Received February 22, 2019, accepted March 28, 2019, date of publication April 11, 2019, date of current version April 22, 2019.
Digital Object Identifier 10.1109/ACCESS.2019.2910079

A Combined Approach for Eliciting Relationships for Educational Ontologies Using General-Purpose Knowledge Bases

ANGEL CONDE[1], MIKEL LARRAÑAGA[2], (Member, IEEE), ANA ARRUARTE[2], AND JON A. ELORRIAGA[2], (Member, IEEE)
[1]IK4-Ikerlan, 20500 Arrasate, Spain
[2]Department of Computer Languages and Systems, University of the Basque Country UPV/EHU, 20018 Donostia, Spain
Corresponding author: Ana Arruarte (a.arruarte@ ehu.eus)

This work was supported by the Basque Government through ADIAN under Grant IT980-16.

ABSTRACT Educational Ontologies describe learning domains by specifying the topics to be learned together with the relationships among the topics. Any technology supported learning system could take advantage of an educational ontology in the process of guiding students during their learning processes. This paper presents LiReWi, a system for the elicitation of relationships for educational ontologies from electronic textbooks. LiReWi combines grammar-based, co-occurrence-based and taxonomy-based methods together with several knowledge bases, such as Wikipedia, WordNet, WikiTaxonomy, WibiTaxonomy, and WikiRelations to elicit isA, partOf, prerequisite and pedagogicallyClose relationships between learning domain topics. LiReWi performs a three-step procedure to fulfill its task: first, all the topics are mapped to the diverse knowledge bases that will be used to identify the relationships; then, several relationship extractors, each using a different approach, are concurrently run to elicit candidate relationships; and, finally, the results are combined and filtered to obtain the final set of pedagogical relationships. LiReWi is designed following a modular approach that enables the inclusion of new relationship extractors. The evaluation conducted to validate the proposal is also reported in this paper.

INDEX TERMS Technology supported learning systems, domain module, educational ontologies, relationship extraction, knowledge bases.

I. INTRODUCTION

Content authoring for any Technology Supported Learning System (TSLS) is a time and effort-consuming task that involves, among other factors, instructional designers and knowledge engineers. The idea of lightening and facilitating the development of TSLS, and the construction of the Domain Module in particular, is not new. The Domain Module is considered the core of any TSLS, as it represents the knowledge about the subject matter to be communicated to the learner. During the 90s, a great effort was made in the development of shells or authoring tools, trying to reduce development costs, and allowing practicing educators to become more involved in the creation of TSLSs and their components [1]. But the results in the area were not as good as expected. Again, most tools were oriented to computer-skilled users or knowledge engineers and, therefore, they became too complicated for average teachers, who may give up on the development of their own Domain Modules.

The ontological approach is one of the formalisms proposed in the literature for the representation of the Domain Module [2]. In the proposal presented in this paper, a Learning Domain Ontology (LDO) describes the topics to be mastered, along with the pedagogical relationships between the topics. The LDO can be leveraged for pedagogical purposes in educational contexts. A TSLS can take advantage of the LDO, not only in the process of guiding students during their learning sessions –traditional ITSs and blended systems– but also by offering mechanisms to guide students during informal learning scenarios. The advances in the area of Natural Language Processing and Machine Learning have fairly consolidated the field of Ontology Learning, which is concerned with the development of methods that can induce relevant ontological knowledge from data. The field of Ontology

The associate editor coordinating the review of this manuscript and approving it for publication was Bora Onat.

Learning can now look back on more than ten years of intense research [3].

In particular, the paper presents LiReWi, a system that combines different knowledge bases and methods for the acquisition of pedagogical relationships between topics from textbooks. Specifically, LiReWi extracts *isA, partOf, prerequisite* and *pedagogicallyClose* relationships. In the last few years, the authors have been working on the automatization of the Domain Module construction. In a previous work, LiTeWi, a tool that combines unsupervised term extraction and entity linking methods to identify the topics to be mastered by the students in a particular textbook, was presented [4]. This new work focuses on the elicitation of relationships between the topics, which may be used by the TSLSs to plan and schedule the learning sessions; in fact, relationships have been traditionally used in TSLSs to guide students during their learning processes [5]. The combination of LiTeWi and LiReWi will contribute automatic knowledge extraction from textbooks framing the resulting LDO in educational contexts. This automatization will lighten the workload of teachers when preparing their courses for TSLSs. The entire process of the Learning Domain Ontology elicitation corresponds to the work done by Conde in his doctoral thesis [6]. The extraction of topics is exhaustively described in [4] and, as mentioned above, this time the authors focus on deeply describing the extraction of relationships.

LiReWi is a modular system based on individual relationship extractors that are combined to obtain more accurate results. This modularity allows new extractors to be integrated.

Both LiTeWi and LiReWi are intended to be domain independent, i.e., they must be applicable in any knowledge area. Therefore, any domain specific resources have been discarded and only general-purpose knowledge bases (e.g., Wikipedia, Wordnet…) for the knowledge elicitation are used. To our knowledge, there is no standard dataset to evaluate the generation of educational ontologies. So, two textbooks on two domains, *Object-oriented programming* and *Astronomy*, have been used to train and evaluate LiReWi, respectively. The evaluation combines gold standard and expert validation, two of the most common used evaluation approaches on ontology learning [7].

The manuscript has been structured as follows. First, a review of the use of ontologies for education is presented along with a review of the extraction of relationships in the literature, in general, and in educational contexts, in particular. Next, LiReWi is described. Then, the experiment conducted to evaluate LiReWi is presented. Finally, the conclusions and some future work lines are depicted.

II. RELATED WORK

This section briefly describes the two main research fields involved in the work presented throughout this paper: Ontologies in Education and the Elicitation of relationships for educational purposes.

A. ONTOLOGIES IN EDUCATION

In times when ontologies have been adopted in many research communities as a way to share, reuse, and process domain knowledge, the Technology Supported Learning Systems research community is not an exception. In this community, ontologies present new opportunities, as they provide great potential by allowing the sharing and reusing of information across learning systems and enabling personalized learner support. The use of ontological engineering, which aims at providing a basis for building models of all things in which computer science is interested [8], was proposed by Mizoguchi and Bourdeau to overcome common problems in the Artificial Intelligence in Education area [9]. Mizoguchi and Bourdeau argue that the sharing or reusing of knowledge and components could benefit from the use of ontology-based architectures and appropriate educational ontologies. More recently, Mizoguchi and Bourdeau presented the trends and perspectives in ontology engineering for Artificial Intelligence in Education and discussed the achievements obtained in the last 10 years [10]. Other authors, e.g., Jensen [11], focused their attention on Semantic Web Technologies and reviewed the fraction of the research that has been done regarding the Semantic Web and its derivative technologies in the educational sphere.

Dicheva *et al.* presented one of the first overviews of ontologies for education [12]. They collect and classify the information available in the field and build the Ontologies for the Education O4E Web Portal. However, what do researchers of educational communities understand by ontology? Although there is no consensus of what it refers to, it can be stated that Educational Ontologies refer to ontologies aimed at being used with educational purposes inside a TSLS. Fok and Ip define an Educational Ontology as an ontology that can help to retrieve, organise, and recommend educational resources for personalized learning [13]. The idea behind an Educational Ontology is that it can be reused by other learning systems with a wide range of teaching/learning methodologies.

Regarding the use of ontologies in the area of TSLSs, they have been mainly used as a means to represent the Domain Module [14]–[25], as a mechanism to describe instructional theories [26], or to build reusable and scrutable student models [27], [28].

The construction of ontologies and their population, with instantiations of both topics and relationships, has been commonly called Ontology Learning [3]. Ontology Learning refers to the application of a set of methods and techniques to enable the (semi-) automatic population of ontologies or the construction of ontologies from scratch from diverse information sources [29]–[36]. Although most of the projects in the area are aimed at extending and populating general purpose ontologies such as WordNet [37], there is a lack of initiatives which focus on the learning of ontologies from scratch, especially in the case of Educational Ontologies. Kaya and Altun present a review regarding the ontologies in educational domains and point out the challenges and difficulties that their development process implies [38].

B. ELICITATION OF RELATIONSHIPS FOR EDUCATIONAL PURPOSES

In recent years, there has been a lot of focus on ontology learning, including the elicitation of relationships. Most efforts have addressed the extraction of taxonomic relationships but there are some approaches that have considered a larger set of relationships. Next, a review of different approaches is presented. The review includes some general-purpose approaches and those more focused on the construction of Educational Ontologies. The review mainly concentrates on those relationships which have been traditionally used in Domain Modules for TSLSs.

Ponzetto and Strube derived a taxonomy from the *Wikipedia* category system, the *WikiTaxonomy* [39]. In order to build up such a taxonomy, they defined an algorithm for the elicitation of *isA* relationships from the *Wikipedia* categories. Syntactic patterns are used on those categories to infer the relationships. For instance, the head of the category allows us to identify that a *British computer scientist* isA *Computer scientist*. The algorithm uses additional patterns, such as Hearst patterns [40], for the extraction of relationships. Finally, inference-based methods are applied to propagate the relationships identified in the previous step through the hierarchy, considering both multiple inheritance and transitivity. For example, as a ''Leek'' is known to be an ''Edible Plant'', and an ''Edible Plant'' is a ''Plant'', these methods would infer that *Leek* isA *Plant*, taking advantage of the transitivity.

KOG [41], which stands for Kylin Ontology Generator, is an autonomous system that builds an ontology by combining the information in *Wikipedia* infoboxes with WordNet, using Machine Learning techniques. Infoboxes are tables containing meaningful information about the article in the form of attribute-value pairs in *Wikipedia* articles. Each infobox is considered a class and all its attribute-value pairs are represented by class slots. KOG uses Machine Learning, in particular a joint inference approach based on Markov logic, to infer *isA* relationships between pairs of classes. To this end, it uses several features such as: 1) similarity measures between the classes, 2) class-name string inclusion, 3) category tags, 4) whether or not the class-names appear in Hearst patterns in Google queries, and 5) *WordNet* mappings, which takes advantage of the defined hypernyms. For example, KOG would infer that *Earth* isA *Planet* from the ''*planets such as Earth''* text fragment.

Flati and Vanella presented an approach for the automatic creation of an integrated taxonomy of *Wikipedia* articles/categories, i.e., a taxonomy of Wikipedia articles aligned to a taxonomy of categories [42]. The main idea of integrating both article and category taxonomies is that this leads to a finer-grained taxonomy with higher coverage, the *WibiTaxonomy*. This enhanced taxonomy is built in three phases. First, an initial article taxonomy is built parsing articles to extract the textual definitions which the articles include. The hierarchical taxonomy is generated by identifying hypernym relationships between the extracted definitions. Secondly, the system iterates over each extracted hypernym in the article taxonomy. Using the category links of each article, the category taxonomy is inferred. In each iteration of this algorithm, the links of the article taxonomy are used to discover category hypernyms, and these are used to discover more hypernyms. Finally, the category taxonomy is improved using structural heuristics, which will provide broader coverage to the taxonomy. For example, given an uncovered category ''c'' which does not have any connection to ''c0'', ''c0'' being the only direct super-category of ''c'' in *Wikipedia*, a link between them will be inferred.

MENTA [43] is a multilingual taxonomy derived from *Wikipedia*. Unlike previous approaches, it was also built by analyzing Wikipedia for languages other than just English. Therefore, it includes local information not covered by the English Wikipedia, examples of this local information being places, people, local laws, etc. The information is organized into a coherent taxonomy using both Wikipedia and WordNet structures as references. To build the MENTA taxonomy, the following procedure was carried out. First, for each article, the parent categories are extracted. In addition, a small gloss, usually found in the first paragraph of the article, and the labels that are associated to the article are also gathered. The next step entails finding connections between all the gathered articles using several linking functions. For example, the cross lingual linker will connect two articles where a link between articles of different *Wikipedias* exists. Another example is *WordNet* hypernyms, which defines connections between articles when the articles have a connection in *WordNet*. Finally, the last step involves aggregating the taxonomic information gathered and applying different filters to produce a clean taxonomy and make it even more consistent. For example, a filter that removes cycles of subclass relationships given that all entities in the cycle are equivalent is used to clean the taxonomy. Educational textbooks have also been used to extract taxonomic relationships. For example, in [44] an approach for the topic hierarchy extraction from textbooks that uses the document structure and *Wikipedia* is presented.

Nastase and Strube mined the *Wikipedia* category network to extract different types of relationships including *isA* and *partOf* relationships [45]. The novelty of their approach is that they focus specifically on using category names to extract the information and propagate this knowledge towards the articles connected to these categories. The process followed by the authors in order to extract relationships entails: 1) identifying the domain constituent of category names; for example, Chairmen of the county councils of Norway has three constituents: chairmen, county councils and Norway, the dominating one being chairmen, 2) extracting relationships from all the articles below the processed category, using syntactic patterns applied to the category name that will infer the relationship encoded in the category name. Figure 1 shows

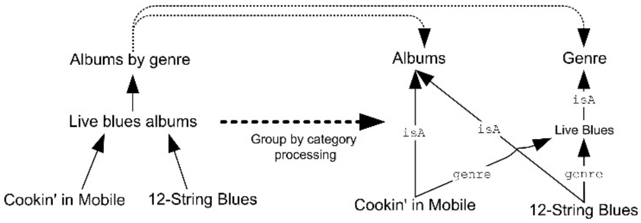

FIGURE 1. Example of the knowledge extracted from WikiRelations.

a complete example of the process. Albums by genre and Live blues albums categories are processed and the corresponding relationships are inferred connecting the articles *Cookin' in Mobile* and *12-String Blues* with the corresponding extracted relationships (*isA* and *genre*).

Arnold and Rahm proposed and evaluated an approach to extract semantic topic relations from un-structured Wikipedia articles [46]. The approach focuses on the analysis of the definition sentence of *Wikipedia* articles and uses finite state machines to extract semantic relation patterns and their operands to discover semantic relations such as *isA, partOf, hasA* or *Equal*.

In addition to the classical taxonomic (*isA*) and non-taxonomic (*partOf*) relationships, Educational Ontologies also can take advantage of more specific pedagogical relationships such as *prerequisite* relationships.

Roy exposes the automatic extraction of pedagogic metadata for document understanding [47]. As regards the pedagogical relationships, she deals with the identification of prerequisite topics to understand the document. To identify these prerequisite topics, she works at sentence level and uses a shallow parsing approach to identify the defined topic list –topic defined/explained in the sentence– and the used topic list –topics used to define/explain the defined topic. The defined topic will constitute the learning outcome and all the remaining noun phrases in the document, i.e., the used topic list, will be considered as a prerequisite to understand the document. Any topic included in the used topic list which is also listed in a defined topic list will be removed from the used topic list and, as a consequence, not considered as a prerequisite to understand the document.

Liang *et al.* proposed a metric to determine *prerequisite* relationships between pairs of topics [48]. To determine if a *prerequisite* relationship exists between two topics A and B, the metric computes the difference of the weighted references from the topics related to topic A to topic B, and the references from topics related to topic B to topic A. If the score goes beyond a threshold, a *prerequisite* relationship exists. The authors used Wikipedia to look for the references and determine the weights of the topics. They reported an average accuracy of 61%.

The identification of *prerequisite* relationships among Learning Objects has been addressed in [49], [50]. Learning Objects are associated to Wikipedia pages (topics), and their dependency is obtained using the classification of those topics supported by Wikipedia Miner [51]. Later, the same authors use a machine learning based approach to identify *prerequisite* relationships. The model used for the classification is learned by considering a training set of instances that usually consists of pairs of Learning Objects with known *prerequisite* relationships [52].

DOMSortze [53], a system that uses natural language processing techniques, heuristic reasoning, and ontologies for the semiautomatic construction of the Domain Module from electronic textbooks was able to extract 4 types of relationships: *isA, partOf, prerequisite* and *next*.

C. CONCLUSION

To sum up, topics and relationships between topics are the main elements in Ontologies, in general, and in Educational Ontologies, in particular. While some types of relationships are generic and not specific to pedagogical vocabularies, e.g., isA or partOf, other types of relationships, e.g., prerequisite or next, are more related to educational contexts. Anyway, the relevance of these relationships for Instructional Design purposes is widely recognized [5] and they have been traditionally used in TSLSs, either individually or in combination, to guide students during their learning processes. Those relationships can help, either the TSLS or the students that are learning on their own, to determine what topics should be studied together or the sequence of the topics to be learnt, and even to personalize the learning path according to their preferences, improving the learning process.

Assuming that ontologies can play a major role in learning applications, the next step is to solve "the quandary of the cognizance acquisition bottleneckness. A semiautomatic approach must be used to develop a domain ontology for retrieval of static and dynamic content as it reduces the cost effectively" [54].

III. LIREWI: A RELATIONSHIP EXTRACTOR FOR EDUCATIONAL ONTOLOGIES

Throughout this paper, the authors present an approach in which Ontology Learning is used to lighten the workload in the construction of Educational Ontologies from electronic textbooks. In particular, this work focuses on the elicitation of relationships among the topics of the ontology, which represent the topics to be mastered by the students during the learning process. To this end, LiReWi takes as input the topics among which the relationships must be inferred. In the work presented throughout this paper, the topics have been extracted from an electronic textbook using LiTeWi [4]. LiTeWi extracts the set of topics from the textbook along with a measure of the domain relatedness of each topic.

LiReWi relies on the approach proposed in [53] to represent the Domain Module: an educational ontology, the Learning Domain Ontology (LDO) describes the topics to be mastered, along with the pedagogical relationships between the topics, whilst a set of Learning Objects (LOs), called Learning Object Base (LOB), includes the didactic resources that can be used to learn each domain topic.

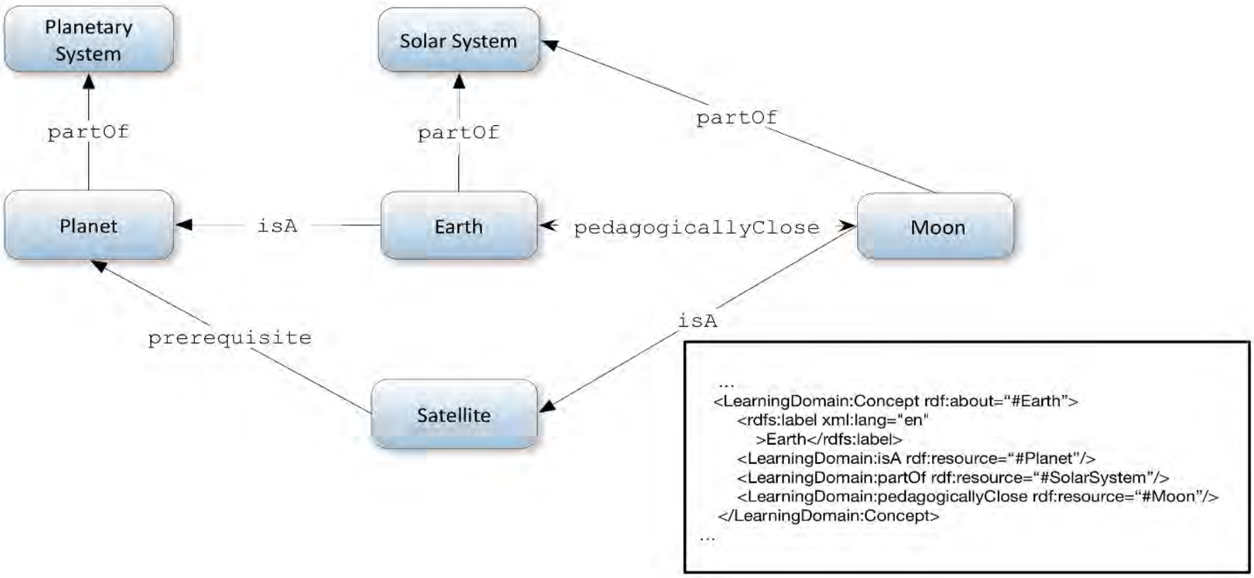

FIGURE 2. **A Learning domain ontology example.**

More specifically, the LDO comprises two kinds of pedagogical relationships, structural relationships —*isA* and *partOf*— and sequential relationships —*prerequisite* and *pedagogicallyClose*. Figure 2 shows a fragment of a LDO to illustrate the semantics of these relationships. The topics "Earth" and "Moon" are *partOf* the "Solar System", i.e., they are lower granularity elements that are constituents of the more general topic "Solar System". "Earth" is related to "Planet" by the *isA* relationship; in other words, "Earth" is a particular instance of the "Planet" topic. The *prerequisite* relationship between "Satellite" and "Planet" expresses that the latter should be learnt before attempting to learn "Satellite". Finally, the *pedagogicallyClose* relationship expressed between "Earth" and "Moon" shows that those topics are strongly related and they could be learnt at the same time. The LDO is formalized in OWL, as can be seen in the code fragment included in Figure 2.

The relationship extraction approaches described in the previous section mainly address the identification of structural relationships between the topics of an ontology. Although the final goal of the authors is to facilitate the development of Domain Modules for Technology Supported Learning Systems –Learning Domain Ontology (topics and relationships) and the Learning Objects Base–, this paper focuses on the extraction of pedagogic relationships for previously identified topics. LiReWi combines some of the approaches and sources of information described in the previous section to elicit the pedagogical relationships (*isA, partOf, pedagogicallyClose* and *prerequisite*) that will be used to build the Learning Domain Ontology (LDO). LiReWi is intended to be usable on documents of any domain. Thus, any domain-dependent technique has been discarded. To cope with the relationship extraction process, LiReWi requires that the electronic textbook is previously processed in order to extract the domain topics, to which end LiTeWi is used. Next, LiReWi elicits the pedagogical relationships between the topics that will be used to build the Educational Ontology.

To elicit the pedagogical relationships between the domain topics, LiReWi follows the following procedure (see Figure 3). First, all the topics are mapped to the diverse knowledge bases (e.g., *Wikipedia*, *WordNet* and others derived from both) that will be used to identify the relationships. Then, several relationship extractors, each using a different approach, are concurrently run to elicit candidate relationships. Finally, the results are combined and filtered to obtain the final set of pedagogical relationships. In the next subsections, each step is described in more detail.

A. APPROACH

LiReWi uses different knowledge bases such as Wikipedia, WordNet, and knowledge bases derived from Wikipedia (WikiTaxonomy, WibiTaxonomy and WikiRelations), and a set of methods for relationship extraction, ranging from grammar-based to methods that mine relationships from the paths included in the taxonomies of those knowledge bases or co-occurrence based methods. In this work, only general-purpose knowledge bases have been used in order to keep LiReWi domain independent.

Every relationship extractor gathers a set of relationships and determines the confidence of the extracted relationships using a specific formula (presented below). Some extractors, e.g., path-based extractors, require threshold values to be defined before being used. The parameters used to compute the confidence of the extracted relationships must also be determined. Therefore, LiReWi needs to be tuned up before being used to extract the relationships from a document. Once the set-up process has been carried out, once only, LiReWi

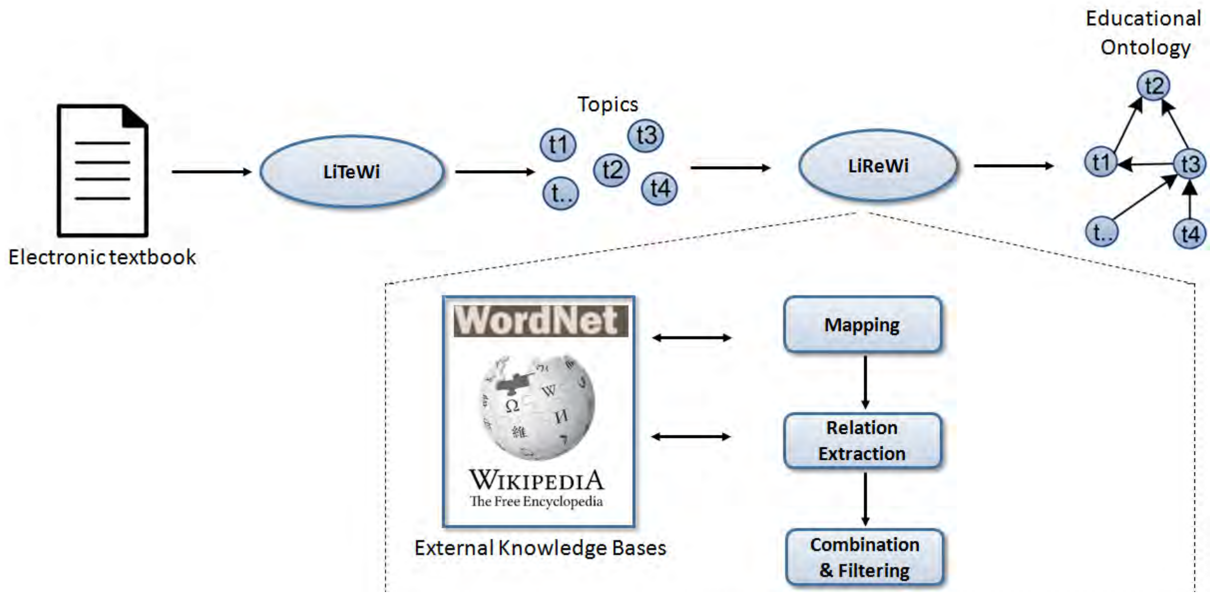

FIGURE 3. The general overview process.

will be ready to be used in production, that is, ready to extract relationships from any document.

In this work, a heuristic approach has been followed to tune up each extractor and determine both the thresholds and parameters of the formulas to compute the confidence scores. The thresholds have been inferred empirically during the tuning up of the system. The formulas that calculate the confidence/trust of the system in the correctness of the extracted relationships contain specific parameters that will be detailed while describing each extractor (presented below). However, there is a shared parameter called "base confidence" that depends on each extractor and represents the trust of the extractor itself, and therefore it is the starting point for calculating the confidence of the relationships identified by the extractor. The Principles of Object-Oriented Programming [55] text-book, which consists of 67 pages and over 30,000 words, has been used in the tune-up process. To optimize the thresholds and parameter values, LiReWi was tested on the book. Recall and precision have been considered to this end; the precision has been prioritized over the recall while determining the appropriate values, as we considered discarding wrong relationships more overwhelming than defining missing relationships. Figure 4 shows the results of the tune-up phase that allowed the optimal path-length of the extractors to be determined.

Once LiReWi was tuned up, its performance was evaluated on the Introduction to Astronomy [56] textbook. This book consists of 150 pages of plain text and over 110,000 words.

B. MAPPING TOPICS TO KNOWLEDGE BASES

To extract pedagogical relationships between topics, LiReWi uses, in addition to shallow parsing techniques, several knowledge bases such as *Wikipedia*, *WordNet*, *WikiTaxonomy*, *WibiTaxonomy* and *WikiRelations*. To this end, it is necessary to map every topic to its corresponding entries in those knowledge bases. The topics identified by LiTeWi are already disambiguated and mapped to *Wikipedia* articles. As *WikiTaxonomy*, *WikiRelations* and *WibiTaxonomy* are based on *Wikipedia*, the topics are already mapped to these knowledge bases. However, to be able to use *WordNet*, the topics must still be mapped to *WordNet* entries. *WordNet* organizes words (nouns, verbs, adjectives and adverbs) into cognitive synonyms called synsets. Each synset refers to a distinct topic that can be referred to using different forms. Navigli & Ponzetto [57] and Fernando [58] faced a similar problem and defined the mappings or equivalences between *Wikipedia* articles and *Wordnet* synsets.

Aiming at carrying out an efficient mapping process, the mapper looks first for the appropriate equivalent synset in those mappings identified in BabelNet project [57], and also in those mappings discovered by Fernando [58]. If the same synset is found in both cases, the mapper assumes that there are no ambiguity problems and returns the identified synset. Otherwise, a disambiguation process is carried out to identify which of the candidate synsets is the appropriate one. Bearing this in mind, a Page Rank Mapping Disambiguation step is carried out using UKB [59], a tool for Word Sense Disambiguation and for determining lexical similarity using a pre-existing knowledge base such as *Wikipedia* or *WordNet*. UKB requires a context to fulfil its goal. The context is obtained from the topics extracted by LiTeWi along with the domain relatedness LiTeWi assigned to each of them. The topics with highest domain relatedness score and with a unique sense in *WordNet* constitute the context that allows the synset for the topic to be chosen. For example, when mapping the topic syntax In WordNet, which is related to

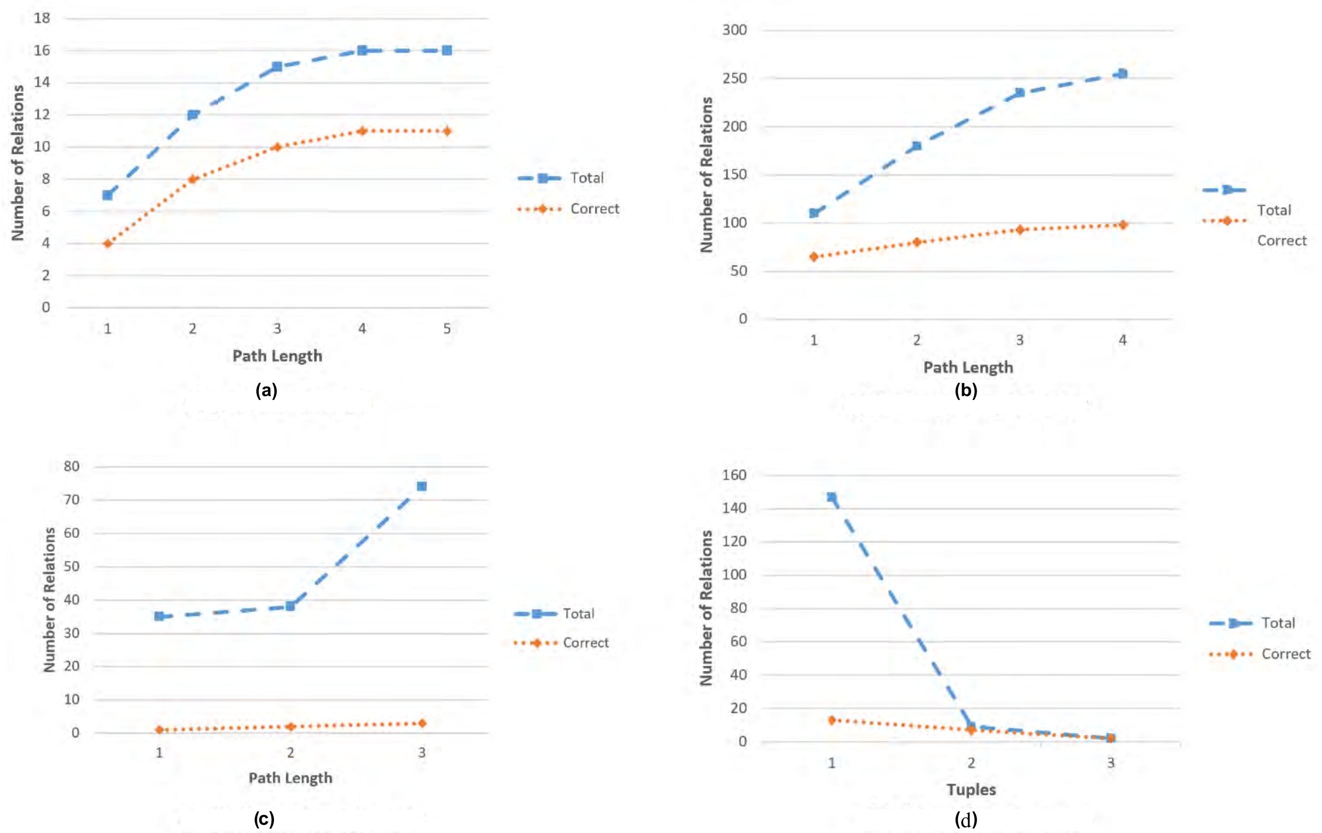

FIGURE 4. **Total and Correct Relationships using different paths lengths for WordNet Extractor, WibiTaxonomy Extractor, WikiTaxonomy Extractor, and WikiRelations Extractor.**

Computer Science, the mapped synsets returned by [57] and Fernando [58] are different. Therefore, the Page Rank Mapping Disambiguation step is carried out to determine the final synset of *syntax* in *WordNet*. The context used in the example entails topics such as *Programming*, *Menu Bar* and *Java*. The Page Rank Mapping Disambiguation mechanism could select a different synsets from those proposed by [57] and Fernando [58].

C. A HYBRID APPROACH TO RELATIONSHIP EXTRACTION

To extract the pedagogical relationships, LiReWi exploits different sources of information and techniques that include taxonomy-based, grammar-based, and co-occurrence-based methods. In Figure 5, the relationship extractors used in LiReWi, along with the type of extracted relationships, and the knowledge bases used to this end by each of them are shown. Each of the extractors identifies a set of candidate relationships along with their confidence, i.e., the trust the extractor has in that relationship being correct. Taxonomy-based methods –WordNet Extractor, WibiTaxonomy Extractor and WikiTaxonomy Extractor– use (1) for calculating the confidence while the other methods employ their own formula. This information –the candidate relationships and their confidence– is used in the Relationship Combination and Filtering step (presented below). Some of the extractors rely on empirically defined thresholds to fulfill their task. Next, each extractor is described.

1) WORDNET EXTRACTOR

WordNet [37] can be considered as a huge graph of topics connected by semantic relationships. LiReWi uses WordNet to infer relationships from the hypernym relationships (isA) and meronym relationships (*partOf*) between the synsets. The processed topics are those identified by LiTeWi. The procedure of extracting relationships with WordNet is described next. First, a Deep-first Search (DFS) is carried out for each input topic to find the shortest upwards path between the topic and other important topics in WordNet. This search is done to gain information from the transitivity of the relationships between the topics. To prevent WordNet from eliciting relationships from paths which are too long, the maximum length of the path can be set. By default, the maximum length is restricted to 3 levels of distance as it produces a balanced output in terms of the number of identified relationships and their correctness and the computational load. This path length threshold has been empirically determined in the setting up of the system.

Finally, the system determines the confidence of the relationship considering the length of the path. The shorter the path, the greater the confidence of the relationship is.

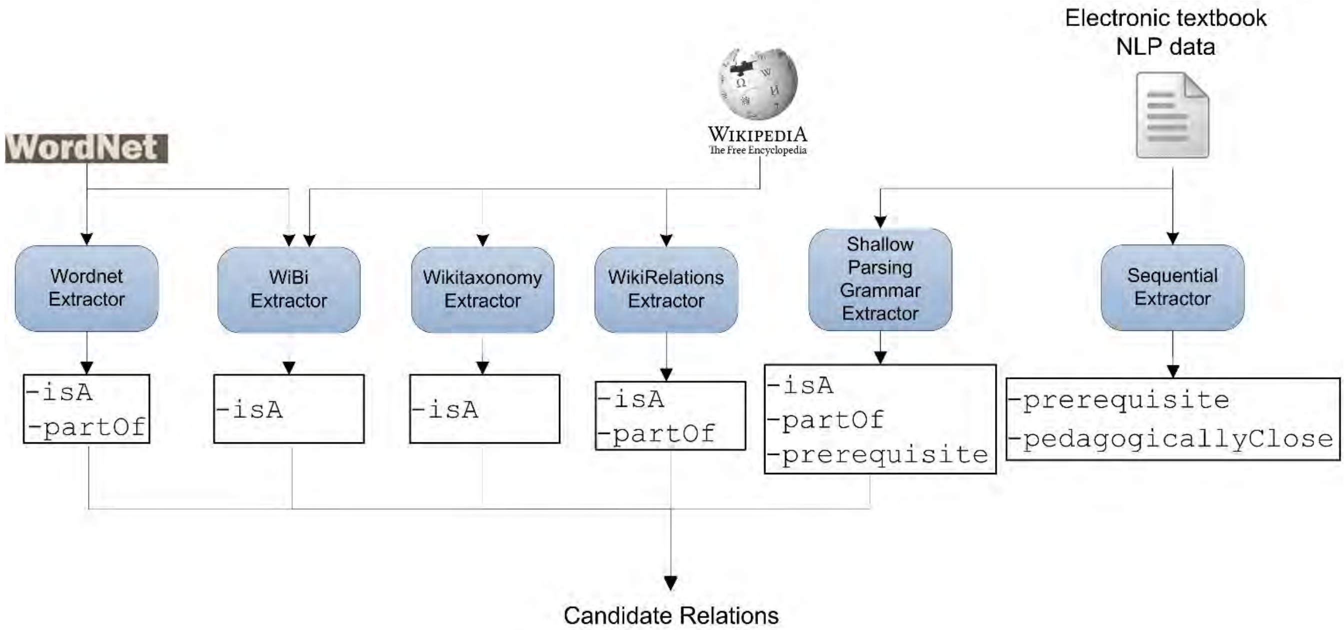

FIGURE 5. Relationship Extrators used in LiReWi.

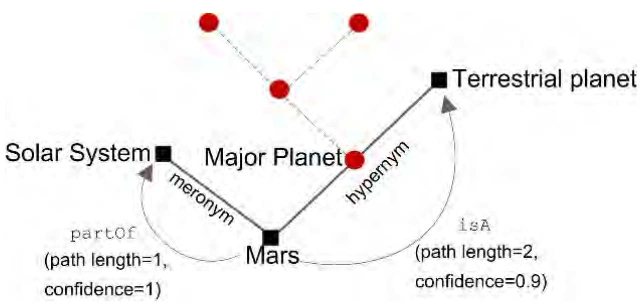

FIGURE 6. Example of the application of WordNet Extractor.

TABLE 1. Relationships extracted by WordNet Extractor.

Relationship	*Confidence*
Earth partOf Solar system	1
Earth isA Planet	1
Jupiter partOf Solar System	1
Saturn partOf Heliosphere	0.9
Sun partOf Heliosphere	0.9
Jupiter parOf Milky Way	0.8
Saturn partOf Milky Way	0.8

Equation (1) is used to calculate the confidence,

$$Confidence = \max(0, b - (0.1 \times (p - 1))) \tag{1}$$

where b is the base confidence, which is 1 for WordNet Extractor, and p represents the path length.

An example of the application of WordNet Extractor can be seen Figure 6. In the figure, the nodes represent the *WordNet* synsets that are connected with each other via semantic relationships such as hypernym relationships or meronym relationships. The rectangles represent topics that are mapped to *WordNet* synsets, whereas the circles represent *WordNet* synsets not mapped to input topics. When a path including only relationships of the same kind between two topics and which is also shorter than the maximum length is found, a pedagogical relationship is defined. In the figure, it can be observed that "Mars" and "Terrestrial Planet" are linked by a hypernym relationship-based path. Therefore, isA relationship is inferred between those topics. On the other hand, "Mars" and "Solar System" are related through meronym relationships. In this case, *partOf* is inferred between those topics. The confidence of the *Mars* partOf *Solar System* relationship is calculated using only the base confidence parameter, which is 1 in WordNet Extractor. *Mars* isA *Terrestrial planet* relationship confidence is calculated with a path length of 2, therefore, the resulting confidence is 0.9.

Examples of relationships extracted by the WordNet Extractor with their assigned confidence from the *Introduction to Astronomy* textbook are shown in Table 1.

2) WIBITAXONOMY EXTRACTOR

WibiTaxonomy [42] is a knowledge base that comprises two interconnected taxonomies, the *Wikipedia* article taxonomy and the category taxonomy. Extracting relationships from *WibiTaxonomy* entails two steps. First, each topic is mapped to the taxonomy of articles using the mapped *Wikipedia* article of each topic. Then, each topic is also mapped to the taxonomy of categories using the parent categories of the topic

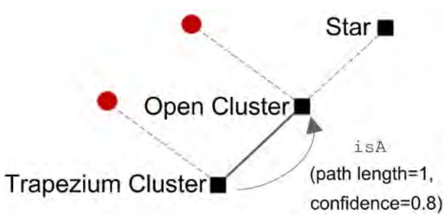

FIGURE 7. Example of the application of WibiTaxonomy Extractor.

TABLE 2. Relationships extracted by WibiTaxonomy Extractor.

Relationship	*Confidence*
Earth's rotation `isA` Rotation	0.8
Trapezium Cluster `isA` Open cluster	0.8
Nix (moon) `isA` Natural satellite	0.8
Tau neutrino `isA` Elementary particle	0.8
Earth's rotation `isA` Motion (physics)	0.6
Trapezium Cluster `isA` Star	0.6

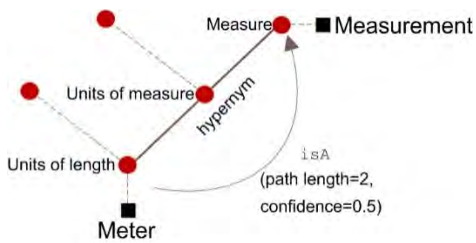

FIGURE 8. Example of the application of WikiTaxonomy Extractor.

TABLE 3. Relationships extracted by WikiTaxonomy Extractor.

Relationship	*Confidence*
Equinox `isA` Astrometry	0.6
Optical aberration `isA` Optics	0.6
Electron `isA` Matter	0.5
Hydrogen `isA` Matter	0.5
International Atomic Time `isA` Measurement	0.5

in *Wikipedia*. In the second step, using a similar procedure to that used by the WordNet Extractor, this extractor looks for paths of a limited length to infer the relationships from the taxonomies of both articles and categories. The confidence of each extracted relationship is also adjusted using the (1) with a 0.8 base confidence.

In Figure 7, a graphical example of a relationship derived using the WibiTaxonomy Extractor can be seen. This example shows that the extractor inferred that *Trapezium Cluster* isA *Open Cluster* with 1-step path and 0.8 confidence.

In Table 2, some examples of extracted relationships inferred by WibiTaxonomy Extractor can be seen with their assigned confidence from the *Introduction to Astronomy* textbook.

3) WIKITAXONOMY EXTRACTOR

The *WikiTaxonomy* [39] is a huge taxonomy derived from the *Wikipedia* category system where all the links between categories are represented by isA relationships. Moreover, *WikiTaxonomy* contains a dictionary where the articles are mapped to the corresponding category entries in the taxonomy. The *WikiTaxonomy* extractor carries out the following procedure to elicit the taxonomic relationships between topics. First, each topic is mapped to its corresponding *WikiTaxonomy* categories. Then, a DFS is carried out to find the shortest upwards path between the topics considering the categories in the *WikiTaxonomy*. Once again, the maximum admissible path length was configured. This has been empirically determined making a set of tests like in WordNet and WibiTaxonomy Extractors. The tests show that the optimal results are obtained using 1 level as a limit.

Moreover, the confidence on the relationship is likewise computed considering the path distance using (1) with the previously empirically determined base confidence (0.8). Figure 8 shows an example in which the WikiTaxonomy Extractor identifies the isA relationship between "Astronomical Unit" and "Measurement" with 0.5 confidence. In this figure, the squares represent the input topics and the circles represent the WikiTaxonomy network section where a path of length 2 between both topics has been found.

Some examples of extracted relationships using WikiTaxonomy Extractor with their assigned confidence from the Introduction to Astronomy textbook are shown in Table 3. The relationships below the dashed line correspond to candidates that did not achieve the empirically determined threshold.

4) WIKIRELATION EXTRACTOR

The *WikiRelations* knowledge base [45] comprises a big set of tuples defining the relationships between *Wikipedia* categories. It contains several kinds of relationships. In this work, only the subset of tuples containing *isA* or *partOf* relationships has been employed. The WikiRelations Extractor carries out the procedure shown in Figure 9. First, for each topic, it gets the corresponding *Wikipedia* article to extract parent categories associated with that article and map them to one or more *WikiRelations* tuples. In the example, "Light" and "Electromagnetic radiation" are each associated with categories that appear in two *WikiRelations* tuples. Then, it filters the tuples where topics are mapped. Whenever a tuple contains two of the input topics, a relationship between those topics is inferred.

As tuples containing the same categories can be found more than once in WikiRelations, this fact is considered to calculate the confidence of each relationship. In the example, it can be seen that WikiRelations has inferred a partOf relation two times. So, the confidence of the relationship is adjusted accordingly using (2), where b represents the base confidence

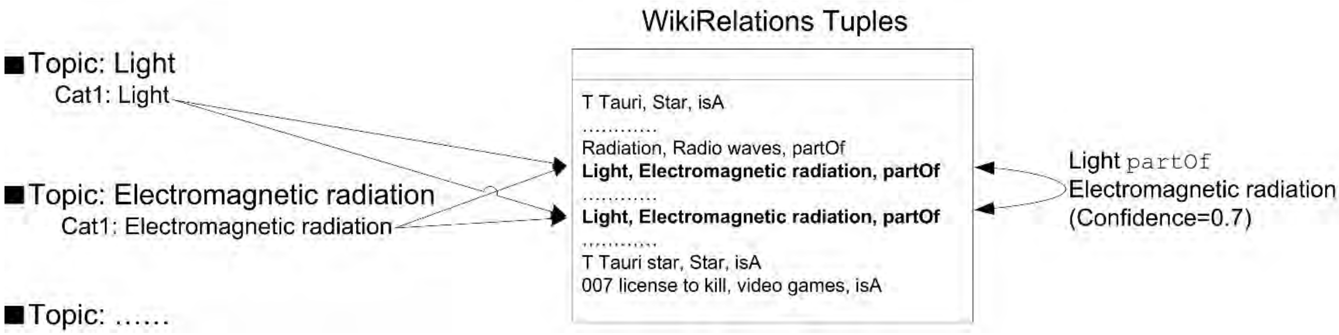

FIGURE 9. Example of the application of WikiRelations Extractor.

TABLE 4. Relationships extracted by WikiRelations Extractor.

Relationship	*Confidence*
Lithium `isA` Metal	1.0
Microwave `partOf` Electromagnetic spectrum	1.0
Earth `partOf` Solar System	1.0
Alpha Centaur `partOf` Centaury	1.0
Solar wind `partOf` Solar System	0.8

(0.6), and n is the number of tuples.

$$\textbf{Confidence} = \textbf{min}(b + (\textbf{0.1} \times n)) \tag{2}$$

The confidence threshold was empirically determined in the tune-up phase of LiTeWi (see Figure 4(d)). In this case the threshold filters out those relationships that have been inferred only once (those that have only one tuple in WikiRelations).

In Table 4 some examples of relationships with their corresponding confidence are shown from the Introduction to Astronomy textbook.

5) SHADOW PARSING GRAMMAR EXTRACTOR

The Shallow Parsing Grammar Extractor can infer *isA, partOf* and *prerequisite* relationships applying a grammar on the *part-of-speech* information of the input textbook. Larrañaga *et al.* [53] defined a grammar for the extraction of pedagogical relationships applied to the Basque language. This grammar is applied to morphological information using the CG3 parser [60]. In this work, a similar grammar has been developed for English. The grammar consists of a set of rules that are triggered when the corresponding pattern is met. Some of those patterns are shown in Table 5.

TABLE 5. Examples of patterns for relationships extraction.

Pattern	*Example*
TOPIC +called|referred to as + TOPIC	Scientists believe that the **galaxy** referred to as **Milky Way** has over 100 billion stars.
TOPIC + TO BE + [det] + TOPIC	**Earth** is a **planet**.
TOPIC + consist + of + TOPIC	**Galaxy** consists of **stars**.
TOPIC + to be component(s) + of +[det] + @ONT-TOPIC	**Galaxies** are the main components of the **universe**.
TOPIC + of + [det] + TOPIC	The **movements** of the **planets**.

Next, the process followed by LiReWi to extract relationships using the Shallow Parsing Grammar is described (see Figure 10). First, the extractor identifies those sentences in which the input topics are referred. In addition, the topics being referred are annotated with the *part-of-speech*(POS) information of the sentence. As some of the input topics might subsume others, e.g., *sun-eclipse – eclipse,*the system resolves this situation by considering a simple matching algorithm where those compound terms have prevalence over the simple ones. The sentences containing more than one mention of input topics will be selected as they may suggest a relationship between the involved topics. Next, the shallow parsing grammar is applied to the sentences extracted in the previous step. Finally, taking into account the information of each triggered rule, specifically, the type of the relationship, the direction of the relationship and the topics that triggered the rule, a relationship is inferred between those topics, also obtaining the confidence of the triggered rule. The confidence of each rule was previously determined from its precision after testing it with a set of examples applied to the *Principles of Object-Oriented Programming* [55] textbook.

6) SEQUENTIAL EXTRACTOR

This extractor aims to obtain sequential relationships such as prerequisite and pedagogicallyClose. The Sequential Extractor uses the information contained in the processed textbook along with information gathered from *Wikipedia* to extract these kinds of relationships. In particular, it uses the co-occurrences of the topics within the sentences along with the *Wikipedia* link structure between articles. To use the information of the link structure between articles, this module

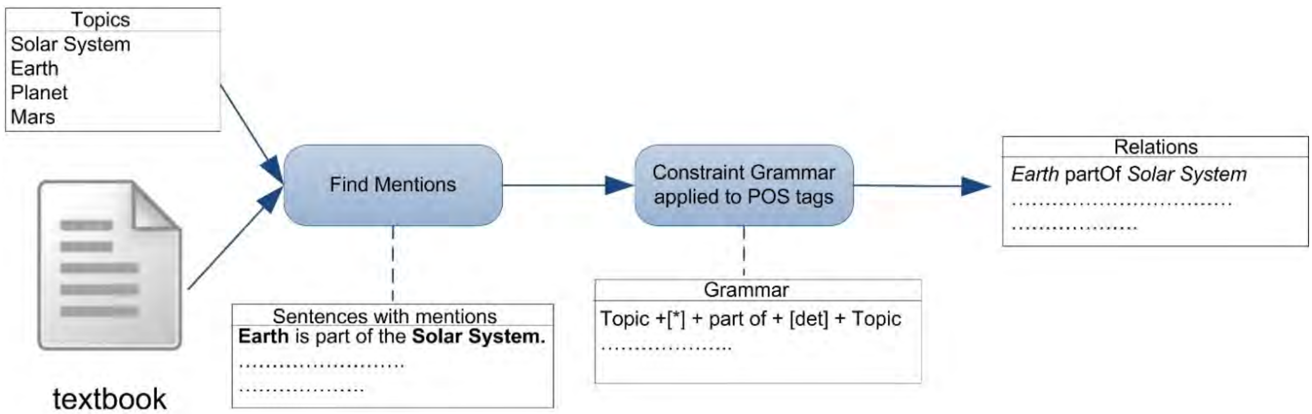

FIGURE 10. Example of the application of Shallow Parsing Grammar Extractor.

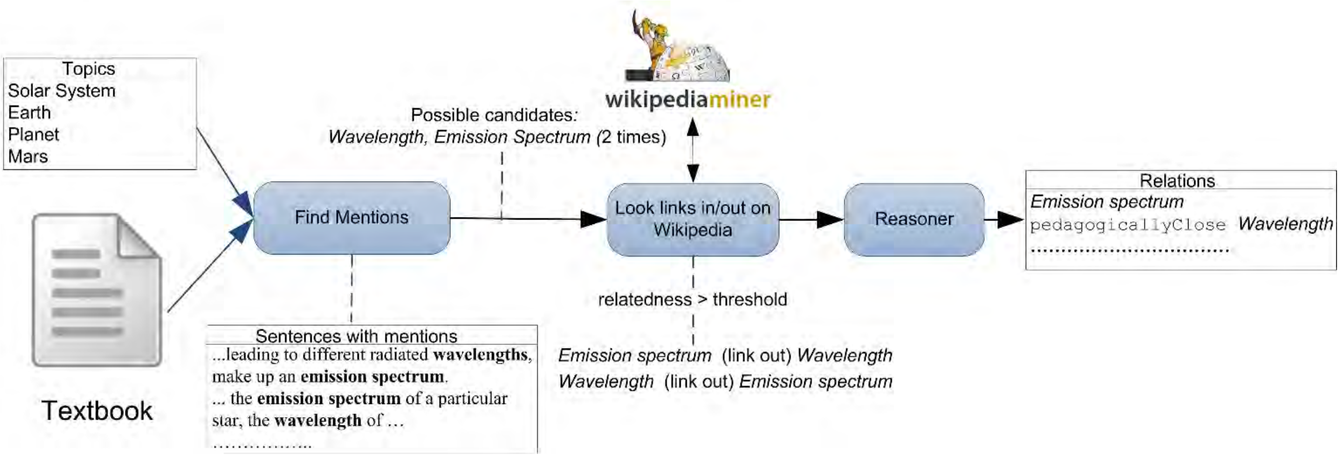

FIGURE 11. Example of the application of Sequential Extractor.

uses *Wikipedia Miner* [51]. Next, the procedure is described (see Figure 11).

First, as occurs in the Shallow Parsing Grammar Extractor, the extractor identifies the topics that are being referred in the text. Once again, the system applies a simple matching algorithm where the compound terms have prevalence over the simple ones. The output of this process is a list of sentences that contain mentions of the input topics. Next, for each of those sentences, a reference relationship is defined between each pair of topics appearing in the sentence if the first topic refers to the second. A topic is considered to refer to another if a link out from the first topic to the second exists in Wikipedia with a relatedness score beyond an empirically gathered threshold. LiReWi uses Wikipedia Miner to compute the relatedness score of two topics.

Finally, for each linked topic pair, a sequential relationship is inferred. If the links between both topics are balanced, i.e., the number of links from the first topic to the second is similar to the number of links from the second to the first, a pedagogicallyClose relationship between both topics is inferred. Otherwise, a prerequisite relationship is inferred from the topic with the highest number of outgoing links to the topic with the highest incoming links. Table 6 shows two examples in which a pedagogicallyClose and a prerequisite relationships are inferred using this procedure.

The confidence of extracted relationships is calculated using (3), where *b* is the base confidence (0.6), *top1m* is the number of links from the first topic, *top2m*is the number of links from the second topic and *low* is the threshold

TABLE 6. Relationships extracted by the Sequential Extractor.

Relationship	*Confidence*
Emission spectrum `pedagogicallyClose` Wavelength	1.0
Wavelength `pedagogicallyClose` Emission spectrum	1.0
Proton `pedagogicallyClose` Electron	1.0
Helium `pedagogicallyClose` Atom	0.9
Radiation pressure `prerequisite` Comet tail	0.6
Space `prerequisite` Planet	0.6

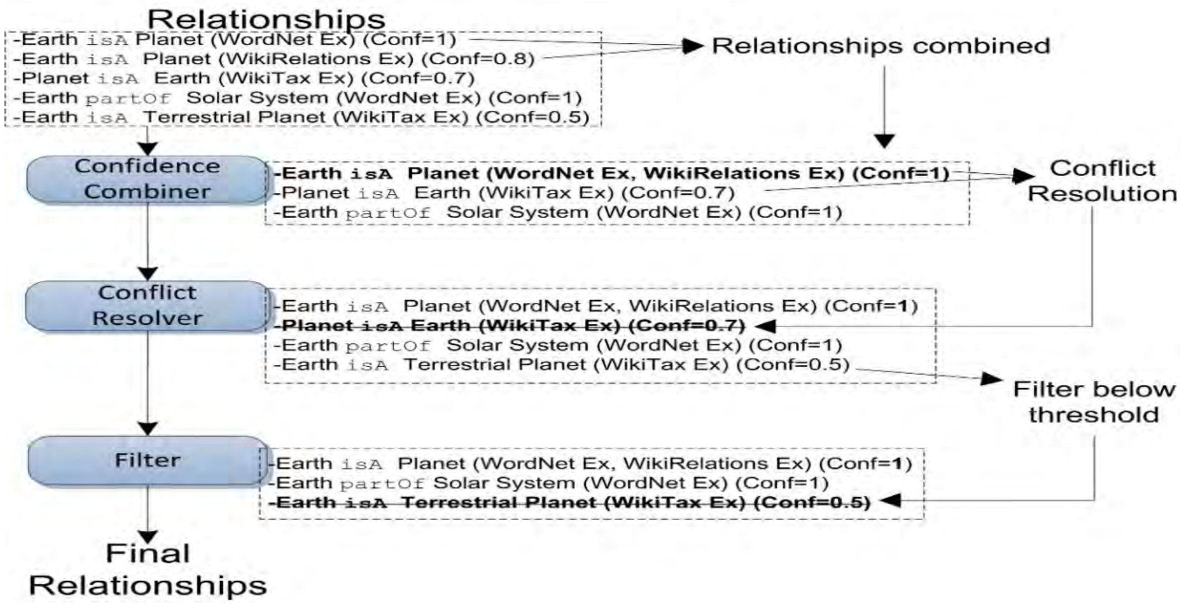

FIGURE 12. Example of getting a mention from a sentence.

determining the minimum number of links for a relationship to be inferred, 2 in this case.

$$Confidence = min(b + (top1m + top2m - low \times 0.05)\ 1) \tag{3}$$

D. COMBINING AND FILTERING RELATIONSHIPS

In the last phase, following a three-step process, LiReWi obtains the final set of relationships from the relationship candidates obtained using the extractors described above. It starts by combining and adjusting the confidence of those relationships inferred by more than one extractor, to which end (4) is used, where c_i is the confidence of extractor i, n is the number of extractors that identified the relationship and α is a constant (1.1) that promotes relationships identified by several extractors. The more extractors infer a relationship, the higher the confidence in that relationship is.

$$Final\ Confidence = \min(\frac{1}{n}\sum\nolimits_{i=1}^{n} c_i \times \alpha, 1) \tag{4}$$

Next, LiReWi detects and solves conflicts between relationship candidates of the same kind. In this step, relationships with inconsistencies and erroneous relationships are removed. For example, when a relationship has the same topic as source and destination, it is removed. Furthermore, some relationships may form a so-called loop. For example, one relationship involving two topics may be inferred in both directions. In those cases, LiReWi carries out a solving process selecting the final relationships using the confidence as a criterion and the link structure in Wikipedia. In the final step, those relationships that have a confidence below an empirically gathered threshold (0.6) are deleted to improve the consistency of the generated LDO. When two different relationships are identified between two topics, say *isA* and *partOf,* the assertion with highest confidence is accepted. In the final step, those relationships that have a confidence below the threshold empirically gathered in the tune-up process (0.6) are deleted to improve the consistency of the generated LDO.

Figure 12 illustrates the process described above. Firstly, the confidences of the relationships elicited by two or more extractors are combined. For example, *Earth* isA *Planet* is combined and adjusted accordingly.

In the second step, a conflict is found between *Earth* isA *Planet* and *Planet* isA *Earth* proposals. The system looks at the link structure of the topics in Wikipedia, along with the confidence of the extracted relationships, to determine the final relationship. In the figure, *Earth* isA *Planet* has higher confidence than *Planet* isA *Earth*. In addition, *Earth* has a link to *planet* in Wikipedia, whereas "Planet" does not have a link to "Earth". Therefore, the system decides to discard *Planet* isA *Earth* (Figure 12).

Finally, the system deletes those relationships that have less confidence than the predetermined threshold. In this case, *Earth* isA *Terrestrial planet* is deleted because its confidence is lower than the threshold.

IV. EVALUATION

In this section, the experiment conducted to evaluate LiReWi is depicted. First, an evaluation of the mapping techniques is depicted. Then, the evaluation of the candidate relationship extraction is presented and, finally, the evaluation of the combination and filtering is described.

LiReWi requires a set of topics as input. Therefore, the *Introduction to Astronomy* textbook has been processed with LiTeWi to obtain the topic set. Next, the topics with

TABLE 7. LiReWi vs. Dom-Sortze reported performance.

	Number of relations	Gold Standard			Expert Validation
		Precision (%)	Recall (%)	F1-Score (%)	Correctness (%)
WordNet	35	**77.14**	15.51	25.83	**100**
WikiTaxonomy	45	8.88	2.29	3.64	8.88
WikiRelations	26	69.23	10.34	17.99	76.92
WibiTaxonomy	**138**	39.85	**31.6**	**35.25**	50.72
Shallow Parsing	11	0	0	0	36.36
Sequential	15	53.33	4.59	8.45	60

highest relatedness with the domain of the textbook have been selected and used as input for the relationship elicitation. The relatedness value used for this purpose was the CValue [61] score computed by LiTeWi for the extracted topics. The C-Value is a domain-independent approach for the automatic elicitation of multiword terms that combines linguistic filters with statistical information in the form of a measure (also called C-Value). Nested terms (e.g., eclipse and sun-eclipse might be candidates in the term elicitation). This measure determines the termhood of the candidate considering: (1) the total frequency of occurrence of the candidate string, (2) the frequency of the candidate string as part of another longer candidate, (3) the number of the longer candidates, (4) the length of the candidate string in words. In this work, we assumed that the multiword terms with highest score are the most representative and, thus, the C-Value has been used to select the most related terms. In the experiment, the input set entailed 199 topics.

The evaluation procedure is a combined one where a gold standard and expert validation are conducted to measure the performance of the system.

For the gold standard evaluation, four experts stated the set of gold relationships (*isA, partOf, prerequisite,* and *pedagogicallyClose*) between the 199 input topics. The gold standard entails 174 relationships, being 15 *pedagogicallyClose,* 10 *prerequisite,* 69 *partOf* and 80 *isA.*Then, the results obtained by the different extractors were compared with the gold standard.

Regarding the expert validation, once again 4 experts have manually checked the correctness of the extracted relationships. Fleiss's kappa [62] coefficient was computed to measure the inter-rater agreement. The experts agreed on 270 of 295 total extracted relationships, with 0.974 weighted kappa score. This value shows an almost perfect agreement between the experts [63]. The expert validation only considers the relationships agreed on by all the experts.

A. RESULTS OF THE MAPPING

In order to map topics to knowledge bases, namely Wordnet, LiReWi relies on previously identified mapping resources −BabelNet [57] and Fernando's [58]− and includes a Page Rank Mapping step using UKB [59]. The performance of the mapping was tested on the *Introduction to Astronomy*textbook, achieving 23.68% recall with 97.82% precision. Using those resources speeds up the identification of mappings, whereas including the disambiguation step resulted in a 5 points increase on the recall compared to the recall achieved when using only one of the resources, BabelNet or Fernando's.

B. RESULTS OF THE CANDIDATE RELATIONSHIPS EXTRACTORS

In this section, the performance of each extractor is depicted. For each extractor, the performance is reported by comparing the relationships it has extracted against the gold standard (precision, recall and F1-score). In addition, the expert validation results, i.e., the percentages of correct relationships according to the experts (correctness) are included. The performance of the extractors is summarized in Table 7.

The WordNet Extractor identified 35 relations achieving 77.14% recall with 15.51% precision. The expert validation resulted in 100% of the identified relations being valid. The WikiTaxonomy Extractor extracted 45 relations from the selected topics. The extractor obtained 8.88% precision and 2.29% recall for gold standard validation. The expert validation shown that only 4 of them (8.88%) were valid. The WikiRelations Extractor identified 26 relations, obtaining 69.23% precision, with 10.34% recall. The expert validation resulted in being 20 correct (76.92%). The WibiTaxonomy Extractor identified 138 relations achieving 39.85% precision with 31.6% recall for the textbook. The expert validation shows that 70 (50.72%) of the identified relations were valid. The Shallow Parsing Grammar Extractor identified 11 relations, none of them being part of the Gold Standard. The expert validation determined that 4 (36.36%) of the identified relations were valid. Finally, the Sequential Extractor identified 15 relationships. This method achieved 53.33% precision and 4.59% recall considering the Gold Standard. The expert validation determined that 9 of them (60%) were correct.

The WordNet Extractor shows the best performance in terms of the expert validation. This result was quite predictable considering that WordNet contains manually defined relationships. However, WordNet may be currently limited in terms of recall as it is not actively updated.

TABLE 8. LiReWi performance in introduction to astronomy textbook.

	Number of relations	*Gold Standard*			*Expert Validation*
		Precision (%)	*Recall (%)*	*F1-Score (%)*	*Precision (%)*
LiReWi	266	36.21	50.57	42.2	43.98

TABLE 9. LiReWi extractors performance.

	Number of relations	*Gold Standard*			*Expert Validation*
		Precision (%)	*Recall (%)*	*F1-Score (%)*	*Precision (%)*
`isA`	213	30.38	76.25	43.44	40.84
`partOf`	37	51.34	27.54	35.85	51.36
`prerequisite`	10	30	30	30	80
`pedagogicallyClose`	6	50	33	39.75	50

The extractors based on Wikipedia (WikiTaxonomy, WikiRelations and WibiTaxonomy) showed diverse behavior; the newer the underlying method, the better the results are. WibiTaxonomy Extractor showed the best performance among those extractors based on Wikipedia. On the other hand, WikiTaxonomy produced the worst results. The Shallow Parsing Extractor did not extract any relations considered for the Gold Standard. However, the expert validation shows that it extracted valuable relations from the selected topics. The Sequential Extractor achieved remarkable results in terms of precision and correctness.

C. RESULTS OF LIREWI

In this section, the overall results of LiReWi are depicted. Once all the extractors were processed in parallel on the Introduction to Astronomy textbook, their results were combined and filtered as described in Section 4.3. The performance of this step is presented in Table 8. 266 different relations were inferred by LiReWi using all the extractors. Considering the Gold-standard, 36.21% precision and 50.57% recall were achieved. The experts considered that 117 of the 266 (43.98%) identified relationships were correct.

Next the results for each kind of relationship are depicted (see Table 9). LiReWi extracted a total of 213 isA relationships achieving 30.38% precision with 76.25% recall. The expert validation resulted in 87 of the identified relations being valid (40.84%). 37 *partOf* relationships from the selected topics have been inferred by LiReWi, obtaining 51.34% precision and 27.54% recall for Gold-standard validation. The expert validation shows that 19 of them were valid (51.36%). Regarding the prerequisite relationships, 10 have been extracted, obtaining 30% precision and with 30% recall. The expert validation resulted in 8 of them being correct (80%). Finally, LiReWi extracted 6 pedagogicallyClose relationships achieving 50% precision with 33% recall for the textbook. The expert validation shows that 3 of the identified relations were valid (50%). It can be observed that the number of sequential relationships identified is low, future work should be directed to increase the number of such relationships. However, it must be considered that prerequisite relationship implies a strong restriction for further use in TSLSs, so their quantity should be minimized to the essentials in order to obtain more flexible TSLSs.

TABLE 10. LiReWi vs. Dom-Sortze reported performance.

	Gold Standard		
	Precision (%)	*Recall (%)*	*F1-Score (%)*
LiReWi	36.21	50.57	**42.2**
Dom-Sortze	63.27	20.74	31.24

In comparison with each Extractor, LiReWi outperforms the best extractor (WibiTaxonomy Extractor) by almost 20 percentage points in terms of recall. Taking into account the F1-score, LiReWi outperforms the best extractor (WibiTaxonomy Extractor) by 7 percentage points.

Comparing LiReWi with DOMSortze,[1] the system referenced in the Related Work Section which is, to our knowledge, the system closest to LiReWi, DOMSortze reported 63.27% precision with 20.74% recall in the elicitation of relationships [53]. As can be observed in Table 10, LiReWi outperforms DOMSortze considering recall and F1-Score.

The result of the overall process shows that LiReWi can take advantage of different methods in order to extract pedagogical relationships from topics.

V. CONCLUSIONS AND FUTURE WORK

Throughout this paper, LiReWi, a tool that combines different methods for relationship extraction in order to build Educational Ontologies from electronic textbooks has been presented. Using as input an electronic textbook and a set of topics, LiReWi identifies 4 types of pedagogical relationships

[1]No expert validation was carried out for DOMSortze regarding the elicitation of pedagogical relationships.

(*isA, partOf, prerequisite,* and *pedagogicallyClose)* existing between those topics. To this end, LiReWi uses different knowledge bases (Wikipedia and WordNet) and different methods for relationship extraction (grammar-based, co-occurrence-based and taxonomy-based methods) and combines them to achieve its goal. Some are extractors developed by the research group itself, others, on the contrary, are external extractors that have required a previous tuning up phase. LiReWi runs the extractors in parallel and combines and refines their outcomes to obtain the final set of pedagogical relationships. LiReWi is based on a modular architecture and it has been designed so that it can easily incorporate not only additional extractors but also new knowledge bases. These characteristics allow future work aimed at improving the results of LiReWi using other general-purpose knowledge bases such as Google knowledge graph, Wikidata, DBpedia and so on.

To assess its performance, LiReWi has been firstly tested on the programming domain, using the *Principles of Object Oriented Programming*[55] textbook to determine its optimal set-up and then, evaluated on the astronomy domain, using the *Introduction to Astronomy* [56] textbook. The evaluation was carried out in three phases: first, an evaluation of the mapping techniques was depicted; then, each individual relation extractor was tested and their performances were compared with each other; finally, the overall process, where all the extractors are combined, was also evaluated obtaining better results than both the individual extractors and also the only system that shares a similar aim, DOMSortze [53]. However, results show that more effort should be made in the identification of sequential relationships, for example, the outcome of the Shallow Parsing Extractors should be analyzed to identify the failures in order to construct more accurate patterns.

LiReWi can be easily extended to support the relationship extraction from documents written in new languages. LiReWi can take advantage of the mappings to WordNet and Wikipedia articles and use the links to the English article versions to extract the pedagogical relationships following the approach described above. In addition, multilingual knowledge bases such as MENTA [43], BabelNet [57], Yago3 [64] or Multilingual WordNet [65] could provide new means to elicit pedagogical relationships. In a similar way, new approaches and resources [66] could be integrated in LiReWi to enhance topic mapping to lexical resources such as WordNet.

Another research line that deserves to be explored is the application of other paradigms to the combination and filtering of the candidate relationships. Probabilistic approaches such as Bayesians networks as well as connectionist systems could obtain better results.

LiReWi, in combination with LiTeWi, could be used to automatically generate the whole educational ontology from a textbook. This would allow average teachers to profit from ontology learning techniques in the development of their own course. They could thus limit their work to the revision and adaptation of the automatically generated ontology using the graphical user interface that will be developed to this end.

REFERENCES

[1] T. Murray, "EON: Knowledge based tutor authoring tools for content, instructional, strategy, student model and interface design," in *Authoring Tools for Advanced Technology Learning Environments*. Dordrecht, The Netherlands: Kluwer Academic Publishers, 2003.

[2] R. Nkambou, "Modeling the domain: An introduction to the expert module," in *Advances in Intelligent Tutoring Systems*, vol. 308, R. Nkambou, J. Bourdeau, and R. Mizoguchi, Eds. Heidelberg, Germany: Springer, 2010, pp. 15–32.

[3] Cimiano, Philipp, "Perspectives on Ontology Learning: Foreword," in *Perspectives on Ontology Learning*. Amsterdam, The Netherlands: IOS Press, 2014, pp. 1–8.

[4] A. Conde, M. Larrañaga, A. Arruarte, J. A. Elorriaga, and D. Roth, "LiTeWi: A combined term extraction and entity linking method for eliciting educational Ontologies from textbooks," *J. Assoc. Inf. Sci. Technol.*, vol. 67, no. 2, pp. 380–399, Feb. 2016.

[5] C. M. Reigeluth, M. D. Merrill, and C. V. Bunderson, "The structure of subject matter content and its instructional design implications," *Instr. Sci.*, vol. 7, no. 2, pp. 107–126, Apr. 1978.

[6] Ã. Conde, "LiDom builder: Automatising the construction of multilingual domain modules," Ph.D. dissertation, Univ. Basque Country UPV/EHU, Donostia, Spain, 2016.

[7] M. N. Asim, M. Wasim, M. U. G. Khan, W. Mahmood, and H. M. Abbasi, "A survey of ontology learning techniques and applications," *Database*, vol. 2018, pp. 1–24, Jan. 2018.

[8] R. Mizoguchi and M. Ikeda, "Towards ontology engineering," in *Proc. Joint Pacific Asian Conf. Expert Syst./Singap. Int. Conf. Intell. Syst.*, 1997, pp. 259–266.

[9] R. Mizoguchi and J. Bourdeau, "Using ontological engineering to overcome common AI-ED problems," *J. Artif. Intell. Educ.*, vol. 11, pp. 107–121, Feb. 2000.

[10] R. Mizoguchi and J. Bourdeau, "Using ontological engineering to overcome AI-ED problems: Contribution, impact and perspectives," *Int. J. Artif. Intell. Educ.*, vol. 26, no. 1, pp. 91–106, Mar. 2016.

[11] J. Jensen, "A systematic literature review of the use of Semantic Web technologies in formal education," *Bri. J. Educ. Technol.*, vol. 50, no. 2, pp. 505–517, Aug. 2017.

[12] D. Dicheva, S. Sosnovsky, T. Gavrilova, and P. Brusilovsky, "Ontological web portal for educational ontologies," in *Proc. Appl. Semantic Web Technol. E-Learn. Workshop*, Jul. 2005, pp. 19–29.

[13] A. W. P. Fok and H. H. S. Ip, "Educational ontologies construction for personalized learning on the web," in *Evolution of Teaching and Learning Paradigms in Intelligent Environment*, vol. 62, 2007, pp. 47–82.

[14] T. Murray, "Authoring knowledge-based tutors: Tools for content, instructional strategy, student model, and interface design," *J. Learn. Sci.*, vol. 7, no. 1, pp. 5–64, Jan. 1998.

[15] T. Murray, S. Blessing, and S. Ainsworth, Eds., *Authoring Tools for Advanced Technology Learning Environments*. Norwell, MA, USA: Kluwer Academic Publishers, 2003.

[16] A. Mitrovic, K. R. Koedinger, and B. Martin, "A comparative analysis of cognitive tutoring and constraint-based modeling," in *Proc. User Model.*, Jun. 2003, pp. 313–322.

[17] B. Martin and A. Mitrovic, "ITS domain modelling: Art or science," in *Proc. Int. Conf. Artif. Intell. Educ.*, Jul. 2003, pp. 183–190.

[18] A. Mitrovic, P. Suraweera, B. Martin, and A. Weerasinghe, "DB-suite: Experiences with three intelligent, Web-based database tutors," *J. Interact. Learn. Res.*, vol. 15, no. 4, pp. 409–432, 2004.

[19] P. Suraweera, A. Mitrovic, and B. Martin, "The role of domain ontology in knowledge acquisition for ITSs," in *Proc. 7th Int. Conf. Intell. Tutoring Syst.*, Aug. 2004, pp. 207–216.

[20] S. Sosnovsky and T. Gavrilova, "Development of educational ontology for C-programming," *Int. J. Inf. Theor. Appl.*, vol. 13, no. 4, pp. 303–308, 2006.

[21] L. N. Cassel, G. Davies, R. LeBlanc, L. Snyder, and H. Topi, "Using a computing ontology as a foundation for curriculum development," in *Proc. 6th Int. Workshop Ontologies Semantic web E-Learn.*, 2008, pp. 21–29.

[22] L. Jiang, Z. Yang, Q. Liu, and C. Zhao, "The use of concept maps in educational ontology development for computer networks," in *Proc. Int. Conf. Granular Comput.*, Aug. 2008, pp. 346–349.

[23] G. Ganapathi, R. Lourdusamy, and V. Rajaram, "Towards ontology development for teaching programming language," *Proc. World Congr. Eng.*, 2011, p. 2192.
[24] C. R. Robin and G. V. Uma, "An ontology based linguistic infrastructure to represent software risk identification knowledge," in *Proc. Int. Conf. Workshop Emerg. Trends Technol.*, Feb. 2011, pp. 744–751.
[25] A. Karkar and J. A. Ja, "am, "An educational ontology-based M-learning system," *Int. J. Interact. Mob. Technol. IJIM*, vol. 10, no. 4, pp. 48–56, Oct. 2016.
[26] J. Bourdeau, R. Mizoguchi, V. PsychÃĨ, and R. Nkambou, "Selecting theories in an ontology-based ITS authoring environment," in *Proc. 7th Int. Conf. Intell. Tutoring Syst.*, Aug. 2004, pp. 150–161.
[27] J. Kay, "Ontologies for reusable and scrutable student models," in *Proc. AIED Workshop W*, 1999, pp. 72–77.
[28] J. Kay and A. Lum, "Ontologies for scrutable student modelling in adaptive E-learning," in *Proc. Adapt. Hypermedia Adapt. Web-Based Syst. Workshop Semantic Web E-Learn.*, Aug. 2004, pp. 12–35.
[29] P. Buitelaar, P. Cimiano, and B. Magnini, "Ontology learning from text: An overview," *Ontol. Learn. Text Methods, Appl. Eval.*, vol. 123, pp. 3–12, May 2005.
[30] M. T. Pazienza and A. Stellato, Eds., *Semi-Automatic Ontology Development: Processes and Resources*. Hershey, PA, USA: Inf. Sci. Reference, 2012.
[31] M. Hazman, S. R. El-Beltagy, and A. Rafea, "A survey of ontology learning approaches," *Database*, vol. 7, no. 6, pp. 36–43, 2011.
[32] A. Scheuermann and J. Obermann, "The status quo of ontology learning from unstructured knowledge sources for knowledge management," in *Proc. IFIP Int. Workshop Artif. Intell. Knowl. Manage.*, Aug. 2012, pp. 72–94.
[33] A. Lenci and G. Benotto, "Identifying hypernyms in distributional semantic spaces," in *Proc. 1st Joint Conf. Lexical Comput. Semantics*, Stroudsburg, PA, USA, Aug. 2012, pp. 75–79.
[34] G. Bordea, P. Buitelaar, S. Faralli, and R. Navigli, "SemEval-2015 task 17: Taxonomy extraction evaluation (TExEval)," in *Proc. 9th Int. Workshop Semantic Eval.* , Denver, CO, USA, May 2015, pp. 902–910.
[35] E. Lefever, "A hybrid approach to domain-independent taxonomy learning," *Appl. Ontol.*, vol. 11, no. 3, pp. 255–278, Oct. 2016.
[36] V. Shwartz, Y. Goldberg, and I. Dagan, "Improving Hypernymy detection with an integrated path-based and distributional method," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, Berlin, Germany, Jan. 2016, pp. 2389–2398.
[37] C. Fellbaum, *WordNet: An Electronic Lexical Database*. Cambridge, MA, USA: MIT Press, 1998.
[38] G. Kaya and A. Altun, "Educational ontology development," in *Encyclopedia of Information Science and Technology*, 4th ed. State College, PA, USA: IGI Global, 2018, pp. 1441–1450.
[39] S. P. Ponzetto and M. Strube, "Deriving a large scale taxonomy from Wikipedia," in *Proc. 22nd Nat. Conf. Artif. Intell.*, vol. 2, Aug. 2007, pp. 1440–1445.
[40] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in *Proc. 14th Conf. Comput. Linguistics*, Feb. 1992, pp. 539–545.
[41] F. Wu and D. S. Weld, "Automatically refining the wikipedia infobox ontology," in *Proc. 17th Int. Conf. World Wide Web*, May 2008, p. 635.
[42] [42] T. Flati, D. Vannella, T. Pasini, and R. Navigli, "Two is bigger (and better) than one: The Wikipedia bitaxonomy project. Baltimore," *Maryland Assoc. Comput. Linguistics*, vol. 1, pp. 945–955, Jun. 2014.
[43] G. de Melo and G. Weikum, "Taxonomic data integration from multilingual Wikipedia editions," *Knowl. Inf. Syst.*, vol. 39, no. 1, pp. 1–39, Apr. 2014.
[44] S. Wang *et al.*, "Concept hierarchy extraction from textbooks," in *Proc. ACM Symp. Document Eng.*, New York, NY, USA, Sep. 2015, pp. 147–156.
[45] V. Nastase and M. Strube, "Decoding Wikipedia categories for knowledge acquisition," in *Proc. 23rd National Conf. Artif. Intell.*, vol. 2, Chicago, IL, USA, Feb. 2008, pp. 1219–1224.
[46] P. Arnold and E. Rahm, "Extracting semantic concept relations from Wikipedia," in *Proc. 4th Int. Conf. Web Intell., Mining Semantics*, Jun. 2014, pp. 1–11.
[47] D. Roy, *Automatic Annotation of Learning Materials for E-Learning*. India: Indian Inst. Technol., 2006.
[48] C. Liang, Z. Wu, W. Huang, and C. L. Giles, "Measuring prerequisite relations among concepts," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Lisbon, Portugal, Feb. 2015, pp. 1668–1674.
[49] F. Gasparetti, C. Limongelli, and F. Sciarrone, "Exploiting wikipedia for discovering prerequisite relationships among learning objects," in *Proc. Int. Conf. Inf. Technol. Based Higher Edu. Training (ITHET)*, Jun. 2015, pp. 1–6.
[50] C. de Medio, F. Gasparetti, C. Limongelli, F. Sciarrone, and M. Temperini, "Automatic extraction of prerequisites among learning objects using wikipedia-based content analysis," in *Proc. Int. Conf. Intell. Tutoring Syst.*, 2016, pp. 375–381.
[51] D. Milne and I. H. Witten, "An open-source toolkit for mining Wikipedia," *Artif. Intell.*, vol. 194, pp. 222–239, Jun. 2013.
[52] F. Gasparetti, C. de Medio, C. Limongelli, F. Sciarrone, and M. Temperini, "Prerequisites between learning objects: Automatic extraction based on a machine learning approach," *Telemat. Inform.*, vol. 35, no. 3, pp. 595–610, Jun. 2018.
[53] M. Larrañaga, A. Conde, I. Calvo, J. A. Elorriaga, and A. Arruarte, "Automatic generation of the domain module from electronic textbooks: Method and validation," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 1, pp. 69–82, Jan. 2014.
[54] M. Rani, K. V. Srivastava, and O. P. Vyas, "An ontological learning management system," *Comput. Appl. Eng. Educ.*, vol. 24, no. 5, pp. 706–722, Sep. 2016.
[55] S. Wong and D. Nguyen, *Principles of Object-Oriented Programming*. Gainesville, FL, USA: Univ. Press of Florida, 2010.
[56] I. Morison, *Introduction to Astronomy and Cosmology*. Hoboken, NJ, USA: Wiley, 2008.
[57] R. Navigli and S. P. Ponzetto, "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network," *Artif. Intell.*, vol. 193, pp. 217–250, Dec. 2012.
[58] S. Fernando, *Enriching Lexical Knowledge Bases with Encyclopedic Relations*. UK London, U.K.: University Sheffield, 2013.
[59] E. Agirre and A. Soroa, "Personalizing pagerank for word sense disambiguation," in *Proc. 12th Conf. Eur. Chapter Assoc. Comput. Linguistics*, Feb. 2009, pp. 33–41.
[60] E. Bick and T. Didriksen. (2015). *CG-3 -Beyond Classical Constraint Grammar*. [Online]. Available: https://www.google.com/search?client=firefox-b-d&q=CG-3+-Beyond+Classical+Constraint+Grammar
[61] K. Frantzi, S. Ananiadou, and H. Mima, "Automatic recognition of multi-word terms: The C-value/NC-value method," *Int. J. Digital Libraries*, vol. 3, no. 2, pp. 115–130, 2000.
[62] J. L. Fleiss, "Measuring nominal scale agreement among many raters," *Psychol. Bull.*, vol. 76, no. 5, p. 378, 1971.
[63] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 1, pp. 159–174, Feb. 1977.
[64] F. Mahdisoltani, J. Biega, and F. M. Suchanek, "YAGO3: A knowledge base from multilingual Wikipedias," in *Proc. 7th Biennial Conf. Innov. Data Syst. Res.*, Asilomar, CA, USA, Jan. 2015. [Online]. Available: www.cidrdb.org
[65] F. Bond and R. Foster, "Linking and extending an open multilingual wordnet," in *Proc. 51st Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2013, pp. 1352–1362.
[66] J. P. McCrae, "Mapping wordnet instances to Wikipedia," in *Proc. 9th Global Wordnet Conf.*, Singapore, 2018, pp. 12–36.

ANGEL CONDE received the Ph.D. degree in computer science from the University of the Basque Country, in 2016. He is currently a Researcher with IK4-Ikerlan and collaborates with the GaLan Research Group, the Group for Adaptive Teaching-Learning Environment, and the University of the Basque Country UPV/EHU.

MIKEL LARRAÑAGA received the Ph.D. degree in computer science from the University of the Basque Country UPV/EHU, in 2012, where he is currently a Faculty Member with the Department of Computer Languages and Sciences. He has been involved in the computer-based education area for the last 15 years. His current research interests include knowledge acquisition, concept mapping, learning analytics, recommender systems, and intelligent tutoring systems.

ANA ARRUARTE received the Ph.D. degree in computer science from the University of the Basque Country UPV/EHU, in 1998, where she is currently an Associate Professor with the Computer Languages and Systems Department. She is a member of the GaLan Research Group and the Group for Adaptive Teaching-Learning Environment.

JON A. ELORRIAGA received the Ph.D. degree in computer science from the University of the Basque Country UPV/EHU, in 1998. Since 1996, he has been a Faculty Member with the Department of Computer Languages and Systems, University of the Basque Country UPV/EHU. He has been with the GaLan Research Group involved in the computer-based education area, since 1991.