APPROACH FOR DISCOVERING CONCEPT PREREQUISITE RELATIONS FROM WIKIPEDIA

Kui Xiao [1] *Kun Li*[1] *Yan Zhang*[1]

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

ABSTRACT

This study addresses the importance of concept prerequisite relations, determining the learning sequence of knowledge concepts and aiding students' self-study. Although vital for personalized learning paths, manually annotating concept prerequisite relations is time-consuming and inconsistent. To tackle this, we introduce a weakly supervised approach utilizing Wikipedia. Concepts are represented by article titles, and BERT and Pointwise Mutual Information (PMI) are employed to calculate concept similarity. Weak labels of concept pairs are used to create an initial concept graph. A graph attentional layer integrates neighboring concepts' features, enhancing individual concept representations. The Variational Graph Auto-Encoder (VGAE) reconstructs the concept graph, revealing real prerequisite relations. Extensive experimental results on English and Chinese datasets prove the method's effectiveness, comparable to supervised learning approaches. This technique offers a promising solution for discovering concept prerequisite relations efficiently and accurately, benefiting educational applications.

Index Terms— Prerequisite relation, Graph attentional layer, VGAE, Weakly supervised learning, Wikipedia

1. INTRODUCTION

With the continuous advancement of online learning platforms, an increasing number of individuals are opting for internet-based education, leading to a rapid proliferation of online learning resources. However, this surge in resources poses a new challenge: learners struggle to comprehend the sequence of learning materials, particularly when navigating materials from diverse courses or sources. Consequently, learners find it challenging to select appropriate resources tailored to their needs, hindering their educational progress. Learners frequently encounter difficulties in resource selection, lacking clarity on necessary prerequisites before delving into new topics or discerning subsequent materials to study after completing a resource. Even with textbooks, learners often discover that comprehension diminishes as they progress. For instance, someone studying "Neural Networks and Deep Learning" must grasp foundational concepts in linear algebra, calculus, numerical optimization, probability theory, and mathematical statistics. Yet, these fundamentals cannot be entirely covered within one book. The vague dependency relationships between learning resources emerge as a significant hurdle faced by learners.

A learning resource, like a video in a MOOC or a chapter in a textbook, typically introduces multiple key knowledge concepts. The connections between learning resources are usually established through the prerequisite relations among these key concepts. When considering a pair of concepts (A, B), if understanding concept A is necessary before learning concept B, then concept A becomes a prerequisite for concept B. Resource-level dependencies are essentially built upon these concept-level prerequisite relations. Fig 1 illustrates an example of such concept prerequisite relations. For instance, students must grasp the concept of "Arithmetic mean" before delving into "Regression analysis", making "Arithmetic mean" a prerequisite concept for "Regression analysis". Similarly, "Regression analysis" is a prerequisite concept for "Poisson regression". These prerequisite relations among main concepts form the foundation for dependency relationships between learning resources, ultimately determining the sequence of learning materials. Presently, these concept prerequisite relations find applications in diverse educational contexts, including intelligent tutoring systems [1, 2], curriculum planning [3–5], learning resource sequencing [6, 7], reading list generation [8,9], learning path generation [10,11], and knowledge tracing [12–14].

In recent years, numerous studies have delved into the realm of concept prerequisite relations, broadly categorized into two types. The first type involves extracting knowledge concepts from sources such as university courses, MOOCs, textbooks, and scientific corpora. These concepts are then linked to Wikipedia pages to deduce relations between them. The second type employs Wikipedia article titles as concepts to infer prerequisite relations, establishing the reading order of articles. Evidently, Wikipedia plays a pivotal role in uncovering concept prerequisite relations. The platform hosts a wealth of articles, spanning various languages, edited by users. It boasts dense hyperlinks between pages and a well-organized category hierarchy, furnishing a robust foundation for exploring concept prerequisite relations.

While there have been studies on discovering prerequisite relations, they often come with shortcomings. Much of

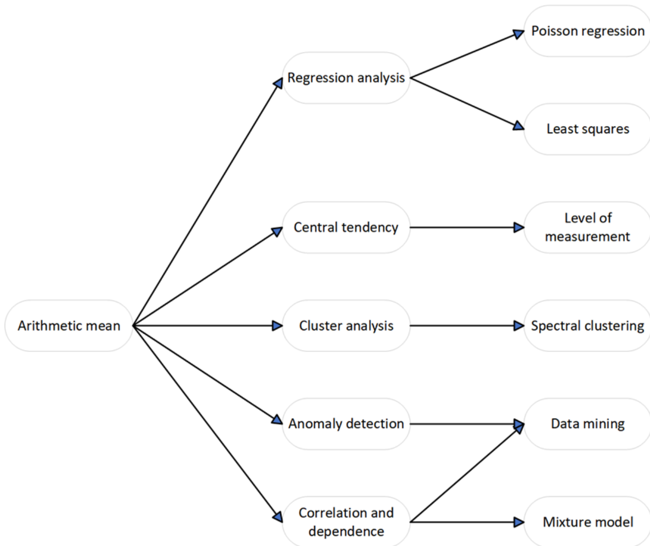

Fig. 1. Example of concept prerequisite relations.

the existing research relies on supervised or semi-supervised learning methods, necessitating labeled data for model training. Manual annotation of concept relations is time-consuming and resource-intensive. Moreover, ensuring consistency and accuracy in annotation results proves challenging due to varying interpretations among different experts regarding the same pair of concept relations. Unsupervised learning methods have been proposed to identify concept prerequisite relations, but these approaches require using learning resource dependencies, which also entail substantial costs, as demonstrated in [15].

To overcome the challenges mentioned above, we propose a weakly supervised learning approach for discovering prerequisite relations among Wikipedia concepts, eliminating the need for labeled data. Initially, we treat each Wikipedia article title as a concept and employ BERT and Pointwise Mutual Information (PMI) to calculate the similarity between concept pairs. Weak labels for concept pairs are generated, forming an original concept graph. Subsequently, we introduce a graph attentional layer to merge feature information from neighboring nodes within the graph, updating the concept representations. Finally, we reconstruct the concept graph using the Variational Graph Auto-Encoder (VGAE) model [16] to establish prerequisite relations between concepts. To address the issue of edge directionality in the reconstructed graph, we divide the graph's adjacency matrix into upper and lower triangular matrices, inputting them separately into the VGAE model. The resulting outputs are combined to infer edge orientation.

Our main contributions include:

1. A novel weakly supervised learning approach that is able to discover the prerequisite relations between Wikipedia concepts without the need for labeled data in the training process.

2. A new method for inferring the directions of concept prerequisite relations in the reconstructed graphs generated by the VGAE model.

3. A Chinese dataset containing 5364 pairs of concepts, which is used to verify the effectiveness of the proposed approach.

The rest of this paper is organized as follows: Section 2 provides an overview of related works. In Section 3, we define the problem of discovering concept prerequisite relations in Wikipedia. Section 4 outlines the details of our proposed approach. Section 5 presents the experimental results of our approach. Finally, Section 6 concludes the paper and outlines directions for future work.

2. RELATED WORK

In recent years, numerous studies have explored learning concept prerequisite relations, primarily employing supervised methods. Talukdar and Cohen [17] pioneered modeling concept prerequisites in Wikipedia, utilizing Hyperlinks, Edits, and PageContent to define features and employing the MaxEnt classifier. Liang et al. [18] proposed a hyperlink-based metric using Reference Distance (RefD) to infer prerequisite relations. Pan et al. [19] investigated inferring relations in MOOCs, devising diverse features like contextual, structural, and semantic aspects. Sayyadiharikandeh et al. [20] employed Wikipedia clickstream data, emphasizing navigation from concepts to their prerequisites. Miaschi et al. [21] introduced a deep learning method relying solely on linguistic features from Wikipedia articles, a departure from structured data. Other supervised methods include [22–26].

Additionally, there are semi-supervised learning methods for discovering concept prerequisite relations. Roy et al. [27] introduced PREREQ, employing latent representations from the Pairwise Latent Dirichlet Allocation model and a Siamese network to infer prerequisite relations. Zhang et al. [28] proposed MHAVGAE, an end-to-end framework utilizing the resource-concept graph's rich feature information for learning concept prerequisites. Other semi-supervised methods include [29–31].

In contrast, weakly supervised and unsupervised learning methods are comparatively scarce. Li et al. [32] introduced a graph neural model that employs deep representations to extract concept prerequisite relations from learning resources. Similarly, Zhang et al. [33] proposed a weakly supervised approach called wMHAVGAE for discovering these relations. They used the RPRD metric to generate preliminary concept prerequisite relations and extended MHAVGAE [28] in a weakly supervised context. Additionally, there are other weakly supervised and unsupervised methods, including [15, 34, 35].

3. PROBLEM STATEMENT

For a pair of concepts (A, B), if people must understand the meaning of concept A before learning concept B. It means that B depends on A, and A is a prerequisite of B, denoted as $A \mapsto B$. The concept prerequisite relation can be formally defined as:

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

It is generally believed that prerequisite relation has the following properties:

- Non-reflexivity: A concept cannot be a prerequisite of itself. A prerequisite relation is only possible between two distinct concepts.
- Anti-symmetry: Given a pair of concepts (A, B), if A is a prerequisite of B, then B cannot be a prerequisite of A. A prerequisite relation represents the dependency relationship between two concepts and also determines the order of the two concepts. Therefore, it is an anti-symmetric relation.

Given a set of concepts of a domain $C = [c_1, c_2, \dots\dots, c_n]$. Each concept corresponds to an article in Wikipedia. Our goal is to learn the prerequisite relations between the concepts, and generates a directed graph, in which nodes represent concepts and edges represent prerequisite relations between concepts. The concept graph learning can be formulated as a link prediction problem. Our unsupervised setting is to exclude any direct concept relations during training, and we wish to predict these edges indirectly through concept similarities.

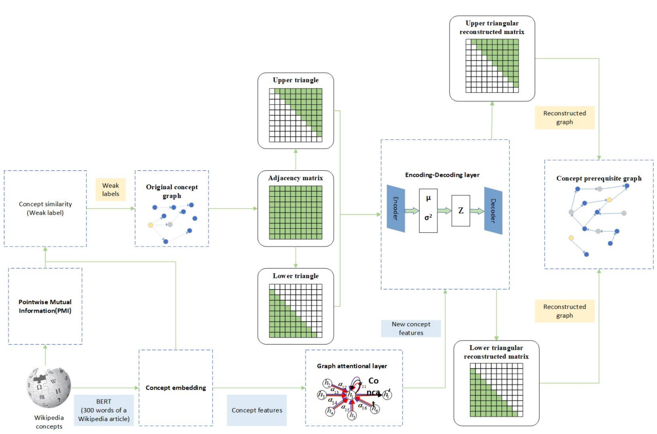

Fig. 2. The architecture of the proposed approach.

4. PROPOSED METHOD

4.1. Overview of the approach

In this section, we present the architecture of our proposed approach, illustrated in Fig 2. First, we treat a Wikipedia article's title as a concept. Using PMI-based and BERT-based methods, we calculate concept similarities and generate weak labels for concept pairs, constructing an initial concept graph. We believe that if two concepts have a prerequisite relation, they share some semantic similarity. Second, employing the graph attention mechanism, we fuse information from neighboring nodes to update each concept node's representation vector, enhancing our ability to discover concept prerequisite relations. Third, the original concept graph undergoes reconstruction using the VGAE model to learn the true concept prerequisite relations. VGAE operates with undirected graphs, yet prerequisite relations must be oriented, directly influencing the knowledge learning sequence. To address this, our weakly labeled concept map is directional, yielding an asymmetric adjacency matrix. We divide this matrix into upper and lower triangular matrices, generating two symmetric adjacency matrices. These matrices, when input into VGAE, yield two reconstructed symmetric adjacency matrices. Combining the upper and lower triangular elements, we obtain an asymmetric adjacency matrix representing the directed concept graph.

4.2. Concept Similarity Calculation

To explore concept prerequisite relations through weakly supervised methods, generating weak labels for concept pairs is crucial. These labels represent incomplete, imprecise, and inaccurate prerequisite relations. In this paper, predicting unlabeled data information relies on training datasets with coarse-grained labels that aren't always ground truth.

The key lies in how we generate weak labels for concept pairs. We believe that if two concepts have a prerequisite relation, they exhibit some degree of semantic similarity. To this end, we employed both PMI-based and BERT-based methods to calculate concept similarities.

4.2.1. PMI based Similarity

PMI provides a measure of concept association within information theory [36], indicating the likelihood of two concepts occurring together compared to their independent occurrences in the corpus. Commonly used in natural language processing, PMI analyzes correlations between concepts. A high PMI value signifies closely related concepts with a high degree of semantic similarity, while a low PMI value indicates less related concepts.

In this paper, to calculate the PMI similarity between concepts, we apply the citations between pages in Wikipedia. We define the PMI similarity of two concepts (c_1, c_2) as follows:

$$PMI(c_1, c_2) = max(0, \frac{\log[p(c_1) \cdot p(c_2)]}{\log p(c_1, c_2)} - 1) \tag{2}$$

with

$$p(c_1) = \frac{R_1 + 1}{T} \tag{3}$$

$$p(c_2) = \frac{R_2 + 1}{T} \tag{4}$$

$$p(c_1, c_2) = \frac{R_{12} + 1}{T} \tag{5}$$

Here, R_1 represents the number of articles that cite c_1 but not c_2, i.e. the probability that an article cites concept c_1 but not c_2. R_2 denotes the number of articles that cite c_2 but not c_1, i.e. the probability that an article cites concept c_2 but not c_1. R_{12} means the number of articles that refer both c_1 and c_2, i.e. the probability that an article cites both concepts c_1 and c_2. T stands for the total number of Wikipedia articles. R_1, R_2 and R_{12} all add "1" for the convenience of logarithmic calculation. Finally, the PMI similarity for all concept pairs is normalized to ensure that the PMI value is between 0 and 1.

4.2.2. BERT based Similarity

The second similarity is determined by calculating cosine similarity between concept vectors, generated using the BERT model. BERT, a language model based on the transformer architecture, was introduced by Google researchers in 2018 [37]. In contrast to other language models like Word2vec [38], Phrase2vec [39, 40], GloVe [41], BERT demonstrates superior performance in various tasks. Its strength lies in capturing precise word context within sentences, yielding more accurate semantic representations.

In this paper, we input the first 300 words in the Wikipedia article corresponding to a concept into BERT to generate a concept vector. The reason is that a concept is a word or a phrase, which may be ambiguous. If a piece of text is used to represent a concept, there will be no ambiguity. We use the BERT model to obtain a 768-dimensional embedding vector for each concept, and the cosine similarity between two concepts (c_1, c_2) is defined as follows:

$$BCS(c_1, c_2) = \frac{\vec{v_1} \cdot \vec{v_2}}{\|\vec{v_1}\|\|\vec{v_2}\|} \tag{6}$$

Here, $\vec{v_1}$ and $\vec{v_2}$ represent the vectors of concepts c_1 and c_2 respectively.

4.2.3. Final Similarity Calculation

We merge the above two similarities of each concept pair to get its final similarity. It can be defined as follows:

$$\begin{gathered} Sim(c_1, c_2) = \alpha \cdot PMI(c_1, c_2) + (1 - \alpha) \cdot BCS(c_1, c_2) \\ s.t.\ 0 \leq \alpha \leq 1 \end{gathered} \tag{7}$$

Here, the parameter α is used to set the weight of the two similarities, and the value of α is between 0 and 1. We will also set a threshold t_1, and the concept pair whose similarity is greater than the threshold is considered to have a weak prerequisite relation between the two concepts of the pair.

4.3. Concept Graph Construction

In this manuscript, we create a concept graph where domain concepts serve as nodes and concept relations act as edges. We denote $V = [v_1, v_2, \ldots\ldots, v_n]$ as the set of nodes within the concept graph of a domain. The node v_i represents the concept c_i. Furthermore, we have $\vec{V} = [\vec{v_1}, \vec{v_2}, \ldots, \vec{v_N}] \in \mathbb{R}^{F \times N}$, which encompasses the feature vectors for all nodes within the concept graph. Here, $\vec{v_i}$ signifies the feature vector for concept c_i, and $\vec{v_i} \in \mathbb{R}^F$, where F stands for the number of features in the concept vector. These concept vectors are essentially generated by the BERT model. N represents the total number of concepts within the domain. Additionally, $A \in \mathbb{R}^{N \times N}$ stands as the adjacency matrix for the concept graph, outlining the relations between concept pairs or the edges connecting the nodes.

4.4. Graph Attentional Layer

There should not be only one embedded representation for each concept. A concept should have different embedding representations in different contexts to meet the requirements of various tasks. In our work, the context of each concept is the concept graph in which it resides. We should analyze the relationship between the current concept and all other concepts in the concept graph to discover prerequisite relations between concepts. However, Due to the limitation of computing power, it is obviously impossible to deal with all the node pairs in the concept graph. For each node, we have to use limited computing resources to analyze those nodes that are most closely related to it, not all nodes in the graph. We believe that the related nodes can play a more important role in finding prerequisite concepts. Therefore, we introduce the attention mechanism [42] to evaluate the importance of the related nodes to the current node. Inspired by [28, 43], the proposed approach uses a graph attentional layer to aggregate the meaningful neighbors of each node and extract the latent representation of the node.

For the nodes in the concept graph, we use self-attention to learn their attention coefficients. The input sequence to our graph attentional layer is $\vec{V}$, i.e. the feature vectors of all the nodes. The graph attentional layer generate a new set of feature vectors, $X = [x_1, \ldots, x_N] \in \mathbb{R}^{F' \times N}$, as its output.

First of all, in order to transform the input features into higher-level features, a shared linear transformation, parametrized by a weight matrix, $W \in \mathbb{R}^{F^{'} \times F}$, is applied to every node in the concept graph. After that, we perform self-attention on all the nodes. Given a pair of nodes (v_i, v_j), we can calculate the attention coefficient e_{ij} for the pair. e_{ij} indicates the importance of the node v_j to the node v_i. The attention coefficient is calculated as

$$e_{ij} = \alpha(W\vec{v_i}, W\vec{v_j}) \tag{8}$$

where $\alpha \in \mathbb{R}^{F^{'} \times F^{'}}$ represents the shared attentional mechanism. Generally speaking, the model allows every node to attend on any other nodes in the concept graph. However, in terms of computing resources and the importance of nodes, we will only focus on the pairs of nodes that are linked to each other. In other words, we only compute e_{ij} for the nodes $v_j \in N_i$, where N_i is the set of neighbors of the node v_i (including itself) in the concept graph. To make attention coefficients easily comparable across different nodes, we normalize them across all choices of v_j using the softmax function:

$$\begin{aligned} \alpha_{ij} &= softmax(e_{ij}) \\ &= \frac{\exp(e_{ij})}{\sum_{k \in N_i} \exp(e_{ik})} \\ &= \frac{\exp(LeakyReLU(\vec{\alpha}^T [W\vec{v_i} \| W\vec{v_j}]))}{\sum_{k \in N_i} \exp(LeakyReLU(\vec{\alpha}^T [W\vec{v_i} \| W\vec{v_j}]))} \end{aligned} \tag{9}$$

Here, the attention mechanism α in Eq.(8) is a single-layer feedforward neural network, parametrized by a weight vector $\vec{\alpha} \in \mathbb{R}^{2F^{'}}$, and applying the *LeakyReLU* as nonlinearity activation function. Besides, $\bullet^T$ denotes transposition, and $\|$ is the concatenation operation.

It should be pointed out that the attention coefficients e_{ij} is asymmetric. In other words, e_{ij} is not equal to e_{ji}. Because the set of neighbor nodes of v_i and v_j may be different. Accordingly, the importance of v_i to v_j is different from the importance of v_j to v_i.

After obtaining the normalized attention coefficients between node pairs, we then begin to calculate a linear combination of the feature vectors of nodes $v_j \in N_i$, to serve as the higher-level feature vector for v_i:

$$h_i = \sigma(\sum_{v_j \in N_i} \alpha_{ij} W\vec{v_j}) \tag{10}$$

where h_i is the higher-level feature vector of v_i, σ is the sigmoid activation function.

In order to stabilize the learning process of self-attention, we extend attention mechanism to multi-head self-attention. We repeat the transformation of Eq.(10) for K times, and then concatenate their feature vectors. Then the output feature vectors are updated as follows:

$$h_i = \Big\|_{k=1}^{K} \sigma(\sum_{v_j \in N_i} \alpha_{ij}^k W_k \vec{v_j}) \tag{11}$$

where $\|$ stands for the concatenation operation. α_{ij}^k are normalized attention coefficients computed by the k-th attention mechanism. W_k is the corresponding input linear transformation's weight matrix. Accordingly, the output feature vectors h_i will consist of $KF^{'}$ features for every node, rather than $F^{'}$. In order to reduce the dimension of node features from $KF^{'}$ to $F^{'}$, we use a fully connected neural network to obtain the final feature vector of a node.

$$x_i = ReLU(h_i W_r + b_r) \tag{12}$$

where $W_r \in \mathbb{R}^{KF^{'} \times F^{'}}$, $b_r \in \mathbb{R}^{F^{'}}$. All these parameters can be learned during the training process, and the calculation results are activated by the *ReLU* function. Finally, we can get the new feature vectors of all the concepts.

4.5. Encoding-Decoding Layer

In this section, we will introduce the Encoding-Decoding layer which utilize node feature vectors X and the adjacency matrix A to generate a reconstructed concept graph and obtain candidate concept prerequisite relations. The Encoding-Decoding layer is implemented by a Variational Graph Auto-Encoders (VGAE) model. The VGAE is a framework for unsupervised learning on graph-structured data based on variational auto-encoders [44], which can learn interpretable latent representations for undirected graph.

In the encoding stage, we take the node feature vectors X and the adjacency matrix A as input, and try to recover the graph adjacency matrix by the hidden layer embeddings Z. The encoder is composed of a two-layer GCN:

$$GCN(X, A) = \tilde{A} ReLU(AXW_0)W_1 \tag{13}$$

where $\tilde{A} = D^{\frac{1}{2}} A D^{\frac{1}{2}}$ is the new adjacency matrix at the second graph layer, and D is the degree matrix of the graph. W_0 and W_1 are the parameters of the first and second layers, respectively.

Then, in the variational graph auto-encoder, the goal is to sample the hidden layer embeddings Z via a normal distribution, that is

$$q(Z|X, A) = \prod_{i=1}^{N} q(z_i|X, A) \tag{14}$$

with

$$q(z_i|X, A) = N(z_i|\mu_i, diag(\sigma_i^2)) \tag{15}$$

Here, $\mu = GCN_\mu(X, A)$ is the matrix of mean vectors, and $\log \sigma = GCN_\sigma(X, A)$.

In the decoding stage, the reconstructed adjacency matrix is the inner product of the latent parameters Z, that is

$$p(\hat{A}|Z) = \prod_{i=1}^{N} \prod_{j=1}^{N} p(A_{ij}|z_i, z_j) \tag{16}$$

with

$$p(A_{ij} = 1|z_i, z_j) = \sigma(z_i^T z_j) \tag{17}$$

Here, A_{ij} are the elements of the reconstructed adjacency matrix $\hat{A}$. And $\sigma(\bullet)$ is the logistic sigmoid function.

We optimize the variational lower bound L with respect to the variational parameters W_i:

$$L = E_{q(Z|X,A)}[\log p(A|Z)] - KL[q(Z|X,A)\|p(Z)] \tag{18}$$

where $KL[q(\bullet)\|p(\bullet)]$ is the Kullback-Leibler divergence between $q(\bullet)$ and $p(\bullet)$.

4.6. Edge Direction Inferring

The reconstructed graph generated by the Encoding-Decoding layer is an undirected graph. If two nodes in the reconstructed graph are connected by an edge, it indicates that there is a prerequisite relation between the concepts of the two nodes. But we do not know which one is the prerequisite concept, because the edge has no direction. In practical, the direction of an edges is very important for the prerequisite relation. In applications such as learning resources sequencing and learning path generation, the direction of a prerequisite relation directly determines the order in which concepts are learned.

In order to solve the problem that the edges in the reconstructed graph have no direction, we split the adjacency matrix of the original concept graph into an upper triangular matrix and a lower triangular matrix, which are respectively input into the VGAE model, and finally two reconstruction matrices will be obtained. After that, we merge the part above the main diagonal of the former with the part below the main diagonal of the latter into one matrix, denoted as M. The combined result M will not be a symmetric matrix, and M_{ij} represents the degree of dependence of concept c_i on concept c_j. Similarly, M_{ji} represents the degree of dependence of c_j on c_i.

However, M_{ij} and M_{ji} cannot exist at the same time according to the Anti-symmetry of prerequisite relation mentioned above. Therefore, we keep the element with the larger absolute value of the two elements, and set the other element to 0, so that we can determine the direction of all node pairs in the graph. Besides, we will not convert all the elements in the matrix M whose absolute value is greater than 0 into edges between nodes. The absolute value of some elements is too small, indicating that the dependence between the two concepts is not strong enough. So, we're going to set a threshold t_2. All elements whose absolute value is less than the threshold will be discarded when constructing the concept graph.

5. PERFORMANCE ANALYSIS

5.1. Datasets

In this paper, we verify the effectiveness of the proposed approach on both English and Chinese datasets. The English dataset, AL-CPL, is an English concept prerequisite relation dataset proposed by Liang et al. [31]. This dataset contains prerequisite pairs in four domains, namely Data Mining, Geometry, Physics, and Precalculus. These English concepts are originally from several textbooks [45], and each concept corresponds to a Wikipedia article. Based on their work, Liang et al. [31] further expanded the concept prerequisite pairs and corrected some of the original mislabels in the initial datasets, and created the AL-CPL dataset.

On the basis of the AL-CPL dataset, we also created a Chinese dataset ZH-AL-CPL through the cross-language links of Wikipedia. We tried to find a corresponding Chinese concept for each English concept in the AL-CPL datasets from the Chinese Wikipedia. If an English concept does not have a corresponding Chinese concept, then the concept pair to which it belongs will be discarded when creating the Chinese datasets. Table 1 summarizes the statistics of the final datasets. The code and data are available at https://github.com/586542484/experiment.

Table 1. Datasets statistics

Dataset	Domain	#Concepts	#Pairs	#Prerequisites
AL-CPL	Data Mining	120	826	292
	Geometry	89	1681	524
	Physics	153	1962	487
	Precalculus	224	2060	699
ZH-AL-CPL	Data Mining	89	558	201
	Geometry	78	1391	449
	Physics	133	1657	390
	Precalculus	197	1758	590

5.2. Metrics

In this section, we will evaluate the performance of the proposed approach in terms of its ability to discover concept prerequisite relations. We use Precision, Recall and F1-score to evaluate the performance of the proposed approach. It can be calculated as follows:

$$Precision = \frac{\text{\# of correctly directed edges in learned model}}{\text{\# of directed edges in learned model}} \tag{19}$$

$$Recall = \frac{\text{\# of correctly directed edges in learned model}}{\text{\# of directed edges in true model}} \tag{20}$$

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{21}$$

In addition, we also use the Area Under the Curve of ROC (AUC) which is popular in link pridiction evaluations.

5.3. Experimental Settings

According to our previous experiments, we set the parameters in the experiment as follows: the concept similarity weight parameter α is set to 0.5, the two thresholds t1 and t2 are both set to 0.6, the learning rate is set to 0.0001, the weight decay is set to 0.04, and the number of attention heads is set to 3. All the experiments were performed on a single PC with Intel Core i5 CPU 2.5GHz and 16 GB memory, running Windows 11 operating system.

To verify the effectiveness of the proposed approach, we applied the following five different experimental settings:

(1) wLabel: We use all the concept pairs with weak supervision relationship as the training set, and all other concept pairs with the real concept prerequisite relations as the positive examples of the test set, and then randomly add the same number of negative samples to the test set.

(2) Label_-half: We randomly remove half of the mislabels in weak supervision labels to verify whether reducing mislabels can affect the performance of the weakly supervised learning method.

(3) Label_-all: We remove all mislabels in weak supervision labels for the same reason mentioned above.

(4) Label_15%: To compare the weakly supervised learning setting with the semi-supervised learning setting, we added 15% of the concept pairs with true prerequisite relations to the training set to observe the change in the performance of the proposed method. We did not remove weak labels in the original training set. However, if the weak labels conflict with the newly added real labels, we will replace the weak labels with real labels.

(5) Label_30%: In the same way as the previous setting, we added 30% of the concept pairs with true prerequisite relations to the training set.

For each of the above experiments, we did five times for each experiment setting, and took the average value as its result.

5.4. Compare with Baselines

We use the following state-of-the-art approaches as baselines. It must be noted that the baselines are all supervised learning methods. As mentioned above, there are still some semi supervised, weakly supervised and unsupervised learning methods, but these methods need the support of learning resources when inferring concept relations. This is different from the hypothesis of our work, so these methods cannot be used here. The baseline approaches are as follows.

(1) **Linguistically-Driven Strategy**: The first work is presented by Miaschi et al. [21]. The authors defined three models: M1 (only using embedding features), M2 (only using handcraft features), M3 (merging M1 with M2). So, there are actually three baseline methods. For M1, the pre-trained word embeddings computed for the first 400 words of a Wikipedia page, i.e. Wikipedia concept, are inputted into a LSTM network and generate the embedding feature of the concept. For M2, the authors define 16 concept pair features for concept prerequisite learning.

Table 2. Comparison with baselines on the English datasets

Methods	Metrics	Data Mining	Geometry	Physics	Precalculus
M1 [21]	P	0.736	0.890	0.764	0.842
	R	0.747	0.850	0.699	0.816
	F1	0.741	0.869	0.730	0.829
	AUC	0.897	0.964	0.921	0.944
M2 [21]	P	0.741	0.780	0.707	0.806
	R	0.708	0.731	0.557	0.729
	F1	0.723	0.754	0.621	0.765
	AUC	0.855	0.876	0.829	0.886
M3 [21]	P	0.819	0.917	0.775	0.875
	R	0.815	**0.876**	**0.718**	0.865
	F1	0.817	**0.896**	**0.743**	0.870
	AUC	**0.953**	**0.979**	0.930	0.961
AT [31]	P	0.807	**0.950**	**0.852**	**0.902**
	R	0.733	0.847	0.593	**0.871**
	F1	0.767	0.895	0.699	**0.886**
	AUC	0.922	0.978	**0.939**	**0.975**
NN [46]	P	0.627	0.706	0.530	0.699
	R	0.682	0.760	0.623	0.769
	F1	0.651	0.730	0.571	0.731
	AUC	0.801	0.892	0.828	0.895
RefD [18]	P	0.517	0.424	0.499	0.751
	R	0.762	0.623	0.496	0.694
	F1	0.614	0.504	0.494	0.721
	AUC	0.695	0.624	0.677	0.792
Proposed					
wLabel	P	0.664	0.756	0.562	0.624
	R	0.653	0.530	0.467	0.470
	F1	0.656	0.617	0.509	0.530
	AUC	0.650	0.682	0.538	0.588
Label_half	P	0.760	0.763	0.692	0.735
	R	0.737	0.530	0.659	0.762
	F1	0.747	0.620	0.674	0.748
	AUC	0.774	0.685	0.725	0.804
Label_all	P	**0.913**	0.662	0.771	0.781
	R	**0.937**	0.641	0.701	0.813
	F1	**0.925**	0.652	0.734	0.796
	AUC	**0.953**	0.733	0.770	0.844
Label_15%	P	0.709	0.749	0.679	0.726
	R	0.575	0.544	0.538	0.466
	F1	0.633	0.621	0.589	0.560
	AUC	0.651	0.678	0.650	0.658
Label_30%	P	0.656	0.716	0.718	0.726
	R	0.719	0.618	0.691	0.661
	F1	0.685	0.655	0.704	0.687
	AUC	0.709	0.720	0.753	0.744

(2) **Active Learning (AT)**: This method is proposed by Liang et al. [31]. It uses 15 link-based and 17 text-based features to learn the prerequisite relations among Wikipedia concepts. The authors also tested the active learning strategies in concept prerequisite learning tasks. We choose the results reported by the random forest classifier for comparison, which is also the best classifier in their paper.

(3) **Neural Network (NN)**: This approach is proposed by the UNIGE_SE team in the EVALITA 2020 shared task on Prerequisite Relation Learning (PRELEARN) [46]. The authors developed a neural network classifier that exploits features extracted both from raw text and the structure of the Wikipedia pages, so as to capture concept prerequisite relations.

(4) **Reference Distance (RefD)**: This metric is also proposed by Liang et al. [18]. It measures how differently two

concepts refer to each other in order to capture prerequisite relations among Wikipedia concepts. A lot of studies use RefD as a baseline. Also, there are many studies that use the values of RefD as a feature of concept pairs to identify prerequisite relations.

For the sake of fairness, we repeat the experiments of each baseline method five times, and take the average value as their respective experimental results. The experimental results on the English and Chinese datasets are shown in Table 2 and Table 3, respectively.

On the English dataset, the average F1 value of wLabel is 57.8%. When half and all of the mislabels in the training set are removed, the performance of the proposed approach is improved by 11.9% and 19.9%, respectively. When 15% and 30% of the real labels are added to the training set, the performance of the proposed approach is improved by 2.3% and 10.5% respectively. Among these five methods, the performance of Label_-all is the best. In terms of average F1, it also outperforms M2, NN, and RefD by 6.10%, 10.6%, and 19.3%, respectively. Especially on the Data Mining dataset, the performance of Label_-all even exceeds all six supervised learning methods.

On the Chinese dataset, the average F1 value of wLabel is 59.3%. When half and all of the mislabels in the training set are removed, the performance of the proposed approach is improved by 3.5% and 12%, respectively. When 15% and 30% of the real labels are added to the training set, the performance of the proposed approach is improved by 2.5% and 8.3% respectively. Among the five proposed methods, the performance of Label_-all is the best. In terms of average F1, it also outperforms M2, NN, and RefD by 3.5%, 6%, and 11.4%, respectively.

Table 3. Comparison with baselines on the Chinese datasets

Methods	Metric	Data Mining	Geometry	Physics	Precalculus
M1 [21]	P	0.733	0.848	0.722	0.792
	R	0.708	0.837	0.631	0.774
	F1	0.718	0.842	0.673	0.782
	AUC	0.855	0.928	0.833	0.914
M2 [21]	P	0.614	0.759	0.698	0.816
	R	0.545	0.715	0.545	0.767
	F1	0.575	0.735	0.611	0.79
	AUC	0.745	0.836	0.818	0.891
M3 [21]	P	0.788	**0.894**	0.762	0.876
	R	0.758	0.883	0.705	0.844
	F1	0.772	**0.888**	0.732	0.859
	AUC	0.872	**0.963**	0.865	0.939
AT [31]	P	**0.874**	0.893	**0.873**	**0.912**
	R	**0.801**	0.875	0.812	**0.893**
	F1	**0.836**	0.884	**0.841**	**0.902**
	AUC	**0.903**	0.957	**0.895**	**0.967**
NN [46]	P	0.601	0.722	0.684	0.772
	R	0.517	0.724	0.538	0.725
	F1	0.543	0.722	0.602	0.746
	AUC	0.711	0.754	0.734	0.803
RefD [18]	P	0.509	0.614	0.582	0.676
	R	0.524	0.638	0.547	0.704
	F1	0.516	0.626	0.564	0.69
	AUC	0.563	0.743	0.634	0.722
Proposed					
wLabel	P	0.598	0.772	0.710	0.734
	R	0.499	0.521	0.446	0.610
	F1	0.543	0.620	0.543	0.665
	AUC	0.608	0.709	0.616	0.713
Label_-half	P	0.646	0.755	0.684	0.769
	R	0.481	0.570	0.570	0.638
	F1	0.548	0.648	0.621	0.696
	AUC	0.617	0.711	0.674	0.763
Label_-all	P	0.768	0.771	0.729	0.842
	R	0.668	0.549	0.709	0.726
	F1	0.714	0.640	0.719	0.779
	AUC	0.743	0.701	0.753	0.837
Label_15%	P	0.631	0.793	0.730	0.767
	R	0.504	0.580	0.451	0.635
	F1	0.557	0.665	0.555	0.693
	AUC	0.633	0.733	0.632	0.739
Label_30%	P	0.638	0.790	0.727	0.858
	R	0.562	0.642	0.567	0.694
	F1	0.597	0.706	0.635	0.767
	AUC	0.627	0.743	0.667	0.800

5.5. Case Study

In this section, we further studied the examples of concept relations recovered by our wLabel method. The concept pairs are all from the English dataset Precalculus. Table **??** lists both correct and incorrect examples. The column "Concept graph" denotes that whether there is an edge between two concepts in the original concept graph. The column "Learned model" represents directed edges generated by the learned model. The column "True model" stands for the real concept prerequisite relations. If there is a prerequisite relation between two concepts, the value is 1; otherwise, the value is 0.

Looking closely at the correct examples, although there is no edge between two concepts in the original concept graph, the proposed method can still correctly predict the prerequisite relations between them, for example $\langle Radius, Geometry \rangle$. There are also some concept pairs with prerequisite relations, which also have edges in the original concept graph. The proposed method can also correctly discover the prerequisite relations, for instance $\langle Exponential\_function, Arithmetic \rangle$. Besides, there are also some concept prerequisite relations that our method did not correctly capture. These concepts are often advanced concepts, and there are very few prerequisites related to them. It is often difficult to identify the prerequisite relations of such concepts.

6. CONCLUSION

In this paper, we study the problem of concept prerequisite relation learning. This problem has important research value, because concept prerequisite relations can be applied in various educational applications, such as intelligent tutoring systems, curriculum planning, learning resources sequencing, reading list generation, learning path generation, and knowledge tracing, etc. We propose a weakly supervised learning approach for concept prerequisite relation discovering. Al-

Table 4. Examples of correct and incorrect examples

	Concept pairs	Concept graph	Learned model	True model
Correct examples	⟨ Spherical_law_of_cosines, Mathematics ⟩	0	1	1
	⟨ Radius, Geometry ⟩	0	1	1
	⟨ Polynomial_long_division, Arithmetic ⟩	0	1	1
	⟨ Descartes'_rule_of_signs, Number ⟩	0	1	1
	⟨ Exponential_function, Arithmetic ⟩	1	1	1
	⟨ Analytic_geometry, Mathematics ⟩	1	1	1
	⟨ Synthetic_division, Multiplication ⟩	1	1	1
	⟨ Circle, Mathematics ⟩	1	1	1
Incorrect examples	⟨ Sine_wave, Trigonometry ⟩	0	0	1
	⟨ Linear_inequality, Number ⟩	1	0	1
	⟨ Area, Geometry ⟩	0	1	0

though, weakly supervised learning methods do not perform as well as supervised learning methods in most cases. However, in many real environments, labeled data is often scarce. Manual annotating data takes a lot of time and cost. Therefore, semi-supervised, weakly supervised, and unsupervised learning methods are usually more reasonable choices.

In the future, we will combine Wikipedia page links and clickstream data to improve the accuracy of weak labels to better capture concept prerequisite relations. We will also investigate how to make our model easy to use to support more downstream educational applications.

7. REFERENCES

[1] Elham Mousavinasab, Nahid Zarifsanaiey, Sharareh R. Niakan Kalhori, Mahnaz Rakhshan, Leila Keikha, and Marjan Ghazi Saeedi, "Intelligent tutoring systems: a systematic review of characteristics, applications, and evaluation methods," *Interactive Learning Environments*, vol. 29, no. 1, pp. 142–163, 2021.

[2] José Paladines and Jaime Ramirez, "A systematic literature review of intelligent tutoring systems with dialogue in natural language," *IEEE Access*, vol. 8, pp. 164246–164267, 2020.

[3] Carla Limongelli, Filippo Sciarrone, Paolo Starace, and Marco Temperini, "An ontology-driven olap system to help teachers in the analysis of web learning object repositories," *Information Systems Management*, vol. 27, no. 3, pp. 198–206, 2010.

[4] Carla Limongelli, Matteo Lombardi, Alessandro Marani, Filippo Sciarrone, and Marco Temperini, "A recommendation module to help teachers build courses through the moodle learning management system," *New Review of Hypermedia and Multimedia*, vol. 22, pp. 58–82, 01 2016.

[5] Olga Revilla Muñoz, Francisco Alpiste Penalba, and Joaquín Fernández Sánchez, "The skills, competences, and attitude toward information and communications technology recommender system: an online support program for teachers with personalized recommendations," *New Review of Hypermedia and Multimedia*, vol. 22, no. 1-2, pp. 83–110, 2016.

[6] Peter Brusilovsky and Julita Vassileva, "Course sequencing techniques for large-scale web-based education," *International Journal of Continuing Engineering Education and Life Long Learning*, vol. 13, no. 1-2, pp. 75–94, 2003.

[7] Carla Limongelli, Fabio Gasparetti, and Filippo Sciarrone, "Wiki course builder: A system for retrieving and sequencing didactic materials from wikipedia," in *2015 International Conference on Information Technology Based Higher Education and Training (ITHET)*, 2015, pp. 1–6.

[8] Georgia Koutrika, Lei Liu, and Steve Simske, "Generating reading orders over document collections," in *2015 IEEE 31st International Conference on Data Engineering*, 2015, pp. 507–518.

[9] Jonathan Gordon, Linhong Zhu, Aram Galstyan, Prem Natarajan, and Gully Burns, "Modeling concept dependencies in a scientific corpus," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany, Aug. 2016, pp. 866–875, Association for Computational Linguistics.

[10] Fuhua Lin, Sabine Graf, Rory McGreal, et al., "Adaptive and intelligent web-based educational systems," *Guest Editorial Preface*, 2004.

[11] Haoran Xie, Hui-Chun Chu, Gwo-Jen Hwang, and Chun-Chieh Wang, "Trends and development in technology-enhanced adaptive/personalized learning: A systematic review of journal publications from 2007 to 2017," *Computers & Education*, vol. 140, pp. 103599, 2019.

[12] Shuting Wang and Lei Liu, "Prerequisite concept maps extraction for automaticassessment," in *Proceedings of the 25th International Conference Companion on World Wide Web*, Republic and Canton of Geneva, CHE, 2016, WWW '16 Companion, p. 519–521, International World Wide Web Conferences Steering Committee.

[13] Yetian Chen, José P González-Brenes, and Jin Tian, "Joint discovery of skill prerequisite graphs and student models.," *International Educational Data Mining Society*, 2016.

[14] Penghe Chen, Yu Lu, Vincent W. Zheng, and Yang Pian, "Prerequisite-driven deep knowledge tracing," in *2018 IEEE International Conference on Data Mining (ICDM)*, 2018, pp. 39–48.

[15] Chen Liang, Jianbo Ye, Zhaohui Wu, Bart Pursel, and C. Giles, "Recovering concept prerequisite relations from university course dependencies," *Proceedings of*

the AAAI Conference on Artificial Intelligence, vol. 31, no. 1, Feb. 2017.

[16] Thomas N Kipf and Max Welling, "Variational graph auto-encoders," *arXiv preprint arXiv:1611.07308*, 2016.

[17] Partha Talukdar and William Cohen, "Crowdsourced comprehension: Predicting prerequisite structure in Wikipedia," in *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, Montréal, Canada, June 2012, pp. 307–315, Association for Computational Linguistics.

[18] Chen Liang, Zhaohui Wu, Wenyi Huang, and C. Lee Giles, "Measuring prerequisite relations among concepts," in *Conference Proceedings - EMNLP 2015*. 2015, Conference Proceedings - EMNLP 2015: Conference on Empirical Methods in Natural Language Processing, pp. 1668–1674, Association for Computational Linguistics (ACL), Publisher Copyright: © 2015 Association for Computational Linguistics.; Conference on Empirical Methods in Natural Language Processing, EMNLP 2015 ; Conference date: 17-09-2015 Through 21-09-2015.

[19] Liangming Pan, Chengjiang Li, Juanzi Li, and Jie Tang, "Prerequisite relation learning for concepts in moocs," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 1447–1456.

[20] Mohsen Sayyadiharikandeh, Jonathan Gordon, Jose-Luis Ambite, and Kristina Lerman, "Finding prerequisite relations using the wikipedia clickstream," in *Companion Proceedings of The 2019 World Wide Web Conference*, New York, NY, USA, 2019, WWW '19, p. 1240–1247, Association for Computing Machinery.

[21] Alessio Miaschi, Chiara Alzetta, Franco Alberto Cardillo, and Felice Dell'Orletta, "Linguistically-driven strategy for concept prerequisites learning on italian," in *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 2019, pp. 285–295.

[22] Yang Zhou and Kui Xiao, "Extracting prerequisite relations among concepts in wikipedia," in *2019 International Joint Conference on Neural Networks (IJCNN)*, 2019, pp. 1–8.

[23] Rubén Manrique, Bernardo Pereira, and Olga Mariño, "Exploring knowledge graphs for the identification of concept prerequisites," *Smart Learning Environments*, vol. 6, no. 1, pp. 1–18, 2019.

[24] Haoyu Wen, Xinning Zhu, Moyu Zhang, Chunhong Zhang, and Changchuan Yin, "Combining wikipedia to identify prerequisite relations of concepts in moocs," in *Neural Information Processing: 28th International Conference, ICONIP 2021, Sanur, Bali, Indonesia, December 8–12, 2021, Proceedings, Part V 28*. Springer, 2021, pp. 739–747.

[25] Cheng Hu, Kui Xiao, Zesong Wang, Shihui Wang, and Qifeng Li, "Extracting prerequisite relations among wikipedia concepts using the clickstream data," in *Knowledge Science, Engineering and Management: 14th International Conference, KSEM 2021, Tokyo, Japan, August 14–16, 2021, Proceedings, Part I*. Springer, 2021, pp. 13–26.

[26] Kui Xiao, Youheng Bai, and Zesong Wang, "Extracting prerequisite relations among concepts from the course descriptions," *International Journal of Software Engineering and Knowledge Engineering*, vol. 32, no. 04, pp. 503–523, 2022.

[27] Sudeshna Roy, Meghana Madhyastha, Sheril Lawrence, and Vaibhav Rajan, "Inferring concept prerequisite relations from online educational resources," in *Proceedings of the AAAI conference on artificial intelligence*, 2019, vol. 33, pp. 9589–9594.

[28] Juntao Zhang, Nanzhou Lin, Xuelong Zhang, Wei Song, Xiandi Yang, and Zhiyong Peng, "Learning concept prerequisite relations from educational data via multi-head attention variational graph auto-encoders," in *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, New York, NY, USA, 2022, WSDM '22, p. 1377–1385, Association for Computing Machinery.

[29] Hao Sun, Yuntao Li, and Yan Zhang, "Conlearn: Contextual-knowledge-aware concept prerequisite relation learning with graph neural network," in *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*. SIAM, 2022, pp. 118–126.

[30] Chen Liang, Jianbo Ye, Han Zhao, Bart Pursel, and C. Lee Giles, "Active learning of strict partial orders: A case study on concept prerequisite relations," 2018.

[31] Chen Liang, Jianbo Ye, Shuting Wang, Bart Pursel, and C Lee Giles, "Investigating active learning for concept prerequisite learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, vol. 32.

[32] Irene Li, Alexander Fabbri, Swapnil Hingmire, and Dragomir Radev, "R-vgae: Relational-variational graph autoencoder for unsupervised prerequisite chain learning," *arXiv preprint arXiv:2004.10610*, 2020.

[33] Juntao Zhang, Hai Lan, Xiandi Yang, Shuaichao Zhang, Wei Song, and Zhiyong Peng, "Weakly supervised setting for learning concept prerequisite relations using multi-head attention variational graph auto-encoders," *Know.-Based Syst.*, vol. 247, no. C, jul 2022.

[34] Haonian Wang, Xinyu Tang, Yurou Liu, and Zhichun Wang, "Extracting isa relations of concepts from books via weakly supervised learning," in *Proceedings of the 11th International Joint Conference on Knowledge Graphs*, New York, NY, USA, 2023, IJCKG '22, p. 91–98, Association for Computing Machinery.

[35] Weiming Lu, Yangfan Zhou, Jiale Yu, and Chenhao Jia, "Concept extraction and prerequisite relation learning from educational data," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, vol. 33, pp. 9678–9685.

[36] Gerlof Bouma, "Normalized (pointwise) mutual information in collocation extraction," *Proceedings of the Biennial GSCL Conference 2009*, 01 2009.

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

[38] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[39] Mikel Artetxe, Gorka Labaka, and Eneko Agirre, "Unsupervised statistical machine translation," *arXiv preprint arXiv:1809.01272*, 2018.

[40] Yongliang Wu, Shuliang Zhao, and Wenbin Li, "Phrase2vec: Phrase embedding based on parsing," *Information Sciences*, vol. 517, pp. 100–127, 2020.

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

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[43] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[44] Diederik P Kingma and Max Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[45] Shuting Wang, Alexander Ororbia, Zhaohui Wu, Kyle Williams, Chen Liang, Bart Pursel, and C. Lee Giles, "Using prerequisites to extract concept maps fromtextbooks," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, New York, NY, USA, 2016, CIKM '16, p. 317–326, Association for Computing Machinery.

[46] Alessio Moggio and Andrea Parizzi, *UNIGE_SE @ PRELEARN: Utility for Automatic Prerequisite Learning from Italian Wikipedia*, pp. 376–380, 01 2020.