

Research Article

Mining Precedence Relations among Lecture Videos in MOOCs via Concept Prerequisite Learning

Kui Xiao ^{1,2}, Youheng Bai ^{1,2} and Shihui Wang ^{1,2}

¹School of Computer Science and Information Engineering, Hubei University, Wuhan 430062, China

²Hubei Province Educational Informatization Engineering Research Center, Hubei University, Wuhan 430062, China

Correspondence should be addressed to Kui Xiao; xiaokui@hubu.edu.cn

Received 23 April 2021; Revised 22 July 2021; Accepted 6 August 2021; Published 13 August 2021

Academic Editor: Zhen Liu

Copyright © 2021 Kui Xiao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In recent years, MOOC has gradually become an important way for people to learn knowledge. But the knowledge background of different people is quite different. Moreover, the precedence relations between lecture videos in a MOOC are often not clearly explained. As a result, some people may encounter obstacles due to lack of background knowledge when learning a MOOC. In this paper, we proposed an approach for mining precedence relations between lecture videos in a MOOC automatically. First, we extracted main concepts from video captions automatically. And then, an LSTM-based neural network model was used to measure prerequisite relations among the main concepts. Finally, the precedence relations between lecture videos were identified based on concept prerequisite relations. Experiments showed that our concept prerequisite learning method outperforms the existing methods and helps accurately identify the precedence relations between lecture videos in a MOOC.

1. Introduction

Nowadays, more and more people around the world are learning through online education platforms. The number of learners and courses of the top MOOC providers has also been rising rapidly. As of December 2020, 76 million learners around the world have already registered on Coursera, and the provider has also launched over 4600 courses (<https://www.classcentral.com/report/mooc-stats-2020/>).

As is well known, a MOOC consists of a series of lecture videos, and each lecture videos is an independent learning resource. The sequence of the learning resources in a MOOC is fixed. But every online learner has a different background, and not everyone wants to learn from the first lecture to the last one. Some learners prefer to choose videos of interest for learning, but they are not sure whether they have the appropriate background knowledge. The required background knowledge may be addressed in some former lectures, but the precedence relations between lecture videos in a course are usually ambiguous, and as a result, the learners do not know where to start.

A precedence relation between a pair of lecture videos establishes which one addresses more simple concepts and thus should be delivered first. In other words, the precedence relation between two resources is determined by the prerequisite relations among the concepts addressed by these resources. Here, a prerequisite refers to a concept or requirement before one can proceed to the next one. The prerequisite relation exists as a natural precedence among concepts in cognitive processes when people learn, organize, apply, and generate knowledge [1]. Concept prerequisite learning has played an important role in many fields, including precedence relation identification [2–4], concept graph extraction [5, 6], reading list generation [7, 8], curriculum planning [9], course sequencing [10], course design [11], knowledge tracing [12], and automatic assessment [13].

In this paper, we present a new approach for identifying precedence relations between lecture videos in a MOOC based on concept prerequisite learning. First of all, we extracted main concepts from the captions of every lecture video, and then measure prerequisite relations among concepts with an LSTM-based neural network model. After

that, the precedence relations between lecture videos were inferred according to the concept prerequisite relations.

What we do can help learners not only know which main concepts are addressed in each lecture video but also understand the dependencies between different lecture videos. The dependencies between these videos can let learners know what background knowledge is needed before learning a video and let them choose the previous related videos to learn according to their personal circumstances.

Our main contributions are as follows:

- (i) A new approach to measure concept prerequisite relations outperforms other baseline methods.
- (ii) A novel metric to identify precedence relations between lecture videos in a MOOC based on concept prerequisite relations.
- (iii) A new dataset containing 1873 video pairs in “Food and Health,” “Algorithms,” “Calculus,” and “Global History.”

The rest of the paper is organized as follows. Section 2 discusses related work; Section 3 gives a brief introduction of our approach; Section 4 introduces the concept prerequisite learning algorithms; Section 5 describes the method for identifying precedence relations between lecture videos in a MOOC; Section 6 describes the new datasets we collected for this paper and reports our experimental results; in Section 7, there is the conclusion and a discussion of future work.

2. Related Work

Mining precedence relations between learning resources is a task that has gained much attention in recent years. Gasparetti et al. [2] presented an approach to predict a prerequisite relation among two learning resources. Given a pair of resources, a feature vector is built based on the Wikipedia pages of the concepts found in the learning resource text. Chen et al. [3] divided each lecture video into a length of 20-second-long segments and inferred precedence relations between the segments. Manrique et al. [4] also identified precedence relations between MOOC videos with their main concepts. However, their main concepts were extracted by experts manually. Obviously, it is a time-consuming work. This manual approach is not feasible when the number of videos is large.

The problem of discovering prerequisite relations between concepts has also been addressed by some researchers. Talukdar and Cohen [14] proposed an approach to model the prerequisite structure between Wikipedia concepts. They assumed that hyperlinks between Wikipedia pages indicate a prerequisite relation and designed several features, such as PageRank score and Random Walk with Restart (RWR) score via the hyperlink network. At last, they used a Maximum Entropy (MaxEnt) classifier to predict prerequisite relations between concepts. Liang et al. [15] also use the hyperlinks for measuring prerequisite relations between Wikipedia concepts. In their work, each concept was represented by a list of related concepts, and the authors calculated the reference distance (RefD) between two concepts

to infer whether one concept is a prerequisite of another or not. Sayyadharikandeh et al. [16] introduced an approach that inferred prerequisite relations between concepts with Wikipedia clickstream data, but a limitation of the approach is that Wikipedia clickstream data usually cannot cover all the concept pairs. Miaschi et al. [17] utilized linguistic features extracted from Wikipedia articles to measure concept prerequisite relations. To the best of our knowledge, it is the first work to use LSTM-based neural networks for concept prerequisite learning. It is a good inspiration for our work. Liang et al. [18] created 15 graph-based features and 17 text-based features for Wikipedia concept pairs and then employed four widely used binary classifiers: Naïve Bayes, Logistic Regression (LR), Support Vector Machine (SVM), and Random Forest (RF), to discover prerequisite relations between concepts. The UNIGE_SE team [19] developed a neural network classifier that exploited features extracted both from raw text and the structure of the Wikipedia pages to achieving automatic prerequisite learning for Italian concepts. Most of the above methods will be used as the baselines in our experiments.

In addition to Wikipedia, other types of learning resources have also been studied to measure prerequisite relations between concepts, such as university courses [20], textbooks [6], and MOOCs [21]. These works provide the foundation for a wide range of education fields, including learning resources sequencing, learning resource recommendation, and curriculum planning.

3. Our Approach

To identify precedence relations between videos in a MOOC, we need to extract the main concepts from videos, and then measure prerequisite relations between the main concepts. Based on concept prerequisite relations, we can identify the precedence relations between videos in a MOOC. The detailed steps are as follows:

- (1) Extracting captions from lecture videos in a MOOC
- (2) Selecting main concepts from the captions of every lecture video
- (3) Inferring concept prerequisite relations via a LSTM-based neural network model
- (4) Mining precedence relations between lecture videos based on the concept prerequisite relations

It should be noted that only existing Wikipedia concepts were considered as the main concepts of a video in our work. A Wikipedia concept is the title of a Wikipedia article. To the best of our knowledge, Wikipedia is the largest multilingual online encyclopedia on the Internet, and its content can provide services for MOOCs in different disciplines.

4. Concept Prerequisite Extraction

In this section, we will introduce how to measure prerequisite relations among concepts. Before introducing the details of concept prerequisite learning, we first define some relevant elements. The details are shown in Table 1.

TABLE 1: Elements used in the approach.

Elements	Descriptions
A	A Wikipedia concept
R_A	A related concept of A . It means that there is a hyperlink in the content of A 's article, users can access the content of R_A by clicking the hyperlink.
$f_L^{(A)}$	Set of A 's related concepts
V_A	Vector corresponding to the concept A
$V_{f_L^{(A)}}$	Vector corresponding to the set $f_L^{(A)}$

In this paper, concept prerequisite learning is a binary classification task. Given a pair of concepts (A, B) , the classification result (equation (1)) is whether B is a prerequisite of A or not.

$$\text{Preq}(A, B) = \begin{cases} 1, & B \text{ is a prerequisite of } A, \\ 0, & \text{else.} \end{cases} \quad (1)$$

Many concepts prerequisite learning algorithms require using artificial design features, such as the method proposed by Liang et al. [18]. In recent years, deep learning technology has been widely used in image recognition, voice processing, natural language processing, and other fields. Deep learning is essentially a feature learning method, which transforms the original data into higher-level feature representation through a non-linear model, to obtain more abstract expression. In this paper, we employ an LSTM model to extract higher-level features of concept pairs from the raw input.

According to [15], a concept could be represented by its related concepts in Wikipedia's concept space. Therefore, when measuring prerequisite relations of (A, B) , we should not only analyze the two concepts themselves but also analyze their related concepts.

The specific classification process is as follows: First of all, the two concepts A and B , as well as their sets $f_L^{(A)}$ and $f_L^{(B)}$, were respectively converted into four 128-dimensional vectors, i.e., $V_A, V_B, V_{f_L^{(A)}}, V_{f_L^{(B)}}$. Secondly, these vectors were input into four identical LSTM-based subnetworks with 32 units, and the four LSTM outputs were then concatenated ($O_A \oplus O_B \oplus O_{f_L^{(A)}} \oplus O_{f_L^{(B)}}$) and passed to a last Dense Layer. And we use Sigmoid as the activation function. Finally, we got the classification results of concept prerequisite learning. The architecture of the classifier is shown in Figure 1.

For a concept A , there must be a corresponding Wikipedia article. To get the vector V_A , each of the first 400 words in the article was converted into a 128-dimensional vector, and the vectors were built with word2vec [22]. We generated the vectors using the ukWac corpora [23]. We also removed the stop words in the article in advance. At last, V_A was equal to the mean of the 400 vectors.

Similarly, for each R_A of concept A , we also calculated its vector according to the above method. Then, vector $V_{f_L^{(A)}}$ was equal to the mean of all R_A vectors.

5. Precedence Relations Identification

In this section, we use concept prerequisites to help identify precedence relations between lecture videos in a MOOC. First, the lecture videos we used were all from the MOOC provider Coursera. For every lecture video, we got video captions with the toolkit coursera-dl (<https://github.com/coursera-dl/coursera-dl>). After that, we utilized another toolkit TextRazor to extract the main concepts from video captions. As mentioned above, the main concepts are all Wikipedia concepts.

However, in order to reduce the amount of calculation, we only extracted a few concepts from the subtitles of each video. TextRazor associates two attributes with each entity, called `relevance_score` and `Wikipedia_link` (<https://www.textrazor.com/docs/python#Entity>), which are used to estimate the "relevance" of Wikipedia entities to the input text. Here, the input text is lecture video captions, and the parameter `relevance_score` represents the relevance of the entity to the input text. For the extracted concepts, we sort them in reverse order according to the `relevance_score` and select the top k concepts. We will also set different values for the parameter k to study the impact of this parameter on the prediction of video dependencies.

A MOOC often consists of a series of lecture videos, as $C = \{\dots, v_j, \dots, v_i, \dots\}$, given two lecture videos v_i and v_j . Normally, if v_i depends on v_j , v_j must be ranked before v_i . There are three possibilities for the relation between the two videos: (1) v_i depends on v_j ; it means v_j contains some background knowledge for learning v_i ; (2) the two lecture videos are related but have no precedence relation; and (3) they are unrelated.

The precedence relation between two lecture videos can be explained in terms of prerequisites among the concepts that they address. If v_j contains some prerequisite concepts that are needed to learn v_i , then it can be said that v_i depends on v_j . In fact, the prerequisite concepts are the background knowledge needed to learn v_i . On the other hand, if there is no prerequisite relation among the concepts of the two videos, then there will be no precedence between the two videos. Note that, if two videos contain too many duplicate concepts, it is not easy to judge the precedence between them. So, we defined a parameter θ , if the number of

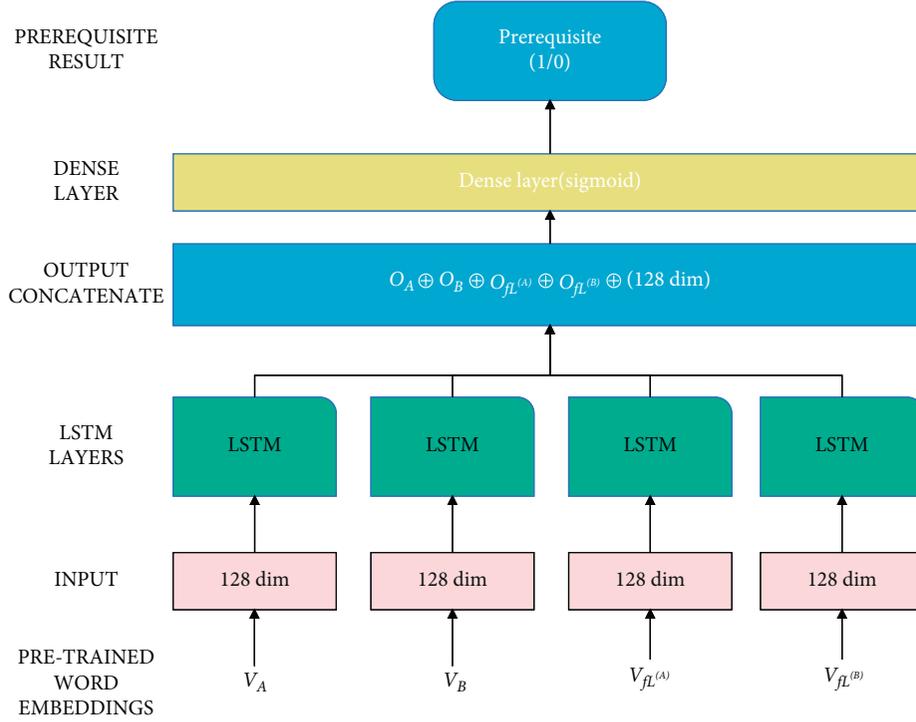


FIGURE 1: Architecture of concept prerequisite learning. (V_A) vector corresponding to the concept A. (V_B) vector corresponding to the concept B. ($V_{f_L^{(A)}}$) vector corresponding to the set $f_L^{(A)}$. ($V_{f_L^{(B)}}$) vector corresponding to the set $f_L^{(B)}$.

duplicate concepts of a video pair is greater than or equal to θ , the video pair will not be considered in our work.

Given a concept-based representation for each lecture video $v_i = \{A_1, A_2, \dots, A_n\}$, $v_j = \{B_1, B_2, \dots, B_m\}$, we calculate precedence score between two videos as

$$\text{Scr}(v_i, v_j) = \frac{1}{|P_c(v_i, v_j)|} \sum_{(A,B) \in P_c(v_i, v_j)} \text{Preq}(A, B),$$

$$\text{s.t. } P_c(v_i, v_j) = \{(A, B) | A \in v_i \wedge B \in v_j, A \neq B\}, \quad |v_i \cap v_j| < \theta. \quad (2)$$

Here, $\text{Scr}(v_i, v_j)$ represents the degree of precedence of video v_i on v_j . Ultimately, whether v_i depends on v_j will be determined by

$$\text{Dep}(v_i, v_j) = \begin{cases} 1, & \text{if } \text{Scr}(v_i, v_j) > \text{Scr}(v_j, v_i), \\ 0, & \text{else.} \end{cases} \quad (3)$$

6. Experiments

In this section, we tested our approach from two aspects, one is the concept prerequisite learning method, and another one is the precedence relations mining method.

6.1. Concept Prerequisite Learning Experiments. First of all, we evaluate the performance of our concept prerequisite learning method on the AL-CPL dataset [24]. The AL-CPL dataset consists of binary-labeled concept pairs from four different domains, including data mining, geometry,

physics, and precalculus. To perform concept prerequisite relation predicting experiments, we trained and tested the classifiers on concept pairs belonging to the same domain. The evaluation is performed using 5-fold cross validation. The performance was quantified with Accuracy (A), Precision (P), Recall (R), F1-score (F1), and Area under the ROC Curve (AUC).

Besides, we compare our model with five state-of-the-art models:

- (1) M1. The model was proposed by Miaschi et al. [17], the authors tested a neural network model (M1) which learned to classify the binary labels using two LSTM-based subnetworks. The embedding vectors of concepts A and B, i.e., V_A and V_B , were inputted into the two subnetworks, respectively. The two LSTM outputs were then concatenated ($O_A \oplus O_B$) and passed to a last dense layer (with Sigmoid activation).
- (2) NN. To verify whether the LSTM part is useful, we also omitted the LSTM part and directly concatenated the A's and B's embedding vectors ($V_A \oplus V_B$) and pass them to a last dense layer (with Sigmoid activation).
- (3) RefD. This baseline was proposed by Liang et al. [15]. In terms of the weight of a referred article, we chose the TFIDF weight to calculate the RefD of two concepts, which is better than the EQUAL weight.
- (4) BERT. This model, developed by Angel [25], corresponds to a single-layer neural network using concept pairs and its Wikipedia description

extracted from the Italian-BERT model. Likewise, we obtain two 786-dimensional embeddings for a concept pair and the single-layer neural network maps the embeddings to an output space of dimension 2 as a precondition for the concepts.

- (5) GE. The concept prerequisite relations of video can be considered as a graph structure. Currently, in the deep graph learning areas, many models were proposed, which can use node, edge, and neighbor information together to generate features embedding. In our experiments, we also used a Graph embedding (GE) method, PyTorch-BigGraph (PBG) [26], as a baseline to compare our model with graph learning methods.

Table 2 shows that in terms of average $F1$ scores, our method generally performs better than the baseline method, reaching +4.8% for M1, +14.8% for NN, +25.3% for RefD, +1% for BERT, and +3.1% for GE. Also, we see that the performance of BERT is very close to ours. More specifically, the BERT method has higher $F1$ on the precalculus domain, which indicates that fine-tuned BERT is sufficient to obtain the prerequisite between concepts in this domain.

The proposed method is superior to the M1 method, because the M1 method only considers the text content of Wikipedia concepts but ignores the fact that related concepts can also provide rich background knowledge for understanding concepts. Besides, the proposed method is also better than the NN method, and we will explain the reason for this problem later. In addition, the RefD method also performs poorly, because the RefD method only uses one feature to predict the prerequisite relations of concepts. Furthermore, thanks to the excellent pre-training of the large-scale corpus, both BERT and GE perform well in concept prerequisite relation learning. But compared with our method, these models lack the word embedding of the main concepts, so our method performs better in several domains such as data mining, geometry, and physics.

On the other hand, because we use a LSTM model in our proposed method, we also investigate two questions about the LSTM model:

- Q1. Does the LSTM model help predict the prerequisite relations between concepts?
 Q2. Is the four-input LSTM model better than the two-input LSTM model?

We use two baseline methods, NN and M1, to investigate these questions. In the NN method, there is not an LSTM model. Given a pair of concepts (A, B) , the embedding vectors of A and B are directly concatenated and then passed to a last dense layer to predict whether there is a prerequisite relation between the two concepts. In the M1 method [17], the embedding vectors of A and B are inputted into two LSTM-based subnetworks. And then the outputs of the two LSTM subnetworks are concatenated ($O_A \oplus O_B$) and passed to a last dense layer.

Table 2 suggests that both the proposed method and the M1 method perform better than the NN method with respect to $F1$ -score across all four datasets. It shows that the LSTM model does provide some useful features that can be used to predict the prerequisite relations between concepts. Furthermore, the proposed method also outperforms the M1 method on all four datasets concerning $F1$ -score. It means that another two LSTM subnetworks for the embedding vectors of $f_L^{(A)}$ and $f_L^{(B)}$, i.e., $V_{f_L^{(A)}}$ and $V_{f_L^{(B)}}$ are also useful for the concept prerequisite learning task.

Besides, we also calculated the AUPRC for the three methods. AUPRC, namely, the area under the precision-recall curve, is a useful performance metric for a problem setting where people care a lot about finding positive examples (<https://glassboxmedicine.com/2019/03/02/measuring-performance-auprc/>). In Figure 2, we show the AUPRC of the three methods on the four datasets. Still, the proposed method performs best among all three methods on the four datasets. For example, the AUPRC of our method on Data Mining outperforms M1 and NN by 4% and 12%, respectively. We also have the observations that our method and the M1 method also perform very closely on both the Geometry and Precalculus datasets. Looking closely at the two datasets, we find that the related concept sets of the two concepts in a concept pair, i.e., $f_L^{(A)}$ and $f_L^{(B)}$, have a higher degree of overlap. As a result, $V_{f_L^{(A)}}$ and $V_{f_L^{(B)}}$ have little effect on generating higher-level features. This is why the four-input LSTM model and two-input LSTM model perform relatively similarly on the two datasets. Table 3 gives two examples of overlapping related concepts.

6.2. Precedence Relations Mining Experiments. We selected four MOOCs from Coursera, including “Food and Health” (<https://www.coursera.org/learn/food-and-health>), “Algorithms” (<https://www.coursera.org/learn/algorithms-part1>), “Calculus” (<https://www.coursera.org/learn/introduction-to-calculus>), and “Global History” (<https://www.coursera.org/learn/modern-world-2>). The number of lecture videos from these MOOCs is 29, 59, 63, and 51, respectively. Table 4 shows the information of some lecture videos of the MOOC Algorithms. The first column presents the catalog number of the lecture videos in the MOOC, and the second column presents the lecture video titles, while the last column presents some of the main concepts of each lecture video. Moreover, we set the value of parameter k to 6. In other words, we select the top 6 main concepts from every lecture video.

We generated video pairs for every MOOC like this $(v_1, v_2), (v_1, v_3), \dots, (v_1, v_n), (v_2, v_3), \dots, (v_2, v_n), \dots$. Then, to obtain the ground truth of precedence relations of all video pairs in the MOOCs, we asked students of corresponding majors to vote on the final results. Given a pair of videos (a, b) , students had three options: (1) b depends on a (“1”); (2) b does not depend on a (“0”); (3) do not know

TABLE 2: Concept prerequisite predicting results.

Methods	Measure	Data mining	Geometry	Physics	Precalculus	AVG
M1	A	83.5	91.9	85.7	89.5	87.7
	P	77.7	88.2	75.2	86.5	81.9
	R	74.3	85.7	63.3	82.0	76.3
	F1	75.8	86.9	68.5	84.1	78.8
	AUC	90.2	96.4	91.5	95.3	93.4
NN	A	74.3	82.1	87.4	90.1	83.5
	P	61.6	69.3	53.5	71.8	64.1
	R	73.8	77.5	74.0	76.9	75.6
	F1	67.0	73.0	61.6	73.6	68.8
	AUC	84.3	90.1	86.2	90.8	87.9
RefD	A	66.2	61.9	74.9	81.9	71.2
	P	51.7	42.4	49.9	75.1	54.8
	R	76.2	62.3	49.6	69.4	64.4
	F1	61.4	50.4	49.4	72.1	58.3
	AUC	69.5	62.4	67.7	79.2	69.7
BERT	A	86.0	92.8	87.4	91.1	89.3
	P	79.7	88.1	75.6	86.5	82.5
	R	81.1	89.3	73.9	87.9	83.1
	F1	80.3	88.6	74.4	87.1	82.6
	AUC	92.2	96.9	92.3	96.6	94.5
GE	A	85.5	91.2	82.4	90.1	87.3
	P	76.5	86.5	74.4	83.4	80.2
	R	81.5	84.1	75.9	81.9	80.9
	F1	78.9	85.3	74.9	82.6	80.4
	AUC	91.7	97.0	93.3	95.4	94.4
Proposed	A	87.0	93.6	88.2	90.9	89.9
	P	83.7	91.3	75.6	86.7	84.3
	R	78.8	88.7	78.0	86.6	83.0
	F1	81.1	89.8	76.7	86.6	83.6
	AUC	92.8	97.4	93.0	96.0	94.8

Bold fonts represent the maximum value of a measure on a dataset.

(“N”). If there is no option to get a majority vote or option 3 got a majority vote in a video pair, the pair will be discarded. The datasets have been published on GitHub (<https://github.com/Morganbyh/MOOC>).

In Table 4, we report the Accuracy (A), Precision (P), Recall (R), and F1-score (F1) obtained for the MOOCs. From Table 5, we can see that the number of precedence relations in “Food and Health” is the least, and the number of positive samples is less than 10%. In “Calculus,” however, the number of positive samples is close to 50%. In another two MOOCs, the number of positive samples is about 20%.

Further observation revealed that “Food and Health” presented only simple common-sense knowledge. There are few precedence relations between lecture videos. This limits our prediction of positive samples. It has the lowest recall of 71.24%. But the accuracy of this domain is generally better than the other three domains. We suppose that this is due to a large number of negative samples and the imbalance in classes. The highest F1 value appeared in “Calculus,” with 81.82%. For the other two MOOC courses, their F1 scores are all over 78%. Overall, our method successfully identifies the precedence relations of the MOOC lecture videos in four domains.

On the other hand, we also want to know whether the value of parameter k has any effect on the prediction of video precedence relations. Figure 3 shows the effect of different

values of k (1, 3, 6, 10, and 15) on the prediction tasks. For every lecture video, we extracted k main concepts from their captions and used the concept prerequisite learning results to identify the lecture video precedence relations.

We extracted 279, 414, 408, and 611 concepts from four MOOCs, respectively ($k=15$). When the parameter k is valued at 1, 3, 6, and 10, we will only use some of these concepts. Looking at four domains, we can see that when the value of k is too large or too small, the performance of precedence relation prediction is not very good. When the parameter k is set to 6, our prediction performances in three MOOCs are the best, including “Food and Health,” “Algorithms,” and “Calculus.” For “Global History,” when the parameter k is set to 3, the prediction performance of the precedence relations is the best. But when the parameter k is set to 6, the prediction performance is also better.

In fact, when the parameter k is set to 1, we may miss some of the main concepts of a lecture video. When the parameter k is valued at 15, we may consider some unimportant words as the main concept of the video. The relevance scores of the words may be very low. Take “Algorithms” as an example, when the k is 15, there will be some general concepts in every lecture video, such as “Data structures,” “Algorithms,” “Classes,” “Sorting,” and “Arrays.” These general concepts are not helpful for precedence relation prediction.

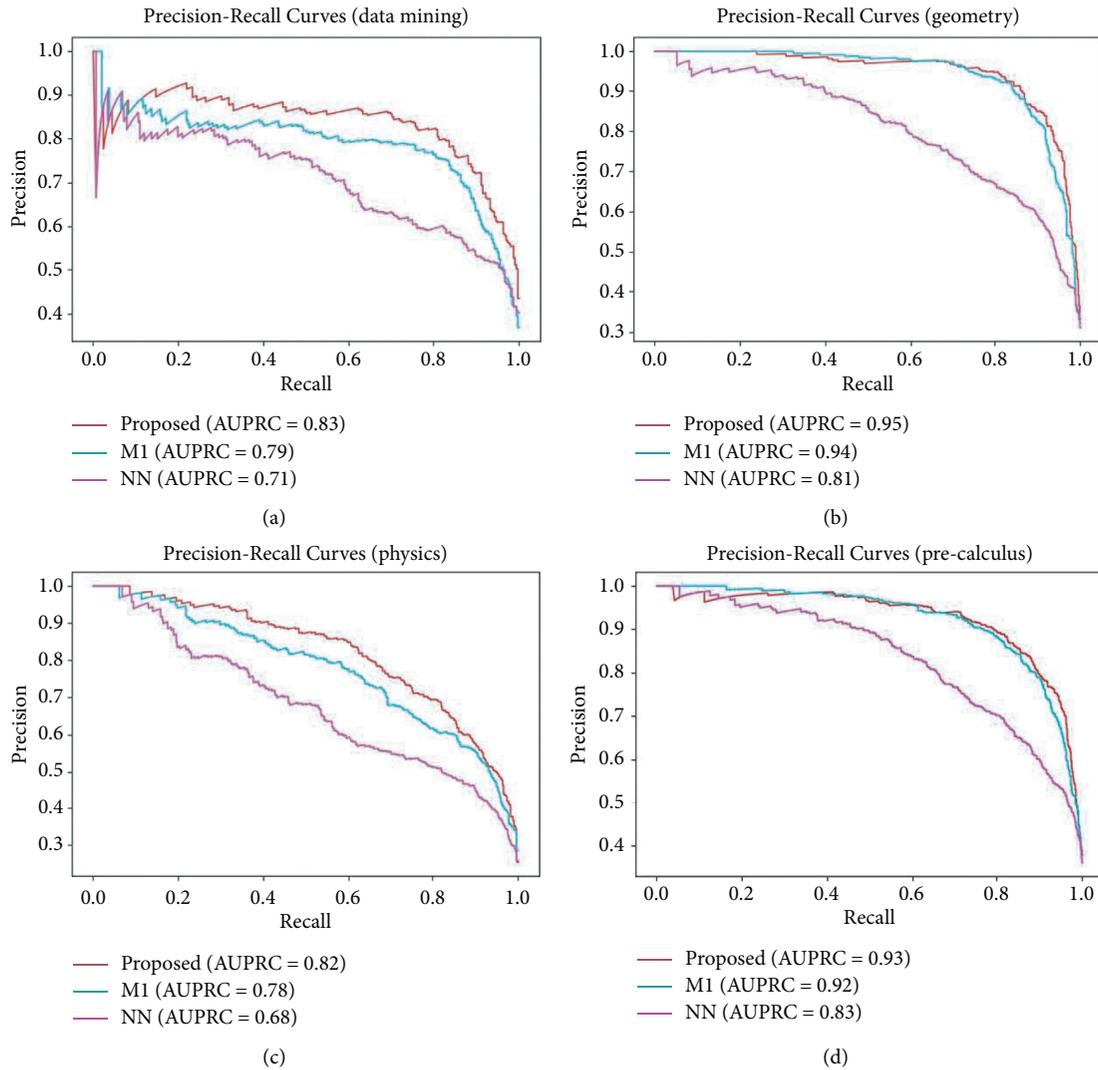


FIGURE 2: Comparison of LSTM models. (a) Data mining. (b) Geometry. (c) Physics. (d) Precalculus.

TABLE 3: Examples of related concept overlap of two concepts.

Domains	Geometry	Precalculus
Concept pair (A, B)	(Slope, secant line)	(Elementary matrix, identity matrix)
#Related concepts of (A, B)	(204, 186)	(185, 190)
#Overlapping related concepts	155	172
Examples of overlapping related concepts	“(ϵ, δ)-definition of limit,” “Abel’s test,” “adequality,” “alternating series,” “alternating series test,” “antiderivative,” “arc length,” “arithmetic-geometric sequence,” “Bernoulli number,” “binomial series”	“Adjacency matrix,” “adjugate matrix,” “alternant matrix,” “alternating sign matrix,” “anti-diagonal matrix,” “arrowhead matrix,” “augmented matrix,” “band matrix,” “Bernoulli matrix,” “biadjacency matrix”

TABLE 4: Lecture videos in the MOOC Algorithms.

Rank	Lecture videos	Main concepts
1	Course introduction	Sorting, Merge sort, Search algorithm, Algorithm, Spanning tree, Sorting algorithm
5	Quick union improvements	Path compression, Linear time, Log, Connected component, Node, Constructor
11	Theory of algorithms	Theory of algorithms, Constant time, Big theta, Big o, Worst case, Binary search
16	Generics	Generics, Generic, Generic type, Type checking, Data type, Bug
20	Selection sort	Selection sort, Array, Quadratic time, Sorting, For loop, Pointer
25	Merge sort	Merge sort, Sort, Stable sort, Recursion, Loop, Sequence
29	Stability	Insertion sort, Selection sort, Sorting algorithm, Shell Sort, Sort, Merge sort
32	Duplicate keys	Sort, Merge sort, Quick sort, Time complexity, Pointer, Array
36	Heapsort	Heap, Sorting algorithms, Heap sort, Heap data structure, Max heap, Binary heap
48	Line segment intersection	Priority queue, Binary search tree, $N \log n$, Brute force method, Interval, Binary search

TABLE 5: Precedence relations predicting results.

	<i>A</i>	<i>P</i>	<i>R</i>	<i>F1</i>	#Video pairs	#Precedence relations
Food and health	95.40	69.23	71.24	70.22	174	13
Algorithms	84.34	81.34	76.63	78.91	437	96
Calculus	87.87	81.32	82.33	81.82	781	378
Global history	79.51	80.13	76.98	78.52	441	84

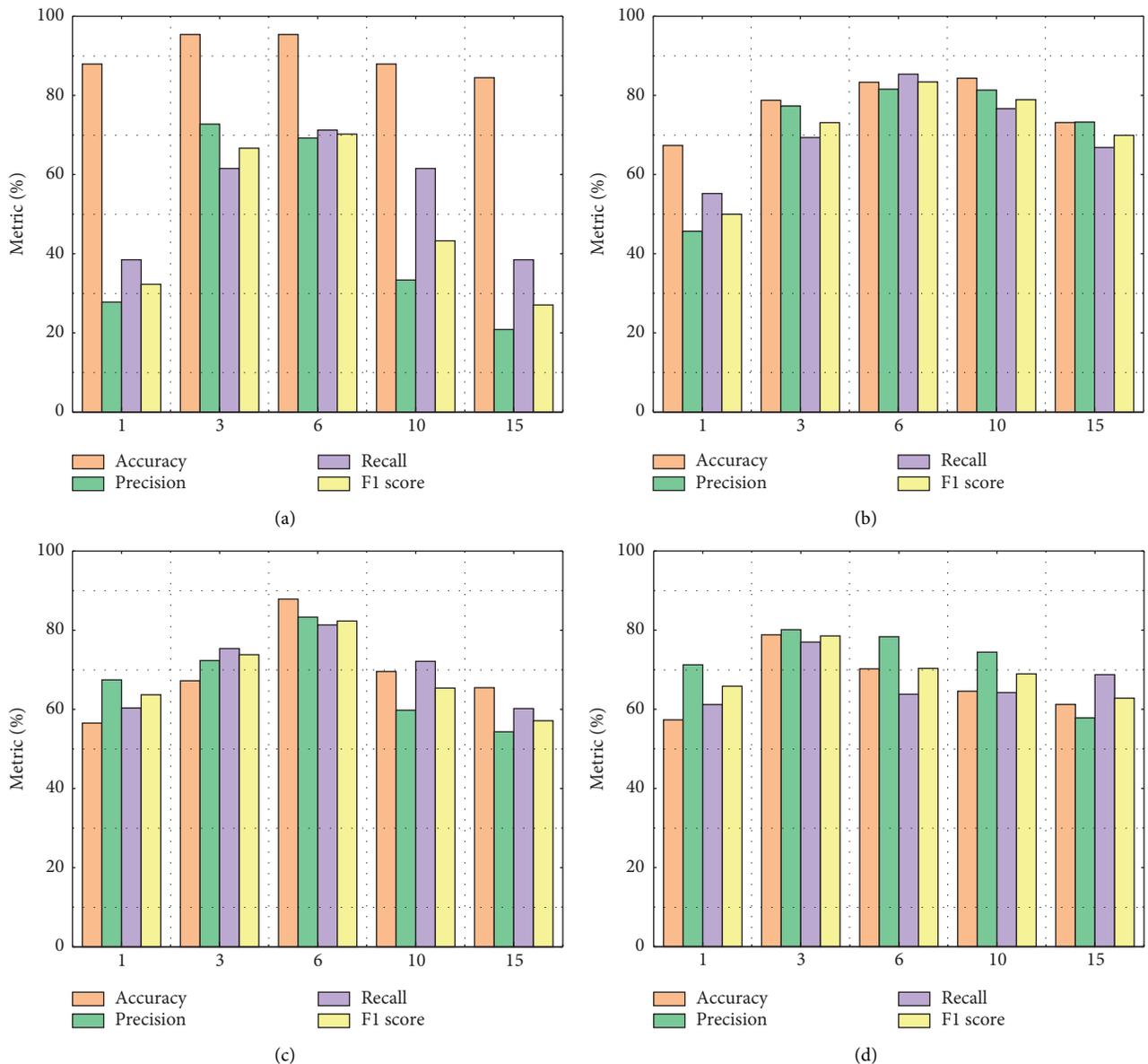


FIGURE 3: Effect of the number of extracted concepts from lecture videos on the identification of precedence relations. (a) Food and health. (b) Algorithms. (c) Calculus. (d) Global history.

7. Conclusions and Future Work

We studied the problem of automatically mining precedence relations between lecture videos in a MOOC. We extracted the main concepts from video captions and analyzed the prerequisite relations between the concepts. After that, the concept prerequisite relations were used to infer the precedence relations between lecture videos. Experiments showed that our concept prerequisite learning method outperforms the existing methods and helps accurately identify the precedence relations between lecture videos in a MOOC.

Promising future direction would be using Wikipedia independent methods to measure prerequisite relations among concepts, so that more accurate main concepts can be selected from video captions for the precedence relation identification. Besides, we can also use the learner behaviour to identify the precedence relations between lecture videos. When users perform personalized learning on a MOOC platform, the order in which they learn videos often implies the precedence relations between the videos.

Data Availability

The CSV data used to support the findings of this study have been deposited in the GitHub repository (<https://github.com/Morganbyh/MOOC>). These data are the results of user voting for the video precedence relations. There are no restrictions on access to these data.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 61977021), the Technology Innovation Special Program of Hubei Province (Nos. 2018ACA133 and 2019ACA144), and the Teaching Research Project of Hubei University (No. 202008).

References

- [1] S. Laurence and E. Margolis, "Concepts and cognitive science," *Concepts: Core Readings*, MIT Press, Cambridge, MA, USA, 1999.
- [2] F. Gasparetti, C. De Medio, C. Limongelli, F. Sciarrone, and M. Temperini, "Prerequisites between learning objects: automatic extraction based on a machine learning approach," *Telematics and Informatics*, vol. 35, no. 3, pp. 595–610, 2018.
- [3] W. Chen, A. S. Lan, D. Cao, C. Brinton, and M. Chiang, "Behavioral analysis at scale: learning course prerequisite structures from learner clickstreams," in *Proceedings of the 11th International Conference on Educational Data Mining Society*, Raleigh, NC, USA, July 2018.
- [4] R. Manrique, J. Sosa, O. Marino, B. P. Nunes, and N. Cardozo, "Investigating learning resources precedence relations via concept prerequisite learning," in *Proceedings of the 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 198–205, IEEE, Santiago, Chile, December 2018.
- [5] Y. Yang, H. Liu, J. Carbonell, and W. Ma, "Concept graph learning from educational data," in *Proceedings of the 8th ACM International Conference on Web Search and Data Mining*, pp. 159–168, Shanghai, China, February 2015.
- [6] S. Wang, A. Ororbia, Z. Wu et al., "Using prerequisites to extract concept maps from textbooks," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pp. 317–326, Indianapolis, IN, USA, October 2016.
- [7] J. Gordon, S. Aguilar, E. Sheng, and G. Burns, "Structured generation of technical reading lists," in *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 261–270, Copenhagen, Denmark, September 2017.
- [8] G. Koutrika, L. Liu, and S. Simske, "Generating reading orders over document collections," in *Proceedings of the 2015 IEEE 31st International Conference on Data Engineering*, pp. 507–518, IEEE, Seoul, Republic of Korea, April 2015.
- [9] R. Agrawal, B. Golshan, and E. Papalexakis, "Toward data-driven design of educational courses: a feasibility study," *Journal of Educational Data Mining*, vol. 8, no. 1, pp. 1–21, 2016.
- [10] A. Vuong, T. Nixon, and B. Towle, "A method for finding prerequisites within a curriculum," in *Proceedings of the 4th International Conference on Educational Data Mining*, pp. 211–216, Eindhoven, Netherlands, July 2011.
- [11] C. Limongelli, F. Gasparetti, and F. Sciarrone, "Wiki course builder: a system for retrieving and sequencing didactic materials from wikipedia," in *Proceedings of the 2015 International Conference on Information Technology Based Higher Education and Training (ITHET)*, pp. 1–6, IEEE, Lisbon, Portugal, June 2015.
- [12] P. Chen, Y. Lu, V. W. Zheng, and Y. Pian, "Prerequisite-driven deep knowledge tracing," in *Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM)*, pp. 39–48, Singapore, November 2018.
- [13] S. Wang and L. Liu, "Prerequisite concept maps extraction for automatic assessment," in *Proceedings of the 25th International Conference Companion on World Wide Web*, pp. 519–521, Montréal, Canada, April 2016.
- [14] P. Talukdar and W. Cohen, "Crowdsourced comprehension: predicting prerequisite structure in wikipedia," in *Proceedings of the 7th Workshop on Building Educational Applications Using NLP*, pp. 307–315, Montréal, Canada, June 2012.
- [15] C. Liang, Z. Wu, W. Huang, and C. L. Giles, "Measuring prerequisite relations among concepts," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1668–1674, Lisbon, Portugal, September 2015.
- [16] M. Sayyadiharikandeh, J. Gordon, J. L. Ambite, and K. Lerman, "Finding prerequisite relations using the wikipedia clickstream," in *Proceedings of the Companion Proceedings of the 2019 World Wide Web Conference*, pp. 1240–1247, San Francisco, CA, USA, May 2019.
- [17] A. Miaschi, C. Alzetta, F. A. Cardillo, and F. Dell'Orletta, "Linguistically-driven strategy for concept prerequisites learning on Italian," in *Proceedings of the 14th Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 285–295, Florence, Italy, August 2019.
- [18] C. Liang, J. Ye, S. Wang, B. Pursel, and C. L. Giles, "Investigating active learning for concept prerequisite learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, New Orleans, LA, USA, February 2018.

- [19] J. Angel, S. Taofeek Aroyehun, and A. Gelbukh, "Nlp-cic @ prelearn: mastering prerequisites relations, from hand-crafted features to embeddings," in *Proceedings of 7th Evaluation Campaign of Natural Language Processing and Speech Tools for 3 Italian. Final Workshop (EVALITA 2020)*, December 2020, <http://www.evalita.it/2020>.
- [20] C. Liang, J. Ye, Z. Wu, B. Pursel, and C. Giles, "Recovering concept prerequisite relations from university course dependencies," in *Proceedings of the AAAI Conference on Artificial Intelligence*, San Francisco, CA, USA, February 2017.
- [21] L. Pan, C. Li, J. Li, and J. Tang, "Prerequisite relation learning for concepts in MOOCs," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pp. 1447–1456, Vancouver, Canada, July 2017.
- [22] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, <https://arxiv.org/abs/1301.3781>.
- [23] A. Ferraresi, E. Zanchetta, M. Baroni, and S. Bernardini, "Introducing and evaluating ukWaC, a very large web-derived corpus of English," in *Proceedings of the 4th Web as Corpus Workshop (WAC-4) Can We Beat Google*, pp. 47–54, Marrakech, Morocco, January 2008.
- [24] C. Liang, J. Ye, H. Zhao, B. Pursel, and C. L. Giles, "Active learning of strict partial orders: a case study on concept prerequisite relations," 2018, <https://arxiv.org/abs/1801.06481>.
- [25] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186, Stroudsburg, PA, USA, May 2019.
- [26] L. Adam, L. Wu, J. Shen et al., "PyTorch-biggraph: a largescale graph embedding system," in *Proceedings of the 2nd SysML Conference*, Palo Alto, CA, USA, April 2019.