An automatic short-answer grading model for semi-open-ended questions

Lishan Zhang, Yuwei Huang, Xi Yang, Shengquan Yu & Fuzhen Zhuang

To cite this article: Lishan Zhang, Yuwei Huang, Xi Yang, Shengquan Yu & Fuzhen Zhuang (2019): An automatic short-answer grading model for semi-open-ended questions, Interactive Learning Environments, DOI: 10.1080/10494820.2019.1648300

To link to this article: https://doi.org/10.1080/10494820.2019.1648300

Published online: 30 Jul 2019.
An automatic short-answer grading model for semi-open-ended questions

Lishan Zhang, Yuwei Huang, Xi Yang, Shengquan Yu and Fuzhen Zhuang

ABSTRACT

Automatic short-answer grading has been studied for more than a decade. The technique has been used for implementing auto assessment as well as building the assessor module for intelligent tutoring systems. Many early works automatically grade mainly based on the similarity between a student answer and the reference answer to the question. This method performs well for closed-ended questions that have single or very limited numbers of correct answers. However, some short-answer questions ask students to express their own thoughts based on various facts; hence, they have no reference answers. Such questions are called semi-open-ended short-answer questions. Questions of this type often appear in reading comprehension assessments. In this paper, we developed an automatic semi-open-ended short-answer grading model that integrates both domain-general and domain-specific information. The model also utilizes a long-short-term-memory recurrent neural network to learn the representation in the classifier so that word sequence information is considered. In experiments on 7 reading comprehension questions and over 16,000 short-answer samples, our proposed automatic grading model demonstrates its advantage over existing models.

ARTICLE HISTORY

Received 18 July 2019
Accepted 23 July 2019

KEYWORDS

Machine learning; short-answer grading; text analysis; classification; auto-grading

1. Introduction

Reading comprehension questions are commonly used to assess students’ understanding of reading materials. Multiple-choice and short-answer reading comprehension questions are two typical types. Multiple-choice questions ask students to recognize the correct answer among several alternatives, while short-answer questions ask students to write answers in several phases or sentences. Generating good alternatives for multiple choice questions is non-trivial (Zhang & VanLehn, 2019). According to Chi’s ICAP framework (Chi & Wylie, 2014), practicing with short-answer questions should also provide more benefits for students’ learning than with multiple-choice questions. However, short answers cannot be autograded directly and manual grading is time-consuming and labor-intensive. Natural language processing (NLP) technology has been extensively applied to the autograding issue. Different from essay grading, which also requires dealing with NLP issues, the length of a short answer ranged from one phrase to one paragraph and the grading criteria focus on the content instead of the style (Higgins et al., 2004; Leacock & Chodorow, 2003; Mohler & Mihalcea, 2009; Pulman & Sukkarieh, 2005).
Typical short-answer questions are closed-ended, namely, the correct answers are expected to match only several facts, which are also called as expectations, that are specified by the question authors. Short-answer questions of this type require students to explain objective facts without expressing their subjective opinions, such as the question below from Zhang, Shah, and Chi (2016).

**Question:** Why are there no potential energies involved in this problem?

**Correct answer:** Because the rock is the only object in the system, there are no potential energies involved.

However, some short-answer questions require students to express their subjective opinions based on a specified context. In this case, it is impossible to list all the expectations of the questions in advance. Hence, student answers must be graded based on a general grading rubric. On the other hand, questions of this type are not completely open-ended either, because students are expected to list specific facts and express their subjective opinions based on the facts. Therefore, we treat these short-answer questions as semi-open-ended questions. Many reading comprehension short-answer questions are semi-open-ended. The main goal of this paper is to propose and evaluate a model that can automatically grade these semi-open-ended short-answer questions. A semi-open-ended short-answer sample question is given below. It is a reading comprehension question with the reading material omitted. The question was originally written in Chinese, but translated into English.

**Question:** Based on the sentence “his lasting appeal of yo-heave-ho has become a kind of cultural landscape of Gumiao town” and your life experience, please explain your understanding about the “yo-heave-ho” of King of sweet ferment rice.

**Sample of correct answer:** I think his yo-heave-ho shows diligent and intangible enthusiasm among people. In our life, vendors’ shouting along street and their insistence on trading no matter how the weather is and what the season can make people feel warm-hearted in everyday life is.

The first sentence of the sample answer explains the meaning of “yo-heave-ho” in the article, which reflects warm-heated relationships among neighbors. The second sentence refers to an example in the student’s personal life that also reflects a relationship of this type. To answer this question correctly, a student must accurately explain the meaning of “yo-heave-ho” and provide a similar example from her personal experience. The first part of a correct answer is highly predictable, but the latter part is flexible. This again explains why this type of question is called semi-open-ended. This characteristic makes autograding difficult.

The remainder of this article is organized as follows: We first review the purpose of automatic grading and its common technologies. Then, we introduce our automatic grading model and evaluate the model with 7 reading comprehension questions and over 16,000 labeled student answers. Last, we present the discussion and our conclusions.

### 2. Literature review

#### 2.1. Purpose of automatic grading

There are two general purposes for implementing automatic grading models: for large-scale assessment and as part of the assessor in an intelligent tutoring system. The Educational Testing Service (ETS) is an industry initiator that pushes forward automatic grading of both short answers (c-rater) and essays (e-rater) (Attali & Burstein, 2004; Leacock & Chodorow, 2003). The ultimate goal of these raters is to accurately classify all student answers into the appropriate categories according to a grading rubric. In contrast, academic researchers are more interested in implementing automatic grading models for instructional purposes and adaptive learning (Luo & Litman, 2016; Zhang & VanLehn, 2017). For example, automatic grading models can be used to build conversational intelligent tutoring systems like AutoTutor, which is driven by expectation and misconception tailored (EMT) dialogue (Graesser et al., 2016; Shi et al., 2018). In each conversational turn, the EMT dialogue requires an assessor to automatically classify student answers into a list of expectations and
misconceptions, so that the intelligent agent knows how to respond (Nye, Graesser, & Hu, 2014). The
goal of the assessor is consistent with that of an automatic grader. They both tried to classify stu-
dents’ text answers into multiple classes, either grades or predefined expectations. Besides these
two general purposes, the underlying text analysis techniques of automatic grading can also
provide new potentials for intelligent tutoring (Liu et al., 2019; Psotka & Chen, 2019).

2.2. Similarity-based automatic grading

As claimed by Burrows, Gurevych, and Stein (2015), short-answer questions typically have one or
several clear correct answers that can be used as the reference and automatic grading can be con-
ducted by calculating the similarity between a student answer and the reference answer (Graesser
et al., 2004). This grading method has been used in intelligent tutoring systems in many domains
(Azevedo et al., 2012; Dzikovska, Farrow, & Moore, 2013; Mcnamara et al., 2007; Rus & Graesser,
2006). Latent semantic analysis and latent Dirichlet allocation (Blei, Ng, & Jordan, 2003) are typically
used to build topic vectors that represent student answers and reference answers that quantify simi-
larities. Of all the similarity calculations, the cosine distance is one of the most common measures.
From the perspective of machine learning, this automatic grading method is similar to the k-
nearest neighbor algorithm, except that only answers with positive labels (correct answers) are typi-
cally provided. Automatic grading algorithms of this type are called similarity-based grading methods
in this paper. The largest advantage of this method is that it requires almost no training data set.
Several sample correct answers are sufficient for a question. However, the grading algorithm often
requires a much larger dataset to calibrate the similarity calculation between students’ answers
and reference answers. The main disadvantage is that a similarity-based grading method can only
recognize the variations of sample reference answers; all other answers will be treated as incorrect
answers. Hence, it is important to ask the question in an appropriate way to ensure that correct
student answers are highly predictable.

2.3. Automatic grading via a machine learning approach

With many data sets of labeled short answers available, researchers have started to build automatic
grading models by using statistical machine learning algorithms. The development of a grading
model via a machine learning approach can be divided into two main steps: feature engineering
and classification. Feature engineering is used to extract features from student answers. It essentially
defines the information that a classification algorithm can consider in making decisions. Most classifi-
cation algorithms treat student answers as a bag of words without considering the word sequence,
such as linear regression (Madnani et al., 2013), support vector machine (SVM) (Hou et al., 2010), and
deep belief network (DBN) (Zhang et al., 2016). In our proposed model, a long short-term memory
recurrent neural network (LSTM) is used to consider word sequence information. Regardless of
which classification algorithm is used, the reference answers become optional in these approaches.
However, machine learning grading methods require a much bigger set of labeled student answers
than similarity-based methods so that automatic grading models can be learned from the data.
Each of the existing machine learning grading methods follows one of two general approaches: (1)
Learn a question general grading model that can grade various questions in the same domain (Bailey
& Meurers, 2008; Gütl, 2008; Nielsen, Ward, & Martin, 2008; Zhang et al., 2016). (2) Learn a question
specific grading model for each question (Luo & Litman, 2016; Madnani et al., 2013; Yang, Zhang, &
Yu, 2017). These two approaches have led to different feature engineering strategies.
While question general grading models are being built, the extracted student answer features
must be question general as well. Hence, the reference answer, which is responsible for providing
question specific information, becomes an essential part of the features. Similarities or differences
between student and reference answers are the most common features. Other typical question
general features are the length of the student answers, question difficulty, and student competency.
Therefore, a machine learning question general grading model is similar to an extended version of a similarity-based grading model.

In contrast, a question specific grading model typically does not require a reference answer, but directly learns the grading mechanism from the labeled student answers. However, this method requires many labeled student answers to train each question specific grading model. Collecting sufficiently manually labeled student answers is a key issue. The size of the set of labeled student answers, which is also called training dataset, may hinder the autograding performance. Since our reading comprehension questions are semi-open-ended and, thus, have no reference answers, we must develop a question specific grading model that is inferred from labeled student answers.

3. Proposed model

In developing a question specific autograding model, the size of the training data set may impede model performance. However, a human can learn how to grade from only a limited number of labeled student answers. This is because a human has a large amount of domain-general knowledge and is able to quickly learn the grading criteria by adapting her domain-general knowledge to the question. This has inspired machine learning researchers to develop many transfer learning algorithms for integrating domain-general and domain-specific information to improve classification accuracy (Pan & Yang, 2010). In our proposed model, domain-general and domain-specific information are integrated in the process of feature engineering. The domain-general information is extracted from Wikipedia and the domain-specific information is the set of labeled student answers.

As discussed above, existing autograding models often neglect word order information in student answers. In our proposed model, LSTM is used to encode word sequence information while building the classifier.

By improving both the feature engineering and the classification algorithm, we aim at answering two research questions:

- Does the integration of domain-general and domain-specific information help improve automatic grading performance?
- Do classifiers that incorporate word sequence information, such as LSTM, outperform classical classifiers that do not consider word sequence information?

The framework of our model is illustrated in Figure 1. In the remainder of this section, we describe how we integrated domain-general and domain-specific information in our feature engineering (left part in Figure 1) and how LSTM was used in our classification algorithm (right part in Figure 1).

3.1. Word embedding in student answers as the features

As Zhang et al. (2016) claimed, the features of an automatic grading model can come from the student’s competency, the question, and the answer. But student’s competency is not always available, especially in the context of assessment. This type of feature is not used in our feature engineering approach.

Words are commonly represented as embedding vectors such that words with similar meanings are close to each other in the vector space. LSA and LDA are the most common methods for transforming words into topic vectors. Recently, a new algorithm, namely, the continuous bag-of-words model (CBOW), was developed, which is based on deep learning technology. CBOW outperforms LSA on preserving linear regularities among words (Mikolov et al., 2013; Zhila et al., 2013). This model is used in our feature engineering approach.
3.1.1. CBOW model

The CBOW model essentially tries to predict the vector of each word based on its context words. It is a three-layered neural network that includes an input layer, projection layer and output layer. The model is illustrated in Figure 2. A brief description of the three layers is given below.

Input layer: For each target word $w_k$ in a sentence, there are $2c$ context words, including $c$ precedent words, which are denoted as $w_{k-c}, \ldots, w_{k-2}, w_{k-1}$, and $c$ posterior words $w_{k+1}, w_{k+2}, \ldots, w_{k+c}$. Each word is initialized as a dimensional vector with a random value. The $2c$ context words are the input of the target word $w_k$. Their values are updated during the learning process.

Figure 1. Automatic grading model overview.

Figure 2. CBOW Model.
Projection layer: Calculate the average of all the vectors from the input layer. The average is treated as the vector of the context information of the target word $w_k$. The formula is shown as (1)

$$v_k = \frac{1}{2c} \sum_{j=k-c, j \neq k}^{k+c} v_i$$

where $v_i$ is the vector of the context word $w_i$.

Output layer: The predicted $v_k$ should be similar to the actual $v_k$, but dissimilar to the vectors of other words, which form the negative set. The CBOW model tries to update the value of each target word to minimize the loss function in (2)

$$loss_{CBOW} = \log \prod_{k=1}^{D} \left\{ \sigma(v_k^T \theta^k) \prod_{j=1}^{|Neg_k|} [1 - \sigma(v_k^T \theta^j)] \right\}$$

where $D$ is the entire word vocabulary of the corpus, $Neg_k$ is a sample of the negative set, $\sigma(v_k^T \theta^k)$ is the likelihood that the predicted vector of the target word is its actual vector, and $\sigma(v_k^T \theta^j)$ is likelihood that the predicted vector is the vector of another word.

### 3.1.2. Integration of domain-general information with domain-specific information

Both domain-general information and domain-specific information are used to generate word vectors. Two training steps are implemented: In the first training step, the CBOW model is trained with Wikipedia data. At the beginning of the training, the word vector values are randomly initialized. After the corresponding loss function has been optimized, the second training step begins. Then, the CBOW model is trained with collected student answers. The main difference is that the initial word vector values are transferred from the result of the first step. Because the training data, namely, Wikipedia data, do not vary among questions, we only need to run the first training step once. The training data for the second step, namely, the set of student answers, vary among questions. Hence, the second training step must be run for each question to be autograded.

### 3.2. Classifier

We used LSTM (Gers, Schmidhuber, & Cummins, 1999) to learn the representation and softmax as our classifier to train the grading model. LSTM is a well-known recurrent neural network model in natural language processing. The largest advantage of this model is that it uses a memory node to consider the word sequence.

![LSTM model](image-url)
The architecture of a unit in LSTM is illustrated in Figure 3. There are three gates, namely, the input, output, and forget gates, which control how the state is stored in the memory cell. Functions \( i_t, o_t, \) and \( f_t \) represent the activations of the input, output and forget gates, respectively, and \( c_t \) is the activation of the memory cell at step \( t - 1 \), which is used to retain the information in the previous step. Vectors \( x_t \) and \( h_t \) are the input and output vectors, respectively, at step \( t \). The activation functions are shown in (3). A softmax layer is used to combine the outputs of all the LSTM cells to predict the score of a student answer, as illustrated in Figure 3.

\[
\begin{align*}
    i_t &= \sigma(W_ix_t + U_i h_{t-1} + b_i) \\
    f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f) \\
    o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o) \\
    c_t &= i_t \cdot \tanh(W_c x_t + U_c h_{t-1} + b_c) + f_t \cdot c_{t-1} \\
    h_t &= o_t \cdot \tanh(c_t)
\end{align*}
\]

where \( W_i, W_f, W_o, \) and \( W_c \) are the weight matrices of the input vector; \( U_i, U_f, U_o, \) and \( U_c \) are the weight matrices of the output vector in the previous step; and \( b_i, b_f, b_o, \) and \( b_c \) are bias vectors.

During the training phase, for each student answer, the classifier outputs the probability of each possible score; these probabilities compose vector \( s_i \). The human-labeled score of the answer is represented by the corresponding vector \( s'_i \), in which the entry that corresponds to the true score is 1 and the remaining entries are 0. The classifier is trained with a cross-entropy loss function, as shown in (4). The Adam algorithm is used to optimize the loss function. After the classifier has been trained, the score with the highest probability is used as the score of a student answer to be graded.

\[
C(s_i, s'_i) = -\frac{1}{n} \sum_{i=1}^{n} [s_i \ln s'_i + (1 - s_i) \ln (1 - s'_i)]
\]

4. Evaluation

To evaluate our proposed model, we conducted experiments on 7 reading comprehension short-answer questions. Those included 5 questions in Chinese and 2 questions in English. Each question contained at least 2000 human graded student answers. In this section, we introduce the data sets and the measures and describe our experimental design and the results.

4.1. Datasets

The 5 reading comprehension questions in Chinese were obtained from the final exams of Grade 8 students. The set of these questions is called the Chinese Reading Comprehension Corpus (CRCC). The first two questions are generated based on two works of fiction that are required by the school syllabus. The remaining three are based on short reading materials that are given in the final exams. The 5 questions are listed in Table 1. We had two experienced teachers grade each

<table>
<thead>
<tr>
<th>ID</th>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRCC 1</td>
<td>Please use your own words to write a recommendation for the fiction “Red Crag”. The recommendation should help the readers grasp the gist of the fiction, and raise their interests</td>
</tr>
<tr>
<td>CRCC 2</td>
<td>“20,000 leagues under the sea” is a fiction. What is the most attractive point of this book to you?</td>
</tr>
<tr>
<td>CRCC 3</td>
<td>We have read “Mr. Bian meets Mr. Cai”, where Mr. Bian failed to convince Mr. Cai. How do these two stories (“Mr. Bian meets Mr. Cai” and the reading material) inspire our interpersonal communication in the current age?</td>
</tr>
<tr>
<td>CRCC 4</td>
<td>Based on the sentence “his lasting appeal of yo-heave-ho has become a kind of cultural landscape of Gumiao town” and your life experience, please explain your understanding about the “yo-heave-ho” of King of sweet ferment rice</td>
</tr>
<tr>
<td>CRCC 5</td>
<td>Mr. Hwang adheres not to extend his business scale, and only sells 30 bottles of rice wine per day. Do you agree with him, and why?</td>
</tr>
</tbody>
</table>
student answer. For each question, they graded 50 student answers together to confirm the grading rubric and individually graded the remaining answers. They discussed with one another to resolve any conflicts. Quadratic-weighted kappa (QWKappa) is used to test their agreement. QWKappa is described in detail in the next section.

The 2 reading comprehension questions in English are from a Kaggle competition. The original Kaggle competition contained 10 questions. However, only two of them are reading comprehension questions. The kappa value is not available for this data set. The 2 questions are listed in Table 2. Table 3 lists the basic statistics of the 7 datasets.

### 4.2. Measures

QWKappa is used to evaluate the agreement among the grades that were assigned by different graders. We adopt QWKappa instead of the typical Cohen’s Kappa because the former is able to capture the scores’ order information. For example, suppose that a question can have up to 3 levels of scores (0,1,2), the first-grader scores an answer as 0, the second-grader scores the same answer as 1, and the third-grader scores the answer as 2. Although both the second and the third-grader disagree with the first-grader, it is clear that the second-grader is more similar to the first-grader than the third grader. The typical Cohen’s Kappa cannot capture this difference, whereas QWKappa can. The details of the calculation of QWKappa can be found on Kaggle’s website.¹

### 4.3. Experiment design

The main objective of the experiment is twofold:

- To test the effectiveness of integrating domain-general and domain-specific information in the grading model;
- To test the effectiveness of our proposed model compared to other existing automatic grading models.

We implemented 7 automatic grading models, including our proposed model. Four out of the 7 grading models extracted features from student answers based on a bag-of-words model. The 4 classifiers are Logistic Regression (LR), which was used in (Madnani et al., 2013); Naïve Bayes (NB), which was used in (Levy et al., 2013); Decision Tree (DT), which was used in (Jimenez, Becerra, & Gelbukh, 2013); and SVM, which was used in (Hou et al., 2010). All 4 classifiers must have a single multi-dimensional vector as their input because these traditional models are unable to take word sequence information into consideration. Each dimension of the input vector represents a word and the value of the dimension is the TF-IDF value of the corresponding word. Hence, the number of dimensions of the input vector is equal to the number of words in the training data set. Only domain-specific information, namely, student answers, is used to train the 4 traditional classifiers. In contrast, the remaining 3 grading models all extracted features based on the CBOW model and respectively use domain-general information only, which is denoted as CBOW-G; domain-specific information only, which is denoted as CBOW-S; and both domain-general and domain-specific information, which is denoted as CBOW-GS.

<table>
<thead>
<tr>
<th>Table 2. Two English reading comprehension questions.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
</tr>
<tr>
<td>----</td>
</tr>
<tr>
<td>Kaggle 1</td>
</tr>
<tr>
<td>Kaggle 2</td>
</tr>
</tbody>
</table>
We performed the same pre-processing step before training for all 7 grading models. We used Jieba to segment the words in the Chinese corpus and scikit-learn to tokenize the English corpus. LR, NB, DT and SVM were implemented with scikit-learn. CBOW-S, CBOW-G and CBOW-GS were implemented with Keras (Chollet, 2015).

QWKappa between the predicted grades and the human resolved grades is calculated for each question. The experiment is conducted with 5-fold cross-validation and the average QWKappa over the 5 folds is used to compare the 7 automatic grading models. We test the following 3 hypotheses:

- **Hypothesis 1.** CBOW-GS outperforms both CBOW-G and CBOW-S because the first model extracts both domain-general and domain-specific information as the features.
- **Hypothesis 2.** CBOW-S and CBOW-G outperform the existing 4 grading models because the former two models take word sequence into consideration.
- **Hypothesis 3.** CBOW-GS performs the best among all 7 grading models.

### 4.4. Results

The performances of the 7 classifiers are detailed in Table 4. Two-tailed paired t-Test was used to test the significance of the difference. However, the performance of a grader is not guaranteed to be normal distributed, and the result reported by Table 4 confirms the view. Therefore, the Wilcoxon signed-rank test was also conducted, which is an alternative to the paired t-Test when the population cannot be assumed to be normally distributed. Table 5 reports both results, because the paired t-Test is more widely used, and the Wilcoxon signed-rank test is more suitable.

**H1:** QWKappa of CBOW-GS is significantly higher than those of CBOW-S and CBOW-G. Hence, hypothesis 1 is confirmed. Integrating both domain specific and domain general information helps improve the autograding performance.

**H2:** CBOW-S did not perform well and was outperformed by two of the bag-of-words grading models. SVM is the best-performing bag-of-words grading model. It significantly outperforms all the others. Although the average QWKappa of CBOW-G exceeds that of SVM, the difference is not significant. Hence, hypothesis 2 is not confirmed. This is probably because the domain general corpus is too sparse. The learned word vectors must be adapted to the domain specific corpus.
Table 5. Pair-by-pair comparison.

<table>
<thead>
<tr>
<th></th>
<th>NB</th>
<th>CBOW_S</th>
<th>CBOW_G</th>
<th>SVM</th>
<th>CBOW_GS</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR</td>
<td>$t = 4.223, p = 0.006$</td>
<td>$t = 3.131, p = 0.020$</td>
<td>$t = -5.030, p = 0.002$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$Z = -2.197, p = 0.028$</td>
<td>$Z = -1.859, p = 0.063$</td>
<td>$Z = -2.366, p = 0.018$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$t = 1.266, p = 0.254$</td>
<td>$t = -6.615, p = 0.001$</td>
<td>$Z = -2.366, p &lt; 0.001$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$Z = -0.845, p = 0.398$</td>
<td>$Z = -2.366, p = 0.018$</td>
<td>$Z = -2.366, p = 0.001$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NB</td>
<td>$t = -2.281, p = 0.063$</td>
<td>$t = -4.977, p = 0.003$</td>
<td>$t = -6.042, p = 0.001$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$Z = -1.859, p = 0.063$</td>
<td>$Z = -2.366, p = 0.018$</td>
<td>$Z = -2.366, p = 0.001$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DT</td>
<td>$t = -0.406, p = 0.699$</td>
<td>$t = -4.763, p = 0.003$</td>
<td>$t = -6.248, p = 0.001$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$Z = -0.676, p = 0.499$</td>
<td>$Z = -2.366, p = 0.018$</td>
<td>$Z = -2.366, p = 0.001$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>$t = 2.417, p = 0.52$</td>
<td>$t = -1.159, p = 0.291$</td>
<td>$t = -4.267, p = 0.005$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$Z = -2.028, p = 0.043$</td>
<td>$Z = -1.183, p = 0.237$</td>
<td>$Z = -2.366, p = 0.018$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CBOW_S</td>
<td>$t = -2.553, p = 0.043$</td>
<td>$Z = -0.028, p = 0.043$</td>
<td>$t = -3.389, p = 0.015$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CBOW_G</td>
<td>$t = -2.371, p = 0.018$</td>
<td>$Z = -2.371, p = 0.018$</td>
<td>$t = -6.767, p = 0.001$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notes: The two-tailed paired t-Test and Wilcoxon signed-rank test are used to compare the results of the autograding models on the 7 questions.

$H3$: CBOW-GS significantly outperformed all the other grading models. Hypothesis 3 is confirmed.

The results demonstrate that both domain-general information and word sequence should be included in the grading model. LR, NB, DT and SVM all use a bag-of-words model in feature engineering. A bag-of-words model builds a dictionary of all the words, where each word is represented as a float value. Then, a student answer is represented as a vector of float values. The sequence of the words is lost. LSTM has the advantage of taking a sequence of word vectors as the input and word vectors can be generated from either domain-specific or domain-general knowledge, or both. Hence, we are able to build a higher-performing grading model than the existing ones by combining CBOW and LSTM.

CBOW-S does not perform well, probably because the domain-specific corpus is not large enough to train reliable word vectors. Indeed, researchers usually train word vectors with large-scale auxiliary data sets. This also explains why CBOW-G performs much better.

We take two questions as examples below to show the advantages of CBOW and LSTM.

**Question 1.** Based on the sentence “his lasting appeal of yo-heave-ho has become a kind of cultural landscape of Gumiaozhao town” and your life experience, please explain your understanding about the “yo-heave-ho” of King of sweet ferment rice.

**Answer 1.1** The “yo-heave-ho” of King of sweet ferment rice reflects that he is rustic and hardworking (勤劳). In my life, my mom calls me up every morning. Her voice is rustic as well. (correct)

**Answer 1.2** The “yo-heave-ho” of King of sweet ferment rice represents his diligence (勤奋). This kind of diligence is worth to be inherited to our age of life (correct).

**Question 2.** We have read “Mr. Bian meets Mr. Cai”, where Mr. Bian failed to convince Mr. Cai. How do these two stories (“Mr. Bian meets Mr. Cai” and the reading material) inspire our interpersonal communication in the current age?

**Answer 2.1** The way of Mr. Bian’s talking is too harsh. In contrast, Mr. Zou (邹忌) (the figure in the reading material) convince his king by analogy, instead of direct criticizing. Therefore, the king accepts Mr. Zou’s suggestion. (correct)
Answer 2.2 The way of Mr. Zou's talking is too harsh. In contrast, Mr. Bian (the figure in the reading material) convince his king by analogy, instead of direct criticizing. Therefore, the king accepts Mr. Zou’s suggestion. (incorrect)

Note: Both the questions and the answers were originally written in Chinese. The most important key words are underlined and their original Chinese key words are given in parenthesis.

The two answers to question 1 demonstrate the advantage of CBOW model when both domain-general and domain-specific information are being used. The bag-of-words based grading model can recognize answer 1.1 as a correct answer due to the existence of the keyword [勤劳], but failed in grading answer 1.2 because the word [勤劳] is replaced with [勤奋]. However, the CBOW model can help recognize that [勤劳] and [勤奋] have similar meaning and grade answer 1.2 correctly as well.

The two answers to question 2 show the advantage of LSTM when the word’s sequence is taken into consideration. Answer 2.1 is correct but answer 2.2 is incorrect because [扁鹊] and [邹忌] are exchanged. Vectors that represent the two answers have no difference when a bag-of-words model is adopted. However, LSTM is able to capture the difference and grade them correctly.

5. Discussion

The automatic grading model has less information for making decisions when a student’s answer is short. It is similar to a topic modeling algorithm that often performs worse when it is used for Tweet analysis compared to article analysis (Mehrotra et al., 2013; Xiang et al., 2012). Additionally, the single reference answer is absent when the question is relatively open. Hence, short answer grading for semi-open-ended questions is a difficult problem. By integrating domain-general and domain-specific information, we significantly improved the performance of our automatic short-answer grading model. Although we only run experiments on Chinese and English reading comprehension short-answer questions, the proposed model can be potentially extended to any short-answer grading task that has no reference answer and many graded student answers.

Our proposed grading model performed differently on the 5 Chinese short-answer questions. The grading model performed poorly on the first two questions, but much better on the remaining three. This is probably because the first two questions asked students to write their answers based on an entire book, whereas the remaining three were based on several paragraphs of text. As a result, the student answers to the first two questions were more diverse. The diversity of the answers increased the grading difficulty. According to Table 3, the numbers of distinct words in the first two questions exceed those of the remaining questions. This is consistent with our conjecture. These differences also reflect the specificity of semi-open-ended short-answer questions compared to closed-ended short-answer questions. Closed-ended short-answer questions elicit specific facts; hence, substantial diversity among the answers is not expected, regardless of whether the answers are derived from an entire book or a short text. For example, a question in computer literacy may ask students to explain how an operating system loads a text file from hard disk to memory. The question covers multiple chapters in a textbook. However, the expectations of the question are specific; hence, the answers are closed-ended. In contrast, students have more freedom to express their own thoughts in answering semi-open-ended questions. Therefore, the diversity of the answers increases with the scope of the question on which they are based. In this case, the corresponding automatic grading model requires more labeled data to capture the diversity.

The largest disadvantage of our proposed model is the requirement of many graded student answers for each target question. As a result, the grading model may be only useful for large-scale applications that may have millions of student answers to a limited number of questions. For example, an instructor should consider similarity-based grading methods instead of our proposed model if she wants to autograde only one class of students. However, intelligent tutoring systems...
or learning assistant system designers should consider integrating our proposed model into their system as an assessor (Sottilare et al., 2018; Vanlehn, 2006), if the tutoring system is going to benefit hundreds of thousands of students. For example, the WISE project, which has been used in many countries, recently implemented an English short-answer grader in the system so that instructors can immediately assess a student’s progress (Tansomboon et al., 2017). In addition, our proposed grading model can be potentially used in large-scale assessment. However, according to our reported measurements, human raters still significantly outperformed our grading models. The grading model must be further improved to fulfill the requirements of a prestigious assessment. Collecting more data may help further improve the performance.

6. Conclusions and future work

We defined semi-open-ended short-answer questions that ask students to express their subjective thoughts based on some facts in their answers. To build an automatic grading model for semi-open-ended short-answer questions, we integrated domain-general information from Wikipedia and domain-specific knowledge from the graded student short answers with the help of CBOW and generated word vectors to feed our LSTM-based classifier. Our experiments demonstrated that integrating both domain-general and domain-specific information significantly improved the automatic grading performance on semi-open-ended questions. Taking advantage of word sequence information by using LSTM improved the grading accuracy as well. As we discussed, training the grading model requires many graded student answers. More answers are needed when the corresponding semi-open-ended questions are based on a book rather than a short text. Since no grading rubric is currently used in our grading model, in the next step, we will explore the integration of information from a grading rubric to further improve the performance of the autograding model.

Notes


Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by National Natural Science Foundation of China: [grant number 61807004] and Philosophy and Social Sciences Research of the Chinese Ministry of Education: [grant number 16JZD043].

Notes on contributors

Lishan Zhang is an associate professor at Central China Normal University. He received a PhD in Computer Science from Arizona State University. He has published over 20 peer-reviewed academic papers. His research interests include intelligent tutoring systems, student modeling for personalized learning and educational data mining.

Yuwei Huang is an NLP engineer in Sunny Education Inc. He received the BS degree from Beijing University of Chemical Technology in 2018.

Xi Yang is an AI Researcher in Sunny Education Inc. She received the PhD degree from Peking University in 2013. She has published over 10 peer-reviewed academic papers. Her research interests include machine learning, natural language processing and artificial intelligence in education.

Shengquan Yu is a Professor at Beijing Normal University. He received a PhD in Educational Technology from Beijing Normal University. His research fields include mobile and ubiquitous learning, ICT and curriculum integration,
network learning technology, and education informatization policy. He has published about 100 peer-reviewed academic papers, four popular science books and three scholarly monographs.

Lishan Zhang is an associate professor in the Institute of Computing Technology, Chinese Academy of Sciences. He received his PhD degree in Computer Software and Theory from Chinese Academy of Sciences. His research interests include transfer learning, machine learning, data mining, multi-task learning and recommendation systems. He has published more than 80 papers in some prestigious refereed journals and conference proceedings.

ORCID
Lishan Zhang http://orcid.org/0000-0003-0830-2399

References

Luo, W., & Litman, D. J. (2016). Determining the quality of a student reflective response. FLAIRS.