

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/356016588>

Curriculum Design Assistant System Based on Automatically-Generated Educational Ontologies

Article in *Revista Iberoamericana de Tecnologías del Aprendizaje* · November 2021

DOI: 10.1109/RITA.2021.3126583

CITATIONS

0

READS

83

3 authors, including:



[Angel Fiallos Ordoñez](#)

Escuela Superior Politécnica del Litoral (ESPOL)

25 PUBLICATIONS 69 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Social Network Analysis in Ecuador [View project](#)

Curriculum Design Assistant System Based on Automatically-Generated Educational Ontologies

Angel Fiallos Ordóñez^{ib}, *Member, IEEE*, Katherine Chiluiza García, *Member, IEEE*,
and Xavier Ochoa Chehab^{ib}, *Member, IEEE*

Abstract—This work describes the design and implementation of a functional assistant system to support instructors in designing or re-designing their courses' curriculum. The system uses automatically generated ontologies from textbooks that reflect the structure of a specific body of knowledge. Twelve instructors participated in the generation of curricula for a programming fundamentals course using the assistant system. An initial evaluation of this tool suggests that the system successfully helped novel instructors design a new curriculum.

Index Terms—Ontologies, natural language processing, education, learning analytics, deep learning, intelligent curricula, word2vec.

I. INTRODUCTION

PREVIOUS studies have already identified the importance of developing ontologies in the field of education [1]. Ontologies create a conceptual skeleton in various domains, which serves a variety of educational purposes, such as better understanding, creation, sharing, and reuse of knowledge [2], collaborative learning [3], intelligent tutorials [4], problem-solving, and skills improvement [5].

Curriculum management and development can be improved using ontologies in curriculum tasks like aligning, comparing, and matching between universities, educational systems, or relevant disciplines. Following this approach, we built a functional assistant system for teachers that use automatically-generated course-based ontologies to help teachers creating Intelligent Curricula. The application shows a preliminary ontology, and teachers could improve it via additional material or direct manipulation. Next, with the teachers' information, a mapping process identified the most relevant concepts using terms available or not in the domain ontology nodes to perform learning resources suggestions.

Teachers can also evaluate the quality of intelligent curricula and validate whether their documents' content is covered by

Manuscript received April 5, 2021; revised May 20, 2021; accepted June 4, 2021. Date of publication November 8, 2021; date of current version December 30, 2021. (*Corresponding author: Angel Fiallos Ordóñez.*)

Angel Fiallos Ordóñez and Katherine Chiluiza García are with the Facultad de Ingeniería Eléctrica y Computación, Escuela Superior Politécnica del Litoral (ESPOL), Guayaquil 90112, Ecuador (e-mail: anfiallos@fec.espol.edu.ec; kchilui@fec.espol.edu.ec).

Xavier Ochoa Chehab is with the Department of Administration, Leadership, and Technology, NYU Steinhardt School of Culture, Education, and Human Development, New York, NY 10003 USA (e-mail: xavier.ochoa@nyu.edu).

There exists a Spanish version of this article available at <http://rita.det.uvigo.es/VAEPRITA/V9N3/A2.pdf>

Digital Object Identifier 10.1109/RITA.2021.3126583

curricular standards, such as the computer science curriculum, CS2013 [6]. Thus, the system uses a multi-level classification model implemented through a combination of Machine Learning and Natural Language Processing techniques.

The main contribution of this work is to evaluate teachers' perception about the usefulness, easiness, engagement, and other aspects related to the domain ontologies offered by our proposed system. To achieve this, we prepared an experimental test focused on teachers who preferably have not taught the programming fundamentals course since the previous semester or more.

The main research questions were as follows:

- RQ1: Can the generated ontologies be useful in an assistant tool to design course curricula?

- RQ2: Which functionality had the highest contribution during course curricula design?

This article is structured as follows. First, a related work section presents a brief discussion of educational ontologies and intelligent curriculum research and our contribution to state of the art. Next, the methodology section describes the processes and functionalities of the system. Subsequently, the experimental design section describes the participants, the experimental context, and other research design details. Also, we present the results and discussion of our findings. Finally, the article ends with conclusions and possible uses in the field of education.

II. LITERATURE REVIEW

Conceptual model visualization methods, such as ontologies, are widely and effectively used in education. Many educational ontologies have been developed for many disciplines [7], [8] and related approaches, such as adaptive educational systems [9], engineering requirements for education [10], authoring tools [11], and intelligent curriculum design [12]. Intelligent curricula have been identified as one of the enablers of data-driven decision support systems in education [13]. It can be defined as representing the domain of knowledge that is generally taught in a course that can be understood and processed by a computational system.

The most common representation that meets this requirement is an ontology. Many educational and learning analytics applications use intelligent curricula represented as course ontologies to recommend learning materials [14] to automatically sequence learning activities [15] to evaluate the quality of

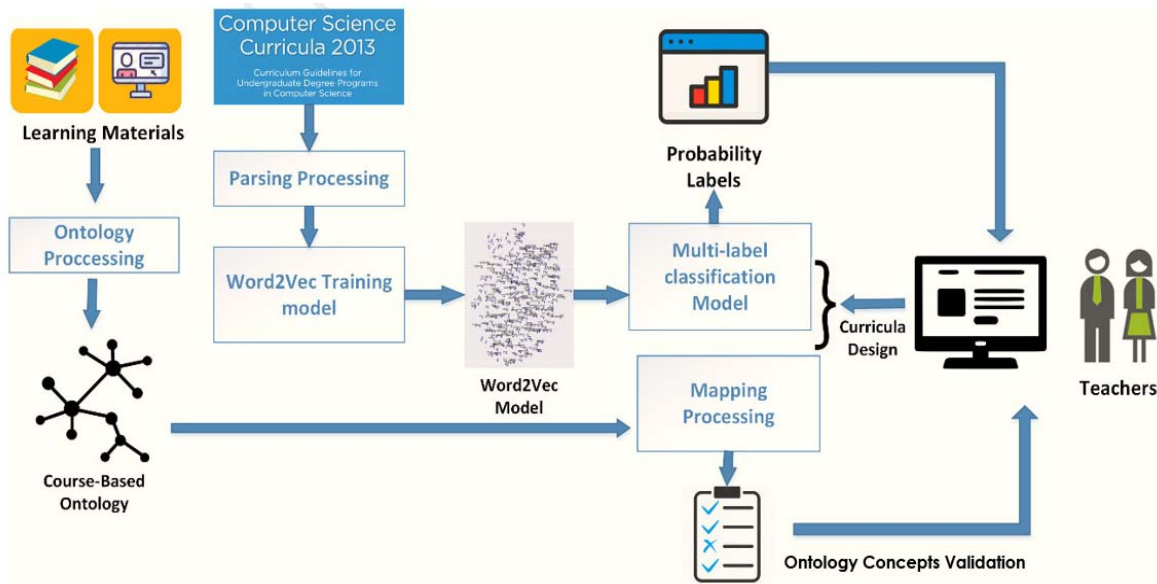


Fig. 1. Curricula Design assistant system Architecture.

contributions in online forums [16]. Furthermore, to monitor the authoring process and resolve inconsistencies [17] or provide visual feedback to students on their progress [18].

However, the creation of intelligent curricula from educational ontologies, including automatically extracted ones, is a field with few research studies and many application possibilities within education and learning analytics. Moreover, the cost of manually creating these course-oriented ontologies is high [19] mainly because knowledge analysts are rarely experts in semantic technologies and vice versa.

This study seeks to shed light on creating intelligent curricula by exploring the use of automatically generated course ontologies as a basis for the design and evaluation of curricula through a curriculum design support system. While instructors can design curricula and add learning content, our system recommends a preliminary ontology for the current course, and the user can improve it through direct manipulation.

III. METHODOLOGY

Figure 1 shows the phases followed for the implementation of the curriculum design support system. The process starts with the collection of data from digital books. It continues with the techniques and models used to build the course ontologies, which will be used by the system's functionalities and explained below.

A. Curricula Design Assistant System

The system through web interfaces allows teachers to register, edit and validate curricula. The system accesses course ontologies to provide guidance and content suggestions to users. Java and JavaScript languages were used to code the text editor and libraries such as JGraphT [20] and Stanford Core [21] to develop parsing algorithms and visualization of graph structures.

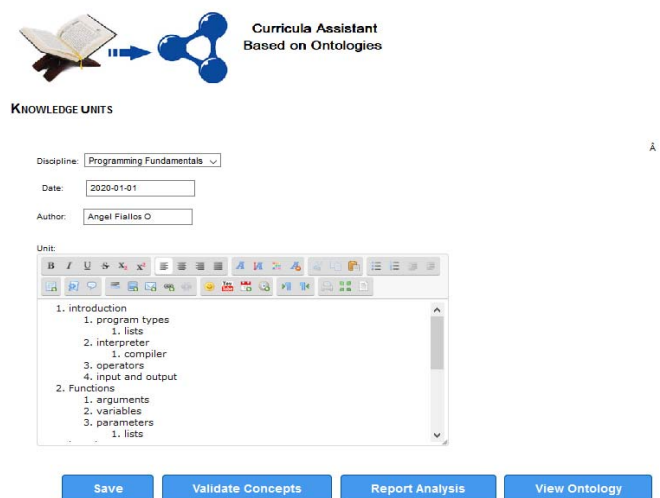


Fig. 2. Curricula Design Assistant System interface.

Figure 2 shows an interface to the curriculum design support system consisting of a course list to access the pre-configured ontologies, a text editor to record the course curriculum, and the functional options available to teachers and instructors.

B. Course Ontologies Generation

The authors explored a methodology for semi-automatic generation from digital texts to obtain course ontologies. Similar to the work proposed by Guerra, two digital books were selected from the fundamental programming course recommended by professors who teach it. (“Learn to Program with Python”, and “Think Python”). A parsing process was applied to divide the documents into their hierarchical components (chapters, sections, subsections) related to similar sections from other documents, using as a metric the similarity between both sections’ texts [22].

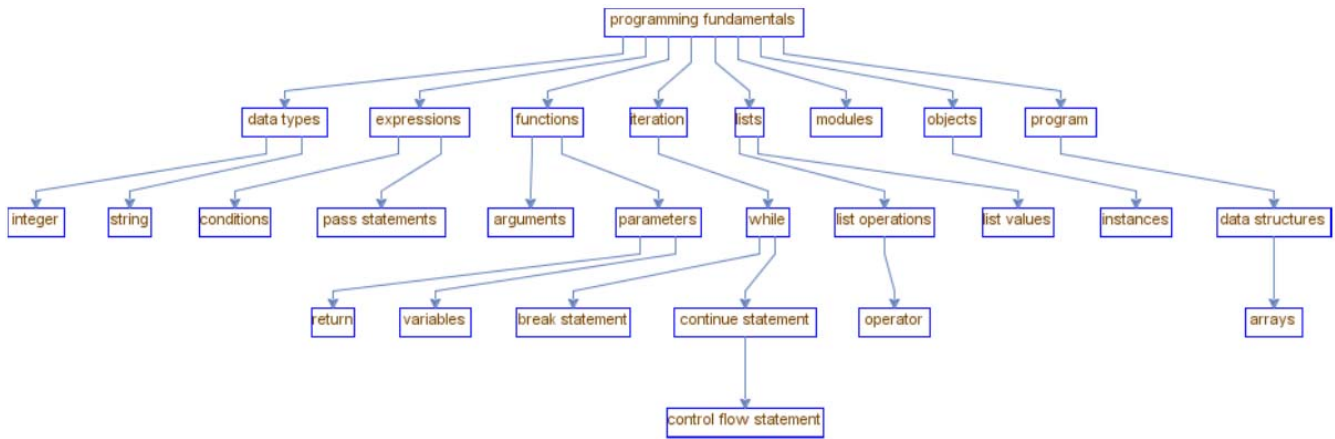


Fig. 3. Ontology course “Programming Fundamentals.”

Next, the Latent Dirichlet Allocation (LDA) topic model [23] was used to determine the most relevant topics in the selected book sections with the highest similarity value. Using OpenIE [24] (library for triple extraction), the words with the highest probability in the identified topics were connected according to their semantic relations and the hierarchical structure present in the books.

The result was an ontology with 74 concepts for the programming fundamentals course, similar to the one shown in Figure 3. Then, through the libraries provided by the editor Protégé [25], the ontology was configured in OWL format. The system can access it and visualize the model’s most relevant concepts in the course ontology nodes.

C. Curricula Concepts Validation Process

It was possible to extract label-delimited text and construct abstract representations of the course curricula through a parsing process. Other elements, such as bullets, punctuation, and line breaks, were also used to identify the plain text’s relevant information. Linguistic analysis using Stanford NLP libraries and part-of-speech tagging (POST) techniques were then applied to determine the nouns and keywords available in each document to obtain a list of the terms used by teachers. The ontology serves the function of a model for knowledge representation, i.e., its nodes represent the domain’s concepts, and the links determine the relationship between them. A mapping process used these links to locate ontology nodes associated with specific concepts available in the course curricula registered by teachers.

Once the mapping process is done, the identified and unidentified concepts and keywords are shown to teachers with a special marking to help them select terms for curriculum design. Figure 4 shows a screenshot of the functionality for concept validation. The colors’ difference can be seen, yellow for concepts not included in the ontology and blue for concepts included.

D. Report of Classification Analysis

This functionality aims to ensure the quality content of course curricula registered by teachers through a multi-layer

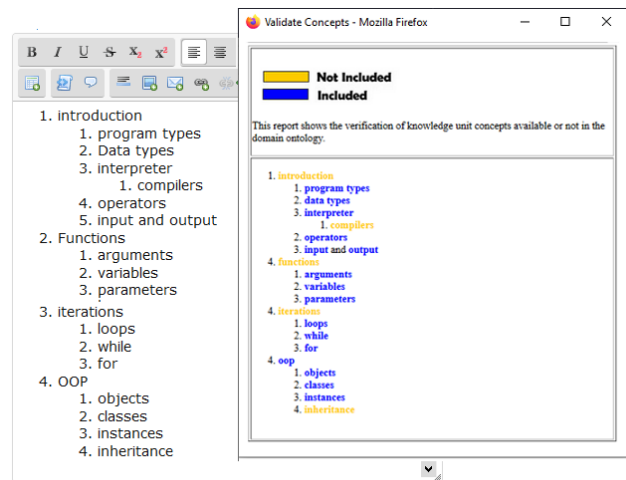


Fig. 4. Concept validation functionality.

classification model implemented using convolutional neural network (CNN) models [26] and pre-trained word vectors. First, semi-structured data from the following courses’ knowledge units were used: programming fundamentals, computer architecture and organization, information management, programming languages, algorithms, operating systems, and artificial intelligence, available in the CS-2013 curriculum standard.

Then, to build the final corpus, a four-level hierarchical structure was designed. The first, second, and third-level sentences were obtained from the selected courses’ knowledge units in the CS-2013 curriculum standard. The last level sentences were obtained from Wikipedia pages, linked to the third level’s main concepts. Table I shows an example of excerpts of sentences extracted from the CS-2013 curriculum standard and Wikipedia pages for each course.

Following Kim’s approach [27], a simple convolutional neural network was designed. It consists of an input layer with five different window sizes (with ngrams) and a convolution layer of word vectors obtained using the unsupervised Word2Vec model [28]. These vector representations are extractors that encode semantic features of words in their dimensions to extract patterns.

TABLE I
CURRICULUM STANDARD CS2013 SENTENCES
AND WIKIPEDIA CONCEPTS

Course	Level	Content
Programming Fundamentals	1	Fundamental Data Structures
Programming Fundamentals	2	Arrays
Programming Fundamentals	3	Stacks
Programming Fundamentals	4	Queues
Programming Fundamentals	4	Queues are common in computer programs, where they are implemented as data structures coupled with access routines
Software Engineering	1	Software Processes
Software Engineering	2	Team participation. Roles and responsibilities in a software team
Software Engineering	3	Team processes including responsibilities for tasks, meeting structure, and work schedule
Software Engineering	3	Roles and responsibilities in a software team
Software Engineering	4	software engineering that deals with establishing the needs of stakeholders that are solved by software.
Artificial Intelligence	1	Basic Machine Learning
Artificial Intelligence	2	Definition and examples of broad variety of machine learning tasks, including classification
Artificial Intelligence	2	Simple statistical-based learning, such as Naive Bayesian Classifier, decision trees
Artificial Intelligence	3	Support vector machines (SVMs)
Artificial Intelligence	3	Learning decision trees,
Artificial Intelligence	4	Support vector machines (SVMs), are a set of related supervised learning methods used for classification and regression.

For implementing the model, sentences extracted from the CS2013 standard were trained using word-embeddings with 100 dimensions. Then, Gensim [29] and Keras [30] libraries were employed to build the convolutional neural network, using as parameters a learning ratio of 0.2 and an epochs count of 20. For the model setup, data was divided into training and test sets in a ratio of 80/20. The model evaluated itself after each epoch to adjust the parameters according to its loss function. The result was a trained model that can classify new values.

The trained model uses the confidence values to produce a ranking. At the end of the convolutional network, a sigmoid activation function was used to treat the labels independently. This multi-label ranking model was applied to the key concepts and phrases identified by Part of Speech (POS) tagging techniques in the curriculum content generated by teachers through our system.

As a result, each curriculum model returned in JSON format a set of probability tags related to the CS-2013 standard areas to be displayed to the user. Figure 5 shows an example of a curriculum's content with a greater than 70% probability of being labeled a Programming Fundamentals course in the CS-2013 curriculum standard.

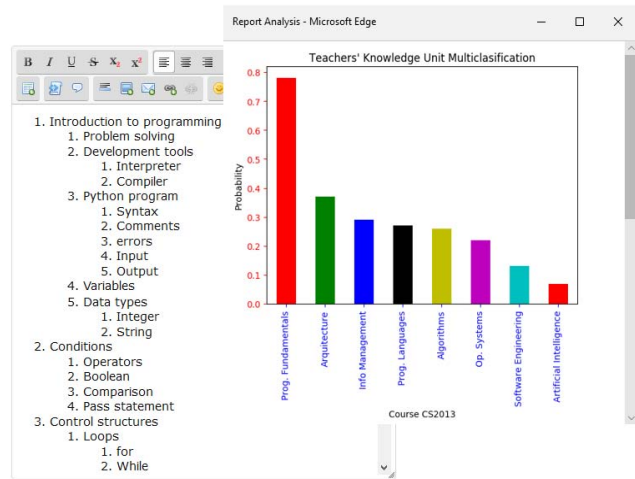


Fig. 5. Functionality “Report of Multi-classification analysis.”

IV. EXPERIMENTAL DESIGN

Twelve teachers belonging to Escuela Superior Politécnica del Litoral, ESPOL were selected for the pilot test. The teachers were young professionals (mean = 30 years), 75% were men, and 25% were women. The average teaching experience was 6.2 years; 25% of them have never taught the programming fundamentals course, 66.7% taught this course two semesters before the experiment, and the remaining 8.3% taught it during the experimental semester.


A. Experimental Context

Teachers who participated in this study were instructed to design curricula for programming fundamentals course with the restriction that it had to be between 30 and 50 concepts in a hierarchical structure. Teachers had access to an online version of the auxiliary curriculum system and its functionalities during the experiment. Also, teachers were informed that the maximum time for the task was 30 minutes.

Teachers were also given access to a digital copy of the programming books and received a brief description of conducting the experiment and an instruction manual. For instance, they were instructed that, in the first iteration, they could use the “View Ontology” option to visualize a course-based programming fundamentals ontology as a content reference. Moreover, the explanation indicated that the remaining system options (“Validate Concepts” and “Report Analysis”) could be enabled once the first version of their Intelligent Curricula was registered in the system. Next, a questionnaire similar to the one in Figure 6 with multiple-choice questions was implemented using Google forms to explore teachers’ perceptions about system functionalities during the curricula design. The link for the questionnaire was sent to the participants once the experiment finished.

For this analysis, the data was divided into two parts. The first part is related to RQ1 and included questions to assess teachers’ perceptions. The second part is associated with the RQ2 and corresponds to questions focusing on teachers’ opinions about the system’s functionalities.

Prototype Evaluation



**Knowledge Unit Recommender
Based in Ontologies**

How old are you as a teacher?

Have you taught in the previous semesters the subject Programming Fundamentals? (Y/N)

How useful was the tool to you during the design of the knowledge unit?

1 2 3 4 5

Nothing useful Very Useful

Rate from 1 to 5, the degree of ease of use of the tool

1 2 3 4 5

Not easy Very Easy

How much do you agree that the tool is used for other courses?

1 2 3 4 5

Totally disagree Totally Agree

Pick one of the following functionalities, that do you think had a greater contribution towards your objective?

View Ontology

Report Analysis

Validate Concepts

Search Knowledge Units

None

Would you consider using a version of the tool during a process of curriculum reform or change? (Y/N)

Fig. 6. Post-Experiment Questionnaire.

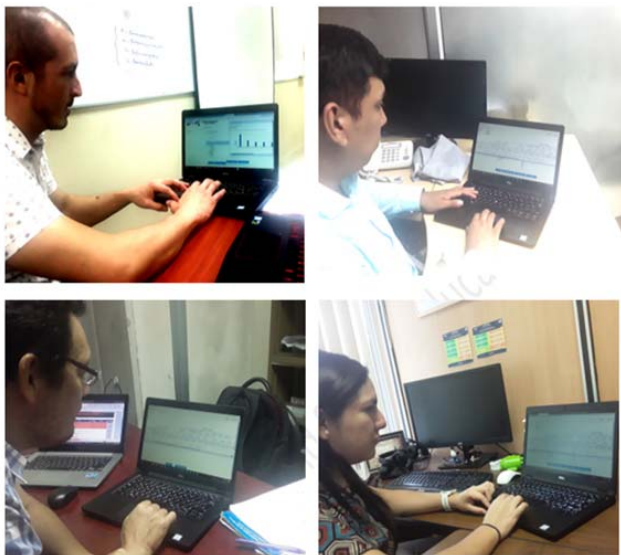


Fig. 7. Teachers using the system during the experiment.

Figure 7 shows screenshots of teachers using the system during the experiment.

Teachers’ perceptions of the usefulness, ease, and recommendation to use the tool for other courses were measured on a five-point Likert scale (One equals strongly disagree and five equals strongly agree).

V. RESULTS AND DISCUSSION

To answer the research questions posed above, we first determined descriptive statistics about the usefulness, ease, and recommendations for the system and its functionalities. Table II presents this information.

As shown in Table II, the assistant system is perceived as useful for curriculum design (mean = 3.9). Regarding the

TABLE II
DESCRIPTIVE STATISTIC: USEFULNESS, EASINESS, RECOMMENDATION OF USAGE

Question Objective	N	Mean	Std Desv.	Variance	Skewness
Usefulness of tool	12	3.9	0.70	0.491	0.123
Easiness of tool	12	3.8	0.75	0.564	0.329
Agreement of use the tool to other courses	12	4.3	0.67	0.455	-0.593

ease of use of the tool, the results were similar for usefulness (mean = 3.8). Most teachers perceived the tool and its functionalities to be relatively easy to use. Teachers also indicated that they would recommend it to others (mean = 4.3). These values allow us to answer RQ1, affirming that knowledge ontologies are useful for designing course curricula from the teachers’ perspective.

The second part of the questionnaire contained questions related to the functionality of the system. The first question investigates which functionality of the system had the most significant contribution for teachers during curriculum design. The results were as follows: “View ontology” with 55%, “Report Analysis” with 36%, and “Validate Concepts” with 9%. These results show that ontology is the functionality that helped teachers the most during the test. These results allow us to answer RQ2, stating that the functionality system that contributes most to intelligent curricula design is the ontology view.

The second question investigates whether teachers would use the tool during a curricular reform in academic curricula. The results were positive, with 91% of teachers indicating that they would use this tool. Also, participants made some comments about their experience with the test. Most of them suggested improvements to the curriculum support system, such as improving the editing process, visualizing ontologies, and the information structure. Table III shows the most relevant comments from teachers about their experience with the curriculum management support system.

Finally, we established a measure of accuracy as the percentage of concepts in each teacher’s curricula included in the CS2013 programming fundamentals knowledge units, available in digital documents. The average accuracy for all teachers’ curricular courses was 73%.

The idea of using visual structuring of information to improve the quality of understanding and thinking among research colleagues is not new [31]. Ontologies are useful structuring tools and provide an organizational axis along which teachers or students can mentally mark their view in the information hyperspace of domain knowledge [32]. These studies may explain why most participants indicated that ontology was the system functionality that helped them the most during testing.

This study is not without limitations. It was conducted on a limited number of participants, and the ontology of one computer science course was tested in the system. Concerning the curricula’ validation, an expert review of the curricula

TABLE III
TEACHERS MOST RELEVANT COMMENTS ABOUT
CURRICULA ASSISTANT SYSTEM

Question Objective	Comment
Usefulness of tool	It would seem like a good idea that the ontology can be grouped and opened so that the user when seeing the information, does not feel stressed from seeing so much text. The system could suggest the list of topics, and the teacher chooses the subset of topics that he wants to include and the order in which he wishes to dictate them.
Easiness of tool	I would like the information presented by the ontology can be dragged towards the system editor, preserving its hierarchy. A text editor with formatting tools may not be the best alternative, but a smaller interface where content is entered, and then it indicates that parts of the document are nested within the text.
Usage of the tool in other courses	The tool speeds up the design of a syllabus and offers immediate feedback on topics that have not been included.

could have been considered and estimate the percentage of concepts available in the CS2013 standard. However, overall, the tool had an adequate assessment of usefulness and ease of use. User comments highlight the few steps that need to be taken to achieve the objectives and request improvements in flexibility in using the ontology.

VI. CONCLUSION

This work explored how useful educational ontologies can be for the generation of intelligent course curricula. The results show that teachers felt that system helped them develop the programming fundamentals course curriculum. The functionality that contributed most to the design of the course was the ontology preview. The average precision of the curricula courses gives an insight into the quality of the content created by each tutor with the help of the assistant system tools.

These results are relevant because by helping teachers improve the curricular content of courses, with the support of ontologies based on digital books, recommended by experts, and content automatic validations, students can get better educational material at the time of development of the courses.

As future work, we can visualize the generation, use, and evaluation of educational ontologies for the curricular design of more academic courses belonging to other areas of knowledge such as social sciences or humanities and increasing the number of participating teachers. Also, we consider exploring other educational and learning analytics applications that consume ontologies for intelligent courses and curricula.

ACKNOWLEDGMENT

The authors would like to thank all the teachers on the Faculty of Electrical Engineering and Computer Science, the Escuela Superior Politécnica del Litoral (ESPOL), who participated in the experiments and surveys.

REFERENCES

- [1] G. Dall'Alba and R. Barnacle, "An ontological turn for higher education," *Stud. Higher Educ.*, vol. 32, no. 6, pp. 679–691, Dec. 2007.
- [2] K.-K. Chu, C.-I. Lee, and R.-S. Tsai, "Ontology technology to assist learners' navigation in the concept map learning system," *Expert Syst. Appl.*, vol. 38, no. 9, pp. 11293–11299, Sep. 2011.
- [3] B. Barros, M. Verdejo, T. Read, and R. Mizoguchi, "Applications of a collaborative learning ontology," in *Proc. 2nd MICAI*, 2002, pp. 103–118.
- [4] W. Yathongchai, T. Angskun, and J. Angskun, "SQL learning object ontology for an intelligent tutoring," *Int. J. e-Educ., e-Bus., e-Manage. e-Learn.*, vol. 3, no. 2, p. 168, 2013.
- [5] D. Fensel, E. Motta, S. Decker and Z. Zdráhal, "Using ontologies for defining tasks, problem-solving methods and their mappings," in *Knowledge Acquisition, Modeling and Management (Lecture Notes in Computer Science)*, vol. 1319. Berlin, Germany: Springer, 1997, pp. 113–128.
- [6] *ACM/IEEE-CS, Joint Task Force on Computing Curricula 2013*, ACM Press and IEEE Computer Society Press, New York, NY, USA, 2013.
- [7] F. Fonseca, C. Davis, and G. Câmara, "Bridging ontologies and conceptual schemas in geographic information integration," *GeoInformatica*, vol. 7, no. 4, pp. 355–378, 2003.
- [8] S. Miranda, M. Gaeta, F. Orciuoli, G. Mangione, and V. Loia, "Unlocking serendipitous learning by means of social Semantic web," in *Proc. 6th Int. Conf. Comput. Supported Educ. (CSEDEU)*, 2014, pp. 285–292.
- [9] S. Sosnovsky, I. Hsiao, and P. Brusilovsky, "Adaptation 'in the wild': Ontology-based personalization of open-corpus learning material," in *Proc. 21st Century Learn. 21st Century Skills*, Berlin, Germany, 2012, pp. 425–431.
- [10] D. Dermeval, J. Vilela, I. B. Bittencourt, J. Castro, S. Isotani, P. Brito, and A. Silva, "Applications of ontologies in requirements engineering: A systematic review of the literature," *Requirements Eng.*, vol. 21, no. 4, pp. 405–437, 2016.
- [11] S. Isotani *et al.*, "A semantic web-based authoring tool to facilitate the planning of collaborative learning scenarios compliant with learning theories," *Comput. Educ.*, vol. 63, pp. 267–284, Apr. 2013.
- [12] W. Lu and J. Wei, "Dynamic visualization of evolutionary curricula model," in *Proc. Int. Conf. Educ. Inf. Technol.*, 2010, pp. 484–488.
- [13] G. Siemens, "Learning analytics: Envisioning a research discipline and a domain of practice," in *Proc. 2nd Int. Conf. Learn. Anal. Knowl. (LAK)*, 2012, pp. 4–8.
- [14] D. Gasevic, D. Djuric, and V. Devedzic, *Model Driven Architecture and Ontology Development*. Berlin, Germany: Springer, 2006.
- [15] Y.-L. Chi, "Ontology-based curriculum content sequencing system with semantic rules," *Expert Syst. Appl.*, vol. 36, no. 4, pp. 7838–7847, May 2009.
- [16] A. Ezen-Can, K. E. Boyer, S. Kellogg, and S. Booth, "Unsupervised modeling for understanding MOOC discussion forums: A learning analytics approach," in *Proc. 5th Int. Conf. Learn. Analytics Knowl.*, Mar. 2015, pp. 146–150.
- [17] L. Aroyo and D. Dicheva, "Authoring support in concept-based web information systems for educational applications," *Int. J. Continuing Eng. Educ. Life-long Learn.*, vol. 14, no. 3, pp. 297–312, 2004.
- [18] E. Duval, "Attention please!: Learning analytics for visualization and recommendation," in *Proc. 1st Int. Conf. Learn. Anal. Knowl. (LAK)*, 2011, pp. 9–17.
- [19] S. Ziebarth, N. Malzahn, and H. Hoppe, "Using data mining techniques to support the creation of competence ontologies," in *Proc. Conf. Artif. Intell. Educ.*, Amsterdam, The Netherlands, 2009, pp. 223–230.
- [20] D. Michail, J. Kinable, B. Naveh, and J. V. Sichi, "JGraphT—A Java library for graph data structures and algorithms," 2019, *arXiv:1904.08355*.
- [21] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics, Syst. Demonstrations*, 2014, pp. 55–60.
- [22] J. Guerra *et al.*, "When one textbook is not enough: Linking multiple textbooks using probabilistic topic models," in *Scaling up Learning for Sustained Impact*, D. Hernández-Leo, Ed. Berlin, Germany: Springer, 2013, pp. 125–138.
- [23] D. Blei, A. Ng, and M. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn.*, vol. 3, pp. 993–1022, Mar. 2003.
- [24] G. Angeli, M. J. J. Premkumar, and C. D. Manning, "Leveraging linguistic structure for open domain information extraction," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process. (Long Papers)*, vol. 1, 2015, pp. 344–354.

- [25] Stanford University. (2016). *Stanford University Protégé*. [Online]. Available: <https://protege.stanford.edu/>
- [26] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
- [27] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1–62.
- [28] N. Azam and J. Yao, "Comparison of term frequency and document frequency based feature selection metrics in text categorization," *Expert Syst. Appl.*, vol. 39, no. 5, pp. 4760–4768, 2012.
- [29] R. Řehůřek and P. Sojka, "Software framework for topic modelling with large corpora," in *Proc. Workshop New Challenges NLP Frameworks (LREC)*, Malta, 2010, pp. 45–50.
- [30] F. Chollet. (2015). *Keras*. [Online]. Available: <https://github.com/fchollet/keras>
- [31] B. Shneiderman, "The eyes have it: A task by data type taxonomy for information visualizations," in *Proc. IEEE Symp. Vis. Lang.*, Jan. 1996, pp. 364–371.
- [32] T. A. Gavrilova and I. A. Leshcheva, "Ontology design and individual cognitive peculiarities: A pilot study," *Expert Syst. Appl.*, vol. 42, no. 8, pp. 3883–3892, May 2015.



Angel Fiallos Ordóñez (Member, IEEE) received the B.S. degree in information systems from the Universidad de Especialidades Espíritu Santo, Ecuador, the master's degree in administration from the University of Guayaquil, and the Ph.D. degree in applied computer science from the Escuela Superior Politécnica del Litoral, Ecuador. He has taught and conducted research in the fields of learning analytics and educational technology. He is currently a senior consultant in data science, big data, social media analytics, and education technologies.



Katherine Chiluiza García (Member, IEEE) received the B.Sc. degree in computer engineering from ESPOL, the M.Sc. degree in engineering from The University of New Orleans, and the Ph.D. degree in pedagogical sciences from Ghent University, Belgium. She started her research in teacher training and educational technology applied to elementary and middle school. She then moved on to human–computer interaction applications in higher education. She is currently a Professor with the Faculty of Electrical Engineering and Computer Science, Escuela Superior Politécnica del Litoral (ESPOL). She is also the Dean of the Faculty of Electrical Engineering and Computer Science, ESPOL.



Xavier Ochoa Chehab (Member, IEEE) received the B.S. degree in computer science from the Escuela Superior Politécnica del Litoral, Ecuador, the M.Sc. degree in applied computer science from Vrije Universiteit Brussels, Belgium, and the Ph.D. degree in computer science from the University of Leuven, Belgium. He is currently an Assistant Professor of learning analytics with the Department of Management, Leadership, and Technology, Steinhardt School of Culture, Education, and Human Development (NYU). He is also a member of the Learning Analytics Research Network (LEARN) at NYU. He is the Vice President of the Society for Learning Analytics Research (SoLAR) and the Editor-in-Chief of the *Journal of Learning Analytics*.