Fine-Grained Similarity Measurement between Educational Videos and Exercises

Xin Wang
Wei Huang
shenai@mail.ustc.edu.cn
ustc0411@mail.ustc.edu.cn
University of Science and Technology of China

Qi Liu*
qiliuql@ustc.edu.cn
School of Computer Science and Technology, University of Science and Technology of China

Yu Yin
Zhenya Huang
yxonic@mail.ustc.edu.cn
huangzhy@ustc.edu.cn
University of Science and Technology of China

Le Wu
lewu.ustc@gmail.com
Hefei University of Technology

Jianhui Ma
jianhui@ustc.edu.cn
University of Science and Technology of China

Xue Wang
1120180725@mail.nankai.edu.cn
Nankai University

ABSTRACT

In online learning systems, measuring the similarity between educational videos and exercises is a fundamental task with great application potentials. In this paper, we explore to measure the fine-grained similarity by leveraging multimodal information. The problem remains pretty much open due to several domain-specific characteristics. First, unlike general videos, educational videos contain not only graphics but also text and formulas, which have a fixed reading order. Both spatial and temporal information embedded in the frames should be modeled. Second, there are semantic associations between adjacent video segments. The semantic associations will affect the similarity and different exercises usually focus on the related context of different ranges. Third, the fine-grained labeled data for training the model is scarce and costly. To tackle the aforementioned challenges, we propose VENet to measure the similarity at both video-level and segment-level by just exploiting the video-level labeled data. Extensive experimental results on real-world data demonstrate the effectiveness of VENet.

CCS CONCEPTS

• **Information systems** → **Multimedia and multimodal retrieval**; • **Computing methodologies** → **Neural networks**; • **Social and professional topics** → **K-12 education**.

KEYWORDS

Multimodal Information; Educational Videos; Exercises; Fine-grained Similarity Measurement

*Corresponding Author.

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413783

ACM Reference Format:
Xin Wang, Wei Huang, Qi Liu, Yu Yin, Zhenya Huang, Le Wu, Jianhui Ma, and Xue Wang. 2020. Fine-Grained Similarity Measurement between Educational Videos and Exercises. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413783

1 INTRODUCTION

The last decade has witnessed the booming of online education platforms, such as Khan Academy[1] and Coursera[2]. As two main types of educational resources, millions of teaching videos and exercises have been generated and collected for learners of all ages [2]. Measuring the similarity between them is a fundamental task with great application potentials, such as bidirectional retrieval and recommendation [6, 41] based on content similarity. Generally, similar videos and exercises are those having common concepts (short for knowledge concepts [26, 45] or knowledge points [15]). Figure 1 shows an example of an educational video with three similar exercises. This video about drawing parabola consists of three segments denoted as S_1, S_2, and S_3. S_1 solves the quadratic equation to find the zeros of the function. S_2 finds its vertice according to the properties of quadratic functions. S_3 graphs the parabola according to the zeros and vertice. Exercise E_2 is completely similar to the whole video because they have the same concepts about graphing a parabola. In most cases, an exercise is only similar to parts of the educational video instead of all of it. For instance, E_1 is only similar to S_1 and E_3 is only similar to S_3. Therefore, it would be of great significance to interpretability and user experience [4, 29] if we could further measure the similarity at segment-level, which we call fine-grained similarity measurement.

Recommender systems have been successfully applied to enhance the quality of service for customers in many fields [19, 47]. Several approaches have been proposed for video segment retrieval and recommendation [24, 39, 44]. Most existing methods based on the text-similarity only consider textual materials while ignoring the visual information. For example, YouEDU [1] automatically recommends related video snippets for the forum posts based on

[1] All Khan Academy content is available for free at www.khanacademy.org
[2] https://www.coursera.org

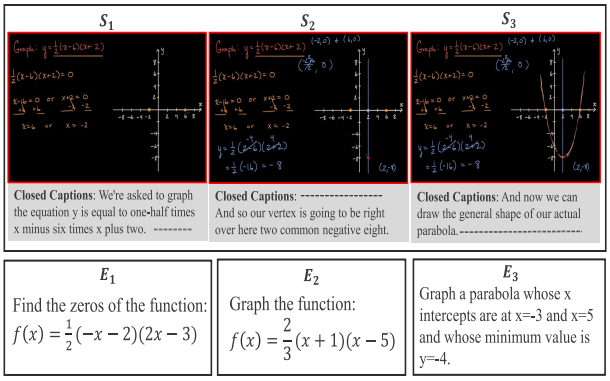

Figure 1: An example of an educational video from Khan Academy and its three similar exercises.

the cosine similarity between the closed captions and the post description. However, the visual information can be used to enhance the sentence semantic understanding [46] and the recommendation performance [38]. Besides, most of them are heavily dependent on the scarce labeled data on segments. In this paper, we explore to make the best of the multimodal information to understand the video accurately and then measure the fine-grained similarity by just exploiting the labeled data on videos.

Despite its value and significance, fine-grained similarity measurement between educationl videos and exercises remains immature due to the following domain-specific challenges: (1) First, unlike general videos, educational videos contain not only graphics but also text and formulas, which have a fixed reading order, i.e., from left to right and from top to bottom. Both spatial structure (graphics) and temporal information (text and formulas) embedded in the frames should be modeled. (2) Second, there are semantic associations between adjacent video segments. The semantic associations will affect the similarity and different exercises usually focus on the related context of different ranges. As shown in Figure 1, when we measure the similarity between S_2 and E_2, the context (i.e. S_1 and S_3) will enhance their similarity. How to perceive and incorporate the context of the appropriate range is one of the biggest obstacles. (3) The segment-level labeled data is scarce and costly, whereas the video-level labeled data is much easier to obtain. How to take full advantage of this coarse-grained labeled data to learn the fine-grained similarity is also a great challenge.

To tackle the aforementioned challenges, we propose a novel method, namely VENet, for the fine-grained similarity measurement task. Specifically, we devise a multimodal representing layer (MRL) to obtain the semantic representation of the heterogeneous data. In MRL, the textual, spatial and temporal information is jointly modeled. Then a multiscale perceptual fusion (MPF) network is used to fuse the context information on multiple scales. Finally, we develop a pairwise training strategy to learn the fine-grained similarity by just exploiting the coarse-grained labeled data. Extensive experimental results on real-world data clearly demonstrate the effectiveness of VENet.

The main contributions of this work are summarized as follows: (1) We explore the promising yet challenging problem of measuring the fine-grained similarity between educational videos and exercises by just exploiting the coarse-grained labeled data. (2) We propose a novel method, namely VENet, to measure the fine-grained similarity by jointly representing the heterogeneous data and capturing their semantic association. (3) We create and show how to create a related dataset using publically available educational services.

2 RELATED WORK

The related work of this study can be summarized into the following three categories: similarity measurement in education, multimodal video representation, and pixel temporal modeling.

2.1 Similarity Measurement in Education

In the literature, several efforts have been made to measure the similarity between the same kind of educational items. For example, Liu et al. [20] developed a novel Multimodal attention-based Neural Network (MANN) framework for finding similar exercises by learning a unified semantic representation from heterogeneous data. MacHardy et al. [22] leveraged an adaptation of traditional Bayesian Knowledge Tracing (BKT) to evaluate the relevance of educational videos. Wang et al. [36] contributed a similarity ranking-based unsupervised approach to measure the originality of coursewares.

However, the problem of measuring the similarity between educational videos and exercises remains pretty much open.The only related work we aware of is YouEDU [1], which automatically recommends video segments to questions based on the cosine similarity between closed captions and question description. Along this line, methods for modeling text pairs [23, 42] can be applied to learn the similarity of video-exercise pairs based on their textual materials. However, the visual information can be used to enhance the sentence semantic understanding [46] and the recommendation performance [38]. Therefore, in VENet, we exploit the visual data as supplement information to the closed captions to accurately understand and represent the video. Moreover, most existing methods are heavily dependent on the segment-level labeled data which is scarce and costly. Comparatively, we aim to measure the similarity by just exploiting the coarse-grained labeled data.

2.2 Multimodal Video Representation

Generally, videos contain multimodal data, such as audio, frames, captions and other auxiliary information. Several efforts have been made to represent the whole video by leveraging these hybrid multimodal data. For example, Ramanishka et al. [28] proposed MMVD (Multimodal Video Description) to exploit frames, audio and text labels for generating video descriptions. Xu et al. [40] proposed a dependency-tree structure model which embeds a sentence into a continuous vector space, and leveraged deep neural networks to capture essential semantic information from frame sequence. Nevertheless, modeling the whole video into a fixed-length semantic vector is not suitable for fine-grained similarity measurement. To handle this problem, an effective framework is to divide the whole video into segments and then encode the semantic information of each segment respectively. For instance, Xu et al. [39] injected text features to help eliminate unlikely clips and then used visual features to modulate the processing of query sentences at the word

Table 1: The statistics of the dataset

Data	Statistics	Values
Exercise	Num of exercises	17,116
	Avg. words per exercise	34.95
	Avg. similar videos per exercise	1.67
Video & Captions	Num of videos	1,053
	Avg. length per video	383.79s
	Total length	404,130s
	Total size	22.6GB
	Avg. words per closed captions	831.78
	Avg. similar exercises per video	17.04
Label	Num of similar pairs	10,679
	Num of dissimilar pairs (negative sampling)	10,679

level in a recurrent neural network. What's more, many multimodal learning methods on handling images (frames) and text can also be applied for modeling segments [3, 7, 17, 21, 27].

However, unlike general videos, educational videos contain not only graphics but also text and formulas, which have a fixed reading order, i.e., from left to right and from top to bottom. Both the spatial structure (graphics) and temporal information (text and formulas) embedded in the frames should be modeled. Therefore, most multimodal learning approaches that only capture the spatial structure of images are infeasible for educational videos.

2.3 Pixel Temporal Modeling

In VENet, one of the key points is to model the temporal information embedded in the frames, which is related to pixel temporal modeling. There have been some efforts to model pixel sequences for various tasks in computer vision. For example, based on the basic RNNs, Graves et al. [10] proposed multi-dimensional recurrent neural networks (MDRNNs) for multi-dimensional sequence data. The basic idea of MDRNNs was to replace the single recurrent connection found in standard RNNs with as many recurrent connections as dimensions in the data. Theis et al. [31] used the conditional distribution of a mixture of GSMs (gaussian scale mixtures) [34, 37] to model the distribution of a pixel given its causal neighborhood. Based on the above work, Theis and Bethge [30] proposed spatial LSTM and produced the promising results in modeling grayscale images and textures. Van Oord et al. [33] proposed Row LSTM and Diagonal BiLSTM to model the temporal dependencies between pixels in different directions.

Unfortunately, the modeling direction of temporal dependencies of the education frame is different from that of the general image. Therefore, these existing methods could not be directly applied to learn the temporal information (text and formulas) embedded in the educational video frames.

3 PRELIMINARIES

In this section, we first give a clear description of the dataset used in this paper and then introduce some important details of data preprocessing. Finally, we give the formal definition of the fine-grained similarity measurement problem.

3.1 Data Description

As far as we know, there is no public dataset of similar video-exercise in education. So we collect the the real-word data from Khan Academy[3] which offers practice exercises and instructional videos, including K-14 and test preparation content. In Khan Academy, learners are allowed to study at their own pace in and outside of the classroom. All of our data is crawled from the math domain, which contains 17,116 math exercises and 1,053 educational videos with closed captions, covering 836 topics. In Khan Academy, most educational videos will be followed by several similar exercises, according to which we obtained 10,679 similar video-exercise pairs. Then we build the dissimilar pairs by negative sampling. Specifically, for a video, we treat the exercises that share no common topics as its dissimilar exercises and randomly sample from them. It is worth noting that all the labeled data is for the entire video, so we did not use any annotation information on segments in the training phase.

Some important statistics are shown in Table 1. We can observe that the dataset is clearly heterogeneous, containing textual materials (captions and exersies) and frames. Besides, even though the closed captions and exercises are both textual materials, there are great differences between them. The closed captions are much closer to oral presentation, while the exercise descriptions are more professional and concise, usually containing many mathematical terms and formulas. So exercises are usually much shorter than closed captions. Due to the differences, it is difficult to measure their semantic similarity just based on the text materials. Therefore, it is necessary to exploit massive visual data as supplement information to enhance the video understanding in a multimodal way.

3.2 Data Preprocessing

3.2.1 Video Preprocessing. As shown in Figure 2, video preprocessing includes the processing of frame sequence and closed captions. For the videos without closed captions, we first transcribe the audio into captions by speech recognition. Then we divide the whole video into several segments by video segmentation. After that, we extract the keyframes from the segments and use them to guide the caption segmentation. Finally, we obtain the video segments by combining the keyframe and its corresponding captions.

You can flexibly configure the video segmentation algorithms (e.g. shot boundary detection [11, 16]) according to the video characteristics. Observing that the educational video pictures are incremental, that is, the next frame can cover the previous one. We argue that the end of this increment usually means the end of complete semantics, which can be used to guide the video segmentation. Based on the above analysis, we develop the Adaptive Block Matching (ABM) which can perform video segmentation and keyframe extraction simultaneously. The main steps of ABM be summarized as follows:

Step 1: Select the last frame as the keyframe of the last segment.
Step 2: Judge whether the current frame can be covered by the previous keyframe from back to front:
  a) Divide the current frame several nonoverlapping blocks.

[3] https://www.khanacademy.org

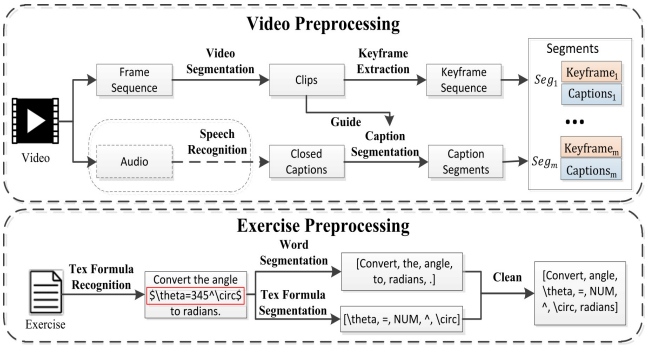

Figure 2: The flow chart of data preprocessing.

b) For each block, calculate the matching score between it and the previous keyframe by the template-matching algorithm of OpenCV [4].
c) If the number of blocks not included in the previous keyframe exceeds the threshold, the current frame can not be covered by the previous key frame, and then take the current frame as the keyframe of the new clip.
d) If the number of blocks that do not match the previous keyframe exceeds the threshold, the current frame is selected as the keyframe for the new clip.

Step 3: Repeat step 2 until all the frames are processed.

Finally, each video is divided into several segments and each segment consists of a keyframe and a corresponding caption. On average, each video contains 3.52 segments, and each caption segment contains 104.9 words.

3.2.2 Exercise Preprocessing. Unlike general text materials, the exercises usually contain many TeX formulas which can not be handled like normal text. As shown in Figure 2, we first identify the TeX formulas by the special symbol '$'. For plain text, we can segment words by spaces. As for the TeX formulas, we develop a TeX parsing tool [5] which treats the TeX commands as special words. Then we clean the word sequences, e.g. remove the stop words and meaningless symbols. Finally, we obtain the word sequences of the exercises.

3.3 Problem Definition

The input of the fine-grained similarity measurement task is heterogeneous data, including a multimodal educational video V and an exercise E. As mentioned above, the video consists of several segments $V = \{Seg_1, \cdots, Seg_m\}$, and each segment Seg_i consist of one keyframe kf_i and a corresponding caption segment c_i, where the kf_i is an image in size $H \times W$ and $c_i = \{w_{i1}, w_{i2}, \cdots, w_{it}\}$ is a sequence of words. The exercise can also be processed into a sequence of words, i.e. $E = \{w_1, w_2, \cdots, w_n\}$.

With the setup stated above, for any educational video V and exercise E, $S_v(V, E)$ denotes the similarity score between the whole video V and exercise E, and $S_s(Seg_i, E)$ denotes the similarity score between the i-th segment and E. Both the similarity scores S_v and S_s are real numbers between 0 and 1, the higher, the more similar. Without loss of generality, we define the problem of fine-grained similarity measurement as follows:

Definition 3.1. (Fine-Grained Similarity Measurement). Given an educational video $V = \{Seg_1, \cdots, Seg_m\}, Seg_i = \{kf_i, c_i\}, c_i = \{w_{i1}, w_{i2}, \cdots, w_{it}\}$ and an exercise $E = \{w_1, w_2, \cdots, w_n\}$, our goal is two-fold: **(1)** Measure the similarity score $S_v(V, E)$ between V and E. **(2)** Measure the similarity score $S_s(Seg_i, E)$ between Seg_i $(i \in 1, \cdots, m)$ and E.

4 VENET FRAMEWORK

In this section, we will introduce VENet model architecture in detail. As shown in Figure 3, VENet mainly contains three parts: Multimodal Representing Layer (MRL), Multiscale Perceptual Fusion (MPF) and Similarity Score Layer (SSL).

4.1 Multimodal Representing Layer

4.1.1 ***Segment Representing Network (SRN).*** The purpose of SRN is to encode the multimodal information of video segments into semantic vectors. As shown in Figure 4, the input of SRN is a segment, including a keyframe and a corresponding closed captions. For the keyframe, we first utilize a CNN architecture with two layers of convolution and max-pooling to get the primary feature map $pr^f \in \mathbb{R}^{p \times q}$. This step abstracts low-level visual features into high-level semantic information and it reduces the resolution of the keyframe, which greatly improves the efficiency of subsequent temporal modeling. For the captions, we first initialize the words with the pre-trained word embedding with *GloVe* [25], and then exploit the LSTM to obtain the semantic vector $r^c \in \mathbb{R}^{d_1}$. The modeling process of the captions is the same as the exercises, which will be described in detail in the Subsection 4.1.2.

Considering that the captions usually focus on only parts of the keyframe and the importance of different areas of the keyframe is different, so the semantic alignment between them is necessary. Attention mechanism is a powerful approach to highlight different parts of the semantic representation [14]. Here we use an attention module (denoted as *F2C Att*) to identify the important areas according to the semantic representation r^c of its corresponding captions. In essence, the attention module is to assign different weights to different areas, which can be expressed as follows:

$$
\begin{aligned}
r^f_{att}[i,j] &= \alpha_{ij} \cdot pr^f_{ij}, \\
\alpha_{ij} &= \frac{\varphi\left(pr^f_{ij}, r^c\right)}{\sum_{k=1}^{p}\sum_{t=1}^{q}\varphi\left(pr^f_{kt}, r^c\right)}, \\
\varphi\left(pr^f_{ij}, r^c\right) &= exp\left(V_{f2c} \cdot tanh\left(W_{f2c} \cdot \left[pr^f_{ij}, r^c\right]\right)\right),
\end{aligned}
\tag{1}
$$

where W_{f2c} and V_{f2c} are learnable parameters, $r^f_{att}[i,j]$ represents the weighted semantic vector at the location (i,j) and $\alpha_{ij} \in [0,1]$ is its weight calculated by normalizing the importance scores φ.

Unlike general videos, educational videos contain not only graphics but also text and formulas, which have a fixed reading order, i.e., from left to right and from top to bottom. Both the spatial structure (graphics) and temporal information (text and formulas) embedded in the keyframes should be modeled. Therefore, we exploit the CNN

[4] OpenCV (https://opencv.org/) is an open source computer vision software library.
[5] The tool is available at https://github.com/bigdata-ustc/stn

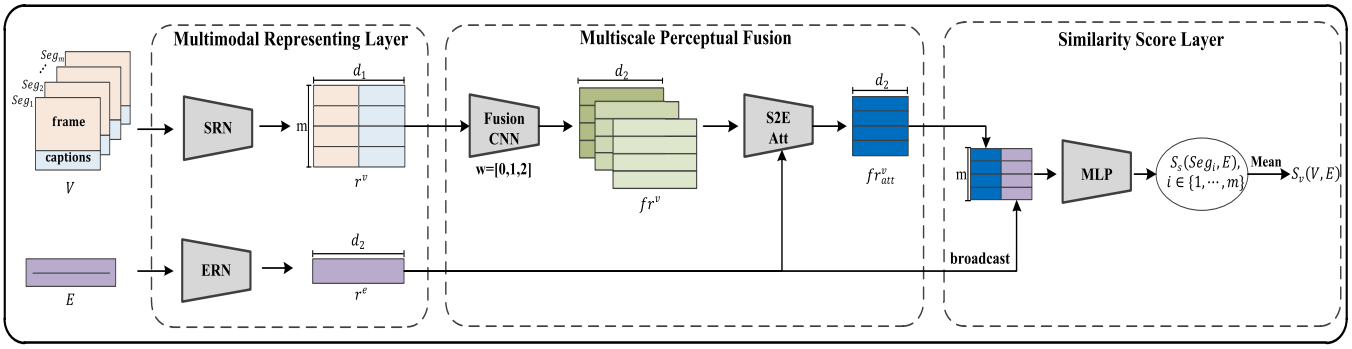

Figure 3: The VENet model architecture consists of three main parts: 1) Multimodal Representing Layer (MRL), 2) Multiscale Perceptual Fusion (MPF), and 3) Similarity Score Layer (SSL). VENet takes a video-exercise pair (V, E) as input and outputs the similarity score $S(Seg_i, E), i \in \{1, \cdots, m\}$ and $S(V, E)$.

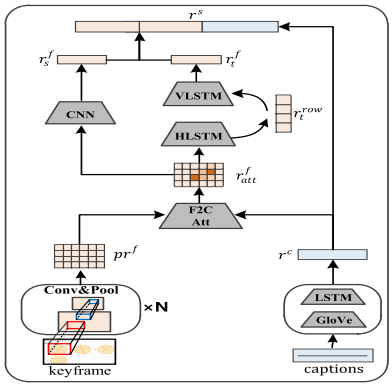

Figure 4: Segment Representation Network.

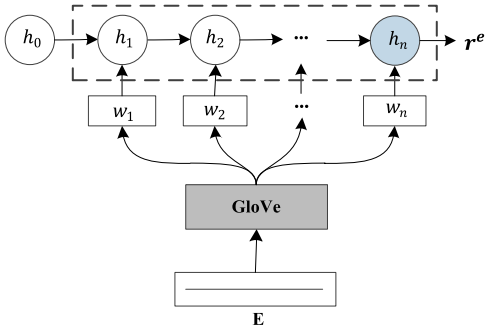

Figure 5: Exercise Representation Network

to capture the spatial information of the keyframe and obtain the spatial semantic representation r_s^f. Then we exploit two LSTM [13] networks successively, which we call Horizontal LSTM (HLSTM) and Vertical LSTM (VLSTM), to model the horizontal and vertical temporal information respectively. Specifically, we first pass each row of r_{att}^f to the HLSTM and take the last hidden state as the row representation r_t^{row}. Then the VLSTM models the temporal dependencies in vertical direction and obtain the temporal representation r_t^f of the keyframe. Finally, we obtain the spatial information r_s^f and the temporal information r_t^f of the keyframe. After that, we concatenate r_s^f, r_t^f, and r^c into the multimodal semantic representation r^s of the video segment.

4.1.2 ***Exercise Representing Network (ERN).*** As mentioned in Subsection 3.2.2, the exercises are preprocessed into word sequences. As shown in Figure 5, the words are initialized by a d_0-dimensional pre-trained word embedding with *GloVe* [25]. After that, we obtain the embedding vector sequence $E = (w_1, w_2, \cdots, w_n)$, where $w_i \in \mathbb{R}^{d_0}$ and n is the length of the exercise. As LSTM [13] can handle temporal sequence and learn long-range dependencies [9], we exploit a LSTM architecture to model the word sequence and obtain the semantic vector r^e of the exercise.

4.2 Multiscale Perceptual Fusion

There are semantic associations between adjacent video segments. The context information is very helpful to accurately understand the semantic content of the target segment. Besides, different exercises usually focus on the different context scales of the target segment. To incorporate the context semantics of the target segment, multiscale perceptual fusion (MPF) fuses adjacent segments on multiple scales and then selects the suitable one for the given exercise by utilizing the attention mechanism.

As shown in Figure 3, after obtaining the video representation r^v consisting of m segments, we fuse the neighboring segments by the Fusion CNN as follows:

$$\begin{aligned} fr_i^v &= ReLU\left(W_{fuse}\widetilde{C}_i + b_{fuse}\right), \\ \widetilde{C}_i &= \left[r_{i-w}^v, \cdots, r_i^v, \cdots, r_{i+w}^v\right], \end{aligned} \tag{2}$$

where W_{fuse} and b_{fuse} are the convolution weight and bias, and w is the perception scale. Here, we use multiple perception scales (i.e. $w = [0, 1, 2]$) and then we obtain the fusional representation $fr^v \in \mathbb{R}^{m \times d_2 \times 3}$ with three channels, each of which is the result from one perception scale.

After that, we exploit another attention module denoted as **S2E** to select the suitable one from the fusional channels according to the exercise. Specifically, we weight the fusional channels according to the exercise and then sum them up to $fr_{att}^v \in \mathbb{R}^{m \times d_2}$. The process

can be formulated as follows:

$$fr^{v}_{att}[i] = \sum_{k=1}^{3} \alpha_{ki} \cdot fr^{v}_{ki}, \quad \alpha_{ki} = \frac{\varphi\left(fr^{v}_{ki}, r^{e}\right)}{\sum_{t=1}^{3} \varphi\left(fr^{v}_{ti} r^{e}\right)}, \tag{3}$$

where the function φ is the same as Equation 1.

4.3 Similarity Score Layer

Similarity Score Layer calculates the similarity score between each segment and the exercise based on their comprehensive semantic representation. As shown in Figure 3, we first broadcast the representation vector r^e of the exercise to all the segments. Then each concatenate vector $\widetilde{Z}_i = \left[fr^{v}_{att}[i], r^e\right]$ is passed to a two-layer MLP (Multilayer Perceptron) with a nonlinear activation function $ReLU(x) = \max(0, x)$ used in the first layer and the sigmoid function for the second one:

$$\begin{aligned} V &= ReLU\left(W_1 \widetilde{Z}_i + b_1\right), \\ S_s(Seg_i, E) &= \sigma(W_2 V + b_2), \end{aligned} \tag{4}$$

where the W_1, b_1, W_2, b_2 are learnable parameters of the MLP. After obtaining the similarity score of each segment, we take the average of them as the similarity score of the whole video.

4.4 Training VENet

As mentioned in Subsection 3.1, we only have binary labels at the video-level. In this subsection, we specify a pairwise loss function for training VENet to learn the similarity at both video-level and segment-level by just exploiting the video-level labels.

For an educational video V, we denote its similar exercise set as SE_v and dissimilar exercise set as DE_v. Given an educational video V, we assume that the similarity score $S_s(Seg_i, E_s)$ should be higher than $S_s(Seg_j, E_ds)$, where Seg_i and Seg_j are both the segments of V, $E_s \in SE_V$, and $E_ds \in DE_v$. Based on the above reasonable assumptions, we formulate the pairwise loss function as follows:

$$\begin{aligned} \mathcal{L}(V, E_s, E_{ds}; \Theta) = \sum_{Seg_i \in V} \sum_{Seg_j \in V} \max(0, \mu - (S_s(Seg_i, E_s) \\ - S_s\left(Seg_j, E_{ds}\right))) + \lambda \|\Theta\|^2, \end{aligned} \tag{5}$$

where Θ denotes all learnable parameters of VENet, λ is the regularization hyperparameter, and μ is the margin forcing $S_s(Seg_i, E_s)$ to be higher than $S_s\left(Seg_j, E_{ds}\right)$ by μ. Finally, we can train VENet by minimizing the loss function $\mathcal{L}$ using Adam.

5 EXPERIMENTS

In this section, we first build a test dataset to assess the performance of VENet comparing with several baselines on the fine-grained similarity measurement task. Then, we conduct an ablation study of VENet to verify the effectiveness of several key modules. Finally, we show the effectiveness of VENet intuitively by a case study.

5.1 Experimental Setup

5.1.1 ***Test Dataset.*** Since the dataset only has video-level labels, we built the test dataset to assess the performance of the comparison methods on the fine-grained similarity measurement task. Specifically, we first randomly selected one hundred educational videos, which have been divided into segments. Then we built the candidate exercises for each test video based on the video-level labels, including five similar and five dissimilar exercises. After that, ten educational experts were invited to score each segment for all of its candidate exercises on a five-point scale (i.e. from 0 to 4). We took the average score of all the segments as the similarity score of the whole video. For both segments and videos, we treated those exercises with similarity score less than 2 as dissimilar exercises. Finally, we obtained the test dataset with fine-grained similarity score on segments. It is worth noting that all the videos and exercises for testing were removed from the training data.

5.1.2 ***Evaluation Metrics.*** We comprehensively evaluated the classification performance and ranking performance of the model at both the video-level and segment-level. At the segment-level, given an exercise, we evaluated the similarity score of all segments of the same video. At the video-level, given a video, we evaluated the similarity score of all candidate exercises. The classification performance is evaluated with the widely used metric AUC [12]:

$$AUC = \frac{1}{|SP| \times |DP|} \sum_{sp \in SP} \sum_{dp \in DP} \delta(S_\bullet(sp) > S_\bullet(dp)), \tag{6}$$

where SP and DP are respectively the sets of similar pairs and dissimilar pairs, $S_\bullet$ is the similarity score S_v or S_s, and $\delta(x)$ is an indicator function that returns 1 iff x is true. The ranking performance is evaluated with the widely used metric NDCG@K [5, 35]:

$$\begin{aligned} NDCG@K &= \frac{DCG@K}{IDCG@K}, \\ DCG@K &= \sum_{i=1}^{k} \frac{2^{s_i} - 1}{\log_2(i+1)}, \end{aligned} \tag{7}$$

where $s_\bullet$ is the similarity scores of the ordering result and IDCG@k is the DCG@k of an ideal ordering result. In the experiment, we set the K equal to 10. Both AUC and NDCG are real numbers from 0 to 1, and the higher the better.

5.1.3 ***VENet Setup.***

(1) **Word Embedding.** The words in the vocabulary were initialized by the pre-trained word embedding of *GloVe* [25] with dimension (d_0) 300 and others (e.g. Tex symbols) were randomly initialized with the same dimension.
(2) **SRN and ERN.** For all the LSTM architecture in SRN and ERN, we set the units number to 100, thus $d_2 = 100$ and $d_1 = 2d_2 = 200$. The kernel sizes of the two-layer convolution were $[5 \times 5]$ and $[3 \times 3]$ respectively. The kernel sizes of the two max-pooling were both set to $[3 \times 3]$.
(3) **MPF and SSL.** In MPF, we utilized three convolution kernel with different size ($w = 0, 1, 2$). In SSL, the number of hidden units for the MLP was set to 100, and we also used dropout with the probability 0.5 to prevent overfitting.
(4) **Training Details.** We initialized parameters of VENet with a truncated normal distribution with the standard deviation 0.1. We set $\mu = 0.3$ and $\lambda = 0.0001$ in Equation 5. The initial learning rate was set to 0.0001 and it decreased every epoch with the decay rate 0.99.

Table 2: Characteristics of the comparison methods

Model	Input		Task	
	Text	Frame	Video-Level	Segment-Level
MaLSTM	✓	×	✓	×
DeepLSTM	✓	×	✓	×
ABCNN	✓	×	✓	×
TextCNN	✓	×	✓	×
DeepLSTM (Seg)	✓	×	✓	✓
TextCNN (Seg)	✓	×	✓	✓
TextualVENet	✓	×	✓	✓
3DCNN	✓	✓	✓	×
JSFusion	✓	✓	✓	×
EarlyFusion	✓	✓	✓	✓
VENet	✓	✓	✓	✓

Table 3: Performance of comparison methods

Model	Video-Level		Segment-Level	
	Auc	NDCG	Auc	NDCG
MaLSTM	0.591	0.635	-	-
DeepLSTM	0.778	0.7503	-	-
ABCNN	0.764	0.7448	-	-
TextCNN	0.792	0.771	-	-
DeepLSTM (Seg)	0.844	0.7728	0.754	0.7437
TextCNN (Seg)	0.806	0.7658	0.7418	0.7415
TextualVENet	0.876	0.832	0.768	0.781
3DCNN	0.654	0.742	-	-
JSFusion	0.826	0.788	-	-
EarlyFusion	0.854	0.7806	0.7863	0.7494
VENet	**0.942**	**0.879**	**0.871**	**0.823**

5.2 Baseline Approaches

In order to demonstrate the effectiveness of our proposed model, we compare VENet with the representative state-of-the-art works including textual and multimodal models:

- **MaLSTM** [23] utilizes the LSTM architecture to learn the semantic representation of the captions and exercises and then the similarity is measured by the Manhattan distance.
- **DeepLSTM** substitutes the SSL of VENet for the Manhattan distance of MaLSTM to calculate the similarity score.
- **ABCNN** [42] utilizes convolutional neural network to model sentence pairs, where the attention mechanism is used at each convolutional layer.
- **TextCNN** [18] is a representative CNN-based model for sentence classification, which is used to represent the closed captions and the exercises.
- **DeepLSTM (Seg)** has the same architecture with DeepLSTM, but its input is video segment rather than the whole video.
- **TextCNN (Seg)** has the same architecture with TextCNN, but its input is video segment rather than the whole video.
- **TextualVENet** is a variant of VENet which only exploits the textual materials and ignores the frames.
- **3DCNN** [32] is a representative approach for video modeling with 3-dimensional convolutional networks. Here, we exploit it to model the keyframe sequence of the video and exploit LSTM to model the closed captions and exercises.
- **JSFusion** [43] is a relatively new model, which can measure semantic similarity between any pairs of multimodal sequence data.
- **EarlyFusion** [39] integrates language and vision more closely using an early fusion scheme, which can be used to model the video segment.

These methods are divided into four categories according to the input and task granularity. The characteristics of them are listed in Table 2. Due to the limited training data, some highly complex models are not included in our baseline, such as BERT [8] and its variant M-BERT [27]. For a fair comparison, all these methods are adjusted to contain approximately the same amount of parameters and all of them are tuned to have the best performance. All models are implemented in Tensorflow and trained on a Linux server with four 2.0GHz Intel Xeon E5-2620 CPUs and a Tesla K20m GPU.

5.3 Experimental Results

5.3.1 ***Performance Comparison.*** Table 3 shows the performance results of all comparison methods. We can easily see that our proposed VENet achieves the best performance at both video-level and segment-level, with a significant improvement on all metrics compared to other methods. Moreover, TextualVENet also performs best in the seven textual models. Further analyzing the results, we can get more observations.

First, the performance of MaLSTM is terrible with the AUC of 0.59. We believe that the main reason is the great differences between the closed captions and the exercises. As mentioned before, the closed captions are closer to oral presentation, while exercise descriptions are more professional and concise, usually containing many mathematical terms and formulas. Therefore, it is difficult to encode them into the same semantic space and measure their similarity by Manhattan distance. Second, comparing DeepLSTM and DeepLSTM (Seg), we can find that dividing video into segments can improve the performance at video-level significantly. We argue that there are two main reasons: 1) Video segmentation greatly reduces the length of video frames and captions, which alleviates the problem of long-range dependencies. 2) More information can be retained by representing a video into several segment vectors instead of one fixed-length vector, which is helpful to discover more partial similarities. Third, the performance of TextualVENet is worse than that of VENet, which shows that the visual data is helpful to accurately understand the video and measure the similarity precisely. However, simply stacking the multimodal information does not necessarily improves the performance (e.g. 3DCNN). The key is to effectively represent the semantic information embedded in multimodal data and align them between various modal data.

5.3.2 ***Ablation Experiments.*** We first studied the effect of the visual and textual information on the similarity measurement. As shown in Table 4, we find that the performance of VisualVENet is much worse than TextualVENet, which indicates the textual material is more important than the visual data. In line with our intuition,

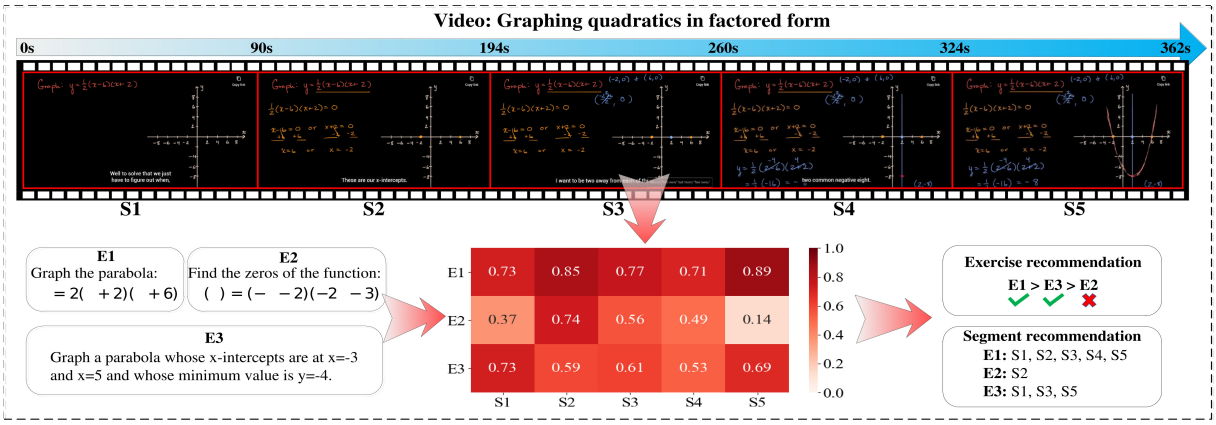

Figure 6: A case study of the similarity measurement for an educational video and three exercises.

closed captions as the detailed description of the video contain most of the semantic information. However, the performance of VisualVENet is better than a random guess (AUC=0.5), which indicates that our method does extract effective information from the visual data to help similarity measurement. Therefore, the VENet which exploits the visual data as supplement information to the closed captions performed better than VisualVENet and TextualVENet.

To further study how each part affects the final results, we design another four variants of VENet, each of which takes out one key module. As shown in Table 4, all the key modules (i.e., F2C, S2E, HVLSTM and MPF) have a significant impact on the final result, which shows the effectiveness of them. Besides, we can find that the performance degradation is greatest when MPF is removed, which indicates the semantic associations between adjacent segments is important to the similarity measurement.

5.3.3 ***Case Study of Similarity Measurement.*** Figure 6 shows a case study of the similarity measurement for a teaching video [6] and three exercises. The video about graphing quadratics in factored form consists of five segments each of which explains one step. The heatmap in Figure 6 shows the similarity scores between each segment and exercise. We can find that the similarity score of E_2 and S_5 is very low. If we go into them, we can see that E_2 is about finding the zeros of the function, and S_5 is about drawing a parabola based on the coordinate points. They are very different and there is almost no intersection between their textual materials.

Based on the similarity score, we can conduct bidirectional recommendation or retrieval between educational videos and exercises. For example, we can recommend the exercises to this educational video by the ranking $< E_1, E_3, E_2 >$, where E_1 and E_3 are similar exercises while E_2 is dissimilar exercise. According to the fine-grained similarity score of the segments, we can further identify the similar segments for each exercise. An interesting finding is that although E_2 is dissimilar to the whole video, VENet can still discover the potential similar segment S_2 for it, which fully shows the great application prospect of VENet.

[6]The complete video titled **graphing quadratics in factored form** is available at https://www.khanacademy.org/math/math2/xe2ae2386aa2e13d6:quad-2.

Table 4: Ablation Experiments

Model	Video-Level		Segment-Level	
	Auc	NDCG	Auc	NDCG
TextualVENet	0.876	0.832	0.768	0.781
VisualVENet	0.624	0.7328	0.6324	0.6931
VENet	**0.942**	**0.879**	**0.871**	**0.823**
VENet-F2C	0.9	0.855	0.8284	0.8198
VENet-S2E	0.91	0.851	0.846	0.8137
VENet-HVLSTM	0.89	0.802	0.803	0.795
VENet-MPF	0.866	0.815	0.789	0.7616

6 CONCLUSION

In this paper, we explore the promising yet challenging problem of measuring the fine-grained similarity between educational videos and exercises by just exploiting the coarse-grained labeled data. We propose a novel method, namely VENet, to handle this problem. VENet exploits the visual data as the supplement information to the closed captions to accurately understand and represent the video. Specifically, VENet models both spatial and temporal information embedded in the keyframes by SRN and then captures the semantic associations between segments by MPF. Finally, we use a pairwise training strategy to learn the similarity at both video-level and segment-level by just exploiting the coarse-grained annotation which is much easier to obtain. The experimental results on real-world data clearly demonstrate the effectiveness of VENet.

Due to dataset limitations, we only verified the effectiveness of VENet on the math subject. In the future, we will collect dataset and conduct experiments to test the performance on other subjects such as Physics. We also plan to consider other meta information, such as topics and titles.

ACKNOWLEDGMENTS

This research was supported by grants from the National Natural Science Foundation of China (Grants No. 61922073, 61672483, U1605251, 61972125). Qi Liu gratefully acknowledges the support of the Youth Innovation Promotion Association of CAS (No. 2014299).

REFERENCES

[1] Akshay Agrawal, Jagadish Venkatraman, Shane Leonard, and Andreas Paepcke. 2015. YouEDU: addressing confusion in MOOC discussion forums by recommending instructional video clips. (2015).

[2] Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. 2014. Engaging with massive online courses. In *Proceedings of the 23rd international conference on World wide web*. ACM, 687–698.

[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] Jingyuan Chen, Hanwang Zhang, Xiangnan He, Liqiang Nie, Wei Liu, and Tat-Seng Chua. 2017. Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*. ACM, 335–344.

[5] Lei Chen, Le Wu, Richang Hong, Kun Zhang, and Meng Wang. 2020. Revisiting Graph Based Collaborative Filtering: A Linear Residual Graph Convolutional Network Approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 27–34.

[6] Matthew Cooper, Jian Zhao, Chidansh Bhatt, and David A Shamma. 2018. MOOCex: Exploring Educational Video via Recommendation. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*. ACM, 521–524.

[7] Peng Cui, Shaowei Liu, and Wenwu Zhu. 2017. General knowledge embedded image representation learning. *IEEE Transactions on Multimedia* 20, 1 (2017), 198–207.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

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

[10] Alex Graves, Santiago Fernández, and Jürgen Schmidhuber. 2007. Multi-dimensional recurrent neural networks. In *International conference on artificial neural networks*. Springer, 549–558.

[11] Rachida Hannane, Abdessamad Elboushaki, Karim Afdel, P Naghabhushan, and Mohammed Javed. 2016. An efficient method for video shot boundary detection and keyframe extraction using SIFT-point distribution histogram. *International Journal of Multimedia Information Retrieval* 5, 2 (2016), 89–104.

[12] Ruining He and Julian McAuley. 2016. VBPR: visual bayesian personalized ranking from implicit feedback. In *Thirtieth AAAI Conference on Artificial Intelligence*.

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

[14] Wei Huang, Enhong Chen, Qi Liu, Yuying Chen, Zai Huang, Yang Liu, Zhou Zhao, Dan Zhang, and Shijin Wang. 2019. Hierarchical multi-label text classification: An attention-based recurrent network approach. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 1051–1060.

[15] Zhenya Huang, Qi Liu, Yuying Chen, Le Wu, Keli Xiao, Enhong Chen, Haiping Ma, and Guoping Hu. 2020. Learning or Forgetting? A Dynamic Approach for Tracking the Knowledge Proficiency of Students. *ACM Transactions on Information Systems (TOIS)* 38, 2 (2020), 1–33.

[16] Nitin J Janwe and Kishor K Bhoyar. 2013. Video shot boundary detection based on JND color histogram. In *2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013)*. IEEE, 476–480.

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

[18] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).

[19] Qi Liu, Yong Ge, Zhongmou Li, Enhong Chen, and Hui Xiong. 2011. Personalized travel package recommendation. In *2011 IEEE 11th International Conference on Data Mining*. IEEE, 407–416.

[20] Qi Liu, Zai Huang, Zhenya Huang, Chuanren Liu, Enhong Chen, Yu Su, and Guoping Hu. 2018. Finding similar exercises in online education systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1821–1830.

[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*.

[22] Zachary MacHardy and Zachary A Pardos. 2015. Evaluating the Relevance of Educational Videos Using BKT and Big Data. *International Educational Data Mining Society* (2015).

[23] Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. In *Thirtieth AAAI Conference on Artificial Intelligence*.

[24] Yuxin Peng and Chong-Wah Ngo. 2006. Clip-based similarity measure for query-dependent clip retrieval and video summarization. *IEEE Transactions on Circuits and Systems for Video Technology* 16, 5 (2006), 612–627.

[25] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[26] q. liu, Z. Huang, Y. Yin, E. Chen, H. Xiong, Y. Su, and G. Hu. 2019. EKT: Exercise-aware Knowledge Tracing for Student Performance Prediction. *IEEE Transactions on Knowledge and Data Engineering* (2019), 1–1.

[27] Wasifur Rahman, Md Kamrul Hasan, Amir Zadeh, Louis-Philippe Morency, and Mohammed Ehsan Hoque. 2019. M-BERT: Injecting Multimodal Information in the BERT Structure. *arXiv preprint arXiv:1908.05787* (2019).

[28] Vasili Ramanishka, Abir Das, Dong Huk Park, Subhashini Venugopalan, Lisa Anne Hendricks, Marcus Rohrbach, and Kate Saenko. 2016. Multimodal video description. In *Proceedings of the 24th ACM international conference on Multimedia*. ACM, 1092–1096.

[29] Bahjat Safadi, Mathilde Sahuguet, and Benoit Huet. 2014. When textual and visual information join forces for multimedia retrieval. In *Proceedings of International Conference on Multimedia Retrieval*. ACM, 265.

[30] Lucas Theis and Matthias Bethge. 2015. Generative image modeling using spatial lstms. In *Advances in Neural Information Processing Systems*. 1927–1935.

[31] Lucas Theis, Reshad Hosseini, and Matthias Bethge. 2012. Mixtures of conditional Gaussian scale mixtures applied to multiscale image representations. *PloS one* 7, 7 (2012), e39857.

[32] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*. 4489–4497.

[33] Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. 2016. Pixel Recurrent Neural Networks. In *International Conference on Machine Learning*. 1747–1756.

[34] Martin J Wainwright and Eero P Simoncelli. 2000. Scale mixtures of Gaussians and the statistics of natural images. In *Advances in neural information processing systems*. 855–861.

[35] Hao Wang, Tong Xu, Qi Liu, Defu Lian, Enhong Chen, Dongfang Du, Han Wu, and Wen Su. 2019. MCNE: An end-to-end framework for learning multiple conditional network representations of social network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1064–1072.

[36] Jiawei Wang, Jiansheng Fang, Jiao Xu, Shifeng Huang, Da Cao, and Ming Yang. 2019. MOC: Measuring the Originality of Courseware in Online Education Systems. In *Proceedings of the 27th ACM International Conference on Multimedia*. 1952–1960.

[37] Yair Weiss and William T Freeman. 2007. What makes a good model of natural images?. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 1–8.

[38] Le Wu, Lei Chen, Richang Hong, Yanjie Fu, Xing Xie, and Meng Wang. 2019. A hierarchical attention model for social contextual image recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2019).

[39] Huijuan Xu, Kun He, Bryan A Plummer, Leonid Sigal, Stan Sclaroff, and Kate Saenko. 2019. Multilevel language and vision integration for text-to-clip retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 9062–9069.

[40] 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*.

[41] Haojin Yang and Christoph Meinel. 2014. Content based lecture video retrieval using speech and video text information. *IEEE Transactions on Learning Technologies* 7, 2 (2014), 142–154.

[42] 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.

[43] Youngjae Yu, Jongseok Kim, and Gunhee Kim. 2018. A joint sequence fusion model for video question answering and retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 471–487.

[44] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. 2017. End-to-end concept word detection for video captioning, retrieval, and question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3165–3173.

[45] 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*. International World Wide Web Conferences Steering Committee, 765–774.

[46] Kun Zhang, Guangyi Lv, Le Wu, Enhong Chen, Qi Liu, Han Wu, Xing Xie, and Fangzhao Wu. 2019. Multilevel Image-Enhanced Sentence Representation Net for Natural Language Inference. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2019).

[47] Lei Zhang, Xin Zhang, Fan Cheng, Xiaoyan Sun, and Hongke Zhao. 2019. Personalized Recommendation for Crowdfunding Platform: A Multi-objective Approach.. In *CEC*. 3316–3324.