Hierarchical Concept Map Generation from Course Data

John Stamper, Bharat Gaind, Karun Thankachan, Huy Nguyen, Steven Moore

Carnegie Mellon University

jstamper@cmu.edu, bgaind@alumni.cmu.edu, kthankac@alumni.cmu.edu, hn1@andrew.cmu.edu, StevenJamesMoore@gmail.com

Abstract

Concept maps are a core feature supporting the design, development, and improvement of online courses and educational technologies. Providing hierarchical ordering of the concepts allows for a more detailed understanding of course content by indicating pre- and post-requisite information. In this research, we implement an end-to-end domain-independent system to generate a concept map from digital texts that needs no additional data augmentation. We extract concepts from digital textbooks on the domains of precalculus, physics, computer networks, and economics. We engineer seven relevant features to identify prerequisite relationships between the concepts. These prerequisites are then used to generate and visualize a hierarchical concept map for each course. Our experiments show that the proposed methodology exceeds the existing baseline performance in existing domains including physics and computer networking, by up to 14.5%. Additionally, human evaluation identified four common errors between the prerequisites found through use of the concept maps. Our findings indicate that our methods, which require minimal data preprocessing, can be used to create more informative concept maps. These concept maps can be leveraged by students, instructors, and course designers to improve the learning process in a variety of domains.

Introduction

Students need prior knowledge for thorough understanding of educational content, which imparts an implicit order of the concepts in the learning process (Chaplot et al. 2016). These concepts or skills are known as knowledge components (KCs), which are formally defined as an acquired unit of cognitive function or structure that can be inferred from performance on a set of related tasks (Koedinger et al. 2012). The KCs represent the knowledge a student needs in order to successfully solve a particular problem. There is a prerequisite relationship between KCs, as a student might need to know certain KCs before they can successfully solve a problem requiring a different KC. For instance, if a student was solving a "multiplication" problem, they might fail because they do not know how to do "addition", which is a prerequisite of the former. This can create problems, as instead of having the student work on problems relating to "multiplication", it would be best to have them first practice and gain an understanding of "addition" (Matsuda et al. 2016).

One form of representing these KCs and their prerequisites is a concept map, which represents domain content and their learning dependencies (Liang et al. 2015). Concept maps have been used to represent the KCs in online courses, such as MOOCs, indicating the prerequisite relationships that exist between them (Watson et al. 2018). These maps can be leveraged by students in browsing course materials, emphasizing important topics in a course and how they are related, to improve students' understanding of the material (ALSaad et al. 2018). Typically a domain expert or experienced instructor manually constructs a concept map (Stamper et al. 2010). However, this traditional way is often time-consuming and not scalable to a large number of concepts or to the amount of online courses being generated (Chen et al. 2018). Additionally, the construction of concept maps this way might be error-prone due to a reliance on expert knowledge to determine the relationship between concepts and their prerequisites from expert blindspot (Nathan et al. 2001). This indicates that an expert's cognition and learner's cognition of certain concepts may not align, causing the manually created concept maps to be suboptimal or misleading for learners.

To assist in avoiding these scaling and expert blindspot problems, recent research has leveraged machine learning methods to automatically identify prerequisites and generate concept maps (Yu et al. 2020). Concept maps have been automatically generated using datasets of educational content gathered from online courses such Coursera, Udacity, and Wikipedia (Liu et al. 2020). However, much of this work is done in the vein of advancing the machine learning methods used in the process, rather than focusing on how the generated concept maps can be effectively leveraged towards educational outcomes. Oftentimes these methods use a variety of datasources in addition to the instructional content, such as student performance data on the material (Chaplot et al. 2016). This can make these methods difficult to use for new courses or courses that do not have thousands of students working through them. While concept maps can be leveraged by recommendation systems and learning analytic systems, there needs to be a way to assess the quality. To evaluate the effectiveness of concept maps as a learning tool, they should be both accurate and easily understandable by both students and instructors when identifying areas of the course to focus on (Gorman & Heinze-Fry 2015).

This paper introduces and analyzes an end-to-end novel system for automatically generating concept hierarchies from digital textbooks irrespective of their domain. We start with concept extraction using a textbook's index and then engineer features from it to train a machine learning model to predict prerequisite relationships between concept pairs. Following this, a hierarchical concept map is generated using the predicted relations between concepts found in textbooks. Finally, we visualize these concepts and have two expert instructors evaluate the results for completeness of topic coverage and usefulness for course design. The primary contributions of this paper are the new features we propose and utilize to generate the prerequisites and concept maps, expanding on previous results (Alzetta et al. 2019). We demonstrate how this approach beats the existing baselines and provide our open-source code for others to use. Finally, we provide generalizable errors identified by human evaluation of the concept maps, with suggestions for overcoming them in future work.

Background

The proposed system is an end-to-end solution to generate concept maps from textbooks belonging to a variety of domains. Prior work in the field essentially splits the process of creating a concept map into extracting concepts and identifying which concepts are prerequisites of another concept. Much of this work exists for the advancement of machine learning techniques, while our system is focused on the educational value of assisting instructors and students.

One of the first steps for generating concept maps and indicating the prerequisite relationships between them is identifying the relevant concepts themselves, which is known as concept extraction (Atapattu et al. 2012). The granularity of these concepts can range from a fine-grained KC (Koedinger et al. 2013) to a high-level concept, such as a chapter found in a textbook, depending on the project (Lu et al. 2019). Traditionally, previous research identifies concepts for extraction at a granularity between these two levels, such as the work of Chen et al. (2018), which derived concepts based on the national standards for mathematics in primary and secondary schools. To perform the concept extraction, Chau et al. (2021b) leveraged a supervised feature-based machine learning model that was built upon several popular methods centered around extracting keyphrases from text. They compared their model's output to human experts and found that the extracted concepts matched with a majority of the expert annotations, even outperforming several baselines for concept extraction. A related study took a different approach and leveraged humans to annotate the concepts found on different web pages related to real estate through different self-review and peer-review processes (Chau et al. 2020). They found that the human annotations served as a quality evaluation of different machine learning models and depending on the model, the human annotations may cause the evaluation metrics to fluctuate.

With the potential concepts identified, the next step in the process involves identifying the prerequisite relationships between them. A concept C1 is generally called a prerequisite to another concept C2 if the knowledge of C1 is necessary to understand C2 (Johnson-Laird 1980). Previous research has leveraged popular machine learning models, such as neural networks and random forest, in order to infer the prerequisite relationships between concepts (Roy et al. 2019). Oftentimes the methods that utilize such models require ample training data, in the form of labeled prerequisites or student data that can be used to infer which concepts might be reliant on others (Li et al. 2020). One method for prerequisite identification by Wang et al. (2016) devises a set of features extracted from both textbooks and Wikipedia, then uses a Support Vector Machine (SVM) to identify the prerequisites. With this approach, they were able to achieve moderately accurate results in the domains of economics, computer networks and physics. Related work by Pan et al. (2017) also proposed a set of features based on syntactic relationships and semantic relatedness of concepts, extracted from course textbooks and videos, to discriminate prerequisite relationships. They carried out a detailed comparison using these features with several machine learning models, including both SVM and random forest. Their results showed that the random forest model was the most effective in prerequisite identification, which informed our choice of using random forest for our prerequisite identification approach.

The extracted concepts and prerequisite relationships identified between them can be visualized in the form of a concept map, which is a graphical representation where concepts are nodes and directed edges between them represent the prerequisite relationship (Mohamad Rasli et al. 2014). Typically visualization software is used to create the concept maps, so they can be evaluated by human experts (Cañas et al. 2005). If human experts are creating them and taking part in the entire process of concept extraction, prerequisite identification, and concept map visualization, then they might do so using tools of their choice, often drawing them by hand (Schwendimann 2016). Much of the prior work on concept extraction, prerequisite learning, and concept maps fails to evaluate the practicality and usefulness of the output concept map (Roy et al. 2019). Concept maps can be fed into educational technologies, to help power adaptive learning and model student learning (Huang et al. 2016). Student models can assist with adaptation and personalization in student learning (Stamper et al. 2007). However, concept maps can also assist instructors in ensuring their course covers essential material or by ensuring that their students have sufficient prior knowledge before exposing them to a new concept (Zeng et al. 2009). Similarly for students, if they are to use a concept map to identify key topics covered in a course or to help guide their plan of study, the concept map needs to be parsable by the student.

Methodology

To build a concept map, we divide our system into several components as shown in the architecture diagram in Figure 1. The first phase is to extract concepts from a given digital textbook, containing instructional text in a single-column format. For each of the extracted concepts, we then extract features that will help us determine if a given concept pair - (C1, C2), if C1 is a prerequisite for C2. Once the features are extracted, we use a machine learning model for identifying prerequisites and then use them to generate a concept map. The following sections provide the technical details behind the proposed system and the dataset used to analyze its performance.

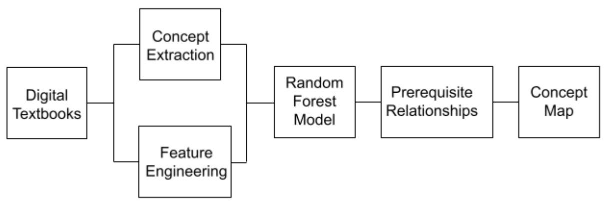
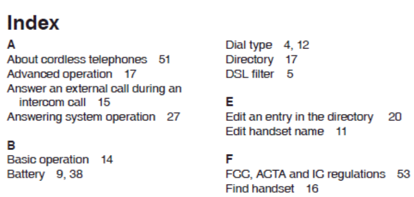


Figure 1: Overview of the end-to-end system architecture

While previous work has focused on concept extraction in textbooks through using section headers, chapter titles, and other book features (Liang et al. 2019), the present work makes use of the index found at the end of the digital textbooks. The index in such textbooks is specifically designed to list the key terms and topics covered throughout the book, along with a page number on where they can be found. Therefore, we parse each word or phrase found in the index of each digital textbook, remove the page numbers that follow it, and treat it as a concept for the purpose of this analysis. Additionally, we pruned concepts that did not occur more than three times throughout the given textbook. An example of the index found in one of the textbooks used in this study can be found in Figure 2.


Index

A
About cordless telephones 51
Advanced operation 17
Answer an external call during an intercom call 15
Answering system operation 27

B
Basic operation 14
Battery 9, 38

Dial type 4, 12
Directory 17
DSL filter 5

E
Edit an entry in the directory 20
Edit handset name 11

F
FCC, ACTA and IC regulations 53
Find handset 16

Figure 2: Sample index from a textbook, where each phrase on a line is treated as concept

Building upon an analysis of related features identified in prior work (Pan et al. 2017)(Wang et al. 2016), we identify the following seven features for pairs of concepts extracted from the digital textbooks:

- **Average Chapter Reference Distance:** This feature represents the average number of textbook chapters found between two concepts. Wang et al. (2016) suggested that a reference distance can indicate how often the two concepts occur together. The intuition for this feature is that the concepts that occur in earlier chapters are likely to be prerequisites of material that occurs in the latter chapters of the textbooks.
- **Average Page Reference Distance:** The average number of chapter pages apart two concepts in a concept pair are, if they occur in the same chapter in a textbook. This feature is constricted by the maximum number of pages found within a chapter, but follows similar logic to the first feature.
- **Average Sentence Reference Distance:** The average number of sentences apart two concepts in a concept pair are, if they occur in the same page in a textbook.
- **Average Position Distance:** The average number of words apart two concepts in a concept pair are, if they occur in the same line in a textbook.
- **Complexity Distance:** Two concepts with a prerequisite relationship typically have a difference in their complexity level, as the prerequisite concept is often basic and the other concept is more advanced. Pan et al. (2017) indicated that the complexity level of a concept is implicit in its distribution across a course or textbook. The more often a concept occurs in the content, the higher likelihood it is for it to be more basic of a concept. Specifically, we can calculate the difference between the number of times two concepts are mentioned in a textbook using this formula:

  Complexity Distance(C1, C2) = Count(C2) - Count(C1)

  Based on the formula above, if the complexity distance value is positive, then it is more likely that C2 is a prerequisite of C1, rather than vice versa.
- **Chapter Concentration:** The F_1 score of the number of chapters each concept occurs in. The intuition is that if a concept occurs in many chapters, it is more likely to be a generic and basic concept in the educational resource, rather than being a strict prerequisite. Note the similarity to the fifth feature, however this one helps to downplay concepts that occur constantly, such as “mean” in a statistics textbook as being assigned a prerequisite to everything. This feature will help identify it as a basic concept, without necessarily being a prerequisite, using the following formula:

  Chapter Concentration=(Ch(C1) Ch(C2))/(Ch(C1) +Ch(C2))*

  where Ch(C1) and Ch(C2) represent the number of chapters concept C1 and C2 respectively occur in.
- **Distributional Asymmetry Distance (DAD):** This feature represents a measure of how many times concept C1 occurs before concept C2 (Pan et al. 2017). The intuition behind this is, if C1 is a prerequisite and occurs before C2, then C1 will be talked about before C2 in the education resources. The following formula, along with the parameters are:

  *DAD = alpha * NCh(C1, C2) + beta * NPage(C1, C2) + gamma * NSent(C1, C2)*

  where, alpha, beta, gamma are hyperparameters, NCh(C1, C2) is number of times C1 occurs in a chapter prior to the chapter containing C2, NPage(C1, C2) is the number of times C1 occurs in a page prior to the page containing C2 if they are in the same chapter, NSent(C1, C2) is the number of times C1 occurs in a sentence prior to the sentence containing C2 if they are on the same page.

These seven features are extracted for each concept pair and fed into a random forest model. Random forest is a supervised learning method that uses a collection of decision trees to predict if the concept pair has a prerequisite relationship or not (Lu et al. 2019). Random forest models can implicitly deal with the class imbalance (i.e. the dataset has more labeled prerequisites pairs that non-prerequisite pairs) in the dataset that we will be using, and was proven to be the most effective for prerequisite learning by Pan et al. (2017) when compared to other traditional machine learning models of SVM, naive bayes, and logistic regression. We used the seven identified features and all of the concepts extracted from the textbook's index to feed into our random forest model. The model then outputs a classification label, corresponding to one or zero, indicating if concept C1 was a prerequisite of concept C2.

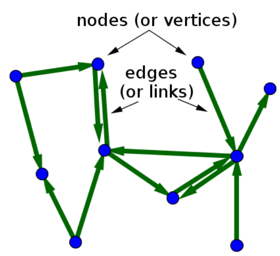


Figure 3: Visualization of a directed graph that represents a concept map, where a node corresponds to a concept and an edge points from prerequisite to the dependent concept

Concept map generation was completed using the output of the random forest model. It contains the concept pairs and their classification as prerequisite or not, which we use to generate a visual representation in the form of a concept map. The concept map takes the form of a directed graph, as shown in Figure 3. A node in the directed graph is a concept and an edge from concept C1 to concept C2 indicates that C1 is a prerequisite for C2. The concept pairs with the predicted label, which indicate whether it is a prerequisite or not, are stored as a CSV file and fed into the visualization software known as Flourish (Liang et al. 2016). This was done to visually represent the concept map for each textbook as a directed graph. Flourish is an online platform to create interactive directed graphs, it takes as input (C1, C2, classification label), where C1 and C2 are the concept name and the classification label indicates if there should be an edge from C1 to C2 that indicates C1 is a prerequisite of C2.

Experimentation

The dataset used for assessing the concept hierarchy generation system was the CMEB dataset (Wang et al. 2016). This dataset contains four textbooks across the domains of Calculus, Economics, Physics and Computer Networks. Each textbook has associated with it a list of manually identified concepts. It also has a list of concept pairs of the form (C1, C2) and a class label indicating if C1 is a prerequisite of C2 (class=1) or not (class=0). While the dataset is relatively small for a machine learning-based system, it is one of the few datasets that provides concepts and prerequisites across multiple domains. This feature was crucial to verify the accuracy and usability of our system for concept map generation and ensuring that it could work for multiple domains with no external data besides the textbook. In the CMEB dataset, shown in Table 1, we see that precalculus has the largest textbook and hence the largest number of concepts. However, we also see that physics is the smallest textbook with the least number of concepts, but it is the most well connected with the highest number of prerequisites identified.

Domain	Unique Concepts	# Prerequisites	# Non Prerequisites
Precalculus	226	418	753
Economics	169	242	635
Physics	160	420	1097
Computer Networks	200	416	1082

Table 1: CMEB dataset distribution spanning four domains

The performance of our system will be analyzed using precision, recall, and F_1 score. Precision will let us know out of all the data we identify as concept/prerequisite pairs, how many are actually concept/prerequisite pairs. Recall helps us identify out of all the actual valid concepts/prerequisite pairs, how many we were able to identify as valid pairs. F_1 score helps us to evaluate the model by giving equal weight to both precision and recall. The performance of the prerequisite learning module is also compared against a baseline model proposed by Wang et al. (2016).

Our heuristic approach of concept extraction from the index was evaluated using precision and recall against the concepts manually extracted by domain experts in the CMEB dataset. In this CMEB dataset, several experts went through the textbooks, and with the help of a knowledge graph constructed from Wikipedia, identified concepts found within the text. The features are then created between all the pairs of concepts. If the concept pair is present in the CMEB dataset with a label 1, then this means C1 is a prerequisite for C2. If a concept pair is present with label 0, then C1 is not a prerequisite for C2. Thus, a dataset with the concept pair, features, and label is created and passed to our prerequisite learning methods which utilizes the random forest model.

In the prerequisite learning model, the dataset is stratified and split based on the label value in a 60-20-20 fashion to create the training, validation, and test dataset.

We utilized the validation dataset to tune the hyperparameters of the random forest model and set the n_estimators as 200. We report our performance on the test dataset in the results and discussion section.

To further evaluate the results of our prerequisite learning model, we also utilized human evaluation. Two researchers with seven or more years of experience in teaching computer science were provided with the visualization for Networks. They were asked to parse through the concept map visualization and identify errors such as unwanted prerequisites, missing prerequisites, and anything else they deemed an error or that may hinder their use of the concept map. A version of the computer networking concept map visualization is made available here: https://public.flourish.studio/visualisation/8954167/.

Results and Discussion

Concept Extraction Validation

Table 2 contains the precision and recall scores of our concept extraction heuristic method, utilizing the index of the textbooks, on the CMEB dataset. We see that precision is low and recall is high across all four domains. The precision is low because every term in the textbooks' index, assuming it occurs more than three times throughout the text, is considered a concept, compared to the CMEB dataset where humans identified concepts based on their knowledge. Also, there are a large number of false positives partially because the index contains non-concepts such as author name, software name, etc. The other reason for low precision is because the index does contain valid concepts, but the manual identification of concepts in the CMEB dataset was not comprehensive enough, as they did not exhaust all of the concepts covered in each textbook. As a result, there are instances where the model accurately detects a concept, but it is missing from the CMEB dataset. The recall is high because a majority of the 200 or so concepts the human evaluators identified in the CMEB dataset are also terms in the index, so our concept extraction methods identified them as well. Ultimately, while this method using the index provides an easy way to extract concepts, the low precision values are not ideal for usage. Note, since the Physics textbook associated with the dataset did not have an index we were unable to run our extraction method on this textbook.

Textbook	Precision	Recall
Computer Networking	0.08	0.84
Precalculus	0.11	0.74
Economics	0.07	0.78

Table 2: Precision and recall of the proposed heuristic method for concept extraction across the four domains in the CMEB dataset

Prerequisite Learning Validation

The precision, recall and F1 score of the random forest model on our test set are reported in Table 3. We passed the concept pairs from the CMEB dataset into our random forest model to compare it to existing approaches using the same concept pairs. We see that our proposed approach beat the baseline approach proposed by Wang et al. (2016). Specifically, our model's results indicate that our methodology exceeds the baseline's performance by 4.5% for the Computer Networking textbook and by 14.5% for the Physics textbook. We also observe that we were able to achieve the best performance on the Physics textbook which also had the largest number of prerequisites, even though it had the least number of unique concepts in the CMEB dataset.

Our model only beat the baseline on two of the four courses, as it did not exceed performance on the Precalculus and Economics courses. Notably, these two courses have a much higher count of non-prerequisite concepts in the CMEB dataset. Our model might be more prone to false positives and thus could be misclassifying the high number of non-prerequisites, causing its average performance.

Textbook	Precision	Recall	F1-Score	F1-Score (Baseline)
Computer Networking	0.75	0.60	0.66	0.63
Physics	0.78	0.81	0.79	0.69
Precalculus	0.71	0.68	0.66	0.66
Economics	0.73	0.68	0.7	0.7

Table 3: Precision/Recall/F_1 Score of the random forest model for prerequisite learning across the four domains.

Analyzing the random forest model for prerequisite learning, we identify the importance of the seven features we utilized, shown in Figure 4. Based on this feature importance, it appears that syntactic features, such as the positioning of concepts in a concept pair across the textbook, are a good indicator of whether two concepts are prerequisites of one another. Some of the implications of this are that the syntactic features such as positioning of concepts in a concept pair across the textbook is a good indicator of whether two concepts are prerequisites of one another.

When we perform error analysis on the average page distance reference feature, ranked a close second in importance, we see that the number of prerequisites and non-prerequisites have a wide range for positional features. This indicates that prerequisite and non-prerequisite concept pairs can be far away or close together, as shown by the green histograms. However, the errors are made in much smaller ranges, meaning discrimination between prerequisites and non-prerequisites begins to fail when concept pairs are located closer to one another. This indicates that as concepts are syntactically located close in the textbook, the feature set we have currently becomes less discriminative. We aim to extend the current feature set in future work to include features that can discriminate between prerequisites and non-prerequisites when concept pairs are positionally nearby each other.

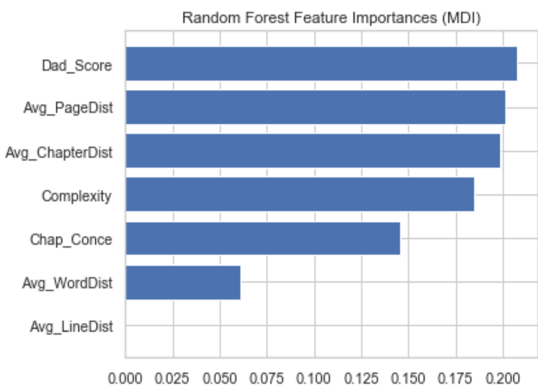


Figure 4: The importance of features used in the random forest model used for prerequisite identification

Human Evaluation

We developed concept maps from the concept pairs with the predicted labels, which are available for public viewing. In Figure 5, we see the portion of the computer networks concept map represented as a directed graph. A node in the directed graph is a concept and an edge from concept C1 to concept C2 indicates that C1 is a prerequisite for C2. e.g. the edge from Protocol Stack to Transmission Control Protocol (TCP) indicates that Protocol stack is a prerequisite to TCP. The color of the node has no meaning and is just used only to tie the concept label to its node in the visualization and help users discern different nodes. Based on the manual evaluation of the interactive concept map for Computer Networks by two experts, they identified four common errors that we aim to address as part of our future work.

- **Absent Error:** The prerequisite for the concept is missing. In Figure 6(a) we see the "TCP", a required prerequisite for "HTTP persistent connection" is missing.
- **Extraneous Error:** An additional prerequisite that is not required is present. In Figure 6(c) we see that "File Transfer protocol" has an unnecessary prerequisite "BitTorrent", which is related, but not a prerequisite.
- **Transitive Error:** The concept map indicates C1 is a prerequisite for C2 and C3. However, the actual mapping should be that C1 is a prerequisite for C2 and C2 is a prerequisite for C3. In Figure 6(b) we see that "2G" is a prerequisite for "3G" and "4G". Ideally, "2G" should be a prerequisite for "3G" and it would inherently be a prerequisite for "4G", as "3G" is a prerequisite for "4G".
- **Direction Error:** The graph shows C1 is a prerequisite for C2, but it should be the other way around. In Figure 6(c) we see that "BitTorrent" is a prerequisite for "Peer-to-peer file sharing", however it should be the other way around.

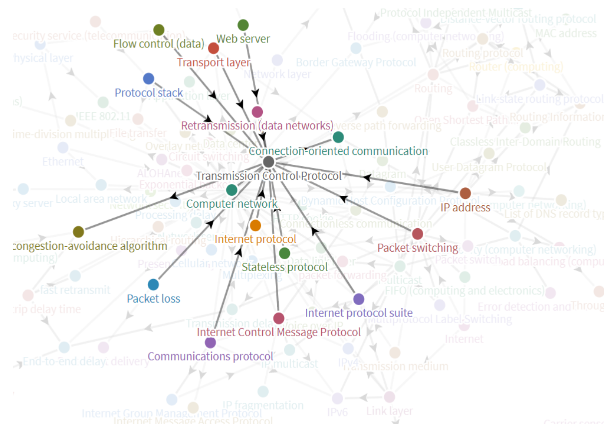
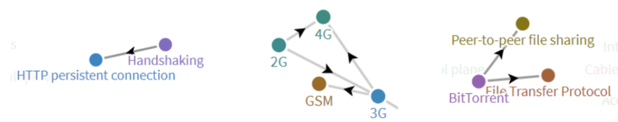


Figure 5: Visualization of concept hierarchy extracted from Computer Networks as an interactive directed graph



(a) (b) (c)

Figure 6: Common error types that researchers identified in the concept hierarchy visualization

Reducing these four common errors is our plan for future work. Minimizing the *Absent* and *Direction Errors* would enable the system to be used by educators more effectively in the course design process. In particular, concepts being absent from the concept map might be the most challenging issues for users. Other problems, such as the arrows being reversed, indicating the prerequisite relationship is backwards, are easier to remedy as they do not require the generation of new content or concepts.

Conclusion

In this paper, we propose a system that assists in the generation of interactive concept maps for a variety of domains. These concept maps can be leveraged by students and instructors alike to identify gaps in their own knowledge and assist in designing comprehensive courses. The implementation of our system is an end-to-end domain-independent system that extracts concepts, engineers features from textbooks, identifies relationships between concepts, and visualizes them as a concept map. Experiments show that our methodology exceeds the previous baseline performance by 4.5% for the "Computer Networking" textbook and by 14.5% for the "Physics" textbook. Human evaluation of the output concept maps also identified four errors that are likely to occur in similar prerequisite and concept map work. While our concept extract method had precision problems, we observed that a significant number of concepts in the index of a textbook are not actually concepts (eg. company names), which motivates future work towards a generic methodology for concept extraction that can be done without excessive preprocessing. As prerequisite learning research continues, we suggest keeping in mind who might be able to make practical use of these visualizations.

References

ALSaad, F., Boughoula, A., Geigle, C., Sundaram, H. and Zhai, C. 2018. Mining MOOC Lecture Transcripts to Construct Concept Dependency Graphs. *International Educational Data Mining Society*. (2018).

Alzetta, C., Miaschi, A., Adorni, G., Dell'Orletta, F., Koceva, F., Passalacqua, S. and Torre, I. 2019. Prerequisite or Not Prerequisite? That's the Problem! An NLP-based Approach for Concept Prerequisite Learning. *CLiC-it* (2019).

Atapattu, T., Falkner, K. and Falkner, N. 2012. Automated extraction of semantic concepts from semi-structured data: Supporting computer-based education through the analysis of lecture notes. *International Conference on Database and Expert Systems Applications (2012)*, 161–175.

Cañas, A.J., Carff, R., Hill, G., Carvalho, M., Arguedas, M., Eskridge, T.C., Lott, J. and Carvajal, R. 2005. Concept maps: Integrating knowledge and information visualization. *Knowledge and information visualization*. Springer. 205–219.

Chaplot, D.S., Yang, Y., Carbonell, J. and Koedinger, K.R. 2016. Data-Driven Automated Induction of Prerequisite Structure Graphs. *International Educational Data Mining Society*. (2016).

Chau, H., Balaneshin, S., Liu, K. and Linda, O. 2020. Understanding the tradeoff between cost and quality of expert annotations for keyphrase extraction. *Proceedings of the 14th Linguistic Annotation Workshop (2020)*, 74–86.

Chau, H., Labutov, I., Thaker, K., He, D. and Brusilovsky, P. 2021. Automatic concept extraction for domain and student modeling in adaptive textbooks. International Journal of Artificial Intelligence in Education. 31, 4 (2021), 820–846.

Chen, P., Lu, Y., Zheng, V.W., Chen, X. and Yang, B. 2018. Knowedu: A system to construct knowledge graph for education. Ieee Access. 6, (2018), 31553–31563.

Gorman, J. and Heinze-Fry, J. 2015. Conceptual Mapping Facilitates Coherence and Critical Thinking in the Science Education System. STEM Education: Concepts, Methodologies, Tools, and Applications. *IGI Global*. 1227–1258.

Huang, X., Yang, K. and Lawrence, V.B. 2015. An efficient data mining approach to concept map generation for adaptive learning. Industrial conference on data mining (2015), 247–260.

Johnson-Laird, P.N. 1980. Mental models in cognitive science. *Cognitive science*. 4, 1 (1980), 71–115.

Koedinger, K.R., Corbett, A.T. and Perfetti, C. 2012. The Knowledge-Learning-Instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive science*. 36, 5 (2012), 757–798.

Koedinger, K. R., Stamper, J. C., Leber, B., & Skogsholm, A. (2013). LearnLab's DataShop: A Data Repository and Analytics Tool Set for Cognitive Science. *Topics in Cognitive Science*, 5(3).

Li, I., Fabbri, A., Hingmire, S. and Radev, D. 2020. R-vgae: Relational-variational graph autoencoder for unsupervised prerequisite chain learning. *arXiv preprint* arXiv:2004.10610. (2020).

Liang, C., Wu, Z., Huang, W. and Giles, C.L. 2015. Measuring prerequisite relations among concepts. *Proceedings of the 2015 conference on empirical methods in natural language processing (2015)*, 1668–1674.

Liang, C., Ye, J., Wang, S., Pursel, B. and Giles, C.L. 2018. Investigating active learning for concept prerequisite learning. *Proceedings of the AAAI Conference on Artificial Intelligence (2018)*.

Liu, L., Lin, F., Wang, B., Li, K., Xiao, M., Xiahou, J. and Wu, P. 2020. Prerequisite relation learning for course concepts based on hyperbolic deep representation. *IEEE Access*. 8, (2020), 49079–49089.

Lu, W., Zhou, Y., Yu, J. and Jia, C. 2019. Concept extraction and prerequisite relation learning from educational data. *Proceedings of the AAAI Conference on Artificial Intelligence (2019)*, 9678–9685.

Matsuda, N., Chandrasekaran, S., & Stamper, J. C. (2016). How quickly can wheel spinning be detected?. In *EDM* (pp. 607-608).

Mohamad Rasli, R., Ahmad, F. and Kamaruddin, S.S. 2014. A comparative study of conceptual graph and concept map. *ARPN Journal of Engineering and Applied Sciences*. 9, 9 (2014), 1442–1446.

Nathan, M.J., Koedinger, K.R. and Alibali, M.W. 2001. Expert blind spot: When content knowledge eclipses pedagogical content knowledge. *Proceedings of the third international conference on cognitive science (2001)*.

Pan, L., Li, C., Li, J. and Tang, J. 2017. Prerequisite relation learning for concepts in moocs. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (2017)*, 1447–1456.

Roy, S., Madhyastha, M., Lawrence, S. and Rajan, V. 2019. Inferring concept prerequisite relations from online educational resources. *Proceedings of the AAAI Conference on Artificial Intelligence (2019)*, 9589–9594.

Sahrawat, D., Mahata, D., Kulkarni, M., Zhang, H., Gosangi, R., Stent, A., Sharma, A., Kumar, Y., Shah, R.R. and Zimmermann, R. 2019. Keyphrase extraction from scholarly articles as sequence labeling using contextualized embeddings. *arXiv preprint arXiv:1910.08840*. (2019).

Schwendimann, B.A. 2016. Comparing expert and novice concept map construction through a talk-aloud protocol. International Conference on Concept Mapping (2016), 16–28.

Stamper, J. C., Barnes, T., & Croy, M. (2007, July). Extracting student models for intelligent tutoring systems. In *AAAI* (pp. 1900-1901).

Stamper, J., Barnes, T., & Croy, M. (2010). Enhancing the automatic generation of hints with expert seeding. In *Intl. Conference on Intelligent Tutoring Systems* (pp. 31-40). Springer.

Wang, S., Ororbia, A., Wu, Z., Williams, K., Liang, C., Pursel, B. and Giles, C.L. 2016. Using prerequisites to extract concept maps from textbooks. *Proceedings of the 25th ACM international conference on information and knowledge management* (2016), 317–326.

Watson, M.K., Barrella, E. and Pelkey, J. 2018. Assessment of conceptual knowledge using a component-based concept map scoring program. *The International journal of engineering education*. 34, 3 (2018), 1025–1037.

Yu, H., Li, H., Mao, D. and Cai, Q. 2020. A relationship extraction method for domain knowledge graph construction. World Wide Web. 23, 2 (2020), 735–753.

Zeng, Q., Zhao, Z. and Liang, Y. 2009. Course ontology-based user's knowledge requirement acquisition from behaviors within e-learning systems. *Computers & Education*. 53, 3 (2009), 809–818.