseline 2. The answer is split into word or phrase and take the word embedding representation by lookup pre-trained word embedding representations. Like FastText[25], the word vectors are combined into sentence vector by linear operation. The similar metric of response answer and reference answer is measured by cosine similarity. The objective function is to minimize the difference between predicted scores and human's scores, as follow:

$$L(y, y') = \frac{1}{N}\sum_{i=1}^{N}(y_i - y_i^{'})^2 + \lambda \mid \Theta \mid^2$$
$$y_i^{'} = \cos(\mathbf{v}_i, \mathbf{u}_i) = \frac{\mathbf{v}_i \cdot \mathbf{u}_i}{\|\mathbf{v}_i\|\|\mathbf{u}_i\|} \quad (8)$$

where $\mathbf{v}_i, \mathbf{u}_i$ are the response answer vector and reference answer vector respectively, $y_i^{'}$ is the predicted score ranging from [0,1], and y_i indicates the score marked by human teacher.

4.3 Model training

We use HanLP[2] tokenizer to divide the answer into words or phrases, and lookup pre-trained word embeddings dataset trained by word2vec/skip-gram model[26] to get word embedding vector. We obtain the word or phrase embedding by querying pre-trained word embedding representations. In our experiments, the size of word embedding is 300 and GRU dimension is 50.

4.4 Result analysis

We conduct two experiments to compare our model with baseline modes. The first experiment compares accuracy on test dataset with square error metric. We employ mean squared error(MSE) and mean absolute error(MAE) measure metric.

In first experiment, compared with MAE 0.316 of Baseline 1 and 0.29 of Baseline 2, the proposed model achieves 0.242 in MAE measure, which has a relative 11.39% increase with Baseline 2 and 15% than Baseline 1, shown in Table 1.

Table 1: The performance for automatic model score

[1] http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf

[2] https://github.com/hankcs/HanLP

Model	MSE	MAE	MAE Improvement(%)
Baseline 1	0.094	0.316	---
Baseline 2	0.081	0.29	**11.39%**
Proposed model	0.052	0.242	**24.05%**

In second experiment, in order to evaluate the effect of attention mechanism, we modify the proposed method by removing attention component and employing the same method that represents reference answer vector to represent response answer vector. The two models are trained and test on same dataset. The result shows that the performance of model with attention component has a relative 6.2% increase than that without attention mechanism. The result shows that attention mechanism is helpful to performance improvement.

Table 2: The performance for automatic model score

Model	MSE	MAE	MAE Improvement(%)
PM(no attention)	0.063	0.258	---
Proposed model	0.052	0.242	6.2%

5. CONCLUSION AND FUTURE WORK

In this paper, we investigate and propose an attention based model for automatic short answer scoring. The model uses bidirectional RNN to learn sentence embedding representations of short answer and employs attention mechanism to learn response answer vector. The response answer vector and reference answer vector are fed into linear layer and logistic regress layer to predict score. Experimental results shows that our model achieve more than 10% increase. Future work will explore automatic scoring model based on Transformer mechanism.

REFERENCES

[1] R. C. Anderson and W. B. Biddle, "On Asking People Questions about What They are Reading 1," Psychology of Learning & Motivation, vol. 9, pp. 89-132, 1975.
[2] C.R. Reynolds,R.B. Livingston, and V. L. Willson, "Measurement and assessment in education," Measurement & Assessment in Education, vol. 12, pp. 109-140, 2006.
[3] C. Leacock and M. Chodorow, "C-rater: Automated Scoring of Short-Answer Questions," Computers & the Humanities, vol. 37, pp. 389-405, 2003.
[4] R. Siddiqi and C. Harrison, "A systematic approach to the automated marking of short-answer questions," in IEEE International Multitopic Conference, 2009.
[5] S. G. Pulman and J. Z. Sukkarieh, "Automatic short answer marking," in Proceedings of the second workshop on Building Educational Applications Using NLP, 2005, pp. 9-16.
[6] S. Burrows, I. Gurevych, and B. Stein, "The eras and trends of automatic short answer grading," International Journal of Artificial Intelligence in Education, vol. 25, pp. 60-117, 2015.
[7] M. Mohler and R. Mihalcea, "Text-to-text semantic similarity for automatic short answer grading," in Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, 2009, pp. 567-575.
[8] S. Basu, C. E. Jacobs, and L. Vanderwende, "Powergrading: A Clustering Approach to Amplify Human Effort for Short Answer Grading," meeting of the association for computational linguistics, vol. 1, pp. 391-402, 2013.
[9] N. Suzen, A. N. Gorban, J. Levesley, and E. M. Mirkes, "Automatic Short Answer Grading and Feedback Using Text Mining Methods," arXiv: Computation and Language, 2018.
[10] F. Dong, Y. Zhang, and J. Yang, "Attention-based Recurrent Convolutional Neural Network for Automatic Essay Scoring," in Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), 2017, pp. 153-162.
[11] D. Alikaniotis, H. Yannakoudakis, and M. Rei, "Automatic text scoring using neural networks," arXiv preprint arXiv:1606.04289, 2016.
[12] K. Taghipour and H. T. Ng, "A neural approach to automated essay scoring," in Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 1882-1891.
[13] Y. Tay, M. C. Phan, L. A. Tuan, and S. C. Hui, "SKIPFLOW: Incorporating Neural Coherence Features for End-to-End Automatic Text Scoring," arXiv preprint arXiv:1711.04981, 2017.
[14] Y. Liu, Q. Ai, J. Guo, and W. B. Croft, "aNMM: Ranking Short Answer Texts with Attention-Based Neural Matching Model," in Acm International on Conference on Information & Knowledge Management, 2016.
[15] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," international conference on learning representations, 2015.
[16] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, et al., "Deep contextualized word representations," 2018.
[17] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2018.
[18] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, pp. 1735-1780, 1997.
[19] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, pp. 436-444, 2015.
[20] D. Alikaniotis, H. Yannakoudakis, and M. Rei, "Automatic Text Scoring Using Neural Networks," 2016.
[21] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of Tricks for Efficient Text Classification," pp. 427-431, 2016.
[22] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from

overfitting," Journal of Machine Learning Research, vol. 15, pp. 1929-1958, 2014.
[23] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," arXiv: Distributed, Parallel, and Cluster Computing, 2016.
[24] T. K. Landauer and S. T. Dumais, "A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge," Psychological Review, vol. 104, pp. 211-240, 1997.
[25] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jegou, and T. Mikolov, "FastText.zip: Compressing text classification models," arXiv: Computation and Language, 2017.
[26] S. Li, Z. Zhao, R. Hu, W. Li, T. Liu, and X. Du, "Analogical Reasoning on Chinese Morphological and Semantic Relations," arXiv preprint arXiv:1805.06504, 2018.