Course Concept Expansion in MOOCs with External Knowledge and Interactive Game

Jifan Yu[1,2], **Chenyu Wang**[3], **Gan Luo**[1,2], **Lei Hou**[1,2]*, **Juanzi Li**[1,2], **Zhiyuan Liu**[1,2], **Jie Tang**[1,2]
[1]Dept. of Computer SCi.& Tech., Tsinghua University, China 100084
[2]KIRC, Institute for Artificial Intelligence, Tsinghua University, China 100084
[3]Shenyuan Honors College, Beihang University, China 100083
{yujf18@mails.,luog18@mails.}tsinghua.edu.cn
{houlei@,lijuanzi@,liuzy@,jietang@}tsinghua.edu.cn
wangchenyu@buaa.edu.cn

Abstract

As Massive Open Online Courses (MOOCs) become increasingly popular, it is promising to automatically provide extracurricular knowledge for MOOC users. Suffering from semantic drifts and lack of knowledge guidance, existing methods can not effectively expand course concepts in complex MOOC environments. In this paper, we first build a novel boundary during searching for new concepts via external knowledge base and then utilize heterogeneous features to verify the high-quality results. In addition, to involve human efforts in our model, we design an interactive optimization mechanism based on a game. Our experiments on the four datasets from Coursera[1] and XuetangX[2] show that the proposed method achieves significant improvements(+0.19 by MAP) over existing methods. The source code[3] and datasets[4] have been published.

1 Introduction

Self-determination theory was first formally proposed by Deci and Ryan in (Deci et al., 1991), suggesting that educators should support students in autonomously discovering and learning course-related knowledge. In fact, in addition to the concepts taught in course, many related concepts are also worthy of learning. Figure 1 shows a real example from Coursera in *Data Structure* course. When the concept *Binary Search Tree* is taught, some other concepts, including its similar structures (*Heap*), applications (*Sorting* and *Priority Queue*) and advanced researches (*Tango Tree*[5]) also benefit students for further course understanding. However, these concepts are not available without specifical mention, especially in the era of Massive Open Online Courses (MOOCs). In

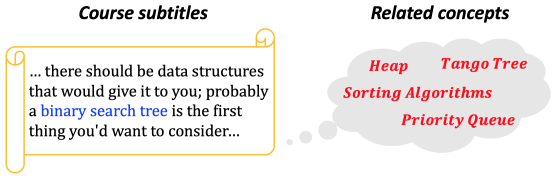


Figure 1: An example of "out-of-teaching" concepts in the course "*Data Structure and Algorithm*".

MOOCs, teachers need to keep a moderate length of the course to face with thousands of students with various backgrounds (Jordan, 2015), making it infeasible to manually pick out these helpful concepts. Therefore, there is a clear need to automatically identify course-related concepts, so that the students can easily acquire additional knowledge and achieve better educational outcomes.

Although much work concerned with extracting course concepts from teaching materials (Kay and Holden, 2002) or course subtitles (Pan et al., 2017b) has been done, the research in finding the concepts absent in course materials, which we call **Course Concept Expansion**, has not been explored. Despite abundant work on related topics, including concept expansion or set expansion (Wang and Cohen, 2007; Wang et al., 2015; Adrian and Manna, 2018), it is far from sufficient to directly apply these methods in the MOOC environments due to the following challenges.

First, unlike the set expansion for a clear general category (e.g., country), course concepts are often the combinations of multiple categories, which is easy to cause semantic drift (Curran et al., 2007) during exploring in different domains (such as Structures: *Heap*, *Binary Tree* and *Algorithms*: *Divide and Conquer*, *Greedy Algorithm*). Second, the features for manifesting course-related concepts are heterogeneous. As shown in Figure 1,

*corresponding author
[1]https://www.coursera.org/
[2]http://www.xuetangx.com/
[3]Source Codes: https://github.com/thukg/concept-expansion-kg
[4]Datasets: http://moocdata.cn
[5]It is an online binary search tree that achieves an $O(\log \log n)$ competitive ratio. (Demaine et al., 2007)

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 4292–4302
Florence, Italy, July 28 - August 2, 2019.

we regard *Heap* as a course concept due to its similar structure while *Binary Search Tree* is a prerequisite concept of *Tango Tree*. Thus mere context information is not enough for effective expansion. Third, as an application-oriented task, it is beneficial to involve human interactions. How to properly leverage the feedback from MOOC users to obtain a better performance for concept expansion remains a challenging issue.

To address the above problems, we propose a three-stage course concept expansion model. Inspired by the idea of concept space (Hori, 1997), we first build an accurate boundary for a given course to alleviate the semantic drift during candidate concept generation from an external knowledge base. Then we transform the expansion into a binary classification problem as previous positive unlabeled learning methods for set expansion (Li et al., 2010; Wang et al., 2017). Three types of features are proposed to incorporate heterogeneous information into classifier to identify high-quality concepts among candidates. Finally, we design a lightweight but attractive top-student game to subtly collect MOOC users' feedback and iteratively optimize the expansion results. For evaluation, we compare the proposed method with 4 representative set expansion methods on real courses from Coursera and XuetangX, and further conduct online evaluation in the game mechanism.

Contributions. Our contributions include: a) the first attempt, to the best of our knowledge, systematically investigate the problem of course concept expansion in MOOCs; b) proposal of an effective three-stage model for course concept expansion using an external knowledge base and interactive game; c) four benchmark datasets using real courses from Coursera and XuetangX.

2 Problem Formulation

In this section, we first give some necessary definitions and then formulate the problem of course concept expansion.

A Course corpus is composed by n courses in the same subject area, denoted as $\mathcal{D} = \{\mathcal{C}_j\}_{j=1}^{n}$, where $\mathcal{C}_j$ is one course. We assume that course $\mathcal{C}_j = \{v_{ij}\}_{i=1}^{m_j}$ consists of m_j course videos, where v_{ij} stands for the i-th video. Following (Pan et al., 2017b), we define **Course Concepts** are the subjects taught in the course denoted as $\mathcal{M} = \{c_i\}_{i=1}^{|\mathcal{M}|}$.

Existing work can extract course concepts $\mathcal{M}$ from course corpus $\mathcal{D}$, but $\mathcal{D}$ could inevitably miss some important course concepts (as illustrated in Figure 1). Therefore, there is a clear need to expand the course concepts beyond $\mathcal{M}$ using external resources. In this paper, we focus on the expansion using external knowledge bases.

Knowledge Base is formally defined as $\mathcal{KB} = (E, R)$, where $E = \{e_i\}_{i=1}^{|E|}$ represents all concepts, $R = \{r_i\}_{i=1}^{|R|}$ represents the relationships between concepts, and (e_i, r_j, e_k) is a triple in $\mathcal{KB}$ meaning e_i has relationship r_j with e_k.

Course Concept Expansion Using Knowledge Base in MOOCs is formally defined as follows. Given the course concepts $\mathcal{M}$ and knowledge base $\mathcal{KB}$, Course Concept Expansion returns a ranked list of expanded concepts $E_c \subset E$, and outputs s_i for $e_i \in E_c$ to indicate its likelihood to be an expanded concept of $\mathcal{D}$.

3 Method

To appropriately expand course concepts in MOOCs, we need to address three crucial problems. 1. How to alleviate semantic drift? 2. How to employ heterogeneous information to identify high-quality expanded concepts? 3. How to properly involve human efforts to optimize the expansion result? In this section, we introduce our novel course concept expansion model in three stages.

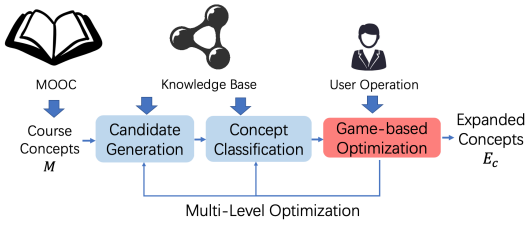


Figure 2: Framework of course concept expansion.

(1)Candidate Generation: To reduce semantic drift, a dynamic boundary is set during our searching for new concepts in $\mathcal{KB}$. We only admit the concepts within the boundary as candidates.

(2)Concept Classification: To leverage heterogeneous information in expansion, we propose three types of novel features to build a classifier, identifying the high-quality expansion concepts among candidates and rerank the result list.

(3)Game-based Optimization: To involve human efforts, we creatively design an interactive online game named top-student game which has been applied in a real MOOC platform to collect

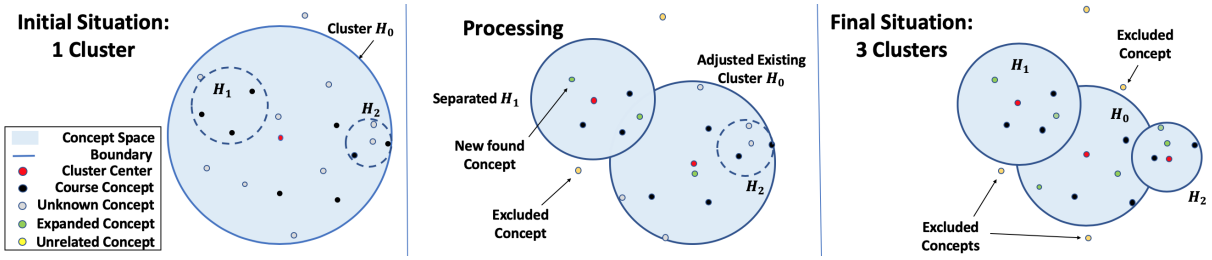


Figure 3: The concept space boundary is fitted in process of searching candidates.

user feedback and cyclically optimize the expansion process at multiple levels.

3.1 Candidate Generation

In this section, we present an unsupervised embedding based algorithm that iteratively generates ranked concept candidates from an external knowledge base $\mathcal{KB}$. In particular, we build a boundary to avoid semantic drift based on the following concept space assumption.

Assumption 3.1 *A course is a concept space which contains one or more concept clusters.*

The idea of concept space was proposed and applied in digital teaching and ontology engineering (Hori, 1997; Cassidy et al., 2006). We extend the assumption by considering concepts' polycentric clustering pattern (e.g., Data Structure course's concepts mainly gather in several clusters, such as graph algorithms, binary trees, etc.). As shown in Figure 3, we initialize the course concepts in one loose cluster and separate the gathering ones into new clusters during candidate generation. An explicit boundary of course is dynamically formed by its clusters to avoid semantic drift.

Given the course corpus $\mathcal{D}$ and knowledge base $\mathcal{KB} = (E, R)$, we first utilize the method in (Pan et al., 2017b,a) to extract course concepts $\mathcal{M}$. Note that we remove the extracted results which do not exist in E, i.e., $\mathcal{M} \subset E$, to facilitate following candidate generation. We use bold-face letters to denote the embeddings of the corresponding terms (i.e., $\mathbf{c_i}$ is the embedding of c_i).

Before introducing the algorithm details, we first define the concept cluster as follows.

Definition 3.1 *A concept cluster H is formed by several semantically related course concepts $\{c_i\}_{i=1,\dots,|H|}$, and is formally represented as a hypersphere $(\mathbf{o}, \gamma)$ with $\mathbf{o}$ and γ denoting its center and radius respectively. Mathematically,*

$$\mathbf{o} = \sum\nolimits_{c_i \in H} \mathbf{c_i}/|H|$$
$$\gamma = \max\nolimits_{c_i \in H} edis(\mathbf{o}, \mathbf{c_i})$$

where $edis(\cdot,\cdot)$ returns the Euclidean distance between the input vectors.

Note that the center $\mathbf{o}$ may be a "virtual" concept, i.e., it does not correspond to any known concept in $\mathcal{M}$ or E. To facilitate the generation process, we introduce a special subset $S_H \subset H$ that includes a fixed size τ (τ is experimentally set to 8) of representative concepts as seeds. We always select the "actual" concepts nearest to the center $\mathbf{o}$, which means that S_H might change dynamically during the generation process. The candidate generation algorithm contains two phases: initialization and searching.

Initialization: We initialize a concept cluster H_0 with all the concepts in $\mathcal{M}$ (as shown in Figure 3), calculate its center $\mathbf{o}_{H_0}$ and radius γ_{H_0}, and select the representative subset S_{H_0}. Then following a predefined order[6], we adapt the single-pass online clustering (Guha et al., 2003) to group the concepts into potential clusters. The clustering algorithm sequentially processes the concepts, one at a time, and grows clusters incrementally. A concept c_i is absorbed by a previously-generated cluster H_i if its Euclidean distance to a concept in H_i is below a predefined threshold[7]; otherwise, the concept is treated a new potential cluster. Finally, we successfully partition the course concepts $\mathcal{M}$ into L potential clusters $H_1, H_2, \dots, H_L$.

Searching: For each concept c_{ij} in a potential cluster H_i, we search its directly-connected concepts in knowledge base $\mathcal{KB}$, e.g., (c_{ij}, r, e) with $e \in E$ and $r \in R$, and use the distance between e and c_{ij} to determine whether to merge it into H_i. Similar to the single-pass clustering, and merge it into H_i if $edis(e, c_{ij}) < \min_{c \in S_{H_i}} edis(\mathbf{o}_{H_i}, \mathbf{c})$.

[6]The course concepts are extracted with the method proposed in Pan et al. (2017b), which also assigns a confidence score for each concept. Here we sort the extracted concepts by the confidence score in descending order.

[7]In experiment, it is set to the minimal distance between representative concepts and the center of H_0, i.e., $\min_{c \in S_{H_0}} edis(\mathbf{o}_{H_0}, \mathbf{c})$

During the process, we separate a cluster from H_0 whenever its size reaches τ (i.e., it is big enough to select representative concepts), update H_0 (including $\mathbf{o}_{H_0}$, γ_{H_0} and S_{H_0}) and calculate its own parameters. For those potential clusters whose size are less than τ, we use $\mathbf{o}_{H_0}$ and and S_{H_0} to make the above decision. When e is merged into H_i, we define the following confidence score s_e to measure its likelihood to be a course concept,

$$s_e = cos(\mathbf{e}, \mathbf{c}_{ij}) + \sum_{c_{ik} \in H_i} cos(\mathbf{c}_{ik}, \mathbf{c}_{ij}) \times cos(\mathbf{e}, \mathbf{c}_{ik}) \quad (1)$$

where $cos(\cdot, \cdot)$ returns the cosine similarity of the input vectors.

After the expansion for all concepts in $\mathcal{M}$, we obtain the expanded concept set $E_c^1 \in E$. Then we sort E_c^1 by s_e in descending order and iteratively repeat the search phase for E_c^1 to obtain E_c^2. The algorithm stops when no concepts in E could be merged. Finally, we achieve $E_c = \bigcup_i E_c^i$ and sort it by s_e in descending order as candidate set.

It's worth noting that each final candidate $e \in E_c$ is directly or indirectly related to a course concept $c \in \mathcal{M}$. e and c are connected by a search path $c \rightarrow r_1 \rightarrow e_1 \rightarrow \ldots \rightarrow e$, where $e_i \in E$ and $r_i \in R$ are concept and relation in $\mathcal{KB}$, and we record such path (denoted as $path(e)$) to hold more semantics. The whole process is summarized in Algorithm 1. Specifically, due to the huge number of operations for finding nearest concepts, we use K-D Tree[8] to store concept vectors, which greatly improves the time efficiency.

Algorithm 1: Candidate Generation

Input: $\mathcal{M}, \mathcal{KB} = (E, R), \tau$
Output: E_c

Sort $\mathcal{M}$, initialize H_0 and further partition into $H_1, H_2, \ldots, H_L$
$t = 0; E_c^0 = \mathcal{M}$
do
  $E_c^{t+1} = \emptyset$
  for $c_{ij} \in H_i \subset E_c^t$ *has related concept* $e \in E$ **do**
    if $edis(e, c_{ij}) < \min_{c \in S_{H_i}} edis(\mathbf{o}_{H_i}, \mathbf{c})$ **then**
      $E_c^{t+1} = E_c^{t+1} \cup \{e\}$
      Merge e into H_i, calculate s_e, record $path(e)$ and update H_i
      if $|H_i| \geq \tau$ **then**
        Separate H_i from H_0 and update H_0
      end
    end
    $E_c = E_c \cup E_c^{t+1}$
  end
  Sort E_c^{t+1} by s_e and $t+ = 1$
while $E_c^t \neq \emptyset$;
Sort E_c by s_e

3.2 Concept Classification

To integrate heterogeneous information in expansion, we propose three features from various sources and rerank the candidates after classification. As a binary classification problem, all existing classifiers can be applied, and we experimentally select XGboost (Chen and Guestrin, 2016). In this section, we introduce the three types of features and how we partially rerank the candidates.

Confidence Score. In accordance to our assumption, the confidence score s_e represents the degree of remoteness between the candidate concept e and the concept cluster H to which it belongs. Thus we select it as the first feature to capture a candidate's basic relevance to the course.

Search Path Encoding. During candidate generation, the search paths insinuate the semantic relations between course concepts. Taking "*Floyd Algorithm*" as an example, its search path, "BFS $\rightarrow$ *InstanceOf* $\rightarrow$ Graph Algorithms $\rightarrow$ *Instance* $\rightarrow$ Floyd Algorithm", indicates that "Floyd Algorithm is a sibling of course concept "BFS". To make effective use of this semantic information, we employ an RNN encoder-decoder neural network (Cho et al., 2014) to encode $path(e)$ for candidate e. Specifically, we train the neural network to take $path(e)$ as input and output the same sequence. Thus, we can obtain a fixed-length vector representation of $path(e)$ from the final hidden state of the RNN encoder.

Prerequisite Features. The course concepts also have an unique relationship called *Prerequisite* (Margolis and Laurence, 1999). Prerequisite concept pair (A, B) means if someone wants to study A, he/she is better to understand B in advance (e.g., *Binary Tree* is a prerequisite concept to *Black-Red Tree*), which indicates how concepts in the course are connected. There are a few previous efforts to extract prerequisite relations from Wikipedia (Talukdar and Cohen, 2012; Liang et al., 2015), textbooks (Yosef et al., 2011; Wang et al., 2016) and MOOCs (Pan et al., 2017a). In this paper, we select five features from (Pan et al., 2017a) that only rely on the course text, and $Pv(a, b)$ is the combination of these five features reflecting the prerequisite likelihood of a to b. Since these features can only measure the relationship between the two concepts that exist in the course, we calculate the prerequisite feature of e

[8] K-D tree is a useful data structure for nearest neighbor searches (Wikipedia).

using its search root phrase c_i as follows:

$$Pf(e) = \frac{cos\,\langle \mathbf{e}, \mathbf{c_i} \rangle * \sum_{c_j \in \mathcal{M}} Pv(c_i, c_j)}{|\mathcal{M}|} \quad (2)$$

Partial Reranking. After feature extraction and classification, each candidate is labeled with a tag P (positive) or N (negative). Then we partially adjust the rankings in low-confidence interval to improve the recall. We define a threshold $\alpha \in [0, 1]$ to control the reranking range and adjust rankings after $\alpha * |E_c|$. Specifically, we sort positive and negative results separately according to their confidence score s and then place the positive results before negative ones. Eventually a reranked expansion list is achieved.

3.3 Game-based Optimization

As an application-oriented task, it is beneficial to properly introduce human efforts to monitor and optimize our expansion model. However, the design of this human-model interaction faces several challenges. For human, we need to ensure the quality and sufficiency of their feedback. For the model, we need to effectively and fully utilize the provided data in model optimization.

Thanks to the multimedia and web platform of MOOCs (Volery and Lord, 2000), we are able to attractively collect feedback from students with diverse backgrounds using an online game, which far exceeds traditional modes in amount. In this section, we introduce the game design and how the collected feedback optimize our model.

Game Design. We design our feedback collector, a game named "Top-Student" by considering its attractiveness and the quality of collected data.

To make game attractive, we place the game under each video v_{ij} of course $\mathcal{D}$, whose basic idea is that the player gains the score and competes with other students of the course by deleting low-quality expanded concepts. It allows users to quickly start with a simple click-to-delete operation. Further, we only show expansion results that are most relevant to the concepts in the video v_{ij}, which increases the affinity of the users who just finished the video. Those design facilitates a wider collection of feedback.

Figure 4 is the game layout in the course of "Introduction to Psychology". The blue concept "IQ" is a course concept c in video, while the orange concepts are its relevant expanded concepts E_c. Users delete low-quality concepts among orange ones and gain different scores while we always record the total deletion times of each expanded concept e_i's (denoted as $del(e_i)$).

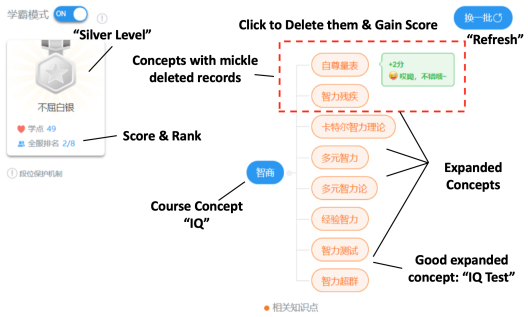


Figure 4: Top-Student Game in course "Introduction to Psychology"

To ensure data quality, we avoid users' irresponsible deletion by employing a group-vote scoring mechanism. Specifically, when a user deletes the expanded concept e_i, he/she gets a score $Q = del(e_i)/\max_{e_j \in E_c} del(e_j) - \frac{1}{2}$, i.e., every user operation corresponds to a score[9] $S \in \left[-\frac{1}{2}, \frac{1}{2}\right]$ based on all existing deletion data, which means that irresponsible operations subject to a penalty.

We set up the game by calculating and storing the expanded concept $e \in E_c$ with highest cosine similarity for each c_i as inputs. Finally we get total deletion $del(e)$ of each e as outputs. The Top-Student Game has been applied in several courses at one of largest Chinese MOOC websites, XuetangX and collected over 10,000 records as of this writing.

Multi-level Optimization. The user feedback affects both candidate generation in Section 3.1 and concept classification in Section 3.2 to perform a multi-level optimization.

We first define and calculate the deletion ratio of a candidate e_i as $Dr(e_i) = del(e_i)/\max_{e_j \in E_c}(del(e_j))$. The value reflects the acceptance of e_i comparing with the other expansion results. Then we present the optimization at two levels.

Candidate Generation: The confidence score s_e in Equation 1 is updated, reducing the likelihood from its directly related concept according to its deletion ratio $Dr(e)$.

$$s_e = cos(\mathbf{e}, \mathbf{c}_{ij}) \times (1 - Dr(e)) + \sum_{c_{ik} \in H_i} cos(\mathbf{c}_{ik}, \mathbf{c}_{ij}) \times cos(\mathbf{e}, \mathbf{c}_{ik}) \quad (3)$$

Concept Classification: We regard deletion ratio $dr(e)$ as a new feature to incorporate user insights into our classifier.

In this way, user feedback is applied to each process of the model, and new results generated after

[9] In real application, the value is enlarged to $[-5, 5]$

optimization are also periodically entered into the game to collect feedback again. Finally it iterates over and gets high-quality expansion results.

4 Experimental Evaluation

4.1 Datasets

Since there is no publicly available dataset for course concept expansion in MOOCs, we use two different domains of Chinese and English courses: "Data Structure and Algorithm" and "Introduction to Psychology" to construct four datasets[10] through a three-stage process.

First, for each domain, we select its most relevant English courses from Coursera and Chinese courses from XuetangX, e.g., for EN-DSA, we select 3 courses[11] of 3 universities and obtain a total of 449 videos. Then, we use the method of Pan (2017b) to extract the course concepts and manually select the high-quality ones as the course concepts $\mathcal{M}$. Finally, we take XLORE (Jin et al., 2018) as $\mathcal{KB}$ to search for related course concepts and manually labeled the reasonable expansion results. For each domain, we collect 100,000 related concepts and record their search path to train the encoder in Section 3.2. But the large amount requires arduous human labeling work, thus we only pick **800** expanded concepts with the highest average cosine similarity to the course concepts to label. For each concept, two human annotators majoring in the corresponding domain are asked to label them as "0: Not related" or "1: Related " based on their own knowledge. Thus, each dataset is doubly annotated, and pearson *correlation* coefficient is applied to assess inter-annotator agreement. A candidate is labeled as a related concept only if the two annotators are in agreement. For each dataset, we split it into training (400), validation (200) and test set (200).

Table 1 presents the detailed statistics, where *#courses*, *#videos*, $|\mathcal{M}|$, *1-Label* and *0-Label* are the number of courses, videos, course concepts, positive and negative labels. We can only obtain *#deletions* from game for Chinese datasets.

	DSA		PSY	
	ZH	EN	ZH	EN
#courses	1	3	1	1
#videos	490	465	57	478
$\lvert\mathcal{M}\rvert$	305	201	575	470
1-Label	398	232	237	246
0-Label	402	568	563	554
correlation	0.696	0.734	0.712	0.681
#deletions	6939	-	4920	-

Table 1: Datasets Statistics

4.2 Experiment Settings

Basic Setting. We choose GloVe (Pennington et al., 2014) as our English word embedding, (Li et al., 2018) as our Chinese word embedding. We follow the same process of (Cho et al., 2014) to train the path encoder and (Pan et al., 2017a) to get prerequisite features for classifier in Section 3.2.

Baseline Methods. We compare our models (simple candidate generation results denoted as MOOC) with four typical methods which employ different similarity metrics. MOOC-C means our model with Classification and MOOC-CG means the whole model added game optimization.

• **PR** Graph based method: We build the candidates and course concepts into a graph. When the similarity between two concepts exceeds a threshold τ_{PR}, there is a link between them. The PageRank score of each candidate is finally used for sorting. A most famous method employing pagerank is SEAL (Wang and Cohen, 2007).

• **SEISA** SEISA(He and Xin, 2011) is an entity set expansion system developed by Microsoft after SEAL and outperforms traditional graph-based methods by an original unsupervised similarity metric. We implement its Dynamic Thresholding algorithm to sort expanded concepts.

• **EBM** Embedding based method mainly utilizes context information to examine the similarity between expanded concepts and seeds like (Mamou et al., 2018). For each expanded concept e, we calculate the pairwise cosine similarity with course concepts $\mathcal{M}$ in word2vec and use the average as golden standard to rank the expanded concept list.

• **PUL** PU learning is a semi-supervised learning model regarding set expansion as a binary classification task. We employ the same setting as (Wang et al., 2017) to classify and sort concepts.

Evaluation Metrics. Our objective is to generate a ranked list of expanded concepts. Thus, to evaluate the ranking result, we use the **Mean Average Precision**(MAP) as our evaluation metric, which is the preferred metric in information retrieval for evaluating ranked lists.

[10]The datasets will be publicly available later.

[11]The three courses: Algorithms (Princeton), Algorithms (Stanford), Data Structure and Algorithm (UC San Diego).

	ZH-PSY	ZH-DSA	ZH-Avg	EN-PSY	EN-DSA	EN-Avg
PR	0.849	0.519	0.684	0.833	0.822	0.827
SEISA	0.877	0.448	0.663	0.814	0.890	0.852
EBM	0.875	0.421	0.648	0.777	0.858	0.817
PUL	0.855	0.680	0.768	0.734	0.795	0.765
MOOC	0.894	0.785	0.839	0.781	0.933	0.857
MOOC-C	0.939	0.804	0.872	**0.922**	**0.976**	**0.949**
MOOC-CG	**0.954**	**0.819**	**0.886**	-	-	-

Table 2: MAP of different methods on datasets. ($S_H = 8, \alpha = 0.4$).

4.3 Overall Evaluation

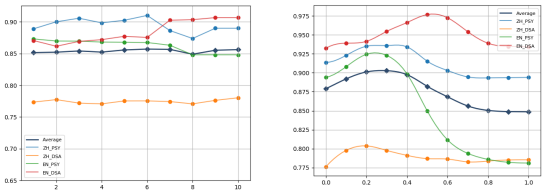
Table 2 summarizes the comparing results of different methods on all datasets, and -avg means the average MAP of datasets in same language. We find that our method outperforms existing methods across all 4 datasets[12]. For example, our whole model surpasses PageRank based method and SEISA by about 0.10 on the average of English courses. Further, we observe the performance in following aspects:

For different datasets, our methods stably perform at a competitive level while existing methods fluctuate fiercely. All methods maintain a better result in English than Chinese. Especially in ZH-DSA, existing methods meet a sharp decline at over 0.17. To find out the reason, we calculate the average pairwise similarity between the extracted concepts $\mathcal{M}$ in each dataset. Results show that ZH-DSA contains the most scattered course concepts at a pairwise distance of 0.60 (ZH-PSY, EN-PSY, EN-DSA at 0.49, 0.50, 0.36). But our expansion achieves a fine result in ZH-DSA at 0.785, indicating it effectively relieves the semantic drift after candidate generation.

For different components of our methods. The pure candidate generation (MOOC) mainly improves the performance by obvious promotion in ZH-DSA. The governing improvement exists after classification (at an average over 0.90), verifying the effectiveness of heterogeneous features we proposed. Moreover, the game-based optimization further improves the performance (+0.14), which proves the power of human efforts and our feedback optimization.

4.4 Result Analysis

The size of seed set τ. The seed set size τ controls how many concepts of E_c and $\mathcal{M}$ should be employed to calculate confidence score. We adjust τ from 1 to 10 and explore the influence of τ on Candidate Generation. Figure 5(a) shows the MAP transmutation. Despite different setting of τ, our model maintain a preeminent competitive performance at an average MAP of 0.85 for English courses and 0.81 for Chinese courses.



(a) The MAP Curve of τ (when $\alpha = 0.4$)

(b) The MAP Curve of α (when $\tau = 8$)

Figure 5: Parameter analysis.

Feature Contribution Analysis. To evaluate the features proposed in Section 3.2 and 3.3,with highest average F1-score at 0.94 as our classifier and run our approach 4 times on the 2 Chinese datasets, with one different feature deleted in each test. Table 3 records the changes of P, R and F_1 for each setting. According to the decrement of F1-scores, we find that all the proposed features are indispensable in classification. Especially, we observe that search path encode Pe plays the most important role, decreasing most F1-score by 7.96%. Besides, user deletion Dr from game also outstandingly increases the precision of classifier by 8.21%.

The ranking threshold α. The ranking threshold α is the parameter controls how much ratio of results in Section 3.1 should be adjusted by classification. As we increase α, less candidates will be adjusted, which weakens the role of the classifier. In Figure 5(b), we set α from 0 to 1 and find that the performance reaches a peak at an average 0.3

[12]The improvements are all statistically significant tested with bootstrap re-sampling with 95% confidence.

Ignored Feature	P	R	F_1
s_e	-2.65%	-1.73%	-1.83%
Pe	-3.97%	**-10.6%**	**-7.96%**
Pf	-7.39%	-1.55%	-3.78%
Dr	**-8.21%**	-3.10%	-4.93%

Table 3: Contribution analysis of different features. s_e, Pe, Pf, Dr are respectively confidence score, search path encoding, prerequisite features and deletion ratio.

	C_r@10	C_r@50	C_r@100
PR	0.034	0.202	0.452
SEISA	0.028	0.210	0.486
EBM	0.043	0.219	0.446
PUL	**0.026**	0.181	0.424
MOOC	0.028	0.166	0.437
MOOC-C	0.028	0.162	0.412
MOOC-CG	0.028	**0.160**	**0.386**

Table 4: Online evaluation results.

of α. The results demonstrate extra information provided by classifier effectively lifts the recall in low-confidence area (latter 70% in average).

4.5 Online Evaluation

In particular, our model uses a gamified form to build a human interface. This interactive design not only optimizes the model, but can also be used to evaluate the effectiveness of the model in practical teaching applications.

By collecting deletion data of the expanded concepts, we can peep whether our expansion results are really helpful to the MOOC users. In order to quantitatively observe the user's feedback on the model, we propose **Average Correction Rate**(C_r) as a game-to-model evaluation metric. This metric is the ratio of the number of times the user deletes the top n concepts to the total deletions, and is formally denoted as $C_r = \sum_{i=1}^{n} del(e_i) / \sum_{j=1}^{|E_c|} del(e_j)$. We set n to 10, 50, 100, and the C_r of each methods are listed in Table 4. Higher C_r indicates less users think the expansion results are helpful. From this perspective, our method is the most helpful to them. For C_r@10, PUL performs a slight advantage at 0.002. However, once the range of observation is broadened to 50 or 100, our method shows an obvious ascendancy. Besides, after adding classifier and game, user satisfaction further increases, reducing the C_r by 0.004 and 0.049 at C_r@50 and C_r@100, reiterating the improvement of these components.

5 Related Works

Our work is based on phrase extraction in MOOCs (Pan et al., 2017b) and is relevant to the set expansion problem, which takes a set of seed entities as input to expand a single category.

Google Sets was an early set expansion system. It spawned quite a few set expansion techniques, such as Bayesian Sets (Ghahramani and Heller, 2006), SEAL (Wang and Cohen, 2007), SEISA (He and Xin, 2011) and others (Sarmento et al., 2007; Wang and Cohen, 2008; Wang et al., 2015). They mainly leverage the similarity between entities measured by their co-occurrences in web texts, wrappers and lists. For example, SEISA employs iterative similarity aggregation and SEAL employs PageRank. Recently, SetExpan (Shen et al., 2017) extend previous works by selecting context features and (Mamou et al., 2018) skillfully employ five different type of context information and gain a very competitive result.

Distinctively, PU-Learning methods (Li et al., 2010; Wang et al., 2017) transform set expansion into a two-class classification problem. A seed set is regarded as a set P of positive examples and candidate set is a set U containing hidden positive and negative cases. The task of filtering the candidate set turns to building a classifier to test if each candidate member is positive or not and this inspires our classification work.

Our approach also benefits from theories of pedagogy. Concept space was first proposed to benefit knowledge comprehension in education(Hori, 1997), and was gradually employed in domain ontology representation; Online games were already used in (Kiili, 2005; Threatt, 2014) education and were also applied for crowdsourcing information collection(Yang et al., 2018). Both of them significantly affected our design of model.

6 Conclusions and Future Work

We conducted a new investigation on automatically course concept expansion in MOOCs. We precisely define the problem and propose an active model to search external knowledge base for candidate concepts and detect high-quality ones with a classifier. Moreover, we design a game-based mechanism to subtly involve human efforts in model optimization. Experimental results on online courses with different domains validate the effectiveness of the proposed method. Promising future directions would be to investigate how

to utilize user interaction in MOOCs more adequately, as well as how attributes of course concepts can help expanding.

Acknowledgments

The work is supported by NSFC key project (U1736204, 61533018, 61661146007), Ministry of Education and China Mobile Joint Fund (MCM20170301), and THUNUS NExT Co-Lab.

Zhiyuan Liu is supported by the National Key Research and Development Program of China (No. 2018YFB1004503).

References

Weronika T. Adrian and Marco Manna. 2018. Navigating online semantic resources forentity set expansion. In *Practical Aspects of Declarative Languages*, pages 170–185, Cham. Springer International Publishing.

Kathryn Cassidy, John Walsh, Brian Coghlan, and Declan Dagger. 2006. Using hyperbolic geometry for visualisation of concept spaces for adaptive e-learning. In *A3H: 1st Inter. Workshop on Authoring of Adaptive & Adaptable Hypermedia*.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

James R Curran, Tara Murphy, and Bernhard Scholz. 2007. Minimising semantic drift with mutual exclusion bootstrapping. In *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics*, volume 6, pages 172–180. Citeseer.

Edward L Deci, Robert J Vallerand, Luc G Pelletier, and Richard M Ryan. 1991. Motivation and education: The self-determination perspective. *Educational psychologist*, 26(3-4):325–346.

Erik D Demaine, Dion Harmon, John Iacono, and Mihai P a traşcu. 2007. Dynamic optimalityalmost. *SIAM Journal on Computing*, 37(1):240–251.

Zoubin Ghahramani and Katherine A Heller. 2006. Bayesian sets. In *Advances in neural information processing systems*, pages 435–442.

Sudipto Guha, Adam Meyerson, Nina Mishra, Rajeev Motwani, and Liadan O'Callaghan. 2003. Clustering data streams: Theory and practice. *IEEE transactions on knowledge and data engineering*, 15(3):515–528.

Yeye He and Dong Xin. 2011. Seisa: set expansion by iterative similarity aggregation. In *Proceedings of the 20th international conference on World wide web*, pages 427–436. ACM.

Koichi Hori. 1997. Concept space connected to knowledge processing for supporting creative design. *Knowledge-Based Systems*, 10(1):29–35.

Hailong Jin, Chengjiang Li, Jing Zhang, Lei Hou, Juanzi Li, and Peng Zhang. 2018. Xlore2: Large-scale cross-lingual knowledge graph construction and application. *Data Intelligence*, 1(1):77–98.

Katy Jordan. 2015. Massive open online course completion rates revisited: Assessment, length and attrition. *The International Review of Research in Open and Distributed Learning*, 16(3).

Judy Kay and Sam Holden. 2002. Automatic extraction of ontologies from teaching document metadata. In *International Conference on Computers in Education, 2002. Proceedings.*, pages 1555–1556. IEEE.

Kristian Kiili. 2005. Digital game-based learning: Towards an experiential gaming model. *The Internet and higher education*, 8(1):13–24.

Shen Li, Zhe Zhao, Renfen Hu, Wensi Li, Tao Liu, and Xiaoyong Du. 2018. Analogical reasoning on chinese morphological and semantic relations. *arXiv preprint arXiv:1805.06504*.

Xiao-Li Li, Lei Zhang, Bing Liu, and See-Kiong Ng. 2010. Distributional similarity vs. pu learning for entity set expansion. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 359–364. Association for Computational Linguistics.

Chen Liang, Zhaohui Wu, Wenyi Huang, and C Lee Giles. 2015. Measuring prerequisite relations among concepts. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1668–1674.

Jonathan Mamou, Oren Pereg, Moshe Wasserblat, Alon Eirew, Yael Green, Shira Guskin, Peter Izsak, and Daniel Korat. 2018. Term set expansion based nlp architect by intel ai lab. *arXiv preprint arXiv:1808.08953*.

Eric Margolis and Stephen Laurence. 1999. *Concepts: core readings*. Mit Press.

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

Liangming Pan, Xiaochen Wang, Chengjiang Li, Juanzi Li, and Jie Tang. 2017b. Course concept extraction in moocs via embedding-based graph propagation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 875–884.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Luis Sarmento, Valentin Jijkuon, Maarten De Rijke, and Eugenio Oliveira. 2007. More like these: growing entity classes from seeds. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 959–962. ACM.

Jiaming Shen, Zeqiu Wu, Dongming Lei, Jingbo Shang, Xiang Ren, and Jiawei Han. 2017. Setexpan: Corpus-based set expansion via context feature selection and rank ensemble. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 288–304. Springer.

Partha Pratim Talukdar and William W Cohen. 2012. Crowdsourced comprehension: predicting prerequisite structure in wikipedia. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 307–315. Association for Computational Linguistics.

Monique Threatt. 2014. 7 things you should know about games and learning.

Thierry Volery and Deborah Lord. 2000. Critical success factors in online education. *International journal of educational management*, 14(5):216–223.

Chi Wang, Kaushik Chakrabarti, Yeye He, Kris Ganjam, Zhimin Chen, and Philip A Bernstein. 2015. Concept expansion using web tables. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1198–1208. International World Wide Web Conferences Steering Committee.

Richard C Wang and William W Cohen. 2007. Language-independent set expansion of named entities using the web. In *Seventh IEEE international conference on data mining (ICDM 2007)*, pages 342–350. IEEE.

Richard C Wang and William W Cohen. 2008. Iterative set expansion of named entities using the web. In *2008 eighth IEEE international conference on data mining*, pages 1091–1096. IEEE.

Shuting Wang, Alexander Ororbia, Zhaohui Wu, Kyle Williams, Chen Liang, Bart Pursel, and C Lee Giles. 2016. Using prerequisites to extract concept maps fromtextbooks. In *Proceedings of the 25th acm international on conference on information and knowledge management*, pages 317–326. ACM.

Yasheng Wang, Yang Zhang, and Bing Liu. 2017. Sentiment lexicon expansion based on neural pu learning, double dictionary lookup, and polarity association. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 553–563.

Jingru Yang, Ju Fan, Zhewei Wei, Guoliang Li, Tongyu Liu, and Xiaoyong Du. 2018. Cost-effective data annotation using game-based crowdsourcing. *Proceedings of the VLDB Endowment*, 12(1):57–70.

Mohamed Amir Yosef, Johannes Hoffart, Ilaria Bordino, Marc Spaniol, and Gerhard Weikum. 2011. Aida: An online tool for accurate disambiguation of named entities in text and tables. *Proceedings of the VLDB Endowment*, 4(12):1450–1453.

A Probe into the different performance across datasets

In our overall evaluation part, methods' performance in different datasets is undulating: 1) English datasets are tend to provide better performance. 2) ZH-DSA is a roadblock of methods, i.e., each method meet a decline in ZH-DSA. To explore the cause of these phenomenons, we take a further observation on the situation of datasets in two aspects.

Looseness of Course concepts.

We calculate the average pairwise similarity of each dataset, which reflect the alienation of course concepts. The results are shown in Figure 6. Obviously the datasets in Chinses contains a more loose $\mathcal{M}$ than English datasets. Thus, when expanding concepts in Chinese courses, the new found concepts are easier to be admitted, for the radius of cluster may be too large to intercept low-quality concepts. In another word, semantic drift is more prone to happen in the two Chinese datasets. Therefore, it is necessary to avoid semantic drifts in real concept expansion of MOOCs.

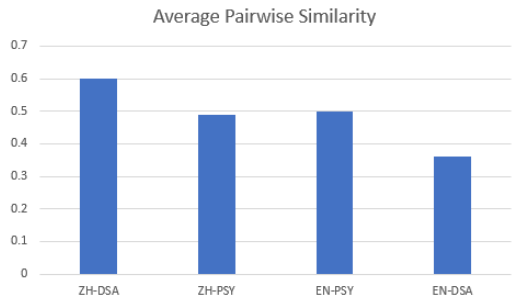
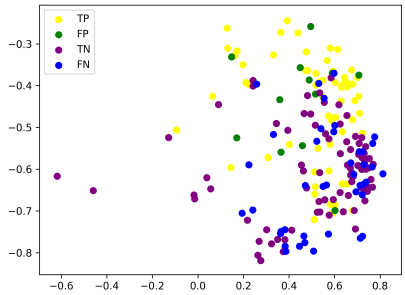
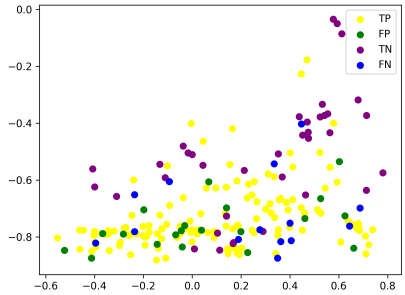


Figure 6: Average pairwise similarity of different datasets.

Specificity of Samples in test set.



(a) Sample distribution of ZH-DSA.



(b) Sample distribution of ZH-PSY.

Figure 7: The sample distribution of two Chinese datasets. TP, FP, TN, FN are respectively *True Positive*, *False Positive*, *True Negative*, *False Negative* samples.

Despite above observation, we still cannot explain the performance decline in ZH-DSA, for ZH-PSY also provide a competitive result. A deeper investigation is formed by finding out the sample distribution of two Chinese classification results. We reduce the feature of all test concepts to a 2-Dimension vector and differentiate their colors according to actual classification results. When comparing the samples in the two test sets, we can obtain two main observations. For one thing ZH-PSY contains more positive samples. For another thing, the positive and negative samples are more blended in ZH-DSA. The fundamental cause of these characteristics may be the nature of the courses. The course of *Data Structure and Algorithm* in Chinese is interdisciplinary of Computer Science and Mathematics while 'Intrduction to Psychology' in Chinese is much more 'pure', only containing one domain. Thus, heterogeneous information is indispensable to some extent, for it can effectively utilize different features to query proper expanded concepts.