PRET: Prerequisite-Enriched Terminology. A Case Study on Educational Texts

Chiara Alzetta, Forsina Koceva, Samuele Passalacqua, Ilaria Torre, Giovanni Adorni
DIBRIS, University of Genoa (Italy)
{chiara.alzetta,frosina.koceva}@edu.unige.it,
samuele.passalacqua@dibris.unige.it,
{ilaria.torre,adorni}@unige.it

Abstract

English. In this paper we present PRET, a gold dataset annotated for prerequisite relations between educational concepts extracted from a computer science textbook, and we describe the language and domain independent approach for the creation of the resource. Additionally, we have created an annotation tool to support, validate and analyze the annotation.

Italiano. *In questo articolo presentiamo PRET, un dataset annotato manualmente rispetto alla relazione di prerequisito fra concetti estratti da un manuale di informatica, e descriviamo la metodologia, indipendente da lingua e dominio, usata per la creazione della risorsa. Per favorire l'annotazione, abbiamo creato uno strumento per il supporto, la validazione e l'analisi dell'annotazione.*

1 Introduction

Educational Concept Maps (ECM) are acyclic graphs which formally represent a domain's knowledge and make explicit the pedagogical dependency relations between concepts (Adorni and Koceva, 2016). A concept, in an ECM, is an atomic piece of knowledge of the subject domain. From a pedagogical point of view, the most important dependency relation between concepts is the prerequisite relation, that explicits which concepts a student has to learn before moving to the next. Several approaches have been proposed to extract prerequisite relations from various educational sources (Vuong et al., 2011; Yang et al., 2015; Gordon et al., 2016; Wang et al., 2016; Liang et al., 2017; Liang et al., 2018; Adorni et al., 2018). Textbooks in particular are a valuable resource for this task since they are designed to support the learning process respecting the prerequisite relation.

In the literature, the evaluation of the extracted prerequisite relations is usually performed through comparison with a gold standard produced by human subjects that annotate relations between concepts (see, among the others, (Talukdar and Cohen, 2012; Liang et al., 2015; Fabbri et al., 2018)). However, most of the evaluations lack a systematic approach or simply lack the details that allow them to be repeated. In this paper, we present our experience in building PRET (Prerequisite-Enriched Terminology), a gold dataset annotated with the prerequisite relation between pairs of concepts. The issues emerged with PRET led us to define a methodology and a tool for manual prerequisite annotation. The goal of the tool is to support the creation of gold datasets for validating automatic extraction of prerequisites. Both the PRET dataset and the tool are available online[1].

PRET was constructed in two main steps: first we exploited computational linguistics methods to extract relevant terms from a textbook[2], then we asked humans to manually identify and annotate the prerequisite relations between educational concepts. Since the terminology creation step was extensively described in Adorni et al. (2018), this paper mainly focuses on the annotation phase.

The annotation task consists in making explicit the prerequisite relations between two distinct concepts if the relation is somehow inferable from the text in question. We represent a concept as a domain–specific term denoting domain entities expressed by either single nominal terms (e.g. *internet, network, software*) or complex nominal structures with modifiers (e.g. *malicious software, trojan horse, HyperText Document*). Figure 1 shows

[1] http://teldh.dibris.unige.it/pret

[2] For the annotation we used chapter 4 of the computer science textbook "**Computer Science: An Overview**" (Brookshear and Brylow, 2015).

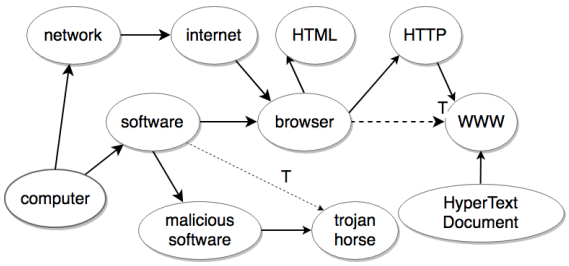

Figure 1: Sample of PRET represented as an ECM.

a sample of the ECM resulting from PRET. According to PRET dataset, an example of prerequisite relation is *network is a prerequisite of internet*, since a student has to know *network* before learning *internet*.

The paper is organized as follows. The related work pertaining to the proposed method is discussed in Section 2. Section 3 describes the methodology used for the creation of the PRET dataset and Section 4 presents the characteristics of the obtained gold dataset and the agreement computed for each pair of annotators together with other statistics about the data. Section 5 describes the main features of the annotation tool we designed. Section 6 concludes the paper.

2 Related Work

Automatic prerequisite identification is a task that gained growing interest in recent years, especially among scholars interested in automatic synthesis of study plans (Gasparetti et al., 2015; Yang et al., 2015; Agrawal et al., 2016; Alsaad et al., 2018). When applying automatic prerequisite extraction methods, a baseline for evaluation is needed. Despite being time consuming, creating manually annotated datasets is more effective and produces gold resources, which are still rare.

To the best of our knowledge, Talukdar and Cohen (2012) is the only case where crowd–sourcing is employed for annotation: they infer prerequisite relationship between concepts by exploiting hyper-links in Wikipedia pages and use crowdsourcing to validate those relations in order to have a gold training dataset for a classifier.

More frequently the annotation of prerequisite relations is performed by domain experts (Liang et al., 2015; Liang et al., 2018; Fabbri et al., 2018) or by students with a certain competence on the domain (Wang et al., 2015; Pan et al., 2017). When annotation is performed by non–experts, agreement usually results very low, so an expert can be consulted (Chaplot et al., 2016; Gordon et al., 2016). Regardless of the annotation methodology, we observe that in the mentioned related works prerequisite relation properties (i.e. irreflexivity, anti–symmetry, etc.) are rarely taken into account in the annotation instructions for annotators. For example, the fact that a concept cannot be annotated as prerequisite of itself is usually left unspecified.

To support the annotation of prerequisites between pairs of concepts, Gordon et al. (2016) developed an interface showing, for each concept of the domain, the list of relevant terms and documents. Although this can be of some support for the annotation providing certain useful information, it cannot be considered an annotation tool itself. According to our knowledge, a tool specifically designed for prerequisite structure annotation which also features agreement metrics is still missing.

3 Annotation Methodology

In Section 4 we will describe the PRET dataset, while here we present the annotation methodology that we used to build PRET and that we refined on the basis of such experience.

Concept identification. Our methodology for prerequisite annotation requires that concepts are extracted from educational materials, that we broadly define Document (D), and provided to annotators. Although we are conscious that a concept, as mental structure, might entail multiple terms, we simplify the problem of concept identification assuming that each relevant term of D represents a concept (Novak and Cañas, 2006). Thus, our list of concepts is a terminology (T) of domain–specific terms (either single or complex nominal structures) ordered according to the first appearance of the terms of T in D and where each concept corresponds to a single term.

For the task of prerequisite annotation, it does not matter if concepts are extracted automatically, manually or semi–automatically. To build PRET, we extracted concepts automatically. To identify our terminology T, we relied on Text-To-Knowledge ($T2K^2$) (Dell'Orletta et al., 2014), a software platform developed at the Institute of Computational Linguistics A. Zampolli of the CNR in Pisa. $T2K^2$ exploits Natural Language Processing, statistical text analysis and machine

learning to extract and organize the domain knowledge from a linguistically annotated text.

We applied T2K^2 to a text of 20,378 tokens distributed over 751 sentences. 185 terms were recognized as concepts of the domain (around 20% of the total number of nouns in the corpus). As expected, the extracted terminology contained both single nominal structures, such as *computer, network* and *software*, and complex nominal structures with modifiers, like *hypertext transfer protocol, world wide web* and *hypertext markup language*. The set of concepts did not go through any post–processing phase.

Annotators selection. The role of annotators is fundamental in order to obtain a gold dataset that represents the pedagogical relations expressed in the educational material. Consequently, the choice of annotators is crucial. As mentioned above, in the literature annotators are often domain experts (Liang et al., 2015; Liang et al., 2018; Fabbri et al., 2018) or students with some knowledge in that domain (Wang et al., 2015; Pan et al., 2017). Based on our experience with different types of annotators, we suggest that annotators should have enough knowledge to understand the content of the educational material. Otherwise, the annotation can be distorted by wrong comprehension of the relations between concepts. On the other hand, experts should not rely on their background knowledge to identify relations, since the goal of the annotation is to capture the knowledge embodied in the educational resource. To build PRET we recruited 6 annotators among professors and PhD students working in fields related to computer science, but eventually 2 of them revealed not to have enough knowledge for the task.

Annotation task. A prerequisite relation between two concepts A and B is defined as a dependency relation which represents what a learner must know/study (concept A), before approaching concept B. Thus, by definition, the prerequisite relation has the following properties: i) asymmetry: if concept A is a prerequisite of concept B, the opposite cannot be true (e.g. *network* is prerequisite of *internet*, so *internet* cannot be prerequisite of *network*); ii) irreflexivity: a concept cannot be prerequisite of itself; iii) transitiveness: if concept A is a prerequisite of concept B, and concept B of concept C, then concept A is also a prerequisite of concept C (e.g. *browser* is prerequisite of *HTTP*, *HTTP* is prerequisite of *WWW*, hence *browser* is prerequisite of *WWW* according to the transitive property).

To keep the annotation as uniform as possible, we provided the annotators with suggestions on how to perform the task together with the book chapter and the terminology extracted from it. Considering the material supplied, we asked annotators to trust the text considering only pairs of distinct concepts of T and annotating the existence of a prerequisite relation between the two concepts only if derivable from D. In our method, annotators should read the text and, for each new concept (i.e. never mentioned in the previous lines), identify all its prerequisites, but, if no prerequisite can be identified, they should not enter any annotation. We also wanted pedagogical relation properties to be preserved, so we asked to respect the irreflexive property not annotating self–prerequisites and to avoid adding transitive relations. Considering the topology of an ECM, we also asked annotators not to enter cycles in the annotation because they represent conceptually wrong relations. To better understand this point, consider the ECM in Figure 1: having a prerequisite relation between *computer* and *network* and between *network* and *internet*, entering a relation where *internet* is prerequisite of *computer* would create a cycle (loop).

The output of the annotation of each annotator is an *enriched terminology*: a set of concepts paired and enhanced with the prerequisite relation. The enriched terminology can be used to create an ECM where each concept is a node and the edges are prerequisite relations identified by humans (see Figure 1).

Annotation validation. Human annotators are not immune from making mistakes and violating the supplied recommendations. The tool we propose addresses this issue by introducing controls to prevent the annotators from making errors (e.g. cycles, reflexive relations, symmetric relations). In the next section we will describe the approach we used to identify some mistakes by using graph analysis algorithms.

Annotators agreement evaluation. Our experience and the literature (Fabbri et al., 2018) show that human judgments about prerequisite identification can vary considerably, even when strict guidelines are provided. This can depend on several factors, including the subjectivity of annotators and the type and complexity of D. Evaluating the annotators' agreement can be useful to assess

Relation Type	Weight	Count (%)
Non–prerequisite	0	33,699 (98.46%)
Prerequisite	All weights	526 (154%)
1 annot.	0.25	293 (55.70%)
2 annot.	0.50	131 (24.90%)
3 annot.	0.75	75 (14.26%)
4 annot.	1	27 (5.13%)
Total number of pairs		34,225

Table 1: Relations and weight distribution in PRET dataset.

if the gold dataset is to be trusted or further annotators are required. Section 4 will describe the measures we used to evaluate annotators' agreement in PRET.

The final combination of the enriched terminologies produced by each annotator is a necessary step to build a gold dataset but, due to space constraints, below we will only present our approach, while a survey on combination metrics is out of the scope of this paper.

4 The PRET Dataset

The PRET gold dataset consists of 34,225 concept pairs obtained by all possible combinations of the elements in the concepts set (excluding self–prerequisites). Pairs vary with respect to the relation weight, computed for each pair by dividing the number of annotators that annotated the pair by the total number of annotators. Only 1.54% (526) of the pairs has a relation weight higher than 0 (i.e. it was annotated as prerequisite by at least one annotator). Details about the distribution of prerequisite relations and respective weights are reported in Table 1.

55.70% (293) of the prerequisite pairs was identified by only one annotator, meaning that it is hard for humans to agree on what a prerequisite is. We further investigate this aspect in section 4.1.

The analysis of the dataset carried out before applying validation checks highlighted some critical issues: some transitive relations were explicitly annotated and some cycles were erroneously added in the dataset, violating the instructions. While cycles are due to distraction, transitive relations are hard to recognize per se, especially when broad terms are involved (e.g. *computer, software, machine*).

In order to study how these issues impact the dataset, each annotation was validated against cycles and transitive relations obtaining 5 dataset variations, in addition to the original annotation. The validation was conducted on the ECM derived from the enriched terminology of each annotator using a graph analysis algorithm. We operated on cycles and transitive relations. In some variations, the latter were added if the pair of concepts in the ECM is connected by a path shorter than a certain threshold, defined by considering the ECM diameter, while cycles were either preserved or removed depending on the variation we wanted to obtain.

Eventually, we obtained the following annotation variations: *no cycles* (removing cycles), *cycles and transitive* (preserving cycles and adding transitive relations), *cycles and non–transitive* (preserving cycles and keeping only direct links), *no cycles and transitive* (removing cycles and adding transitivity) and *no cycles and non–transitive* (removing both cycles and transitivity).

4.1 Annotators Agreement in PRET

Following Artstein and Poesio (2008), we computed the agreement between multiple annotators using Fleiss' k (Fleiss, 1971) and between pairs of annotators using Cohen's k (Cohen, 1960). Using the scale defined by Landis and Koch (1977), Fleiss' k values show *fair agreement*, suggesting that prerequisite annotation is difficult. Similar tasks obtained comparable or lower values, confirming our hypothesis: Gordon et al. (2016) measured the agreement as Pearson Correlation obtaining 36%, while Fabbri et al. (2018) and Chaplot et al. (2016) obtained respectively 30% and 19% of Fleiss' k.

Compared to the other variations, removing cycles and adding transitive relations showed the highest improvement on the agreement, also for pairs of annotators (Table 2). Our results suggest that different competence level entails different annotations and values of agreement, confirming previous results (Gordon et al., 2016): lower agreement can be observed when annotator 4 (quasi–expert) is involved, possibly due to the lower competence level if compared to the other annotators. Annotator 4 is also the one who considered the highest number of transitive relations, producing a more connected ECM: it is likely that when the competence in the domain is lower, a person tends to consider a higher number of prerequisites for each concept. On the other hand, annotators with more experience show even *moderate* (pairs A1-A3 and A2-A3) or *substantial agree-*

Metric		Orig.	No Cycl. & Trans.	Diff
Fleiss's *k*	All raters	38.50%	**39.94%**	+1.44
Cohen's *k*	A1-A2	34.46%	42.81%	**+8.35**
	A1-A3	57.80%	50.84%	**-6.96**
	A1-A4	37.59%	39.29%	+1.70
	A2-A3	56.50%	63.62%	+7.12
	A2-A4	28.02%	29.42%	+1.40
	A3-A4	25.35%	25.71%	+0.36

Table 2: Agreement values and differences for two annotation variations.

ment (pair A2-A3 for the variation). Adding transitive relations and removing cycles generally improves the agreement values also when we consider pairs: we notice an increase of 8.35 points for A1-A2. The only exception is observed for the pair A1-A3, which experienced a decrease of almost 7 points. The cause is though to be the number of transitive relations considered by annotator 3, which is around one third of the transitive relations annotated by annotator 1: the validation creates more distance between the two annotations reducing the agreement.

As a support for the annotation, the experts used a $n \times n$ matrix of the terminology T where they entered a binary value in the intersection between two concepts to indicate the presence of a prerequisite relation. We believe that our results are partially influenced by the instrument we used to perform the annotation: a large matrix structure is likely to cause distraction errors and does not perform validation checks during the annotation. Based on this experience and the encountered issues, we developed an annotation tool able to support and validate the annotation. It will be described in the next section.

5 Annotation and Analysis Tool

We provide a language and domain independent prototype tool which aims on the one hand to support and validate the annotation process and on the other hand to perform annotation analysis. All its main features have been designed taking into account real problems encountered while building PRET. Thus, this tool is highly valuable for annotators because specifically addresses annotators' needs and, at the same time, avoids possible annotation biases. In particular, the tool has three main functionalities: annotation support, annotation representation and analysis of the results.

To support the annotation, the user is provided with the terminology T as a list L of concepts ordered by their first occurrence in the text. This is done in order to give the annotator an overview of the context in which the concept occurs. We observed that the textual context plays a crucial role in deciding which concepts are prerequisites of the one under observation, so for each term we show the list of other terms with visual indication of the progress in the text. Additionally, as said before, the tool validates the map resulting from the annotation against the existence of symmetric relations, transitivity and cycles.

Once the annotation is completed, the user can choose to generate different types of visualization of her/his annotation. The goal of this functionality is to provide information visualization and data summarization for analyzing and exploring the result of the annotation. We provide the following different views: Matrix (ordered by concept frequency, clusters, temporal, occurrence or alphabetic order), Arc Diagram, Graph and Clusters. Furthermore, the Data Synthesis task provides the number of concepts, number of relations, number and list of disconnected nodes and transitive relations.

Lastly, the tool computes the agreement between relations inserted by all annotators who took part in the task (see Section 4.1) and provides visualization of the final dataset, which results as a combination of all users' annotation. This feature also outputs a Data Synthesis that provides the number of relations of every annotator, number of transitive relations and the direction of conflicting relations between annotators.

The demo version of the tool is available online at the URL provided in the Introduction.

6 Conclusion and Future Work

In this paper, we described PRET, a gold dataset manually annotated for prerequisite relations between pairs of concepts; moreover we presented the methodology we adopted and a tool to support prerequisite annotation. The case study, even limited as for the number of annotators and the educational material, was a reasonably good training ground to set the basis to define a methodology for prerequisite annotation and to identify the major issues related to this task. Moreover, the analysis of the annotation provided insights for automatic identification of concepts and prerequisites, that will be investigated in future work.

References

Giovanni Adorni and Frosina Koceva. 2016. Educational concept maps for personalized learning path generation. In *Conference of the Italian Association for Artificial Intelligence*, pages 135–148. Springer.

Giovanni Adorni, Felice Dell'Orletta, Frosina Koceva, Ilaria Torre, and Giulia Venturi. 2018. Extracting dependency relations from digital learning content. In *Italian Research Conference on Digital Libraries*, pages 114–119. Springer.

Rakesh Agrawal, Behzad Golshan, and Evangelos Papalexakis. 2016. Toward data-driven design of educational courses: a feasibility study. *Journal of Educational Data Mining*, 8(1):1–21.

Fareedah Alsaad, Assma Boughoula, Chase Geigle, Hari Sundaram, and Chengxiang Zhai. 2018. Mining MOOC lecture transcripts to construct concept dependency graphs. In *Proceedings of the 11th International Conference on Educational Data Mining, EDM 2018, Buffalo, NY, USA, July 15-18, 2018*.

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Glenn Brookshear and Dennis Brylow, 2015. *Computer Science: An Overview, Global Edition*, chapter 4 Networking and the Internet. Pearson Education Limited.

Devendra Singh Chaplot, Yiming Yang, Jaime G. Carbonell, and Kenneth R. Koedinger. 2016. Data-driven automated induction of prerequisite structure graphs. In *Proceedings of the 9th International Conference on Educational Data Mining, EDM 2016, Raleigh, North Carolina, USA, June 29 - July 2, 2016*, pages 318–323.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Felice Dell'Orletta, Giulia Venturi, Andrea Cimino, and Simonetta Montemagni. 2014. T2k^2: a system for automatically extracting and organizing knowledge from texts. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*.

Alexander R Fabbri, Irene Li, Prawat Trairatvorakul, Yijiao He, Wei Tai Ting, Robert Tung, Caitlin Westerfield, and Dragomir R Radev. 2018. Tutorialbank: A manually-collected corpus for prerequisite chains, survey extraction and resource recommendation. In *ACL*.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Fabio Gasparetti, Carla Limongelli, and Filippo Sciarrone. 2015. Exploiting wikipedia for discovering prerequisite relationships among learning objects. In *2015 International Conference on Information Technology Based Higher Education and Training, ITHET 2015, Lisbon, Portugal, June 11-13, 2015*, pages 1–6.

Jonathan Gordon, Linhong Zhu, Aram Galstyan, Prem Natarajan, and Gully Burns. 2016. Modeling concept dependencies in a scientific corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 866–875.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

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.

Chen Liang, Jianbo Ye, Zhaohui Wu, Bart Pursel, and C Lee Giles. 2017. Recovering concept prerequisite relations from university course dependencies. In *AAAI*, pages 4786–4791.

Chen Liang, Jianbo Ye, Shuting Wang, Bart Pursel, and C Lee Giles. 2018. Investigating active learning for concept prerequisite learning. *Proc. EAAI*.

Joseph D. Novak and Alberto J. Cañas. 2006. The theory underlying concept maps and how to construct and use them. research report 2006-01 Rev 2008-01, Florida Institute for Human and Machine Cognition.

Liangming Pan, Chengjiang Li, Juanzi Li, and Jie Tang. 2017. 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.

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.

Annalies Vuong, Tristan Nixon, and Brendon Towle. 2011. A method for finding prerequisites within a curriculum. In *Proceedings of the 4th International Conference on Educational Data Mining, Eindhoven, The Netherlands, July 6-8, 2011*, pages 211–216.

Shuting Wang, Chen Liang, Zhaohui Wu, Kyle Williams, Bart Pursel, Benjamin Brautigam, Sherwyn Saul, Hannah Williams, Kyle Bowen, and C Lee Giles. 2015. Concept hierarchy extraction from textbooks. In *Proceedings of the 2015 ACM Symposium on Document Engineering*, pages 147–156. ACM.

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.

Yiming Yang, Hanxiao Liu, Jaime Carbonell, and Wanli Ma. 2015. Concept graph learning from educational data. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 159–168. ACM.