Finding Similar Exercises in Online Education Systems

Qi Liu[1], Zai Huang[1], Zhenya Huang[1], Chuanren Liu[2], Enhong Chen[1,*] , Yu Su[3], Guoping Hu[4]
[1]Anhui Province Key Laboratory of Big Data Analysis and Application, School of Computer Science and Technology,
University of Science and Technology of China
{qiliuql,cheneh}@ustc.edu.cn,{huangzai,huangzhy}@mail.ustc.edu.cn
[2]Decision Sciences & MIS Department, Drexel University, chuanren.liu@drexel.edu
[3]School of Computer Science and Technology, Anhui University, yusu@iflytek.com
[4]iFLYTEK Research, gphu@iflytek.com

ABSTRACT

In online education systems, finding similar exercises is a fundamental task of many applications, such as exercise retrieval and student modeling. Several approaches have been proposed for this task by simply using the specific textual content (e.g. the same knowledge concepts or the similar words) in exercises. However, the problem of how to systematically exploit the rich semantic information embedded in multiple heterogenous data (e.g. texts and images) to precisely retrieve similar exercises remains pretty much open. To this end, in this paper, we develop a novel *M*ultimodal *A*ttention-based *N*eural *N*etwork (MANN) framework for finding similar exercises in large-scale online education systems by learning a unified semantic representation from the heterogenous data. In MANN, given exercises with texts, images and knowledge concepts, we first apply a convolutional neural network to extract image representations and use an embedding layer for representing concepts. Then, we design an attention-based long short-term memory network to learn a unified semantic representation of *each exercise* in a multimodal way. Here, two attention strategies are proposed to capture the associations of texts and images, texts and knowledge concepts, respectively. Moreover, with a Similarity Attention, the similar parts in *each exercise pair* are also measured. Finally, we develop a pairwise training strategy for returning similar exercises. Extensive experimental results on real-world data clearly validate the effectiveness and the interpretation power of MANN.

KEYWORDS

Similar exercises, Online education systems, Heterogenous data

ACM Reference Format:
Qi Liu[1], Zai Huang[1], Zhenya Huang[1], Chuanren Liu[2], Enhong Chen[1,*] , Yu Su[3], Guoping Hu[4]. 2018. Finding Similar Exercises in Online Education Systems. In *KDD'18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3219819.3219960

1 INTRODUCTION

Recent years have witnessed the booming of online education systems, such as KhanAcademy.org, Knewton.com, ASSISTments.org and Zhixue.com. In these systems, millions of exercises (or questions) have been collected for numerous applications [2, 4, 19, 20, 32, 40, 41]. For instance, we can retrieve/recommend the similar exercises to students for practicing [2] or conduct the cognitive analysis of students with the help of these exercises [32, 48].

Among exercise-based applications, Finding Similar Exercises (FSE) is a fundamental task [29, 44]. Generally, similar exercises are those having the same purpose [7, 32] that is embedded in the semantics learned from exercise contents (i.e., texts, images and concepts[1]) [16, 35]. For instance, Figure 1 shows three examples of math exercises, where exercise E_1 and its similar exercise E_2 share the same purpose of assessing the mastery of information acquisition on concept C_1 ("Solid geometry") and mathematical calculation on C_2 ("Volume"). Indeed, several efforts have been made on FSE task for finding the similar ones of each given exercise. Specifically, on a small quantity of exercises, manual labeling is usually conducted [18]. However, manual labeling requires strong expertise and takes much time, which is not suitable for FSE task in large-scale online education systems containing millions of exercises that are continuously collected from various sources (e.g. the Internet or schools). Therefore, it is an urgent issue to automatically understand the semantics of exercises from their contents. Along this line, methods based on text similarity (e.g. vector space model) have been applied [9, 36, 37, 44], where the same concepts or the similar words are used to calculate exercise similarity. Unfortunately, to the best of our knowledge, few of existing solutions can synthetically exploit the heterogeneous data (i.e. both texts and images) to precisely understand the semantics of each exercise. Consequently, the dissimilar exercises (e.g. E_1 and E_3 in Figure 1), which share the same concepts or many common words but have different purposes, may be misclassified as similar ones.

In summary, there are still many unique challenges inherent in designing an effective FSE solution. First, exercises contain multiple heterogeneous data, i.e., texts, concepts and images (actually, about 75% of math exercises have at least one image, e.g. geometric figures and equations, as shown experimentally). How to integrate these materials to understand and represent exercises in a multimodal way is a nontrivial problem. Second, in *a single exercise*, different parts/words of the text are usually associated with different concepts (text-concept) or images (text-image). For instance, in E_1 of

[*] Corresponding Author.

[1] Concepts (short for knowledge concepts [35, 46] or knowledge points [4]) are previously labeled (e.g. in ASSISTments [4, 46]) or can be easily obtained by algorithms [26].

KDD'18, August 19–23, 2018, London, United Kingdom
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3219960

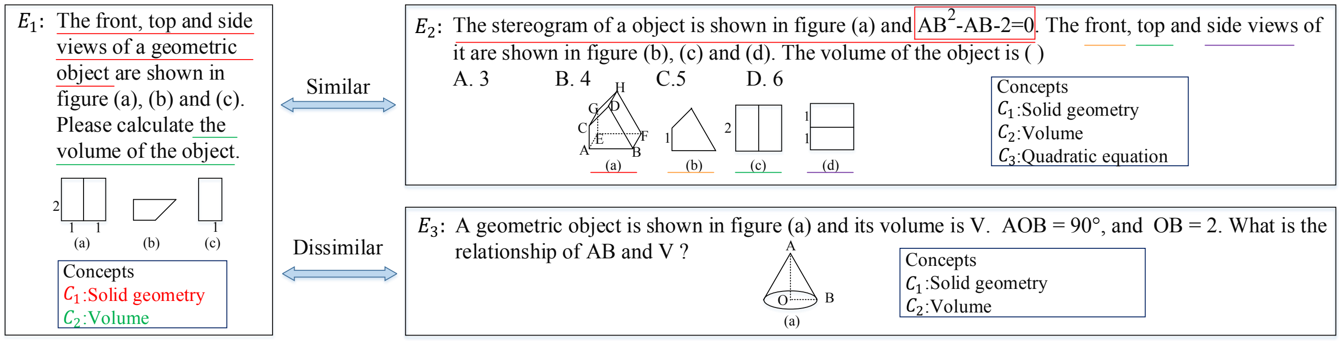

E_1: The front, top and side views of a geometric object are shown in figure (a), (b) and (c). Please calculate the volume of the object.
2
1 1
(a)
(b)
(c)
1
Concepts
C_1:Solid geometry
C_2:Volume
Similar
Dissimilar
E_2: The stereogram of a object is shown in figure (a) and AB^2-AB-2=0. The front, top and side views of it are shown in figure (b), (c) and (d). The volume of the object is ( )
A. 3 B. 4 C.5 D. 6
H
G
D
C
E
F
A
B
(a)
1
(b)
2
(c)
1
1
(d)
Concepts
C_1:Solid geometry
C_2:Volume
C_3:Quadratic equation
E_3: A geometric object is shown in figure (a) and its volume is V. AOB = 90°, and OB = 2. What is the relationship of AB and V ?
A
O
B
(a)
Concepts
C_1:Solid geometry
C_2:Volume

Figure 1: Three examples of math exercises: E_1, E_2 and E_3.

Figure 1, words on the "red" underline focus more on the concept C_1 while the words on the "green" underline concentrate more on C_2. Similarly, in E_2 of Figure 1, the words that describe the same image are noted with the same color underline. Thus, when understanding each exercise, it is necessary to capture these text-concept and text-image associations. Third, *a pair of similar exercises* may consist of different types of texts, images and concepts, such as the similar exercise pair (E_1, E_2) in Figure 1. Thus, for finding similar exercises, it is also critical to measure the similar parts in each exercise pair by deeply interpreting their semantic relations.

To address the challenges mentioned above, in this paper, we develop a novel *M*ultimodal *A*ttention-based *N*eural *N*etwork (MANN) framework for finding similar exercises in large-scale online education systems by learning a unified semantic representation from the heterogenous data. Specifically, given the exercises with texts, images and concepts, we first apply a Convolutional Neural Network (CNN) to extract image representations and use an embedding layer for representing concepts. Then, we design an Attention-based Long Short-Term Memory (Attention-based LSTM) network to learn a unified semantic representation of each exercise by handling its heterogeneous materials in a multimodal way. Here, two attention strategies, i.e. Text-Image Attention and Text-Concept Attention, are proposed to capture the text-image and text-concept associations in *each single exercise*, respectively. Next, we design a Similarity Attention to measure the similar parts in *each exercise pair* with their semantic representations. Finally, a pairwise training strategy is proposed for MANN to find similar exercises. In this way, those candidate exercises having the largest similarity score with the given exercise will be classified as the similar exercises. Extensive experiments on a large-scale real-world dataset reveal that MANN not only significantly outperforms several baselines on the FSE task, but also provides interpretable insights to the similarity information of exercise pairs.

2 RELATED WORK

Generally, the related work can be grouped into the following three categories, i.e. studies on finding similar exercises, multimodal learning and pair modeling.

Studies on FSE. There are several efforts for FSE in the literature. Some prior works leveraged the texts or concepts of exercises to calculate exercise similarity. For example, Vector Space Model (VSM), combining TF-IDF and cosine similarity to measure text similarity of exercises, was a common and effective method in item bank systems [9, 36]. Williams et al. [37] held that similar exercises had core concepts in common, and they used concepts to analyze similarity between two exercises. Yu et al. [44] developed a method combining ontology and VSM to reveal the intrinsic relationship among words for text similarity of exercises. However, few of existing solutions can synthetically exploit the multiple heterogeneous materials (especially, the massive image data) to precisely understand and represent the semantics of each exercise. Recently, another direction made attempts to utilize students' performance data for measuring similar exercises by obtaining clusters of exercises [29]. Unfortunately, the similar performance of students (e.g. the similar percentage of students who answer the two exercises right [13]) usually does not guarantee the similarity of exercises.

Multimodal Learning. In our framework, one of the most important steps is to integrate heterogeneous exercise materials in a multimodal way, and this is related to multimodal learning. Multimodal learning is a powerful approach to deal with multiple heterogeneous data, such as sound and video [24], video and text [42], or image and text [3, 6]. What relate to our approach more closely are the works on handling images and texts. For example, some representative works attempted to map images (or image regions) and texts to a common embedding space and used canonical correlation analysis to obtain relations between images and texts [3, 14]. In another direction, Park et al. [27] developed a coherent recurrent convolutional network architecture to capture the associations between a sequence of images and sentences. Ma et al. [21] designed a CNN architecture with a multimodal convolution layer to learn the joint representations of image questions.

Unfortunately, these existing methods could not be directly applied to learn the semantics of exercises, as understanding exercise purposes has to not only handle multiple heterogeneous data (i.e., texts, images and concepts) but also consider the text-concept and text-image associations in each exercise. What's more, none of these methods can measure the similar parts between two exercises.

Pair Modeling. Modeling exercise pairs is relevant to many researches in pair modeling, such as sentence pair [25], image pair [22] or video-sentence pair [42]. Generally, methods for pair modeling tried to learn the relations between two instances in a pair. For example, Xu et al. [42] designed a joint video-language embedding model to learn the matching relations between the video and its describing sentence in video-sentence pairs. Mueller et al. [25] utilized a LSTM architecture to extract semantic representations for analyzing the similarity relations of sentences in pairs. Yin et al. [43] incorporated attention strategies into CNN to catch related parts of sentence pairs from words, phrases to sentences views. However, these methods do not focus on pair modeling of the instances having multiple heterogeneous data. Therefore, we should design novel solutions for measuring exercise pairs.

3 MANN FRAMEWORK

In this section, we first give the formal definition of the FSE task. Then, we introduce technical details of MANN framework. At last, we specify a pairwise loss function to train MANN.

3.1 Problem Definition

Similar exercises are those having the same purpose [7, 32] which is related with the semantics of exercises [16]. For any two exercises E_a and E_b, we use score $S(E_a, E_b)$ to measure the similarity between E_a and E_b. The higher $S(E_a, E_b)$ is, the more similar E_a and E_b are. Without loss of generality, the problem of finding similar exercises can be formulated as:

Definition 1. *Given a set of exercises with corresponding heterogeneous materials including texts (ET), images (EI) and concepts (EC), our goal is to integrate these heterogeneous materials to learn a model $\mathcal{F}$, which can be used to measure the similarity scores of exercise pairs and find similar exercises for any exercise E by ranking the candidate ones $\mathcal{R}$ with similarity scores, i.e.*

$$\mathcal{F}(E, \mathcal{R}, \Theta) \rightarrow \mathcal{R}^s, \tag{1}$$

where Θ is the parameters of $\mathcal{F}$, $\mathcal{R} = (E_1, E_2, E_3, \dots)$ are the candidate exercises for E and $\mathcal{R}^s = (E_1^s, E_2^s, E_3^s, \dots)$ are the candidates ranked in descending order with their similarity scores ($S(E, E_1^s), S(E, E_2^s), S(E, E_3^s), \dots$). The similar exercises for E are those candidates having the largest similarity score.

For tackling the above problem, we propose a two-stage solution containing a training stage and a testing stage. The flowchart is shown in Figure 2. In the training stage, given exercises with texts, images and concepts, we propose MANN to learn a unified semantic representation for each exercise by handling the heterogeneous materials in a multimodal way, and meanwhile, calculate the similarity score for each pair of exercises. We utilize a pairwise loss function to train MANN, i.e. for an exercises E, its similar pairs (E, E_s) should have higher similarity scores than dissimilar ones (E, E_{ds}). Here, $E_s \in Sim(E)$, $E_{ds} \in DS(E)$ and we suppose the similar exercises $Sim(E)$ for E are previously given, e.g. labeled by the experts, and its dissimilar ones $DS(E)$ can be gotten by sampling. After obtaining the trained MANN, for any exercise E_a in the testing stage, we could find its similar exercises $(E_{a,1}^s, E_{a,2}^s, \dots)$ by ranking the candidate ones according to their similarity scores.

3.2 Details of MANN

In this subsection, we will introduce the technical details of MANN framework. As shown in Figure 3, MANN mainly contains three parts, i.e., Multimodal Exercise Representing Layer (MERL), Similarity Attention (SA) and Similarity Score Layer (SSL). Specifically, MERL outputs a unified semantic representation of each exercise in a multimodal way by utilizing its heterogeneous materials. SA measures similar parts between two exercises with their semantic representations. SSL calculates the similarity scores of exercise pairs, which can be used to rank candidate exercises to find similar ones for any exercise.

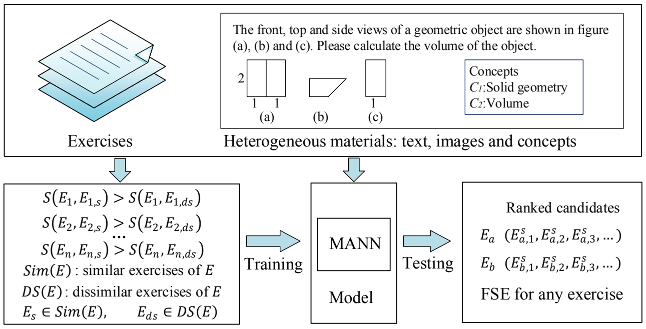

The front, top and side views of a geometric object are shown in figure (a), (b) and (c). Please calculate the volume of the object.
2
1
1
1
(a)
(b)
(c)
Concepts
C1:Solid geometry
C2:Volume
Exercises
Heterogeneous materials: text, images and concepts
$S(E_1, E_{1,s}) > S(E_1, E_{1,ds})$
$S(E_2, E_{2,s}) > S(E_2, E_{2,ds})$
...
$S(E_n, E_{n,s}) > S(E_n, E_{n,ds})$
$Sim(E)$: similar exercises of E
$DS(E)$: dissimilar exercises of E
$E_s \in Sim(E)$, $E_{ds} \in DS(E)$
Training
MANN
Model
Testing
Ranked candidates
E_a $(E_{a,1}^s, E_{a,2}^s, E_{a,3}^s, \dots)$
E_b $(E_{b,1}^s, E_{b,2}^s, E_{b,3}^s, \dots)$
FSE for any exercise

Figure 2: The flowchart of our work.

3.2.1 *Multimodal Exercise Representing Layer*.

Figure 4 shows the details of MERL in MANN framework. With the exercise materials input from Exercise Input, we first utilize Image CNN and Concept Embedding to preprocess the encodings from images and concepts, respectively. Then, we design an Attention-based LSTM to learn a unified semantic representation for each exercise by integrating its heterogeneous materials in a multimodal way.

1) Exercise Input. The input to MERL is the materials of an exercise E, i.e., the text (ET), images (EI) and concepts (EC), as shown in Figure 4. Intuitively, the text ET is formalized as a sequence of N words $ET = (w_1, w_2, \dots, w_N)$, where $w_i \in \mathbb{R}^{d_0}$ is initialized by an d_0-dimensional pre-trained word embedding with *Word2vec* [23]. The concepts in E can be represented by a matrix $EC = (k_1, k_2, \dots, k_L) \in \{0, 1\}^{L \times L_{all}}$, where k_i is a one-hot vector with the dimension equaling to the total number L_{all} of all concepts in the item bank, L is the number of concepts in E. For images EI, similar to the works [12, 31], we convert them to gray images with the resizing size (64×64) and each pixel value in $[0, 1]$. Thus, $EI = (p_1, p_2, \dots, p_M) \in \mathbb{R}^{M \times 64 \times 64}$, where $p_i \in \mathbb{R}^{64 \times 64}$ represents the i-th image and M is the number of images in E. After the initialization from Exercise Input, in the following, we apply Image CNN and Concept Embedding to enhance feature representations of images and concepts, respectively.

2) Image CNN. For the images EI in E, we utilize a CNN architecture, i.e. Image CNN (ImCNN), with five layers of convolution and max pooling, which is similar to the work [17], to get the feature vector for each image. We use an encode-decode architecture for pre-training ImCNN. Specifically, we set ImCNN as the encode and five corresponding deconvolution layers [45] as the decode, DeImCNN, and then pre-train ImCNN with the following loss function:

$$\mathcal{L}_{ImCNN} = \sum_p (DeImCNN(ImCNN(p)) - p)^2, \tag{2}$$

where p is a pre-trained image.

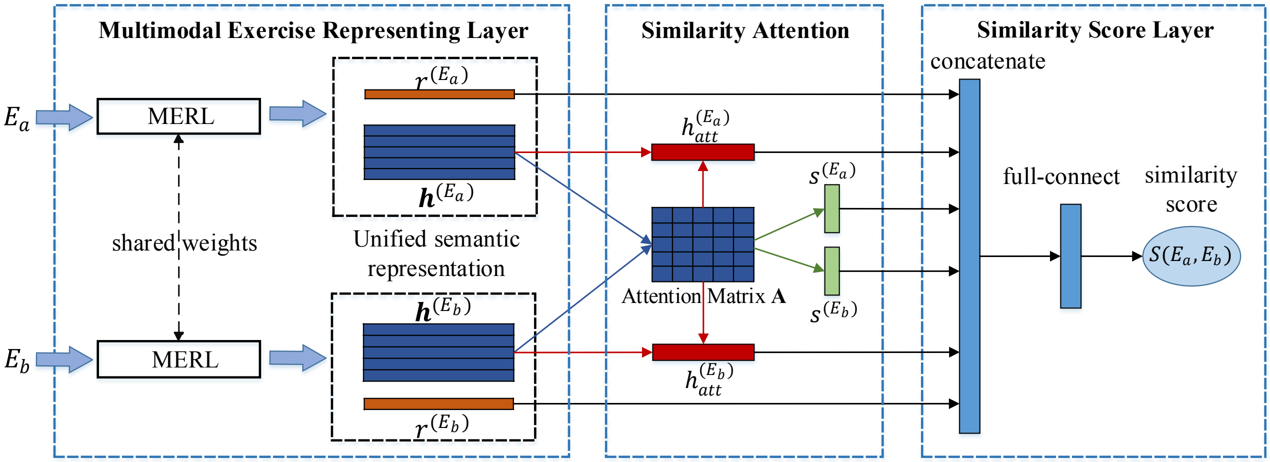


Figure 3: MANN framework with the input of an exercise pair (E_a, E_b) and the output of its similarity score.

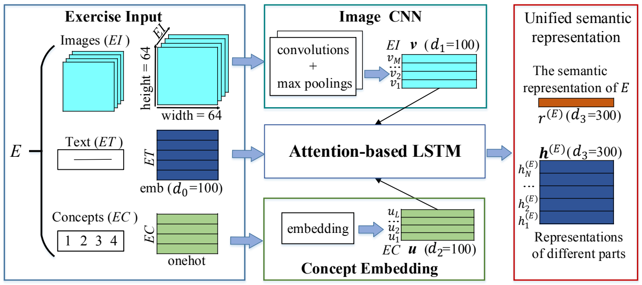


Figure 4: Multimodal Exercise Representing Layer (MERL).

Through ImCNN, each image p_i can be represented by a fixed length vector v_i, which can be expressed as

$$v_i = \sigma(ImCNN(p_i)), \tag{3}$$

where $v_i \in \mathbb{R}^{d_1}$, d_1 is the output dimension of ImCNN, and $\sigma(x)$ is the sigmoid function. As a result, the representation EI of images is transformed into a matrix $\boldsymbol{v} = (v_1, v_2, \ldots, v_M) \in \mathbb{R}^{M \times d_1}$, which is shown in Figure 4.

3) Concept Embedding. Since the one-hot representations for concepts are too sparse to train, we utilize an embedding operation to convert the initialized vectors of concepts into a low-dimensional ones with dense values. Formally, for a concept k_i, the converted vector u_i is expressed as:

$$u_i = k_i \mathbf{W_u}, \tag{4}$$

here, $\mathbf{W_u} \in \mathbb{R}^{L_{all} \times d_2}$ are the parameters of the embedding layer and $u_i \in \mathbb{R}^{d_2}$, where d_2 is the output dimension of it. As a result, the representation EC of concepts is transformed into a matrix $\boldsymbol{u} = (u_1, u_2, \ldots, u_L) \in \mathbb{R}^{L \times d_2}$, which is also shown in Figure 4.

4) Attention-based LSTM. After getting the feature representations of images and concepts, Attention-based LSTM aims at learning a unified semantic representation for an input exercise E by integrating its all heterogeneous materials, i.e., texts, images and concepts. In each exercise, different parts of the text are associated with different concepts and images. As shown in Figure 1, words in E_1 on the "red" underline pay more attention to the concept C_1 while the ones on the "green" underline focus more on C_2. In E_2, the words that describe the same image are on the same color underline. Therefore, we design the Attention-based LSTM architecture to learn representations for each exercise in a multimodal way, where we utilize two attention strategies, i.e. Text-Concept Attention (TCA) and Text-Image Attention (TIA), to capture the text-concept and text-image associations respectively. The architecture of this Attention-based LSTM is shown in Figure 5.

Methodology-wise, Attention-based LSTM is a variant of the traditional LSTM architecture [8, 11] with improvements. As LSTM can handle an any long sequence and learn long range dependencies across the input sequence [8, 11], we use a LSTM-based architecture to learn the representations for exercises with word sequences of any length. Specifically, in this paper, the input to the LSTM network is a sequence $\boldsymbol{x} = (x_1, x_2, \ldots, x_N)$ combined with all materials of each exercise, and then the hidden state h_t at the t-th input step is updated as following formulas:

$$\begin{aligned}
i_t &= \sigma(\mathbf{W_{xi}} x_t + \mathbf{W_{hi}} h_{t-1} + \mathbf{b_i}),\\
f_t &= \sigma(\mathbf{W_{xf}} x_t + \mathbf{W_{hf}} h_{t-1} + \mathbf{b_f}),\\
o_t &= \sigma(\mathbf{W_{xo}} x_t + \mathbf{W_{ho}} h_{t-1} + \mathbf{b_o}),\\
c_t &= f_t c_{t-1} + i_t tanh(\mathbf{W_{xc}} x_t + \mathbf{W_{hc}} h_{t-1} + \mathbf{b_c}),\\
h_t &= o_t tanh(c_t),
\end{aligned} \tag{5}$$

where $i_\bullet$, $f_\bullet$, $c_\bullet$, $o_\bullet$ are the input gate, forget gate, memory cell, output gate of LSTM respectively. $\mathbf{W}_\bullet$ and $\mathbf{b}_\bullet$ are learned weight matrices and biases.

Here we introduce how to obtain the combined sequence input $\boldsymbol{x}$. Obviously, at each input step, x_t is a multimodal vector integrating the text, images and concepts, i.e.

$$x_t = w_t \oplus \hat{u}_t \oplus \hat{v}_t, \tag{6}$$

where "$\oplus$" is the operation that concatenates two vectors into a long vector, w_t is the t-th word representation in the text ET, $\hat{u}_t$ and $\hat{v}_t$ are the representations of the associated concepts and images of this word, which are learned by TCA and TIA respectively.

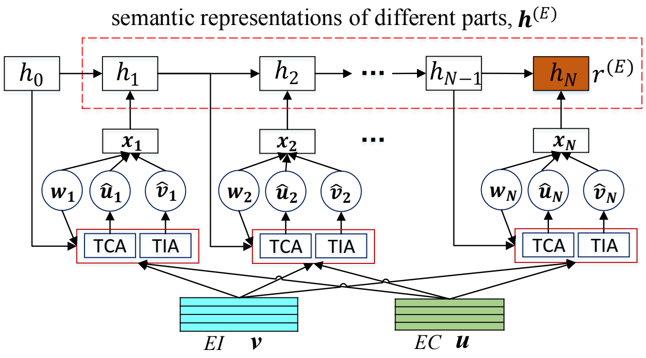

semantic representations of different parts, $\boldsymbol{h}^{(E)}$
h_0
h_1
h_2
h_{N-1}
h_N
$r^{(E)}$
$\boldsymbol{x_1}$
$\boldsymbol{x_2}$
$\boldsymbol{x_N}$
$\boldsymbol{w_1}$ $\boldsymbol{\hat{u}_1}$ $\boldsymbol{\hat{v}_1}$
$\boldsymbol{w_2}$ $\boldsymbol{\hat{u}_2}$ $\boldsymbol{\hat{v}_2}$
$\boldsymbol{w_N}$ $\boldsymbol{\hat{u}_N}$ $\boldsymbol{\hat{v}_N}$
TCA TIA
TCA TIA
TCA TIA
EI $\boldsymbol{v}$
EC $\boldsymbol{u}$

Figure 5: Attention-based LSTM in MERL.

TCA aims at capturing the text-concept associations. In TCA, at the t-th input step, we first measure the association between w_t and each concept representation. As a concept is related to words of a text part, and h_{t-1} holds the information of words before the t-th input step, h_{t-1} should be taken into account during measuring the association. Then, the associated concept representation $\hat{u}_t$ can be modeled as a vector by a weighted sum aggregated result of $\boldsymbol{u}$, which is expressed as

$$\hat{u}_t = \sum_{j=1}^{L} \alpha_j u_j, \quad \alpha_j = \frac{\varphi(u_j, w_t, h_{t-1})}{\sum_{i=1}^{L} \varphi(u_i, w_t, h_{t-1})},$$

$$\varphi(u_j, w_t, h_{t-1}) = \mathbf{V_{ac}} tanh(\mathbf{W_{ac}}[u_j \oplus w_t \oplus h_{t-1}]), \tag{7}$$

where $\mathbf{V_{ac}}$ and $\mathbf{W_{ac}}$ are learned parameters of TCA, $\varphi(u_j, w_t, h_{t-1})$ measures the association between the j-th concept u_j and w_t in E, α_j denotes the attention score of $\varphi(u_j, w_t, h_{t-1})$ after normalization.

TIA aims at capturing the text-image associations. Similar to TCA, in TIA, the associated image representation $\hat{v}_t$ for w_t can be modeled as the form of Eq. (7), where we can simply use v_j and the learned parameters of TIA, i.e. $\mathbf{V_{ai}}$ and $\mathbf{W_{ai}}$, to replace u_j, $\mathbf{V_{ac}}$ and $\mathbf{W_{ac}}$. Please also note that, $\hat{v}_t$ can be a zero vector if the exercise E has no image.

Through Attention-based LSTM, we can get the hidden state sequence $\boldsymbol{h} = (h_1, h_2, \ldots, h_N)$ with the combined input sequence $\boldsymbol{x}$. Furthermore, inspired by the works applying LSTM to natural language processing [8, 25], the final hidden state h_N holds the semantic information of the whole input sequence $\boldsymbol{x}$ of the exercise E, so we employ h_N as the semantic representation $r^{(E)}$ for E, i.e. $r^{(E)} = h_N$. Besides, the t-th hidden state h_t only holds the information of the sequence $(x_1, x_2, \ldots, x_t)$, so different hidden states contain semantic information of different parts of the exercise E. Therefore, we further denote $\boldsymbol{h}^{(E)} = \boldsymbol{h}$ as representations of different parts of E. Thus, we can obtain the unified semantic representation $(r^{(E)}, \boldsymbol{h}^{(E)})$ for E.

3.2.2 *Similarity Attention*.

As shown in Figure 3, for each exercise pair (E_a, E_b) with their unified representations, i.e. $(r^{(E_a)}, \boldsymbol{h}^{(E_a)})$ and $(r^{(E_b)}, \boldsymbol{h}^{(E_b)})$, obtained by MERL, Similarity Attention targets at measuring the similar parts between E_a and E_b with the semantic representations. As shown in Figure 1, though E_1 and E_2 are similar exercises, they have different texts, images and concepts. This evidence indicates that similar exercises may consist of different materials. Thus, when finding similar exercises, it is necessary to catch semantic similar parts of two exercises. Therefore, we design the Similarity Attention to measure similar parts of two exercises and learn attention representations for them.

In Similarity Attention, we use a similarity attention matrix $\boldsymbol{A}$ to measure similar parts of the input pair (E_a, E_b) by calculating *cosine similarities* between each part of E_a and each part of E_b with $\boldsymbol{h}^{(E_a)}$ and $\boldsymbol{h}^{(E_b)}$. $\boldsymbol{A} \in \mathbb{R}^{N_{E_a} \times N_{E_b}}$ can be expressed as

$$A_{i,j} = cos(h_i^{(E_a)}, h_j^{(E_b)}), \tag{8}$$

here, $1 \le i \le N_{E_a}$, $1 \le j \le N_{E_b}$, N_{E_a} and N_{E_b} are the lengths of the word sequences of E_a and E_b respectively. $h_i^{(E_a)}$ is the i-th representation in $\boldsymbol{h}^{(E_a)}$ and $h_j^{(E_b)}$ is the j-th representation in $\boldsymbol{h}^{(E_b)}$. Particularly, the *cosine similarity* score $A_{i,j}$ in $\boldsymbol{A}$ greatly enhances the explanatory power of MANN. It helps us analyze the similar parts in an exercise pair, e.g. by visualization.

With the attention matrix $\boldsymbol{A}$, we can find that the sum score value $s_i^{(E_a)} = \sum_{k=1}^{N_{E_b}} A_{i,k}$ actually measures the sum similarity of the i-th representation in $\boldsymbol{h}^{(E_a)}$ with each one in $\boldsymbol{h}^{(E_b)}$. Similarly, the sum score value $s_j^{(E_b)} = \sum_{k=1}^{N_{E_a}} A_{k,j}$ measures the sum similarity of the j-th representation in $\boldsymbol{h}^{(E_b)}$ with each one in $\boldsymbol{h}^{(E_a)}$. Thus, we denote these two similarity score vectors, i.e. $s^{(E_a)}$ and $s^{(E_b)}$, as the similarity attention representations of E_a and E_b respectively.

Furthermore, as discussed in Attention-based LSTM, $h_{N_{E_a}}^{(E_a)}$ and $h_{N_{E_b}}^{(E_b)}$ hold the whole semantic information of E_a and E_b respectively, so $A_{i,N_{E_b}}$ and $A_{N_{E_a},j}$ actually measure the similarity of the i-th part $(h_i^{(E_a)})$ of E_a with E_b and the similarity of the j-th part $(h_j^{(E_b)})$ of E_b with E_a, respectively. Thus, for exercises E_a and E_b, we can model their semantic attention representations, i.e. $h_{att}^{(E_a)}$ and $h_{att}^{(E_b)}$, by the weighted sum aggregated results of $\boldsymbol{h}^{(E_a)}$ and $\boldsymbol{h}^{(E_b)}$, respectively:

$$h_{att}^{(E_a)} = \sum_{i=1}^{N_{E_a}} A_{i,N_{E_b}} h_i^{(E_a)},$$

$$h_{att}^{(E_b)} = \sum_{j=1}^{N_{E_b}} A_{N_{E_a},j} h_j^{(E_b)}. \tag{9}$$

With the help of Similarity Attention, we can get the attention matrix $\boldsymbol{A}$ and learn the similarity attention representations ($s^{(E_a)}$ and $s^{(E_b)}$) and semantic attention representations ($h_{att}^{(E_a)}$ and $h_{att}^{(E_b)}$) of the input exercise pair (E_a, E_b).

3.2.3 *Similarity Score Layer*.

Similarity Score Layer targets at calculating the similarity score of each exercise pair, which can be used to rank candidate exercises to find similar ones for any exercise. As shown in Figure 3, the similarity score of the input pair (E_a, E_b) is computed by leveraging their semantic representations (i.e. $r^{(E_a)}$ and $r^{(E_b)}$) and attention representations (i.e., $s^{(E_a)}$, $h_{att}^{(E_a)}$, $s^{(E_b)}$ and $h_{att}^{(E_b)}$). Specifically, we first concatenate them to a vector, i.e. $z_{ab} = r^{(E_a)} \oplus r^{(E_b)} \oplus s^{(E_a)} \oplus s^{(E_b)} \oplus h_{att}^{(E_a)} \oplus h_{att}^{(E_b)}$, and then obtain the similarity score $S(E_a, E_b)$ by using two full-connected networks [10] with a nonlinear activation function $ReLU(x) =$

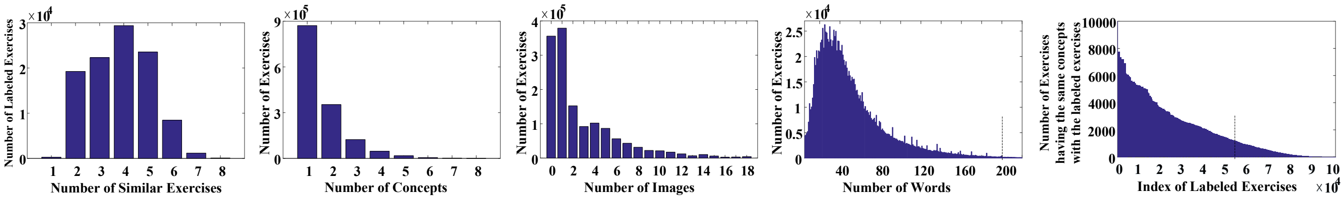


Figure 6: Number distributions of the observed records.

Table 1: Toy examples of the labeled similar exercises.

Labeled exercise	Expert	Similar exercises labeled by expert	Similar exercises Selected in experiments
E_a	exp_1 exp_2 exp_3	E_1, E_2, E_3, E_4, E_5 E_1, E_2, E_3, E_5 E_2, E_3, E_4, E_5	E_2, E_3, E_5
E_b	exp_4 exp_5	E_6, E_7, E_8, E_9, E_{10} E_6, E_7, E_8, E_9	E_6, E_7, E_8, E_9
E_c	exp_6	E_{11}, E_{12}, E_{13}	E_{11}, E_{12}, E_{13}
. . .	. . .	. . .	. . .

$\max(0, x)$ used in the first one and the sigmoid function for the second one:

$$\begin{aligned} \widetilde{o}_{ab} &= ReLU(\mathbf{W}_1 \widetilde{z}_{ab} + \mathbf{b}_1), \\ S(E_a, E_b) &= \sigma(\mathbf{W}_2 \widetilde{o}_{ab} + \mathbf{b}_2), \end{aligned} \qquad (10)$$

where $\mathbf{W}_1$, $\mathbf{b}_1$, $\mathbf{W}_2$, $\mathbf{b}_2$ are parameters of the network. Therefore, this method precisely measures the similarity scores of exercise pairs by leveraging the heterogeneous materials. Those candidates with the largest similarity score will be returned as similar exercises of the given one, e.g. $(E^s_{a,1}, E^s_{a,2}, \ldots)$ for exercise E_a in Figure 2.

3.3 MANN Learning

In this subsection, we specify a pairwise loss function for training MANN. In the training stage, we suppose there are a subset of exercises which have been labeled with several similar ones. For an exercise E, we use $Sim(E)$ to denote its labeled similar exercises and treat unlabeled exercises as its dissimilar ones $DS(E)$. Considering the similar pairs (E, E_s) should have higher similarity scores than the dissimilar ones (E, E_{ds}), where $E_s \in Sim(E)$ and $E_{ds} \in DS(E)$, as shown in Figure 2, we further formulate the pairwise loss function as following:

$$\mathcal{L}(\Theta) = \sum_{E, E_s, E_{ds}} max(0, \mu - (S(E, E_s) - S(E, E_{ds}))) + \lambda_\Theta ||\Theta||^2, \quad (11)$$

where $S(\cdot, \cdot)$ is computed by Eq. (10); Θ denotes all parameters of MANN and λ_Θ is the regularization hyperparameter; μ is a margin, forcing $S(E, E_s)$ to be greater than $S(E, E_{ds})$ by μ. In this way, we can learn MANN by directly minimizing the loss function $\mathcal{L}(\Theta)$ using Adam [15].

As the number of dissimilar exercises of each labeled exercise E is huge, it will take much time to train MANN if using all of them at each iteration of the training process. Therefore, inspired by the work [38, 39], we only sample a number (e.g. 50) of dissimilar exercises as $DS(E)$ for E at each iteration. Specifically, in our work, we have two sampling ways:

Table 2: The statistics of the dataset.

Statistics	Values
number of exercises	1,420,727
number of exercises having images	1,064,964
number of labeled exercises	104,515
number of similar pairs	401,476
number of similar pairs having the same concepts	174,672
Average similar pairs per labeled exercise	3.84
Average concepts per exercise	1.61
Average images per exercise	3.04

Sampling Randomly (Random). At each iteration, for each given exercise E, we randomly select a number of dissimilar exercises from all the dissimilar ones of E.

Sampling by Concepts (Concept). At each iteration, for each given exercise E, we randomly select a number of dissimilar exercises from those having at least one common concept with E.

4 EXPERIMENTS

In this section, we first assess the performance of MANN on the FSE task comparing with several baselines. Then, we conduct a *case study* to visualize the explanatory power of MANN.

4.1 Dataset Description

The experimental dataset supplied by iFLYTEK is collected from Zhixue[2], which is an online education system for providing a series of exercise-based applications to high school students in China. The dataset contains 1,420,727 real-world math exercises[3], which are collected from schools or the Internet. Images in math exercises include geometric figures, some equations and mathematical notations as shown in the red box in E_2 of Figure 1. Moreover, education experts (e.g. teachers) are invited to label similar exercises. Different from crowdsourcing labeling, labeling similar exercises not only is time-consuming and laborious, but also requires strong expertise. As a result, in the dataset, 104,515 exercises are labeled with several similar exercises and each labeled (given) exercise is labeled by at least one expert.

As shown in Table 1, for each labeled exercise, we select those similar exercises labeled by all its labeling experts as its similar exercises, so we get 401,476 similar pairs of exercises totally after pruning. For a labeled exercise E, we denote those which are not its labeled similar exercises as the dissimilar exercises of E. Table 2 shows the basic statistics of the dataset, and Figure 6 illustrates the

[2] http://www.zhixue.com

[3] Please note that MANN solution is a general framework which can also handle the exercises from all the disciplines, e.g. Physics.

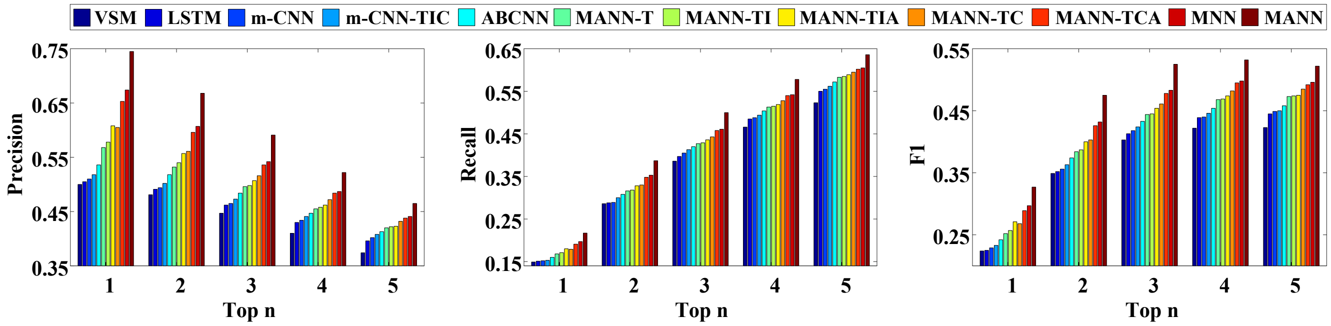


Figure 7: Performance comparison on the FSE task.

number distributions of similar exercises, concepts, images, words in text and the exercises having the same concepts with each labeled exercise. We can observe that: (1) On average 3.84 similar exercises are labeled for the given one; (2) Each exercise consists of about 1.61 concepts and 3.04 images; (3) About 75% exercises have at least one image; (4) 99% exercises contain less than 200 words in the text; (5) More than 55% labeled exercises have the same concepts with at least 1,000 exercises. These observations once again prove that similar exercises cannot be easily identified (e.g. just based on the same concepts or from the similar words of two exercises) in large-scale online education systems, and it is necessary to exploit the multiple heterogeneous data for precisely understanding each exercise on the FSE task.

4.2 Experimental Setup

For validating the effectiveness of MANN, we employ 5-fold cross validation on the labeled exercises in the dataset, where one of five folds is targeted to construct the testing set and the rest for the training set.

Word Embedding Pre-training. The word embedding used in Exercise Input of MERL is trained on texts of math exercises in the dataset, using the *Word2vec* tool [23] with the dimension (d_0) 100.

Image CNN Pre-training. Image CNN (ImCNN) for getting the feature vector of each image in MERL is trained on images in the training set by minimizing the loss function as Eq. (2), and the dimension (d_1) of feature vectors for images is 100.

MANN Setting. We set the size (d_2) of embedding representations for concepts as 100, the dimension (d_3) of hidden states in Attention-based LSTM as 300 and the size of the output of the first full-connected network as 200.

Training Details. We initialize parameters in MANN with a truncated normal distribution with the standard deviation 0.1. We set mini-batches as 64, $\mu = 0.5$ and $\lambda_\Theta = 0.00004$ in Eq. (11) for training MANN, and parameters of MANN, except those in the pre-trained ImCNN and *Word2vec*, can be tuned during the training process. We also use dropout [33] with the probability 0.2 to prevent overfitting and gradient clipping [28] to avoid the gradient explosion problem.

Testing Details. As the similar exercises usually have common core concepts [37], for a given exercise, it is necessary to select those having at least one common concept with it as candidates to find its similar ones. However, the number of those exercises is still very large, and it is impractical to take all of them as candidates, e.g. more than 55% labeled exercises in our dataset have the same concepts with at least 1,000 exercises. Therefore, similar to the training process, for each given exercise in the testing set, we randomly sample m unlabeled exercises having at least one common concept with it and mix them with its labeled similar exercises together as candidates. When measuring the performance of a model, we repeat this process multiple times and report the average results.

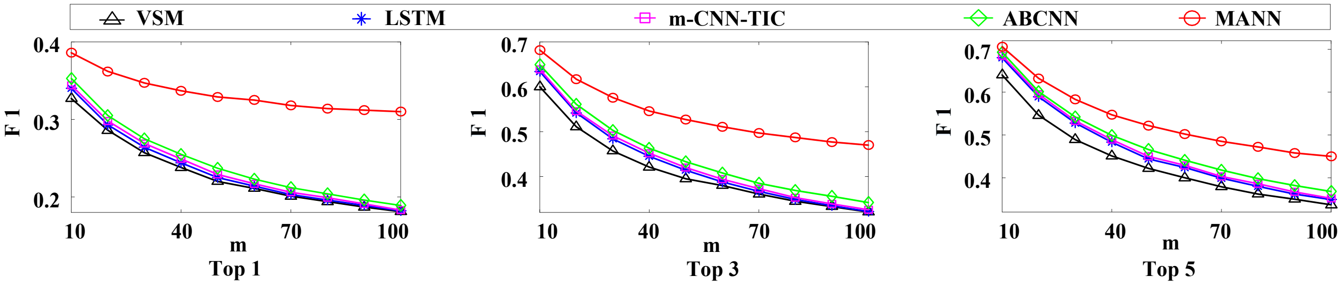
4.3 Baseline Approaches

In order to demonstrate the effectiveness of MANN, we compare it with several methods including some variants of MANN, a traditional method on the FSE task and the models for pair modeling and multimodal learning:

- *VSM*: Vector space model (VSM) combining TF-IDF and cosine similarity based on the texts of exercises, is a simple, effective and unsupervised method. It is widely applied for the FSE task in many educational systems [9, 36, 44].
- *LSTM*: LSTM is applied to learn the semantic similarity between sentences [25], based on texts.
- *ABCNN*: ABCNN is a network architecture based on texts for modeling sentence pairs [43].
- *m-CNN*: m-CNN is a multimodal CNN architecture for integrating texts and images into a vectorial representation [21], which can be applied to obtain the representation for an exercise by using its texts and images.
- *m-CNN-TIC*: We expand m-CNN model to integrate Texts, Images and Concepts in the similar way in m-CNN to obtain the vectorial representation for an exercise.

The variants of MANN are listed as follows:

- *MANN-T*: MANN-T is a variant of MANN by only using the Texts of exercises.
- *MANN-TI*: MANN-TI is a variant of MANN by only using the Texts and Images of exercises without TIA.
- *MANN-TIA*: MANN-TIA is a variant of MANN by only using the Texts and Images of exercises, and considering the text-image association with TIA.
- *MANN-TC*: MANN-TC is a variant of MANN by only using Texts and Concepts of exercises without TCA.



Top 1

Top 3

Top 5

Figure 8: Performance with different *m*.

- *MANN-TCA*: MANN-TCA is a variant of MANN by only using Texts and Concepts of exercises, and considering the text-concept association with TCA.
- *MNN*: MNN is a variant of MANN by using texts, images and concepts of exercises without TIA and TCA.

Note that the above variants of MANN all contain Similarity Attention and Similarity Score Layer. The baselines except VSM are trained with the pairwise loss function as Eq. (11) and implemented in Tensorflow [1]. All the experiments are conducted on a Pascal Titan X GPU.

4.4 Evaluation Metrics

To find similar exercises for a given exercise, the candidates that have the largest similarity score with this exercise will be returned. Thus, we adopt three widely used top-n ranking metrics [5, 30, 39, 47]: *Precision*, *Recall*, and *F1* measure, where n denotes the size of exercises selected from candidate ones. As shown in Table 2 and Figure 6, on average 3.84 similar exercises are labeled for the given one and more than 90% given exercises have less than 6 labeled similar exercises, so we set $n = 1, 2, 3, 4, 5$ in experiments. For the three metrics, the larger, the better.

4.5 Experimental Results

4.5.1 ***Performance Comparison.*** To investigate the performance of MANN and baselines on the FSE task, we train the models in the *Concept* sampling way (Section 3.3). As discussed in *Testing Details*, we set $m = 50$ (50 is enough, for the average number of similar exercises per labeled exercise is only 3.84). The process of constructing the testing set to calculate the three metrics is repeated 10 times and we report the average results.

Figure 7 shows the performance results of all models. We can find that our proposed MANN achieves the best performance, with the improvement by up to 39%, 35% and 35% in *Precision*, *Recall* and *F1* at Top 1 compared to ABCNN. Meanwhile, the variants of MANN also have better performance than other baselines. Specifically, first, VSM does not perform as well as other models because VSM just focuses on common words in exercise pairs but cannot understand exercises semantically. Second, ABCNN performs better than LSTM, m-CNN and m-CNN-TIC, as ABCNN can learn and measure similar parts of an exercise pair but LSTM, m-CNN and m-CNN-TIC cannot. Third, MANN-T performs better than ABCNN, indicating the effectiveness of Similarity Attention to measure similar parts of an exercise pair and that our framework still works well on FSE task just using texts of exercises. Fourth, MANN-TIA beats MANN-T and MANN-TI by additionally utilizing images of exercises and capturing the text-image association with TIA, and MANN-TCA performs better than MANN-T and MANN-TC by additionally utilizing concepts and capturing the text-concept association with TCA. Last but not least, MANN performs best and MNN ranks the second, which suggests that it is more effective for the FSE task by integrating the texts, images and concepts, and further demonstrates the effectiveness of TIA and TCA.

In summary, these evidences indicate that the concepts and images are important materials in exercises and are useful for FSE task. Also, they imply that MANN can more effectively find similar exercises by integrating the texts, concepts and images in a multimodal way, and meanwhile, capturing the text-image association and text-concept association, as well as measuring similar parts between two exercises with their semantic representations.

4.5.2 ***Performance with Different m.*** To further demonstrate the effectiveness of MANN, we set *VSM*, *LSTM*, *ABCNN* and *m-CNN-TIC* as representative baselines, and investigate the performance of models with different number (m) of the sampled unlabeled exercises for each given exercise in the testing set (as discussed in *Testing Details*). We conduct an experiment for different m from the set $\{10, 20, 30, \ldots, 100\}$ and take *F1* measure at top $n = 1, 3, 5$ as the metric. The experimental process is repeated 10 times.

The average results are shown in Figure 8. We can find that with different m, MANN still outperforms baselines and the *F1* value of MANN degrades the most slowly while m increases. That is, the more unlabeled exercises (i.e. negative samples) in the testing set, the more improvement of MANN compared with the baselines could be observed. These results once again indicate that MANN can be more effective and powerful for FSE task by integrating texts, images and concepts, capturing the text-image and text-concept associations, and measuring similar parts between two exercises.

4.5.3 ***Influence of Sampling Ways in Training.*** As discussed in Section 3.3, we have two sampling ways for training MANN, i.e. *Random* and *Concept*. In the following, we use the same testing set to investigate the influence of different sampling ways on the effectiveness of MANN. The experimental process is also repeated 10 times and the average results are reported.

The average results are shown in Figure 9. From this figure, we can observe that the MANN trained in *Concept* performs better than that in *Random*. We guess a possible reason is that during training MANN, for each given exercise, its similar exercises are very

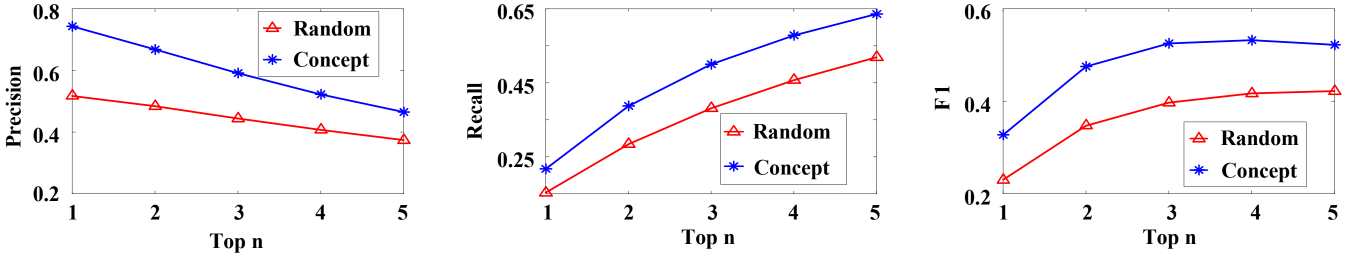
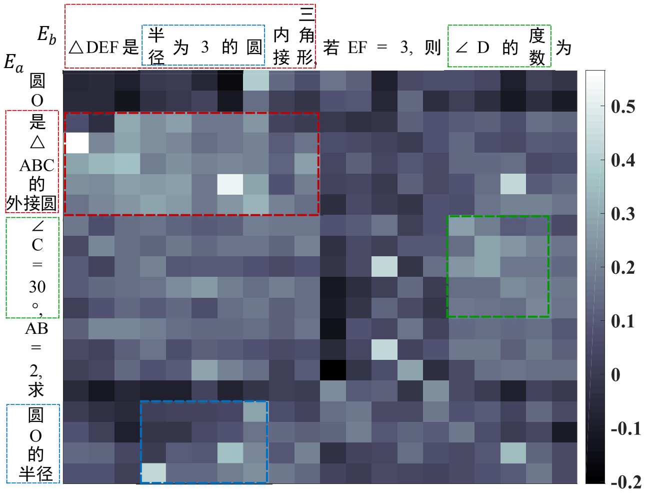


Figure 9: Influence of different sampling ways on MANN.

different from most sampled dissimilar ones in *Random* (sampling randomly) while its similar exercises are close to the dissimilar ones in *Concept* (sampling by concepts), so MANN can focus on the subtle differences between its similar pairs and dissimilar ones in *Concept*. Thus, for the FSE task, the model trained in the sampling way of *Concept* can be more powerful.

4.5.4 ***Case Study.*** One important characteristic of MANN is its explanatory power to measure the similar parts between two exercises from their semantic representations, via visualizing the similarity attention matrix $\boldsymbol{A}$ in Eq. (8). As an example, Figure 10 shows the similar parts between two exercises, E_a and E_b, and these exercises are also translated into English in the bottom of the figure to make them easy to understand. Please note that, we only show the text information of E_a and E_b in Figure 10 (the images and concepts are omitted) for the illustration purposes. The color in Figure 10 changes from white to black while the value of cosine similarity decreases. We can see that the parts in the green box (or blue, red box) in E_a and E_b are the similar parts that express the same meaning. For instance, the similar parts (i.e. “$\angle C = 30°$” and “the number of degrees of $\angle D$”) in the green box both describe the number of degrees of an angle. This implies that MANN provides a good way to capture the similarity information between exercises by the Similarity Attention.



E_a The circle O is a circumcircle of the triangle ABC, ∠C = 30° and AB = 2. Figure out the radius of The circle O.

E_b The triangle DEF is an inscribed triangle of a circle with a radius of 3. If EF = 3, the number of degrees of ∠D is __

Figure 10: Visualization of the similar parts between two example exercises E_a and E_b.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we provided a focused study on finding similar exercises (FSE) in online education systems. For modeling the heterogeneous materials of exercises semantically, a novel *M*ultimodal *A*ttention-based *N*eural *N*etwork (MANN) framework was proposed. Specifically, given the texts, images and concepts of exercises, we first utilized a CNN to generate the image representations and used an embedding layer to obtain representations of concepts. Then, we designed an Attention-based LSTM network to learn a unified semantic representation of each exercise in a multimodal way, where two attention strategies were proposed to capture the text-image and text-concept associations, respectively. Next, we designed a Similarity Attention to measure the similar parts in exercise pairs. Finally, a pairwise training strategy was proposed to return similar exercises. The experimental results on a large-scale real-world dataset clearly demonstrated both the effectiveness and explanatory power of MANN.

In the future, there are still some directions for further studies. First, besides the semantic similarity, we would like to measure the relation of exercises in more aspects, e.g. by considering the difficulty of exercises [13]. Second, as it is not easy to collect a massive number of similarity labels, we will also try to develop the semi-supervised or unsupervised learning methods for the FSE task. Finally, as our MANN is a general framework, we will test its performance on other disciplines (e.g. Physics), and meanwhile, on the similar applications in other domains, such as the measurement of product similarities in e-commerce [34].

6 ACKNOWLEDGEMENTS

This research was partially supported by grants from the National Key Research and Development Program of China (No. 2018YFB1004304), the National Natural Science Foundation of China (No. 61672483) and the Science Foundation of Ministry of Education of China & China Mobile (No. MCM20170507). Qi Liu gratefully acknowledges the support of the Young Elite Scientist Sponsorship Program of CAST and the Youth Innovation Promotion Association of CAS (No. 2014299).

REFERENCES

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* (2016).

[2] Hicham HAGE Esma AIMEUR. 2005. Exam question recommender system. *Artificial Intelligence in Education: Supporting Learning Through Intelligent and Socially Informed Technology* 125 (2005), 249.

[3] Yue Cao, Mingsheng Long, Jianmin Wang, Qiang Yang, and Philip S Yu. 2016. Deep visual-semantic hashing for cross-modal retrieval. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1445–1454.

[4] Yuying Chen, Qi Liu, Zhenya Huang, Le Wu, Enhong Chen, Runze Wu, Yu Su, and Guoping Hu. 2017. Tracking Knowledge Proficiency of Students with Educational Priors. In *ACM International on Conference on Information and Knowledge Management*. ACM, 989–998.

[5] Peng Cui, Shifei Jin, Linyun Yu, Fei Wang, Wenwu Zhu, and Shiqiang Yang. 2013. Cascading outbreak prediction in networks: a data-driven approach. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 901–909.

[6] Peng Cui, Shaowei Liu, and Wenwu Zhu. 2018. General Knowledge Embedded Image Representation Learning. *IEEE Transactions on Multimedia* 20, 1 (2018), 198–207.

[7] Teresa del Solato and Benedict Du Boulay. 1995. Implementation of motivational tactics in tutoring systems. *Journal of Interactive Learning Research* 6, 4 (1995), 337.

[8] Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850* (2013).

[9] Hicham Hage and E Aimeru. 2006. ICE: A system for identification of conflicts in exams. In *Computer Systems and Applications, 2006. IEEE International Conference on*. IEEE, 980–987.

[10] Robert Hecht-Nielsen. 1989. Theory of the backpropagation neural network. In *Neural Networks, 1989. IJCNN., International Joint Conference on*. IEEE, 593–605.

[11] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[12] Andrew G Howard. 2013. Some improvements on deep convolutional neural network based image classification. *arXiv preprint arXiv:1312.5402* (2013).

[13] Zhenya Huang, Qi Liu, Enhong Chen, Hongke Zhao, Mingyong Gao, Si Wei, Yu Su, and Guoping Hu. 2017. Question Difficulty Prediction for READING Problems in Standard Tests.. In *Thirty-First AAAI Conference on Artificial Intelligence*. 1352–1359.

[14] Andrej Karpathy, Armand Joulin, and Fei Fei F Li. 2014. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in neural information processing systems*. 1889–1897.

[15] Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[16] Vlasta Kokol-Voljc. 2000. Exam Questions When Using CAS for School Mathematics Teaching. *Algebra* 7 (2000), 13.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[18] Johan Lithner. 2004. Mathematical reasoning in calculus textbook exercises. *The Journal of Mathematical Behavior* 23, 4 (2004), 405–427.

[19] Qi Liu, Runze Wu, Enhong Chen, Guandong Xu, Yu Su, Zhigang Chen, and Guoping Hu. 2018. Fuzzy cognitive diagnosis for modelling examinee performance. *ACM Transactions on Intelligent Systems and Technology (TIST)* 9, 4 (2018), 48.

[20] Yuping Liu, Qi Liu, Runze Wu, Enhong Chen, Yu Su, Zhigang Chen, and Guoping Hu. 2016. Collaborative learning team formation: a cognitive modeling perspective. In *International Conference on Database Systems for Advanced Applications*. Springer, 383–400.

[21] Lin Ma, Zhengdong Lu, and Hang Li. 2016. Learning to Answer Questions from Image Using Convolutional Neural Network.. In *Thirtieth AAAI Conference on Artificial Intelligence*. 3567–3573.

[22] Iaroslav Melekhov, Juho Kannala, and Esa Rahtu. 2016. Siamese network features for image matching. In *International Conference on Pattern Recognition*. IEEE, 378–383.

[23] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[24] Youssef Mroueh, Etienne Marcheret, and Vaibhava Goel. 2015. Deep multimodal learning for audio-visual speech recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2130–2134.

[25] Jonas Mueller and Aditya Thyagarajan. 2016. Siamese Recurrent Architectures for Learning Sentence Similarity.. In *Thirtieth AAAI Conference on Artificial Intelligence*. 2786–2792.

[26] Zachary A Pardos and Anant Dadu. 2017. Imputing KCs with representations of problem content and context. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*. ACM, 148–155.

[27] Cesc C Park and Gunhee Kim. 2015. Expressing an image stream with a sequence of natural sentences. In *Advances in neural information processing systems*. 73–81.

[28] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *International Conference on Machine Learning*. 1310–1318.

[29] Jirí Rihák and Radek Pelánek. 2017. Measuring Similarity of Educational Items Using Data on Learners' Performance. In *Proceedings of the 10th International Conference on Educational Data Mining*. 16–23.

[30] Shuo Shang, Ruogu Ding, Bo Yuan, Kexin Xie, Kai Zheng, and Panos Kalnis. 2012. User oriented trajectory search for trip recommendation. In *Proceedings of the 15th International Conference on Extending Database Technology*. ACM, 156–167.

[31] Shuo Shang, Jiajun Liu, Kun Zhao, Mingrui Yang, Kai Zheng, and Jirong Wen. 2015. Dimension reduction with meta object-groups for efficient image retrieval. *Neurocomputing* 169 (2015), 50–54.

[32] Mohammad E Shiri, A Esma Aïmeur, and Claude Frasson. 1998. Student modelling by case based Reasoning. In *International Conference on Intelligent Tutoring Systems*. Springer, 394–403.

[33] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research* 15, 1 (2014), 1929–1958.

[34] Armin Stahl. 2006. Combining Case-Based and Similarity-Based Product Recommendation. In *European Conference on Advances in Case-Based Reasoning*. 355–369.

[35] Yu Su, Qingwen Liu, Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Chris Ding, Si Wei, and Guoping Hu. 2018. Exercise-Enhanced Sequential Modeling for Student Performance Prediction. In *Thirty-Second AAAI Conference on Artificial Intelligence*. 2435–2443.

[36] Avgoustos Tsinakos and Ioannis Kazanidis. 2012. Identification of conflicting questions in the PARES system. *The International Review of Research in Open and Distributed Learning* 13, 3 (2012), 297–313.

[37] Adrienne E Williams, Nancy M Aguilar-Roca, Michelle Tsai, Matthew Wong, Marin Moravec Beaupré, and Diane K O'Dowd. 2011. Assessment of learning gains associated with independent exam analysis in introductory biology. *CBE-Life Sciences Education* 10, 4 (2011), 346–356.

[38] Le Wu, Yong Ge, Qi Liu, Enhong Chen, Richang Hong, Junping Du, and Meng Wang. 2017. Modeling the evolution of users' preferences and social links in social networking services. *IEEE Transactions on Knowledge and Data Engineering* 29, 6 (2017), 1240–1253.

[39] Le Wu, Yong Ge, Qi Liu, Enhong Chen, Bai Long, and Zhenya Huang. 2016. Modeling users' preferences and social links in Social Networking Services: a joint-evolving perspective. In *Thirtieth AAAI Conference on Artificial Intelligence*. 279–286.

[40] Runze Wu, Qi Liu, Yuping Liu, Enhong Chen, Yu Su, Zhigang Chen, and Guoping Hu. 2015. Cognitive Modelling for Predicting Examinee Performance.. In *International Joint Conferences on Artificial Intelligence*. 1017–1024.

[41] Runze Wu, Guandong Xu, Enhong Chen, Qi Liu, and Wan Ng. 2017. Knowledge or Gaming?: Cognitive Modelling Based on Multiple-Attempt Response. In *Proceedings of the 26th International Conference on World Wide Web Companion*. International World Wide Web Conferences Steering Committee, 321–329.

[42] Ran Xu, Caiming Xiong, Wei Chen, and Jason J Corso. 2015. Jointly Modeling Deep Video and Compositional Text to Bridge Vision and Language in a Unified Framework.. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*. 2346–2352.

[43] Wenpeng Yin, Hinrich Schütze, Bing Xiang, and Bowen Zhou. 2016. ABCNN: Attention-Based Convolutional Neural Network for Modeling Sentence Pairs. *Transactions of the Association for Computational Linguistics* 4 (2016), 259–272.

[44] Jing Yu, Dongmei Li, Jiajia Hou, Ying Liu, and Zhaoying Yang. 2014. Similarity Measure of Test Questions Based on Ontology and VSM. *Open Automation and Control Systems Journal* 6 (2014), 262–267.

[45] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. 2010. Deconvolutional networks. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2528–2535.

[46] Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. 2017. Dynamic Key-Value Memory Networks for Knowledge Tracing. In *Proceedings of the 26th International Conference on World Wide Web*. 765–774.

[47] Hengshu Zhu, Hui Xiong, Yong Ge, and Enhong Chen. 2014. Mobile app recommendations with security and privacy awareness. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 951–960.

[48] Tianyu Zhu, Qi Liu, Zhenya Huang, Enhong Chen, Defu Lian, Yu Su, and Guoping Hu. 2018. MT-MCD: A Multi-task Cognitive Diagnosis Framework for Student Assessment. In *International Conference on Database Systems for Advanced Applications*. Springer, 318–335.