Extracting Prerequisite Relations Among Wikipedia Concepts Using the Clickstream Data

Cheng Hu[1], Kui Xiao[1(✉)], Zesong Wang[1], Shihui Wang[1], and Qifeng Li[2]

[1] School of Computer Science and Information Engineering, Hubei University, Wuhan, China
xiaokui@hubu.edu.cn

[2] Academic Affairs Office of Hubei University, Wuhan, China

Abstract. A prerequisite relation describes a basic dependency relation between concepts in education, cognition and other fields. Especially, prerequisite relations among concepts play a very important role in various intelligent education applications, such as concept map extraction, learning object sequencing, reading order list generation. In this paper, we investigate the problem of extracting prerequisite relations among Wikipedia concepts. We take advantage of Wikipedia clickstream data and related concept sets to discover prerequisite relations among Wikipedia concepts. Evaluations on two datasets that include nine domains show that the proposed method can cover most of the concept pairs, and achieves significant improvements (+1.7–31.0% by Accuracy) comparing with existing methods.

Keywords: Prerequisite relation · Wikipedia concept · Clickstream data · Set of related concepts · User navigation

1 Introduction

In a textbook, each chapter contains some important knowledge concepts. Similarly, in a MOOC, each video lecture also introduces several main concepts. There are usually some prerequisite relations between the concepts from different chapters or video lectures, which determine the order of the chapters in a textbook or the order of video lectures in a MOOC. Given a pair of concepts (A, B), if a learner has to understand the meaning of concept A before he or she learns concept B, then we say that A is a prerequisite of B. In this paper, we study the problem of prerequisite relation extraction among concepts in Wikipedia.

When users browse the content of a Wikipedia article, they usually cannot understand the content of the article due to lack of background knowledge. But the background knowledge is often contained in other Wikipedia articles. If users know the prerequisite relations between articles, they will be able to quickly find out the articles in which the background knowledge is located. Here a Wikipedia concept refers to the title of a Wikipedia article. If concept A is a prerequisite of

© Springer Nature Switzerland AG 2021
H. Qiu et al. (Eds.): KSEM 2021, LNAI 12815, pp. 13–26, 2021.
https://doi.org/10.1007/978-3-030-82136-4_2

concept B, it means the article of A contains some background knowledge needed to understand the content of the article of B. In addition, some researchers have extracted Wikipedia concepts from learning resources such as textbooks [1–3], MOOCs [4–6], university course introductions [7–9], and scientific reports [10], and inferred precedence relations between learning resources by using concept prerequisite relations.

User navigation data in Wikipedia, i.e. clickstream data, is helpful for discovering concept prerequisite relations. Intuitively, users visit an article they are interested in so that they can understand the corresponding concept. Meanwhile, they will follow links to other articles they believe will support that objective. For example, a user browses the article of concept B, then follows links from the article of B to the article of concept A. This may be because A is a prerequisite of B. Clickstream data refers to the number of times users followed links from one article to the other [14]. Normally, Wikipedia provides clickstream data for the last 30 months for users to download and use.

In this paper, we propose an approach for extracting concept prerequisite relations in Wikipedia based on clickstream data. Clickstream data was used to define different kinds of features for concept pairs, and then we predicted prerequisite relations between concepts with the features. On the other hand, the number of concept pairs covered by clickstream data is always low. To solve this problem, we used the related concept set of every Wikipedia concept, which significantly improved the coverage of concept pairs.

Our main contributions include: (1) A novel metric to extract prerequisite relations among concepts based on clickstream data that outperforms existing baseline methods. (2) A new method to improve the concept pair coverage of clickstream data with the related concept sets of each Wikipedia concept.

The rest of this article is organized as follows. In Sect. 2 we discuss some related work. In Sect. 3, we describe the proposed method for prerequisite relation extraction. Section 4 elaborates our approach for the evaluation. Finally, we conclude our work in Sect. 5.

2 Related Work

Talukdar and Cohen proposed an early attempt to model the prerequisite structure of Wikipedia concepts [11]. For a pair of concepts, the authors used Hyperlinks, Edits and PageContent to define features, and then employed the MaxEnt classifier to infer prerequisite relations between concepts. Liang et al. [12] proposed a hyperlink-based method for inferring prerequisite relations among Wikipedia concepts. They compute prerequisites based on reference distance (RefD), where Wikipedia hyperlinks serve as "reference relations" among concepts. Zhou and Xiao [13] created four groups of features for prerequisite discovering, including link-based features, category-based features, content-based features, and time-based features.

Most of the related methods extracted concept prerequisite relations by using the features based on Wikipedia article content. In contrast, Sayyadiharikandeh et al. [14] proposed a measure for extracting concept prerequisite relations based

on the Wikipedia clickstream data. This is the first time that user navigation information is used to predict prerequisite relations. we will use this method as a baseline method.

In addition, there are also some other studies similar to this article. Liang et al. [15,16] used the Active learning technology in prerequisite relations extraction tasks. Chen et al. [17] studied how to incorporate the knowledge structure information, especially the prerequisite relations between pedagogical concepts, into the knowledge tracing model. Qiu et al. [18–20] proposed an algorithm that integrates syntactic and semantic validations in system integration tasks, which was used in smart personal health advisor systems for comprehensive and intelligent health monitoring and guidance.

3 Concept Prerequisite Extraction

For a pair (A, B), there may be several situations in the relationship between them, including 1) B is a prerequisite of A , 2) A is a prerequisite of B , 3)the two concepts are related, but they don't have any prerequisite relation between them, 4)the two concepts are unrelated [11]. In most of the previous studies, extracting concept prerequisite relations is treated as a binary classification task. In other words, the authors only distinguished whether or not B is a prerequisite of A [11–14]. In this paper, we will also treat this task as a binary classification task.

Given a set of concept pairs $(A_1, B_1), (A_2, B_2), \cdots, (A_n, B_n)$, Wikipedia clickstream data usually cannot cover all concept pairs. In other words, there is a low likelihood that we will have clickstream data for all pairs of concepts in the datasets. Because some concept pairs are not so concerned. In this paper, we use the set of related concepts of each concept to improve the concept pair coverage in clickstream data. Before going into details, we define some basic elements used in this section. The details are shown in Table 1.

Table 1. Definition of basic elements

Terms	Explanation
A	A Wikipedia concept, which is the title of a Wikipedia article
RA	A related concept of A, i.e.the article of A contains a link to the article of RA
$f_L^{(A)}$	Set of related concepts of A
TA	A target concept of A, i.e. users followed links from the article of A to the article of TA in the recorded months of clickstream data
$f_C^{(A)}$	Set of target concepts of A

According to [12], a concept could be represented by its related concepts in the concept space. Related concepts are the concepts linked out in Wikipedia by the current concepts, and the related concepts can be obtained via Wikipedia

APIs[1]. Therefore, the prerequisite relation between $f_L^{(A)}$ and $f_L^{(B)}$ also represents the prerequisite relation between A and B. In order to increase the coverage of concept pairs, besides $''A - B''$, we also consider the clicks between the related concepts of A and B, namely $''RA - B''$, and the clicks between A and the related concepts of B, namely $''A - RB''$, as well as the clicks between the related concepts of A and the related concepts of B, namely $''RA - RB''$. In this way, we can significantly increase the coverage of concept pairs.

With the different types of Wikipedia clickstream data, we define four groups of features for concept pairs.

3.1 Features of $''A - B''$

The concept pair features based on the clickstream data of $''A - B''$ indicate the prerequisite relation between concept A and concept B. In this paper, we use 8 features for this group, all of which were proposed by Sayyadiharikandeh et al. [14]. The details of the features are as follows:

- $Weight_1(A, B)$. Total number of clicks from the article of A to the article of B.
- $Weight_1(B, A)$. Total number of clicks from the article of B to the article of A.
- $Sum_1(A, B)$. Sum of $Weight_1(A, B)$ and $Weight_1(B, A)$.
- $Diff_1(A, B)$. Absolute difference between $Weight_1(A, B)$ and $Weight_1(B, A)$.
- $Norm_1(A)$. Normalized value of $Weight_1(A, B)$.

$$Norm_1(A) = \frac{Weight_1(A, B)}{1 + Sat_1(A)} \tag{1}$$

Here, $Sat_1(i)$ is the sum of clicks from i to all its target concepts. A "1" is added to the denominator to prevent the denominator from being zero.

$$Sat_1(i) = \sum_{t \in f_c^{(i)}} Weight_1(i, t) \tag{2}$$

- $Norm_1(B)$. Normalized value of $Weight_1(B, A)$.
- $Gtm_1(A, B)$. A binary feature indicating whether $Weight_1(A, B)$ is greater than $Mean_1(A)$.

$$Gtm_1(A, B) = \begin{cases} 1, \text{if } Weight_1(A, B) > Mean_1(A) \\ 0, \text{else} \end{cases} \tag{3}$$

Here, $Mean_1(i)$ is the average number of clicks from i to all its target concepts.

$$Mean_1(i) = \frac{Sat_1(i)}{1 + |f_c^{(i)}|} \tag{4}$$

- $Gtm_1(B, A)$. A binary feature indicating whether $Weight_1(B, A)$ is greater than $Mean_1(B)$.

[1] https://en.wikipedia.org/w/api.php.

3.2 Features of "$RA - B$"

The features based on the clickstream data of "$RA - B$" reflect the prerequisite relations between the concepts in $f_L^{(A)}$ and concept B. We define 8 features for this group, which are as follows:

- $Weight_2(A, B)$. The average number of clicks from the concepts in $f_L^{(A)}$ to B.

$$\begin{aligned} &Weight_2(A, B) = \overline{Weight_1(r, B)} \\ &\text{where } r \in f_L^{(A)} \text{ and } Weight_1(r, B) > 0 \end{aligned} \tag{5}$$

- $Weight_2(B, A)$. The average number of clicks from concept B to the concepts in $f_L^{(A)}$.

$$\begin{aligned} &Weight_2(B, A) = \overline{Weight_1(B, r)} \\ &\text{where } r \in f_L^{(A)} \text{ and } Weight_1(B, r) > 0 \end{aligned} \tag{6}$$

- $Sum_2(A, B)$. Sum of $Weight_2(A, B)$ and $Weight_2(B, A)$.
- $Diff_2(A, B)$. Absolute difference between $Weight_2(A, B)$ and $Weight_2(B, A)$.
- $Norm_2(A)$. Normalized value of the sum of $Weight_1(r, B)$ of the concepts in $f_L^{(A)}$.

$$Norm_2(A) = \frac{\sum_{r \in f_L^{(A)}} Weight_1(r, B)}{1 + \sum_{r \in f_L^{(A)}} Sat_1(r)} \tag{7}$$

- $Norm_2(B)$. Normalized value of $Weight_2(B, A)$.

$$Norm_2(B) = \frac{Weight_2(B, A)}{1 + Sat_1(B)} \tag{8}$$

- $Gtm_2(A, B)$. A binary feature indicating whether $Weight_2(A, B)$ is greater than $Mean_2(A)$.

$$Gtm_2(A, B) = \begin{cases} 1, \text{ if } Weight_2(A, B) > Mean_2(A) \\ 0, \text{ else} \end{cases} \tag{9}$$

  Where, $Mean_2(A)$ is the average number of clicks from the concepts in $f_L^{(A)}$ to their target concepts.

$$Mean_2(A) = \frac{\sum_{r \in f_L^{(A)}} Sat_1(r)}{1 + \sum_{r \in f_L^{(A)}} |f_C^{(r)}|} \tag{10}$$

- $Gtm_2(B, A)$. A binary feature indicating whether $Weight_2(B, A)$ is greater than $Mean_1(B)$.

$$Gtm_1(B, A) = \begin{cases} 1, \text{ if } Weight_2(B, A) > Mean_1(B) \\ 0, \text{ else} \end{cases} \tag{11}$$

3.3 Features of $''A - RB''$

The features based on the clickstream data of $''A - RB''$ imply the prerequisite relations between A and the concepts in $f_L^{(B)}$. We also define 8 features for this group, which are as follows:

- $Weight_3(A, B)$. The average number of clicks from concept A to the concepts in $f_L^{(B)}$.

$$\begin{aligned} &Weight_3(A, B) = \overline{Weight_1(A, r)} \\ &\text{where } r \in f_L^{(B)} \text{ and } Weight_1(A, r) > 0 \end{aligned} \tag{12}$$

- $Weight_3(B, A)$. The average number of clicks from the concepts in $f_L^{(B)}$ to concept A.

$$\begin{aligned} &Weight_3(B, A) = \overline{Weight_1(r, A)} \\ &\text{where } r \in f_L^{(B)} \text{ and } Weight_1(r, A) > 0 \end{aligned} \tag{13}$$

- $Sum_3(A, B)$. Sum of $Weight_3(A, B)$ and $Weight_3(B, A)$.
- $Diff_3(A, B)$. Absolute difference between $Weight_3(A, B)$ and $Weight_3(B, A)$.
- $Norm_3(A)$. Normalized value of $Weight_3(A, B)$.

$$Norm_3(A) = \frac{Weight_3(A, B)}{1 + Sat_1(A)} \tag{14}$$

- $Norm_3(B)$. Normalized value of the sum of the number of clicks from the concepts in $f_L^{(B)}$ to concept A.

$$Norm_3(B) = \frac{\sum_{r \in f_L^{(B)}} Weight_1(r, A)}{1 + \sum_{r \in f_L^{(B)}} Sat_1(r)} \tag{15}$$

- $Gtm_3(A, B)$. A binary feature indicating whether $Weight_3(A, B)$ is greater than $Mean_1(A)$.

$$Gtm_3(A, B) = \begin{cases} 1, \text{ if } Weight_3(A, B) > Mean_1(A) \\ 0, \text{ else} \end{cases} \tag{16}$$

- $Gtm_3(B, A)$. A binary feature indicating whether $Weight_3(B, A)$ is greater than $Mean_3(B)$.

$$Gtm_1(B, A) = \begin{cases} 1, \text{ if } Weight_3(B, A) > Mean_3(B) \\ 0, \text{ else} \end{cases} \tag{17}$$

  Here, $Mean_3(B)$ is the average number of clicks from the concepts in $f_L^{(B)}$ to their target concepts.

$$Mean_3(B) = \frac{\sum_{r \in f_L^{(B)}} Sat_1(r)}{1 + \sum_{r \in f_L^{(B)}} |f_C^{(r)}|} \tag{18}$$

3.4 Features of "$RA - RB$"

The features based on the clickstream data of "$RA - RB$" represent the prerequisite relations between the concepts in $f_L^{(A)}$ and the concepts in $f_L^{(B)}$. We define 8 features for this group, too. The features are as follows:

- $Weight_4(A, B)$. The average number of clicks from the concepts in $f_L^{(A)}$ to the concepts in $f_L(B)$.

$$\begin{aligned} &Weight_4(A,B) = \overline{Weight_1(r_1, r_2)} \\ &\text{where} r_1 \in f_L^{(A)} \text{and} r_2 \in f_L^{(B)} \text{and} Weight_1(r_1, r_2) > 0 \end{aligned} \tag{19}$$

- $Weight_4(B, A)$. The average number of clicks from the concepts in $f_L^{(B)}$ to the concepts in $f_L^{(A)}$.

$$\begin{aligned} &Weight_4(B,A) = \overline{Weight_1(r_2, r_1)} \\ &\text{where } r_1 \in f_L^{(A)} \text{ and } r_2 \in f_L^{(B)} \text{ and } Weight_1(r_2, r_1) > 0 \end{aligned} \tag{20}$$

- $Sum_4(A, B)$. Sum of $Weight_4(A, B)$ and $Weight_4(B, A)$.
- $Diff_4(A, B)$. Absolute difference between $Weight_4(A, B)$ and $Weight_4(B, A)$.
- $Norm_4(A)$. Normalized value of the sum of the number of clicks from the concepts in $f_L^{(A)}$ to the concepts in $f_L^{(B)}$.

$$Norm_4(A) = \frac{\sum_{r_1 \in f_L^{(A)}, r_2 \in f_L^{(B)}} Weight_1(r_1, r_2)}{1 + \sum_{r_1 \in f_L^{(A)}} Sat_1(r_1)} \tag{21}$$

- $Norm_4(B)$. Normalized value of the sum of the number of clicks from the concepts in $f_L^{(B)}$ to the concepts in $f_L^{(A)}$.

$$Norm_4(B) = \frac{\sum_{r_1 \in f_L^{(A)}, r_2 \in f_L^{(B)}} Weight_1(r_2, r_1)}{1 + \sum_{r_2 \in f_L^{(B)}} Sat_1(r_2)} \tag{22}$$

- $Gtm_4(A, B)$. A binary feature indicating whether $Weight_4(A, B)$ is greater than $Mean_2(A)$.

$$Gtm_4(A,B) = \begin{cases} 1, \text{ if } Weight_4(A,B) > Mean_2(A) \\ 0, \text{ else} \end{cases} \tag{23}$$

- $Gtm_4(B, A)$. A binary feature indicating whether $Weight_4(B, A)$ is greater than $Mean_3(B)$.

$$Gtm_4(B,A) = \begin{cases} 1, \text{ if } Weight_4(B,A) > Mean_3(B) \\ 0, \text{ else} \end{cases} \tag{24}$$

It is clear that the four group of features have great similarities. The first group of features represent the direct relation between A and B, and the other three groups of features represent the indirect relations between them. On the other hand, features like $f_1(\cdot,\cdot)$ can only cover a few concept pairs. For many concept pairs, the feature $f_1(\cdot,\cdot)$ may not exist, but $f_2(\cdot,\cdot)$, $f_3(\cdot,\cdot)$ and $f_4(\cdot,\cdot)$ may exist. Therefore, we merge these similar features so as to improve the coverage of concept pairs. We define a new type of features as

$$\begin{aligned} f(\cdot,\cdot) = \alpha_1 f_1(\cdot,\cdot) + \alpha_2 f_2(\cdot,\cdot) + \alpha_3 f_3(\cdot,\cdot) + \alpha_4 f_4(\cdot,\cdot) \\ \text{st.} \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1 \\ 0 \leq \alpha_1, \alpha_2, \alpha_3, \alpha_4 \leq 1 \end{aligned} \tag{25}$$

Here, α_1, α_2, α_3, α_4 are the weights of the four similar features, and the four weights should subject to two constraints, $\alpha_1{+}\alpha_2{+}\alpha_3{+}\alpha_4{=}1$ and $0 \leq \alpha_1, \alpha_2, \alpha_3, \alpha_4 \leq 1$. At last, six new features are created in this way, including $Weight(A,B)$, $Weight(B,A)$, $Sum(A,B)$, $Diff(A,B)$, $Norm(A)$, $Norm(B)$.

In addition, since the value of the features $Gtm_*(A,B)$ can only be 0 or 1, it may not be able to get an integer result if combined in the previous manner. Therefore, we define the new feature $Gtm(A,B)$ as

$$Gtm(A,B) = \begin{cases} 1, \text{ if } Gtm_1(A,B) + Gtm_2(A,B) + Gtm_3(A,B) + Gtm_4(A,B) > 2 \\ 0, \text{ else} \end{cases} \tag{26}$$

In a similar way, we also define the feature $Gtm(B,A)$.

Thus, we have 8 new features for each concept pair, including $Weight(A,B)$, $Weight(B,A)$, $Sum(A,B)$, $Diff(A,B)$, $Norm(A)$, $Norm(B)$, $Gtm(A,B)$ and $Gtm(B,A)$. After that, we can use these features to predict the prerequisite relations among Wikipedia concepts.

4 Experiment

4.1 Dataset

In this paper, we use two datasets to evaluate the proposed method. The first one is the CMU dataset created by Talukdar and Cohen [11], which contains 1,547 pairs of Wikipedia concepts in five domains: Global warming, Meiosis, Newton's laws of motion, Parallel postulate, and Public-key cryptography. The second one is the AL-CPL dataset created by Liang et al. [15], which contains 6,529 pairs of Wikipedia concepts in four domains: Data mining, Geometry, Physics and Pre-calculus.

We first obtain clickstream data from November 2017 to April 2020 from the Wikipedia website[2], and then calculated the coverage of concept pairs for each domain. The results are shown in Table 2. The third column in the table is the domain name, the fourth column denotes the number of concept pairs in each

domain, the fifth column stands for the concept pair coverage of the clickstream data of $''A-B''$, the sixth column and the seventh column are the concept pair coverage based on paths with one and two intermediate nodes respectively, for example, $A-M1-B$ and $A-M1-M2-B$. Here, the $M1$ and $M2$ are the intermediate nodes. It is said that intermediate nodes can also help improve concept pair coverage [14]. And the last column is the concept pair coverage of $''A-B''$, $''RA-B''$, $''A-RB''$ and $''RA-RB''$.

It can be seen from the last column that, after using related concept sets, the clickstream data in seven domains cover more than 90% of the concept pairs. Compared with not using related concepts, i.e. the fifth column, the coverage has been greatly improved. On the AL-CPL dataset, however, the concept pair coverage based on related concept sets is slightly lower than the coverage based on one or two intermediate nodes. We suppose it is because the Wikipedia concepts in the AL-CPL dataset are selected from textbooks. Intermediate nodes strengthen the link between concepts, and increase the coverage of concept pairs.

4.2 Evaluation Results

For each domain, we apply 5-fold cross validation to evaluate the performance of the proposed method. In our experiments, we employ 7 different binary classifiers, including Random Forest (RF), NaïveBayes (NB), C4.5 Decision Tree (C4.5), Multi-Layer Perceptron (MLP), SVM with rbf kernel (SVM), Logistic Regression (LR) and AdaBoost (Ada). We measure performance of classifiers on the CMU and AL-CPL datasets in terms of the precision, recall, and F1 scores. In addition, each of the four weights $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ is set to 0.25.

Table 2. Comparison of different kinds of concept pair coverage

Datasets	ID	Domains	#pairs	A-B(%)	A-M1-B(%)	A-M1-M2-B(%)	A-B/RA-B/A-RB/RA-RB(%)
CMU	D1	Global warming	400	27.25	80.50	82.00	72.25
	D2	Meiosis	347	40.35	90.20	90.78	99.71
	D3	Newton's laws of motion	400	37.75	82.75	84.25	97.00
	D4	Parallel postulate	200	29.00	79.00	79.50	94.50
	D5	Public-key cryptography	200	38.50	84.00	86.50	91.00
	Total		**1547**	**34.58**	**83.52**	**84.81**	**90.11**
AL-CPL	D6	Data mining	826	36.44	91.40	93.70	87.18
	D7	Geometry	1681	25.46	93.46	94.11	93.63
	D8	Physics	1962	29.61	98.01	98.57	95.01
	D9	Pre-calculus	2060	29.17	97.28	97.82	96.02
	Total		**6529**	**29.27**	**95.77**	**96.57**	**93.98**

Table 3 shows the evaluation results on the two datasets. It can be seen that RF outperforms other classifiers on four domains, including Meiosis (CMU), Parallel postulate (CMU), Physics (AL-CPL) and Pre-calculus (AL-CPL). Similarly, Ada performs better than other classifiers on Data mining (AL-CPL) and Geometry (AL-CPL); LR performs better than other classifiers on Newton's laws of motion (CMU). For another two domains, Global warming (CMU) and Public-key cryptography (CMU), different classifiers achieve the best P, R and F1 values. Overall, RF performs best in the concept prerequisite relations prediction tasks. On the CMU dataset, its average F1 outperforms NB, C4.5, MLP, SVM, LR and Ada by 12.7%, 8.3%, 9.5%, 2.1%, 4.9% and 0.3% respectively. On the AL-CPL dataset, its average F1 outperforms NB, C4.5, MLP, SVM, LR and Ada by 9.2%, 8.1%, 14.7%, 8.2%, 6.1% and 2.2% respectively. Consequently, we use RF in the following experiments.

Table 3. Classification results of the proposed method (%)

Classifiers		CMU					AL-CPL			
		D1	D2	D3	D4	D5	D6	D7	D8	D9
RF	P	75.9	**91.2**	77.3	**89.3**	71.8	66.0	72.8	**78.6**	**76.1**
	R	81.9	**92.2**	84.3	**90.6**	82.2	67.2	74.2	**81.6**	**76.8**
	F1	78.5	**91.7**	80.5	**89.8**	76.6	65.4	72.8	**78.8**	**76.4**
NB	P	41.7	74.2	80.2	79.5	77.3	46.8	63.8	70.6	64.6
	R	28.5	81.6	83.3	84.0	82.2	55.6	70.2	76.7	71.2
	F1	20.8	77.0	81.0	80.8	**79.1**	46.2	63.0	72.1	63.5
C4.5	P	77.4	76.8	77.7	73.4	77.1	64.0	70.1	73.4	70.2
	R	74.8	75.3	77.8	71.1	75.1	61.9	70.4	73.4	69.8
	F1	75.6	75.9	77.4	71.1	75.7	62.4	70.2	73.3	69.9
MLP	P	74.6	74.6	79.1	75.5	**78.8**	64.3	62.5	71.1	64.0
	R	84.1	76.6	75.1	82.7	79.6	56.5	63.7	72.6	61.3
	F1	78.9	74.3	76.3	78.6	78.7	57.7	60.2	70.5	60.7
SVM	P	73.9	71.4	76.3	80.5	72.1	48.1	64.1	69.6	67.1
	R	**85.8**	84.5	86.4	87.2	**84.8**	68.9	71.1	79.5	71.8
	F1	79.4	77.4	80.6	82.2	77.9	56.5	60.1	70.7	60.8
LR	P	77.2	71.3	**84.1**	82.5	74.4	58.9	68.7	74.3	67.1
	R	85.5	83.2	**87.0**	87.2	84.2	68.0	71.7	79.1	72.0
	F1	**80.3**	76.7	**82.6**	82.9	78.5	57.6	62.8	70.7	62.3
Ada	P	**78.6**	77.5	76.6	80.0	72.9	**69.0**	**73.4**	76.6	74.1
	R	85.1	82.0	84.3	84.0	80.9	**69.8**	**74.5**	79.9	75.2
	F1	80.1	78.6	79.9	81.5	76.6	**66.2**	**73.5**	75.6	74.2

4.3 Comparison with Baselines

We further compare our approach with two representative methods. The first method is the RefD method proposed by Liang et al. [12]. The authors used two different weights for the related concepts of a Wikipedia concept, namely Equal and Tf-idf. In our experiments, we will compare our approach with both of them, i.e. RefD-Equal and RefD-Tfidf.

Another baseline method is proposed by Sayyadiharikandeh et al. [14]. The authors also identified concept prerequisite relations with Wikipedia clickstream data. They used intermediate nodes to improve concept pair coverage, and there were three strategies in their work: Direct link (no intermediate node), 1 intermediate node, and 2 intermediate nodes. In this paper, we will compare our approach with all of the three strategies.

Table 4 shows the comparison results of the proposed method and the baseline methods in terms of accuracy. We find that our method outperforms all the baseline methods on seven of the nine domains. Furthermore, the average accuracy of our method on CMU outperforms RefD-Equal, RefD-Tfidf, Direct link, 1 intermediate node, and 2 intermediate nodes by 31%, 29.5%, 20%, 13.1% and 12.5%, respectively. On the AL-CPL dataset, our method also beats others with best average accuracy. Its average accuracy outperforms RefD-Equal, RefD-Tfidf, Direct link, 1 intermediate node, and 2 intermediate nodes by 10.3%, 9%, 14.7%, 3.1% and 1.7%, respectively.

Table 4. Comparison with baselines (%)

Methods	CMU					AL-CPL			
	D1	D2	D3	D4	D5	D6	D7	D8	D9
RefD-Equal	57.4	53.0	63.7	70.5	55.2	**70.1**	57.1	68.4	72.7
RefD-Tfidf	60.1	55.7	64.6	67.9	57.7	68.4	57.2	66.4	**78.9**
Direct link	81.3	59.3	72.0	72.7	66.7	58.3	58.8	62.1	67.5
1 Intermediate node	78.0	75.4	84.2	70.0	72.7	68.9	75.8	72.4	76.3
2 Intermediate nodes	76.3	75.4	84.2	83.3	66.7	68.8	75.5	78.3	75.4
Proposed method	**91.8**	**92.2**	**89.7**	**90.6**	**83.3**	69.4	**76.4**	**81.6**	76.8

4.4 Feature Contribution Analysis

In order to investigate the importance of each feature in the proposed method, we perform a contribution analysis with different features. Here, we run our approach 8 times on the Pre-calculus (AL-CPL) dataset. In each time, one feature is removed. We record the decrease of accuracy for each removed feature. Table 5 lists the evaluation results after ignoring different features.

According to the decrement of accuracy, we find that all the proposed features are useful in inferring concept prerequisite relations. Especially, we observe that

$Norm(B)$, decreasing our accuracy by 7.3%, plays the most important role. In fact, $Norm(B)$ represents the proportion of the number of clicks from B (or related concepts of B) to A (or related concepts of A) in the number of clicks from B (or related concepts of B) to all their target concepts. The larger the proportion, the stronger the dependence of concept B on concept A. Therefore, after removing this feature, the overall concept prerequisite relations prediction accuracy drops the most. All the relevant code and data of the experiment have been released on github[3].

Table 5. Contribution analysis of different features (%)

Features	Accuracy
$Weight(A, B)$	74.5%(–2.3%)
$Weight(B, A)$	73.7%(–3.1%)
$Sum(A, B)$	74.5%(–2.3%)
$Diff(A, B)$	73.3%(–3.5%)
$Norm(A)$	73.2%(–3.6%)
$Norm(B)$	**69.5%(–7.3%)**
$Gtm(A, B)$	74.1%(–2.7%)
$Gtm(B, A)$	73.8%(–3.0%)

5 Conclusion

This paper studies the problem of concept prerequisite relation extraction in Wikipedia. We used Wikipedia clickstream data and related concept sets to define concept pair features, and then inferred whether there was a prerequisite relation between two concepts. Experiments show that, the proportion of concept pairs covered by clickstream data has been significantly improved. And at the meanwhile, we can also accurately identify prerequisite relations.

However, our current work is only appropriate for Wikipedia concepts. Some of the main concepts of learning resources may not exist in Wikipedia. In the future, we will study the method of prerequisite relation extraction that can be used for both Wikipedia concepts and non-Wikipedia concepts. And then build concept graphs with concept prerequisite relations. The concept graphs can help us improve various intelligent tutoring systems.

Acknowledgement. This work is supported by the National Natural Science Foundation of China (No. 61977021), the Technology Innovation Special Program of Hubei Province (Nos. 2018ACA133 and 2019ACA144).

[3] https://github.com/Little-spider2001/Data-set-and-code-program-of-KSEM-2021-paper.

References

1. Wang, S., 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. 317–226 (2016)
2. Wang, S., Liu, L.: Prerequisite concept maps extraction for automaticassessment. In: Proceedings of the 25th International Conference Companion on World Wide Web, pp. 519–521 (2016)
3. Lu, W., Zhou, Y., Yu, J., Jia, C.: Concept extraction and prerequisite relation learning from educational data. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 9678–9685 (2019)
4. Pan, L., Li, C.J., Li, J.Z., Tang, J.: Prerequisite relation learning for concepts in moocs. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pp. 1447–1456 (2017)
5. Pan, L., Wang, X., Li, C., Li, J., Tang, J.: Course concept extraction in moocs via embedding-based graph propagation. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing, pp. 875–884 (2017)
6. Roy, S., Madhyastha, M., Lawrence, S., Rajan, V.: Inferring concept prerequisite relations from online educational resources. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 9589–9594 (2019)
7. Yang, Y., Liu, H., Carbonell, J., Ma, W.: Concept graph learning from educational data. In: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, pp. 159–168 (2015)
8. Liu, H., Ma, W., Yang, Y., Carbonell, J.: Learning concept graphs from online educational data. J. Artif. Intell. Res. **55**, 1059–1090 (2016)
9. Liang, C., Ye, J., Wu, Z.H., Pursel, B., Giles, C.L.: Recovering concept prerequisite relations from university course dependencies. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
10. Gordon, J., Zhu, L.H., Galstyan, A., Natarajan, P., Burns, G.: Modeling concept dependencies in a scientific corpus. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pp. 866–875 (2016)
11. Talukdar, P., Cohen, W.: Crowdsourced comprehension: predicting prerequisite structure in Wikipedia. In: Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, pp. 307–315. Association for Computational Linguistics (2012)
12. Liang, C., Wu, Z.H., Huang, W.Y., Giles, C.L.: Measuring prerequisite relations among concepts. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1668–1674 (2015)
13. Zhou, Y., Xiao, K.: Extracting prerequisite relations among concepts in Wikipedia. In: 2019 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2019)
14. Sayyadiharikandeh, M., Gordon, J., Ambite, J.L., Lerman, K.: Finding prerequisite relations using the Wikipedia clickstream. In: Companion Proceedings of the 2019 World Wide Web Conference, pp. 1240–1247 (2019)
15. Liang, C., Ye, J., Wang, S., Pursel, B., Giles, C.L.: Investigating active learning for concept prerequisite learning. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
16. Liang, C., Ye, J., Zhao, H., Pursel, B., Giles, C.L.: Active learning of strict partial orders: a case study on concept prerequisite relations. arXiv preprint arXiv:1801.06481(2018)

17. Chen, P., Lu, Y., Zheng, V.W., Pian, Y.: Prerequisite-driven deep knowledge tracing. In: 2018 IEEE International Conference on Data Mining (ICDM), pp. 39–48. IEEE (2018)
18. Chen, M., Zhang, Y., Qiu, M.K., Guizani, N., Hao, Y.X.: SPHA: smart personal health advisor based on deep analytics. IEEE Commun. Mag. **56**(3), 164–169 (2018)
19. Tao, L.X., Golikov, S., Gai, K.K., Qiu, M.K.: A reusable software component for integrated syntax and semantic validation for services computing. In: IEEE Symposium on Service-Oriented System Engineering, pp. 127–132. IEEE (2015)
20. Gai, K., Qiu, M.: Reinforcement learning-based content-centric services in mobile sensing. IEEE Netw. **32**(4), 34–39 (2018)