Validation of Course Ontology Elements for Automatic Question Generation

Noor Hasimah Ibrahim Teo(✉) and Mike Joy

Department of Computer Science,
University of Warwick, Coventry CV4 7AL, UK
{N.H.Ibrahim-Teo,M.S.Joy}@warwick.ac.uk

Abstract. Recent research has led to the emergence of ontology-based question generation and aims to benefit instructors by providing support and intelligent assistance for the automatic generation of questions. However, existing ontologies are not designed mainly for this purpose and the concern is that an ontology will not be competent enough to act as a semantic source for the question generation process. Therefore, the aim of this work is to validate how well the elements represented in a course ontology can be used for the purpose of automatic question generation. In this work, we choose to validate Operating System ontologies and identify related question sources from textbooks on this subject as competency questions. Finally, the result shows that the evaluated ontologies need more modification if they are to be used for question generation and we also suggest a list of concept naming patterns that need to be considered for such ontology modification.

Keywords: Competency question · Course ontology · Question generation

1 Introduction

Ontologies have been widely used in educational environments and the number of evaluations of them is increasing. Various techniques for ontology evaluation have been already proposed, comprising validation and verification of ontologies taxonomies as well as of their content. To develop a *complete* ontology is almost impossible in practice, but as long as an ontology can be used to solve a specific problem, it is considered to be sufficient. A course ontology is a subject domain ontology that represents knowledge of educational learning content and, like other ontologies, it contains concepts and the relationships that exist between those concepts. A course ontology can be used to automatically generate questions related to course content. Therefore, evaluation of existing course ontologies is crucial to determine their coverage, and validation of concepts presented in the ontologies towards real world assessment questions will help to achieve this purpose.

Course ontologies can be categorised as domain ontologies where the scope is limited to delivering educational learning content. There are some course ontologies found in the literature such as Object Oriented Programming [1], Operating Systems [2, 3], Mathematical Logic [4] and Networking [5], but the competency of each ontology to be used as a source of semantic information for automatic question generation is not

G. Vincenti et al. (Eds.): eLeot 2016, LNICST 180, pp. 135–142, 2017.
DOI: 10.1007/978-3-319-49625-2_17

known. Therefore, we consider validation of course ontologies which are used as a source of information for automatic question generation later in this paper. Since validation needs real world examples, real assessment questions will be used as competency questions. The definition of competency question in this paper is slightly different from the one that normally use in other literature. We define a competency question as a question stated in natural language and containing required terms for the particular context. We will discuss the validation of course ontology elements relating to particular concepts and relations, using competency questions to determine the sufficiency of the information represented in the ontology to be used for automatic question generation.

An Operating System course from the Computer Science domain was chosen to begin with due to the availability of this course ontology on the web and the nature of the test questions being mainly factual. Operating System ontologies from Kent University Library [2] and ONKI Library [3] were chosen to be validated and have been named as *OntoA* and *OntoB* respectively throughout this paper. A set of competency questions related to this subject is chosen and discussed in the following section.

2 Competency Questions

Gruninger and Fox [6] had used competency questions as a means to evaluate whether an ontology is sufficient for its intended purposes. These questions are not only used for categorizing an ontology but also to drive the development of new ontologies to fit certain purposes. The use of competency questions is a well-known technique for determining the requirements the ontology should fulfil.

Competency questions used in this research are collected from Operating System review questions in Silberschatz et al.'s textbook [7]. The competency questions used for this ontology evaluation will be used to determine the coverage of concepts in the chosen ontology, as well as to enrich the ontology with missing concepts and relations. These competency questions cover 15 chapters and these are good for identifying which chapters have fewer concepts represented in the ontology.

The list of competency questions is used as an input for this validation, where a string similarity algorithm will be executed to extract any terms in the questions that match a given concept in the course ontology. The detailed discussion about the validation process will be discussed in the following section.

3 Ontology Concepts Validation Process

The evaluation was conducted using Operating System review question in [7]. A total of 259 questions from 15 chapters, which contain short answers and true/false question types, were analyzed. The two ontologies used are *OntoA* containing 97 triples and 97 concepts, and *OntoB*, which contains 1041 triples and 980 concepts. Both ontologies use only hierarchical types of relation. The main task for this validation is to match the extracted terms in each question against the concepts represented in both ontologies.

Dice's Similarity Coefficient (DSC) algorithm is applied by extracting character bigrams to calculate similarity scores of two strings. The algorithm had been modified to allow matching between pairwise words. This is because most of the terms exist in questions are linked pairwise. Therefore, instead of comparing one word with another, the algorithm matches a pair of words used in the question with a pair of words that represents a concept in the ontology. For example, the term 'operating system' appearing in a question can be matched with the 'Operating-System' concept in the ontology. Preprocessing was performed to create a combination of words from each sentence. The first word for each question will have an empty string as a pair and under-score will be added between two words. For example: the question "*What is an Operating System?*" is tokenized into a pairwise string as [" -What, What-is, is-an, an-Operating, Operating-System?"] and stored in an array. Later, each of these tokens will be mapped to the concept in the ontology. DSC is calculated as follows:

$$\text{similarity\_score}(\text{WP1},\ \text{WP2}) = \text{intersection}\ (\text{WP1}, \text{WP2}) * 2 \,/\, \text{total} \tag{1}$$

The similarity between strings WP1 and WP2 will give a similarity score of 1 when both strings have all their bigrams intersecting or matching and 0 if there is no intersection at all. Otherwise, it will have a score that varies between 0 and 1. The following is the algorithm for matching terms in a question with the concept in the ontology.

```
Score = 0
For every question Q do
   Split question Q into token
      While has more tokens
         For every concept C in ontology
              Create pair of word in question
              Calculate similarity score()
              If( similarity score > 0.9)
                   Display matched (M1) and its score
              Endif
          Endfor
      Endloop
Endfor
```

4 Result and Discussion

This section will discuss the results obtained for two evaluations. Concept validation analysis will first give an input to how well the ontology concepts can be used for question generation, and secondly the numbers of questions that can be generated from both ontologies by considering existing concepts and relations.

4.1 Concepts Validation Analysis

This evaluation analyses the numbers of questions that contain terms that matched the concepts present in the ontologies. After running the experiment, we found most terms in both ontologies were not matched with the concepts in the ontologies as the algorithm only detects *pairwise* terms. We investigate this problem further manually and found the result as in Fig. 1. We classify the result found into three matching levels which are M1, M2 and M3 that represent 'Concept is exactly matched with the term in the questions', 'Concept is partially matched with the term in the questions', and 'Concept does not exist in the ontology' respectively.

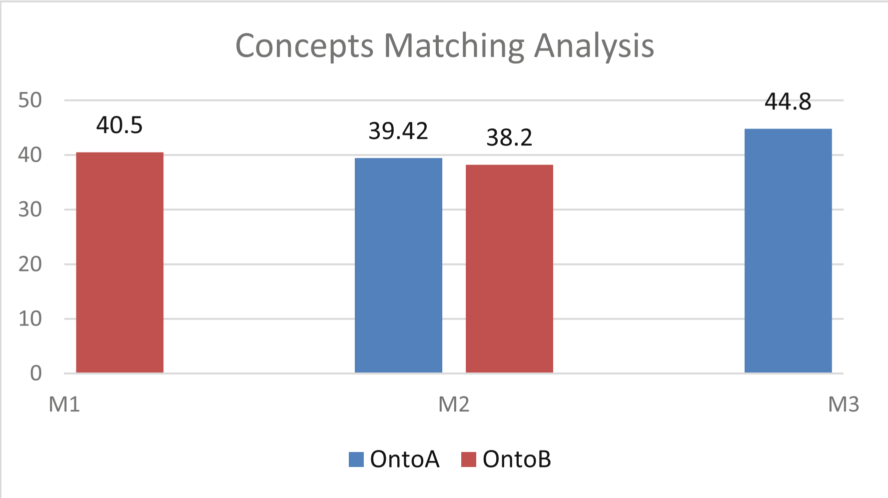

Fig. 1. Percentage of questions with different matching levels

The analysis results in Fig. 1 shows that 78.7 percent of terms in the questions exist in *OntoB* and 54.82 percent of terms in questions exist in *OntoA*. The figure is calculated by the total percentage of categories M1 and M2 combined. M1 is meant for the pairwise terms that have similarity score of 0.9 with the concept existing in the ontology, which is nearly half of the questions with the terms detected in *OntoB* but it appears less than a quarter were detected in *OntoA*. This category calculates similarity scores using methods discussed in the previous section. M2, representing partial matches, gave the highest percentage for both ontologies, which is nearly half of the questions. This level has included single terms, triple terms, multiterms, terms with suffices, terms with a combination of noun and verb phrases, and acronyms. Table 1 shows the numbers of ways of naming concepts and the numbers of each occurrence. The evaluation has been able to identify 9 categories under the M2 level of term pattern that may be useful information to be analyzed for developing a more useful course ontology. The example of all categories in M2 is shown in Table 2.

Table 1. Categories of concept naming and their occurrences.

No.	Categories	*OntoA*	*OntoB*
1	Single terms	9	23
2	Pairwise terms	47	121
3	Triple terms	1	8
4	Multiterms	0	2
5	Terms that form part of the concept	2	13
6	Term combinations	0	4
7	Parts of a concept that have different meanings	4	9
8	Terms with suffices	2	3
9	Combinations of several concepts in the ontology	0	5
10	Acronyms	0	3

Table 2. Term pattern examples in M2 category.

No.	Categories	Example	
		Inside question	Inside ontology
1	Single terms	Interrupts	Interrupts
2	Triple terms	Process control block	Process control block
3	Multiterms	Multiple feedback queue scheduling algorithm	Shortest-job-first-scheduling-algorithm
4	Terms that form part of the concept	Message passing	Message-passing-model
5	Term combinations	Communication + client server system	Communication–in-client-server-system
6	Parts of a concept that have different meanings	RAID	RAID-structure, RAID-level
7	Words with suffices	Process, scheduler	Processes, scheduling
8	Combinations of several concepts in the ontology	Distributed-information-systems	Distributed-information-systems/ distributed-naming-services
9	Acronyms	VFS	Virtual-file-systems’

M3 shows the numbers of terms that do not exist in the ontology and the number is higher in *OntoA* compared to *OntoB* with a difference of 26.3 percent. This may be due to two reasons which are (i) that the number of concepts in *OntoB* is much higher than *OntoA*, and (ii) that the concept representation in *OntoB* was mainly developed for the textbook which provided the questions used for this validation.

From the experimental evaluation, several important observations have been made. The first relates to the scope of the ontology that has been evaluated by means of concept completeness. Concept completeness in this work is defined as whether all important concepts in each course within the syllabus are represented in the ontology – if they are, the ontology is concept complete. Second, the result for *OntoB* has shown to have a better representation compared to *OntoA*. *OntoB* has shown that more than three-quarter of the terms in questions exist in the ontology with half of them identical and another half would need some minor modification. This would mean that the ontology needs only minor effort to be enriched and make it concept complete with only a quarter of new concepts needing to be added to the ontology.

4.2 Numbers of Questions that Can Be Generated from a Course Ontology

This evaluation discussed the number of questions that can be generated from both ontologies. The evaluation is classified into 4 categories and the following is an example to show how the questions are categorized. We assume a question contains 'question word', 'noun' and 'action verb' where noun will be represented as concept and action verb will be represented as a relation in the ontology.

Assume we have triples '`[Y] is-a [X]`' and '`[Z] is-a [X]`', 'question word' are [what, define, explain] and 'action verb' are [the purpose of, the advantage of].

A: Complete
All terms match the concepts in the ontology.
Examples: "What is [X]?" and "Define [X]".

B1: Nearly Complete
Only some terms in question matched the concepts in the ontology.
Example: "Explain [X] in [Q]".

B2: Incomplete
Question contains an action verb which does not exist in the ontology but all terms match the concepts in the ontology.
Example: "Explain the purpose of [X]".

C: Cannot be generated
Question contains an action verb which does not exist in the ontology and no term matches any concept in the ontology.
Example: "Explain the advantage of [Q]".

The result in Fig. 2 shows the outcome of the mapping process between questions and ontology elements, in particular the concepts and relations in the course ontologies.

The result shows that less than 5 percent of the questions can be generated using the ontology and all are questions that only need the existence of a concept and a template question such as "What is X?" to support question generation. Most of the questions can be partly generated from a hierarchical type of ontology. This shows that the ontology needs to have certain kinds of a predicate to relate two concepts in order for it to be able to generate meaningful questions. Category B1 contributes the largest number of cases for both ontologies where each question cannot be generated as the question has not enough concepts. Category B1 shows that significant effort can be made to add concepts to enable the ontology to generate questions. This effort will contribute to the larger number of questions that can be generated. However, for category B2, since both ontologies are hierarchical, adding object property relationships between concepts in ontology is troublesome. Varieties of words need to be considered for relationships and this is quite a tedious task.

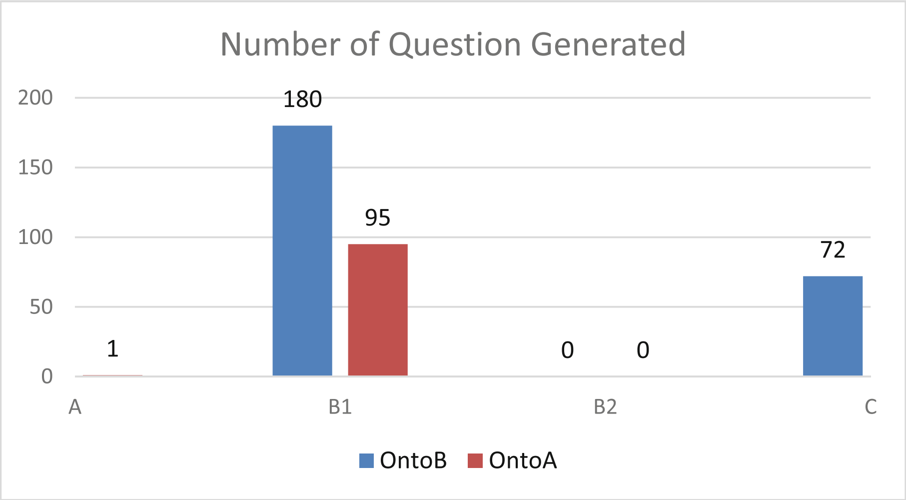

Fig. 2. Number of questions generated

In addition, many of the concepts in *OntoA* did not exist, and significant numbers in *OntoB* as well. Most of the concepts used in these two ontologies intersect and there is no need for combining the two ontologies to make it complete. The ideal way would be to use *OntoA* as a basic ontology for use with automatic question generation.

Furthermore, with regard to the relationship, '*is-a'* type of relation alone might contribute to just a small percentage of questions generated. Missing appropriate relations between concepts may not generate semantically correct questions. For example, in the question 'What *are three components of an Operating System*', and when we remove the word '*components-of'* that act as a relationship in ontology, the question becomes '*What are three Operating Systems*' and now has a different meaning. The result has shown that more than half of the question cannot be generated due to the absence of relations that link between two concepts.

5 Conclusion

The course ontology validation results could suggest different dimensions of improvement to prepare ontologies for automatic question generation. First, the hierarchical type of ontology is not comprehensive enough to use as a source of semantics for the question generation process. It will take a lot of effort to enrich the relations of the ontology especially given the huge size of many ontologies. Second, more than half of the terms in questions exist in the ontologies and this gives a good indication that the ontology scope is sufficient with little effort needed to redefine certain concepts. And finally, there is a need for other strategies to support question generation with ontologies in order to enhance question readability and to enable semantically correct question generation such as question templates. Future work will look into the techniques to enrich information in the course ontology and strategies for question generation.

Acknowledgement. This research was supported by Universiti Teknologi MARA and Ministry of Education, Malaysia.

References

1. Abuhassan, I., Al Mashaykhi, A.: Web Based OOP Domain Ontology (WOOPDO). LAP Lambert Academic Publishing, Deutschland (2012)
2. Khan, J., Hardas, M.S.: A technique for representing course knowledge using ontologies and assessing test problems. In: Wegrzyn-Wolska, K.M., Szczepaniak, P.S. (eds.) Advances in Intelligent Web Mastering. ASC, vol. 43, pp. 174–179. Springer, Heidelberg (2007)
3. Finnish Ontology Library Services. https://onki.fi/en/browser/overview/operating-system
4. Boyatt, R., Joy, M.: Developing an ontology of mathematical logic. Sci. J. Tech. Univ. Civil Eng. **6**, 6–11 (2010). Technical University of Civil Engineering, Bucharest
5. Jiang, L., Zhao, C., Wei, H.: The development of ontology-based course for computer networks. In: International Conference on Computer Science and Software Engineering, Wuhan, pp. 487–490. IEEE (2008)
6. Gruninger, M., Fox, M.S.: The role of competency questions in enterprise engineering. In: Rolstadås, A. (ed.) Benchmarking - Theory and Practice. IFIP Advances in Information and Communication Technology, pp. 22–31. Springer, New York (1999)
7. Silberschatz, A., Galvin, P.B., Gagne, G.: Operating System Concepts. Wiley, New Jersey (2012)