6

A Review of Ontology Development Methodologies: The Way Forward for Robust Ontology Design

Enesi Femi Aminu[1], Ishaq Oyebisi Oyefolahan[2], and Muhammad Bashir Abdullahi[1]

[1]Department of Computer Science, Federal University of Technology, Nigeria
[2]Department of Information & Media Technology, Federal University of Technology, Nigeria
E-mail: enesifa@futminna.edu.ng; o.ishaq@futminna.edu.ng; el.bashir02@futminna.edu.ng

Abstract

In this present age, the application of ontology as a data modeling technique across different fields of study, for example, knowledge management and information retrieval systems, is indispensable. This development is necessary to find viable solutions to the challenges of data heterogeneity and concept mismatch. Therefore, the end goal is geared toward achieving machine-represented data; in other words, the data are being modeled ontologically. There are existing ontology design methodologies; however, a single methodology is often not complete to design a robust ontology. Thus, this research aims to review the existing standard methodologies through concept-based analysis that suggests a way forward to design robust ontology. The analysis of the review is carried out by considering the goals of achieving robust ontology design, such as data integration, accessibility, reusability, and domain granularity. Based on the literature, this review shows that collaborative design with domain experts, application of standard evaluation techniques, modification of existing ontology development methodologies, types of ontology, and ontology-based machine learning models are determinant factors that define the robustness of ontology. Therefore, if an ontology developer pays attention to these criteria to design an implementable model, this would pave way for robust ontology to be designed.

Keywords: Data collaboration, data integration, robust ontology, ontology design, ontology methodologies.

6.1 Introduction

In this current age, while availability of data is no longer an issue, the astronomical growth of these data in heterogeneous forms calls for research attention. This is because the heterogeneous nature of data along with the unstructured state of its repositories poses difficulties to achieve the required collaborative operations on data, such as integration and reusability. Consequently, several data modeling techniques have been employed by researchers to advance data collaboration in the form of metadata. However, there are gaps to bridge in order to ascertain a more robust knowledge management system, such as information retrieval systems, question answering systems, and recommendation systems. In view of this research quest, a more robust data modeling technique called ontology is promising to this effect. Over some decades, researchers have been constantly employing the technique in the knowledge management fields of study as mentioned earlier in this section. An important strength of ontology technique lies on its ability to adapt to any series of modification either during the course of modeling or application. It is flexible toward any form of design approaches; either top down or bottom up [1]. Ontology, as semantic technology, has the potential to map the physical entities into computational entities that would make communication easier between human and machine [2]. This is because ontology has a standard mechanism to characterize domain knowledge.

Ontology is described as the prime stronghold of semantic technologies. As such, researchers in the field of ontology engineering are currently challenged to advance a solution on viable ontology development methodology that can assist to design robust knowledge-based systems [3]. There are required technologies to model ontology, which include but not limited to extensible markup language (XML), resource description framework (RDF), web ontology language (OWL), logic inference, and SPARQL as shown by the popular semantic technologies stack shown in Figure 6.1.

The semantic stack's architecture represented by Figure 6.1 shows the semantic technologies in different layers to realize knowledge systems. The

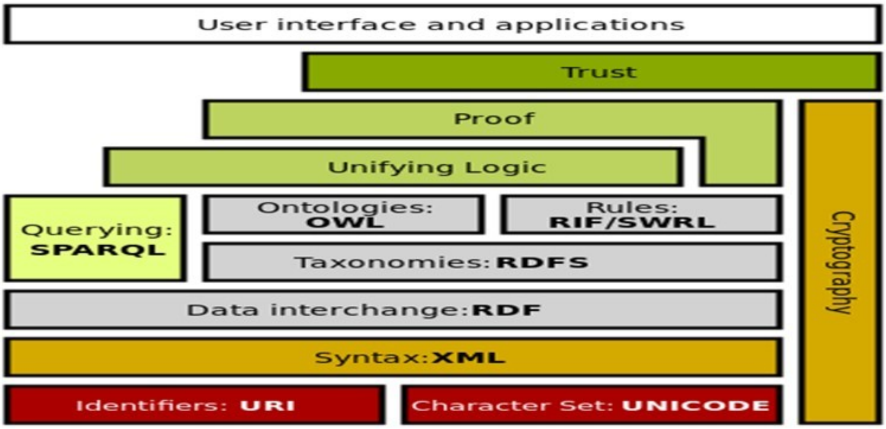

Figure 6.1 The semantic technologies stack [4].

bottom layer consists of basic techniques, which include URIs, Unicode, and XML. The machine-readable format is represented in the syntax of XML with URI as metadata and namespace identifier. The middle layer presents some techniques, which include RDF, RFDS, OWL, and SPARQL. The first three technologies are used to represent knowledge; however, their differences are premised on the degree of expressivity. Axioms and rules are encoded using SWRL or RIF; SPARQL serves as a medium to query data. The top layer, proof, trust, and cryptography, ensures that source documents for the web are from trusted sources. The last layer is the user interface that enables humans to use an application.

Based on the literature, ontology is a data modeling technique that leverages on gathering of entities by taking cognizance of their semantic relations, such as meronyms and holonyms. Also, constraints are enforced among the relations in order to achieve a structured and collaborative data repository [5]. In essence, this is to enhance the data communication between human and machine. The strength of a robust ontology hugely depends on its capacity to infer contextual (hidden) knowledge based on a given literal knowledge. The inference would occur based on the knowledge representation models such as RDF and OWL [6–8]. Similarly, a long-standing acceptable definition of ontology [9, 10] is that of Gruber [11], which defines ontology as an explicit and formal specification of a conceptualization. The term "*explicit*" in the definition suggests that entities of any knowledge must be defined in clear terms; also, the entities' integrity must be precisely defined as well. The term "*formal*" connotes and emphasizes that ontology is a structured knowledge representation model. Another important term in the definition is *conceptualization*, which implies that ontology is an abstract representation of a physical knowledge where the required entities are harvested for modeling.

The physical knowledge includes physical real objects; similarly, abstract objects are also part of the knowledge. Examples of objects include bioinformatics, biomedicine, sports, building construction and design, agriculture, and religion; they are described as domains of ontology [12, 13]. Therefore, following the established knowledge about ontology, it is also viewed as *shared* knowledge of domain to solve the underlying structural and data heterogeneous issues [14]. The term "shared" describes ontology as a mechanism that supports data collaboration and integration. Consequently, ontology is therefore a semantic data model that leverages on a well-defined data collaboration mechanism, thereby ameliorating the issue of data inconsistency [15]. Thus, the role of ontology in knowledge management, data sharing, integration, and reusability against other techniques is described as unmatched [16]. This is because ontology has been severally applied to model complex data of different domains such as biomedical data (gene ontology), informatics data, and agriculture data [17]. This is evident in the work of He *et al.* [18] who employed ontology to model the data of the world ravaging ailment, coronavirus (COVID-19). The next section of this chapter presents the common principles for ontology development and the technologies required.

6.2 Ontology Development

Researchers in knowledge management usually employ ontology to build structure and intelligent knowledge-based systems that assist in knowledge reuse [19]. More so, with the aid of ontology, a system capable to infer the contextual knowledge of a given domain is achieved [20]. There are several reasons behind ontology development; these are but not limited to creating a platform for machine or users to share knowledge, to allow knowledge to be reused, and to define and analyze domain knowledge unambiguously [21, 22]. These reasons are principles that define the robustness of a given ontology if they are successfully attained.

To develop such ontology, a standard methodology, which is an iterative engineering process like the standard software development principle, is required. However, part of the contending issues that surround ontology design is on the appropriate methodology to deploy [23, 24]. Noy and McGuiness [21], in their research, reported that among the existing methodologies, there is no one that can sufficiently claim superiority over others. In other words, each of them has their own design flaws, which makes it inadequate to be used effectively. Therefore, a developer has the prerogative to make a choice based on certain defined factors. Nevertheless, to design ontology with characteristics of shared knowledge and inference capability, the methodology employed must be robust. This implies that ontology development's activities must be spread across the three categories of design, which include predevelopment, development, and postdevelopment stages [15]. Considering the literature reviewed in this work, the most frequently used methodologies include Gruninger−Fox, Methontology, Noy−McGuiness methodology, and NEON [25].

The process of ontology development equally involves the choice of ontology knowledge representation technologies, which include the languages and editors [26]. However, the focus of this review is on the methodology for ontology design that provides structural framework. In addition, the developer determines the language in which the methodology is implemented based on certain criteria; for example, expressivity of language. In this research, ontology representation language is classified into two folds. They are World Wide Web Consortium (W3C) based standard and International Standardization Organization (ISO). Examples of the former include OWL [27–29], RDF [30–32], RDF Schema [33–35], DARPA Agent Markup Language, and Ontology Interchange Language (DAML+OIL) [36, 37]. A classic example of the latter is XML Topic Map (XTM) [38]. OWL is reported as the most commonly used language as a result of its semantic expressivity [30, 39, 40], regardless of the two standards. The ontology editors are Protégé [41–43], FAO AGROVOC Concept Server Workbench Tool [44], OBO-Edit [45], SWOOP [46], Apollo [47], IsaViz [48], and TopBraidComposer [49, 50].

6.3 The Existing Ontology Development Methodology: The Review

In this section, the existing ontology design methodologies are partly reviewed. Methodology specifies a set of procedures on how the identified activities in the process of ontology design are duly carried out. Therefore, the quest to employ an existing methodology or brainstorm on a total new approach is indispensable [51, 52]. This is because the developers are in most cases constrained on the existing methodologies to model a knowledge-based system. The constraint is owing to their shortfalls based on the standard of software development guidelines [53, 54]. The literature of Lenat and Guha [55] reported the first ontology development methodology dubbed as Cyc project. The activities of the methodology are encapsulated into three stages [21]. Dean *et al.* [56], in their work, reviewed some ontology development methodologies. They include TOVE (Toronto Virtual Enterprise) that consists of six approaches: enterprise model approach with four stages of activity and Methontology that consists of seven activities of the development process. Others are SENSUS, MENELAS, ONIONS, Ontolingua KBSI, and IDEF5 consisting of five activities. Similarly, the research of Fernandez-Lopez *et al.* [57] presents what is described as regularly used approaches for ontology design. They are Methontology, Gruninger−Fox, SENSUS, and Uschold−King. The functionalities of these approaches are analyzed against IEEE Standard 1074–1995, which is often described as a software development standard.

There are some relevant questions that normally arise when ontology design is required to address a particular problem. For instance, most often, the ontology developer and domain query the suitable methodology to design ontology. This question has to be diligently addressed because the robustness of ontology is a consequence of ontology methodology, which requires a lot of analysis. Similarly, a question such as "should ontology be developed from scratch or reuse an existing ontology?" normally arises. To address this question, it largely depends on the proposed approach to design ontology [58]. Therefore, the choice of methodology is a rigorous task that requires attention. This postulation is affirmed by the work of Dnyanesh and Rahul [15] who reported that the existing approaches for ontology design are deficient in terms of completeness in design; that is, inability of design activities' wholesomeness for predevelopment, development, and postdevelopment stages. In order to deal with this issue, some literature advocate for hybridized methodology for robust activities. Methontology and Gruninger−Fox [59–61] are some of the methodologies that have benefited from this approach.

Similar to standard software methodologies, the ontology design methodologies are described as iterative ontology engineering process. This is because the developer can effortlessly recall back to the previous activities of methodology whenever the need arises. Most of the domain-based ontologies developed are premised on the engineering process; for example, the soccer ontology in the research of [62]. Obrst *et al.* [63], in their work, equally harped for the significance of methodology to be an iterative process.

Some of the methodologies reported in the literature are Gruninger–Fox, Methontology, Noy–McGuiness (with example of wine ontology), and Uschold–King.

An important activity during the process of ontology development [64, 65] is the identification of ontology's terms or concepts. In order to achieve this goal, there are three strategies that can be employed [66]. They are bottom-up, top-down, and middle-out strategies. The bottom-up strategy works by first identifying the specific terms and generalizing into more abstract terms. On the other hand, the top-down strategy first identifies the most abstract terms and specializes the terms into more specific terms. Lastly and more importantly, the middle-out strategy first identifies the most significant terms, generalizes, and specializes into other terms [58]. Thus, the following subsections specifically review Gruninger and Fox's methodology, Methontology methodology, Noy–McGuiness methodology, and Uschold–Kings methodology.

6.3.1 Gruninger and fox's methodology

This is a methodology that is built based on the technique of first-order logic proposed by Gruninger and Fox [67] to develop a knowledge representation system. As stated earlier, the methodology leverages on the strength of logic because it has the capacity to transform informal scenarios into formal notations. It has five activities; they are identification of motivating scenarios and formalization of informal; specification of ontology's terms in formal language; formulation of competency questions; specification of axioms and rules for the ontology's terms; and creation of conditions for characterizing the completeness of ontology [58, 68, 69]. The methodology was initially conceived to design ontology for business-enterprise-related knowledge. However, the methodology has been constantly employed for different real-life scenarios. For example, Walisadeera *et al.* [23] adapted the methodology to design ontology for the agriculture domain to assist farmers' information needs in Sri Lanka. The implementation was achieved using OWL knowledge representation. Also, the middle-out strategy was employed to specify the core terms of the ontology designed. Similarly, the Gruninger–Fox approach partly constitutes the hybridized methodology for the development of the university ontology [70].

The methodology has an important activity called formulation of competency questions (CQs) that can serve as evaluation mechanism, to validate the correctness of ontology. It equally aids to set the scope of an ontology's domain. As a result of this development, the application of formal notations for machine-represented knowledge is indispensable [71, 72]. Therefore, the capacity of this methodology cannot be overestimated; this is why most often according to literature, it usually formed part of a hybridized approach [59]. However, the methodology is not without notable gaps as identified by the literature [15]; for instance, ontology mapping or versioning cannot be technically achieved. Similarly, the literature [59, 73] equally reported the deficiency of the methodology in terms of completeness of activities for the three stages of design. Another challenging issue is on its capacity to reuse existing ontology.

6.3.2 Methontology methodology

It is a research product of Artificial Intelligence Lab, which supports both ontology design approaches. That is, to design ontology based on reuse of existing ontologies or to entirely create a new ontology from scratch. WebODE and OntoEdit are the management tools that were primarily designed for it [20]. At the initial stage, chemical ontology was the first beneficiary of the methodology, which consists of seven activities. The activities that range from the starting step to the ending step are as follows. The first step is specification, where the objective of the ontology has to be defined. This is followed by knowledge acquisition; the required knowledge to model ontology has to be acquired from related sources. The third activity is conceptualization; the concepts of a domain are defined and classified as subjects, objects, and relations. Integration is another subsequent activity where the concepts are related together as hypernym, hyponym, holonymy, or meronym. The fifth activity is implementation; at this point, knowledge representation language and editor are required. The sixth is evaluation; that is, the consistency of ontology to develop has to be verified and validated. The last activity is documentation of ontology development process [74–76].

According to literature, this methodology has a wide application. The work of [77] designed graduation screen ontology based on this methodology. The ontology was implemented using OWL and protégé as representation language and editor, respectively. Similarly, the methodology was also adopted to develop an active waterfall protection based ontology [78], against the existing methodologies reviewed in the work of [79]. Furthermore, the research of Rizwan *et al.* [80] reaffirmed the standard of the methodology whose activities were claimed to be in total compliance with the procedure of software development life cycle. Even though the reviewed literature of [81] duly acknowledged that no methodology for ontology development is sufficiently robust, the work described Methontology as a promising approach. The work of Ibrahim *et al*. [82] also described the methodology as an outstanding and balanced approach for ontology design.

More so, Methontology was reported as one of the commonly used methodologies for ontology design [83, 84]. Similarly, the research of [85] equally reported the strengths of the methodology; among others is that the activities for the development process of the methodology are in full conformity with the IEEE standard 1074-1999.

Nonetheless, the methodology still has some areas in which an improvement can be made. For instance, a robust activity as that of competency questions stands to improve the validation process.

6.3.3 Noy–McGuiness methodology

Noy–McGuiness is another iterative ontology engineering principle that is designed based on certain rules. It equally works based on seven iterative activities. The first activity is to define the ontology's domain along with precise scope. The second activity is to ascertain the need to develop a new ontology; this is because the methodology encourages the reuse of existing ontology. The next activity is to specify the required and relevant concepts for ontology design. The fourth activity is to declare the classes along with hierarchy, especially when OWL is to be used for implementation. The fifth activity is to also declare the class's property; while the sixth activity is to determine the constraints of the properties. Lastly, individuals or instances (as in the case of OWL) are established [22, 86]. The functionality process of the methodology is explained based on wine ontology. Furthermore, Godspower and Esingbemi [66] adopted the methodology to design a cash crop based ontology for farmers in Nigeria's market. Top-down strategy was employed for terms identification of the ontology and was implemented using the protégé editor. Similarly, the work of Serna and Serna [87], whose aim was to develop software maintenance based ontology, equally employed the Noy–McGuiness methodology.

In a modified form, Tiffani *et al*. [88] adapted Noy–McGuiness and formulated the Arp methodology [89] as a six-activity ontology engineering process to design antimicrobial prescription based ontology. The OWL-Protégé platform was used for the implementation. Similarly, the research of Chen-Huei *et al*. [90] considered the first four activities of Noy and McGuiness approach and merged it with another methodology to design natural disaster management based ontology.

6.3.4 Uschold–King methodology

The idea of this methodology is conceived from an (enterprise) ontology, which comprises four activities [14]. They are as follows: to determine the sole objective of the proposed ontology; to develop the ontology; to perform evaluation measure; and, lastly, to conduct process documentation. However, as the application of the methodology progresses, an improvement was carried out through the three ontology core concept identification strategies [91]. Therefore, the methodology was duly employed to design an ontology for the domain of waste water treatment christened (WaWO) [92].

As partly stated earlier, a robust ontology design is achieved mostly when these methodologies are adapted. Based on literature, some clear cut cases of methodology's modification include the research of Bonanci *et al*. [93] that presents an adapted six-activity ontology design methodology for agriculture knowledge. Also, Aree *et al*. [94] proposed a modified five-activity ontology engineering process for the development of rice crop based ontology. Niu and Issa [95] designed taxonomy for construction domain based on the proposed fused methodology. Gregor *et al*. [96] proposed a logical methodology to design an anticipated intelligence transportation system, ontologically. In the same vein, Dutta *et al*. [97] also presented a ten-activity-based ontology engineering principle for the domain of food. The principle is dubbed as YAMO in their literature. Similarly, Zeb *et al*. [98] equally presented a methodology that consists of ten activities to design ontologies for domains. Another two-activity-based top-down design methodology was also proposed to create ontology in the research of Mezghani *et al*. [99].

Furthermore, there is no doubt that some activities of the existing methodologies are similar. However, in most cases, these activities are combined into three main activities of ontology engineering process [100]. Rayyaan *et al*. [101] examined Methontology and UPON methodologies to develop ontology for the domain of textile supply chain. Based on their results, the researchers pointed out the robustness of UPON for the domain under consideration.

6.4 Way Forward for Robust Ontology Design: The Review

Ontology is said to be robust if it satisfies a set of given design guidelines according to a given standard. More so, the positive outcome of ontology's validity and verification in terms of its content and design largely contributes to the factors that determine the robustness. Primarily, a serious attention has to be paid to the choice of ontology development methodology [102]. However, this is not to state that there are hard rules or guidelines to come up with robust ontology. This is because the research work of [103] emphasizes on collaborative ontology design between ontology engineers and domain experts as one of the reliable means of developing ontology that can stand the test of time. The researchers emphasize on the complete involvement of domain experts at the requirement elicitation level, logical level, and the physical level of ontology design.

John *et al.* [104], in their research, equally faulted the robustness of the existing ontology development methodology. The researchers proposed the hybridization of the procedures or models of software engineering process to the traditional ontological engineering process. Consequently, a software-centric innovative ontology development methodology capable to develop large-scale ontology is targeted. The researchers, in their conclusion, promised to validate the prototype to ascertain the level of accuracy and applicability.

Several methodologies for ontology development have been proposed; however, a robust methodology to address the design of multiple aspect of domain knowledge is largely still in progress. This development propels the research of [105] to propose a four-step methodology that can assist the integration of cross-domain knowledge for multi-aspect ontology design. The major criterion considered by the researchers is on the integration aspect of different domains. A tourism domain is considered to depict the application of the proposed methodology for decision support system based on human machine collective intelligence. Therefore, the researchers aim to experiment the model on multifaceted problems that require knowledge from numerous application domains so as to improve decision making.

The utilization of effective supply chain ontology that aims to ameliorate the interoperability issue often associated with information systems is another aspect of robust ontology design approach to look out for. However, according to the survey work carried out by [106], attention has not been paid to this observation. According to the report, the survey was conducted based on three criteria as a yardstick for measuring the frameworks of six supply chain ontology models. None of the six ontology models considered by the researchers comes without pitfalls. The criteria are follows: scientific paradigm, granularity, and fundamental methodological mechanism. The researchers argued that much effort of ontology development has been concentrated on the organization of human knowledge at the expense of the philosophy of supply chain itself over the year. Therefore, the work suggested that researchers on supply chain ontology should re-channel their efforts on formal ontology. More so, they equally suggested that a more holistic and thorough effort of literature review must be carried out if the dream of robust supply chain ontology methodology has to be achieved.

Furthermore, what gives birth to robust ontology could mean different things to different researchers. The research of [102] strongly argued that if a keen attention is paid to the testing stage of ontology development as a means of measuring quality, a robust ontology would not be an issue. The researchers noted that ontology testing mechanism has the capacity to test the major components of ontology, such as classes, relations, property, and axioms. Consequently, the research was motivated to propose what they called top domain ontology based testing mechanism. Therefore, the efficacy of the proposed mechanism, in terms of semantic matching, would be evaluated.

As noted earlier in the previous section of this work, ontology developer must carefully choose the methodology in order to have effective and efficient ontology that is largely devoid of flaws at minimum. In view of this, the research in [103] proposes a reusable prototype for ontology engineering process, which is premised on the adapted famous NeOn ontology development methodology. Urban Internet of Things was considered as the ontology domain, and the methodology can be applied across other related domains. There are three thematic key points that the method is sitting on, which are level of domain expert participation, logical correctness, and the content performance.

Ontology modeling technique has been identified over time as a reliable means of knowledge management and building of information systems in a contextual form. The commonest technique of modeling ontology is manual, which is very tedious and time consuming; however, the technique is reliable with a good degree of design accuracy. While this strength partly contributes to an effective ontology design, the time spent cannot be easily traded off. Consequently, in order to minimize the human error and design time, Yang *et al.* [107] proposed an ontology learning methodology that can autonomously extract data from extant system engineering standards to form system engineering ontology. The multi-tier methodology consists of a collection of data and pre-processing, natural language preprocessing (NLP) based lexical analysis, and extraction of ontology components. However, the authors submitted that interested researchers have to advance a concerted effort to develop a more robust ontology learning methodology. Currently, the research toward this approach is ongoing as most of the existing prototypes and models suffer from one form of deficiencies or another. For example, they are often incapable to handle implicit terminological and non-terminological relations and data properties.

In a quest to design an ontology that is largely free from errors, the research of [108] described the robustness of ontology as the type of ontology an engineer intends to develop. The researchers argue that most existing domain ontologies are a product of top-level ontology. However, to develop an ontology that is accessible, findable, interoperable, and reusable, a most recent technique called upper ontology alignment is proposed. The technique is an improvement of extreme ontology design methodology, which is based on the ontology design patterns. However, like every other technique, this design pattern equally sought for improvement on the namespace of entity's prefix. Besides, there is a need for robust mechanism to take care of more complex alignment beyond sub-entities (either class or property).

According to the research of [109], there are known limitations among the existing ontology design methodologies; such as vague procedures on how existing ontological and non-ontological collections can be reused. Besides, the issue of usability of finished ontology is yet to attract the needed attention. More importantly, the issue of integration process of various concepts is within a given top-level domain. It implies that ontology's robustness is proportionate to the resources available, and how well can the resources or concepts be integrated within the given domain. Consequently, the researchers proposed a systemic ontology iterative design approach for manufacturing domain herein called *manuService*. The design methodology ranges from requirement analysis to evaluation and feedback. It is a product of popular software development process called rapid application development and extraction of some concepts from existing ontology methodology. The ontology is implemented in OWL using the format of RDF/XML and finally developed using the open source ontology editor, protégé. However, more concepts from the knowledge domain are required to be integrated into the cloud manufacturing based data model.

Similar to the position of [109] on the type of domain, serving as determinant to a methodology to produce robust ontology, Palmirani *et al.* [110] proposed to develop a legal reasoning based ontology named privacy ontology. The researchers on this note hunted for a legal-based design approach called methodology for building legal ontology. This methodology requires ten activities that range from description of the ontology's goal to documentation and collection of feedback. More importantly, the researchers acknowledge the existence of several legal ontologies; however, they lack capacity to integrate with the deontic logic model functioning for legal reasoning. Therefore, the proposed ontology has the capacity for integration. However, the work was reported to be a continuous process as more concepts would be gradually integrated into it.

Yunianta *et al*. [111] argued that the existing ontology design methodologies are not suitable for data integration. Consequently, the research is motivated to develop an enhanced ontology methodology capable for semantic data integration. The enhanced approach called *OntoDI* consists of seven steps, which are categorized in three phases of predevelopment, core development and postdevelopment. The methodology was experimented in the domain of e-learning system. The researchers therefore anticipate that more data would still be integrated; a detailed evaluation would equally be carried out to ascertain the robustness of the methodology.

The fusion of machine learning models with ontology engineering process is gradually receiving attention in the field of knowledge management. Beyond the traditional semantic word representation models, such as word embedding technique (word2Vec), the richness of ontology (especially the OWL-based ontology) has contributed immensely to the robustness of knowledge system. For example, the research of [112] aimed to embed OWL ontology that encrypts the semantic of an ontology by considering its knowledge graph, the lexical knowledge, and the constructors of the knowledge representation model. In other words, each of the ontology's entities such as class, individual, and property would be represented in a vector space – hence termed OWL2Vec. Other ontology-based semantic word representations that literature proposed before the emergence of OWL2Vec are Onto2Vec [113] and OPA2Vec [114]. The methodology for this proposed ontology based on word representation consists of two steps, which are extraction of corpus from ontology and training of word embedding model with the corpus.

Some literature argue that irrespective of the methodology employing to develop ontology, the evaluation technique strongly determines the robustness of ontology. To this end, the research of [115] designed citrus ontology based on the crop production knowledge framework. The researchers employed some standard ontology validation techniques to determine the robustness of the ontology. The evaluation techniques include ontology vocabulary evaluation, structural evaluation based on the eight widely used metrics, antipattern-based evaluation, and ontology competency evaluation. Once a proposed ontology is validated against the technique and the outlook is positive, it implies that such ontology is robust. For example, a high value of schema deepness metric of structural evaluation against the average value of 0.34 indicates that the ontology is deep.

6.5 Proposed Methodology: Determinants for Robust Ontology Design

Presently, the traditional software development methodologies and standards such as waterfall or spiral models are highly deficient to develop an application for some real-life domains such as manufacturing execution systems (MESs). The reason is that most of these domains consist of a complex set of data, and efforts are being made to gear the design standards and functionalities of system-based domains in compliance with the concept of Industry 4.0 requirements [116]. The researchers, in the course of reviewing cutting-edge methodologies and tools to build MES-based Industry 4.0 concept, canvassed for ontology driven based technology. They present the superiority of this technique (that is, the OWL-based ontology-driven approach) over the existing conventional methods of software development. Some of the shortcomings of the conventional approaches highlighted by the authors are as follows: high overhead cost, partially reliable, weak interoperability mechanism, and too much time-consuming exercise.

Conversely, considering the numbers of existing ontologies for domains and continuous growth of heterogeneous data, there is a need to make ontology design process more robust. Therefore, this research aims to employ a concept-based analysis approach to determine the factors that can be responsible for the robustness of ontology design. Table 6.1, based on literature reviewed, summarizes the identified criteria or factors that can lead to the achievement of robust ontology.

Based on the concept-based approach employed in this research, seven criteria are identified to design a robust ontology. More importantly, some literature equally suggested that the goal or objective of ontology design has

Table 6.1 Determinant factors for robust ontology design.

S/N	Determinants for robust ontology
1	Articulate a robust ontology engineering process (for example, modification of existing methodologies)
2	Early collaborative design with domain experts
3	Ontology-based feature learning model (ontology-based word embedding model)
4	Ontology learning
5	Ontology's types: upper ontology alignment (enhancement of extreme ontology design methodology)
6	Standard evaluation techniques
7	Domain types

proportionate effect on the robustness of ontology. This research identifies some of the driven goal of ontology design as data integration [105, 110, 109]; accessible, findable, interoperable, and reusable [108]; granularity, scientific paradigm, and methodological mechanism [106]; testing component [102]. The next section of this article discusses these approaches in detail.

6.6 Discussion and Conclusion

As stated earlier, this section discusses the identified eight criteria as shown in Table 6.1 during the course of this literature review. The criteria are arranged in no particular order of significance.

i Articulate a robust ontology engineering process [102, 110, 111]: It is a common principle that a robust ontology design is a product of a robust ontology engineering process. This is in support of a common computing phenomenon that says garbage in garbage out. Activities or steps of a robust ontology development methodology are to be duly spread across predevelopment, development, and postdevelopment stages. However, since no single existing methodology is self-sufficient, it is expected to modify it. For example, the case of methodologies' hybridization [5].

ii Collaborative design with domain experts [103]: Some literature clearly pointed out that for an ontology engineer to design a robust ontology for any domain, there must be a holistic effort to collaborate with domain experts at the early stage. That is, since ontology developer in most cases has little or no knowledge of the domain, there must be a concerted effort to start the planning with a team of domain experts. Some concepts of real-life domains, such as medicine and agriculture, have a peculiar type of terminologies. Therefore, the experts are well positioned to professionally handle those concepts with their similar lexical concepts in terms of synonyms, hypernyms, hyponyms, meronyms, and holonyms. Consequently, data integration is achieved and expert-based evaluation mechanism would be easily established.

iii Ontology-based learning model: Another angle of literature view robust ontology design via the mechanism of machine learning models. Specifically, the fusion of feature optimization models into the ontology's dataset. For instance, the literature of [112] employs the semantic data representation technique (that is, word embedding technique) nicknamed as OWL2Vec to design a robust ontology-based word embedding system.

iv Ontology learning [107]: Researchers in this aspect of ontology design believe the prospect of this factor in developing a robust ontology. There are efforts to design an automated technique where a given dataset would be autonomously learned by such a technique to model ontology. The motivation behind this goal is that, besides the fact that manual design of ontology is tedious and time consuming, human errors are also unavoidable. Therefore, researchers argue that all these limitations have their consequent effects on a good ontology design.

v Ontology types [108]: Some schools of thought believe that robustness of ontology cannot be ascertained without paying attention to the types of ontologies. This implies that the richness of top-level ontology (upper ontology), for example, cannot in any way be compared with domain ontology (light weight ontology) or task ontology. Therefore, developers are expected to first determine the type of ontology they intend to develop and for what purpose. In some instances, domain ontology is regarded as a rich form of taxonomy system. As a result, the robustness cannot be compared with task or application ontology, which has semantic richness owing to its ability to handle competency questions.

vi Domain's type [109]: The rate of obtaining robustness in ontology design differs across various domains. This is because the rate of obtaining comprehensive data owing to complexity nature of a particular domain differs from another. For example, the data complexity of biomedical domain has to be properly collected and analyzed with a cutting-edge technique in collaboration with experts. Therefore, developers are advised to carry out a thorough feasibility study and collaborative analysis on a proposed domain.

vii Standard evaluation techniques [115]: One of the contending challenges in ontology development has to do with evaluation technique. Consequently, to ascertain the robustness of ontology, some set of evaluation techniques have to be deployed. These include ontology vocabulary evaluation, antipattern-based evaluation, ontology competency evaluation, and structural evaluation. This last technique, the structural-based evaluation, works using the eight widely used metrics. It has an average and median values obtained from a large number of 1413 OWL ontologies as considered by the research of [117].Therefore, the technique is very rich to determine the completeness of ontology. For instance, schema deepness metric one of the eight metrics of the structural based evaluation has the capacity to determine if an ontology design is deep or flat.

Table 6.2 The driven goals for robust ontology design.

S/N	Article details	Proposed methodology	Goal considered	Future work/result
1	Smirnov *et al.* [105]	A four-step methodology	*Integration*	To experiment the model on multifaceted problems domain
2	Sulaeman and Harsono [106]	A review of existing methodology	*Granularity, scientific paradigm, and methodological mechanism*	Suggested a more holistic review to achieve robust supply chain ontology methodology
3	Tebes *et al.* [102]	Reinforcement of Ontology engineering principle	*Testing component*	To evaluate the efficacy of the proposed mechanism in terms of semantic matching
4	Howell *et al.* [103]	Adapted NeOn methodology	*Complete domain expert participation, logical accuracy, and performance*	A reusable engineering process that can be applied in related domain
5	Yang *et al.* [107]	Automation of ontology design (ontology learning)	*Collection of data and preprocessing, NLP-based lexical analysis, and extraction of ontology components*	They are often incapable to handle implicit terminological and non-terminological relations and data properties
6	Dalal [108]	Enhancement of extreme ontology design methodology	*Accessible, findable, interoperable, and reusable*	There should be a robust mechanism to take care of more complex alignment beyond sub-entities (either class or property)
7	Lu *et al.* [109]	Manufacturing ontology approach (combined RAD and existing methodologies' concepts)	*Integration*	It requires more domain concepts to be integrated
8	Palmirani *et al.* [110]	Methodology for building legal ontology (MeLOn)	*Integration*	It requires more domain concepts (deontic logic models) to be integrated
9	Yunianta *et al.* [111]	Enhancement of existing methodology	*Data integration*	More data to be integrated and detailed evaluation technique

Furthermore, this section discusses a fragment of the driven goals behind the development of robust ontology as presented in Table 6.2. Similarly, in order to open up further research drive, it equally presents some areas of research attentions.

Evidently, from the sample of literature presented in Table 6.2 (for instance, S/N 1, 7, 8, and 9), data integration is largely identified as the driven goal to develop robust ontology. It implies that robustness of ontology could be found on its capacity to integrate heterogeneous complex data either within a given ontological domain or across two and more ontological platforms. Therefore, when complex heterogeneous data are modeled for knowledge representation without compromising the sensitivity to contextual meaning of concepts and firmly establishing its relations among concepts, such can be described as robust ontology design. However, an area of research interest open for further work is on the mechanism that can make it possible to integrate more domain concepts without compromising its sensitivity to the factors mentioned earlier. Besides, the existing models for data integration have not been properly experimented. The confirmation to this open problem is the research effort by S/N 3 of Table 6.2 [102] who adapted an ontology engineering process to work on the testing capacity.

Another interesting goal is on the aspect of modifying an existing ontology development methodology to explicitly include the collaboration of domain expert. The work of Howell *et al.* [103] as shown by S/N 4 modified NeOn methodology to pave way for full participation of domain experts. Other driven goals are to define the granularity of the domain concisely, accessibility, and reusability of ontology. The last two goals are very crucial for ontology mapping either for alignment or merging purpose. Robustness of ontology is also obtained when two or more ontologies are mapped.

In conclusion, this review identifies some goals to develop robust ontology and equally points out some areas of research interventions as highlighted by Table 6.1 to design an ontology that can stand the test of time.

References

[1] Viswanath, S., Guntz, S., Dieringer, J., Vaidyaraman, S., Wang, H., & Gounaris, C. (2020) An ontology to describe small molecule pharmaceutical product development and methodology for optimal activity scheduling. *Journal of Pharmaceutical Innovation*, 1-15.

[2] Steinmetz, C., Rettberg, A., Ribeiro, F. G. C., Schroeder, G., Soares, M. S., & Pereira, C. E., (2018) Using Ontology and Standard Middleware for integrating IoT based in the Industry 4.0. *IFAC-PapersOnLine*, *51*(10), 169-174.

[3] Salem, A. B. M. & Parusheva, S. (2018) Developing a web-based ontology for e-business. *International Journal of Electronic Commerce Studies*, *9*(2), 119-132.

[4] Thangaraj, M. & Sujatha, G. (2014) An architectural design for effective information retrieval in semantic web. *Expert Systems with Applications*, *41*(18), 8225–8233. https://doi.org/10.1016/j.eswa.2014.07.017

[5] Aminu, E. F., Oyefolahan, I. O., Abdullahi, M. B. & Salaudeen, M. T. (2020) A Review on Ontology Development Methodologies for Developing Ontological Knowledge Representation Systems for various Domains. *International Journal of Information Engineering and Electronic Business*, *12*(2), 28–39. https://doi.org/10.5815/ijieeb.2020.02.05

[6] Thomas, C. J. (2009) Just what is an ontology, anyway?' *IT Professional,* 11, 22-27.

[7] Wei, Y., Wang, R., Hu, Y., & Wang, X. (2012) From Web Resources to Agricultural Ontology: a Method for Semi-Automatic Construction'*Journalof Integrative Agriculture*, 11(5): 775-783

[8] Meriyem, C., Adil, S., & Hicham, M. (2015) IT Governance ontology building process: example of developing audit ontology. *International Journal of Computer Techniques*, *2*(1), 134-141

[9] Ivanović, M. & Budimac, Z. (2014) An overview of ontologies and data resources in medical domains. *Expert Systems with Applications*, *41*(11), 5158-5166

[10] Barão, A., de Vasconcelos, J. B. Rocha, A., & Pereira, R. (2017) A knowledge management approach to capture organizational learning networks. *International Journal of Information Management*, 37(6), 735-740.

[11] Gruber, T. R., (1993) A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition,* 5(2): 199-220.

[12] Dou, D., Wang, H. & Liu, H. (2015) Semantic data mining: A survey of ontology-based approaches. In *Semantic Computing (ICSC), 2015 IEEE International Conferenc on* (pp. 244-251). IEEE.

[13] Kang, Y. B., Haghighi, P. D., & Burstein, F. (2014) CFinder: An intelligent key concept finder from text for ontology development. *Expert Systems with Applications*, *41*(9), 4494-4504

[14] Uschold, M. & King, M. (1995) Towards a Methodology for Building Ontologies. In IJCAI95 Workshop on Basic Ontological Issues in Knowledge Sharing. Montreal

[15] Dnyanesh, R., & Rahul, C. A. (2011) Generic ontology development framework for data integration and decision support in a distributed environment. *International Journal of Computer Integrated Manufacturing* Vol. 24, No. 2, 154–170

[16] Wang, R., Nellippallil, A. B. Wang, G., Yan, Y., Allen, J. K. & Mistree, F. (2019) Ontology-based uncertainty management approach in designing of robust decision workflows. *Journal of Engineering Design*, *30*(10-12), 726-757.

[17] Aminu E. F., Oyefolahan I. O., Abdullahi M. B., Salaudeen M. T. (2022) Modeling Competency Questions-Based Ontology for the Domain of Maize Crop: SIMcOnto. In: Mandal J. K., Buyya R., De D. (eds) *Proceedings of International Conference on Advanced Computing Applications*. Advances in Intelligent Systems and Computing, vol 1406. (pp. 751-763). Springer, Singapore. https://doi.org/10.1007/978-981-16-5207-3_61

[18] He, Y., Yu, H., Ong, E., Wang, Y., Liu, Y., Huffman, A., & Smith, B. (2020). CIDO, a community-based ontology for coronavirus disease knowledge and data integration, sharing, and analysis. *Scientific data*, *7*(1), 1-5.

[19] Pratibha, G., Sangeeta, D. & Bhanumurthy, K. (2011). Ontology Development Methods. DESIDOC Journal of Library & Information Technology, Vol. 31, No. 2, pp. 77-83

[20] Agyapong-Kodua, K., Niels, L., Robert, D., & Svetan, R. (2013) 'Review of semantic modeling technologies in support of virtual factory design' *International Journal of Production Research,* Vol. 51, No. 14, 4388–4404

[21] Noy, N. F. and McGuinness, D. L. (2001) 'Ontology Development 101: A guide to Creating Your First Ontology', Stanford University, Stanford, CA, USA, pp. 1-25.

[22] Noy, N. F., & McGuinness, D. L. (2016). Ontology development 101: a guide to creating your first ontology Stanford University.

[23] Walisadeera, A. I., Ginige, A., & Wikramanayake, G. N. (2015) 'User centered ontology for SriLanka farmers' *Ecological Informatics*, 140-150

[24] Bhaskar, K. & Savita S. (2010) 'A Comparative Study Ontology Building Tools for Semantic Web Applications' International journal of Web

& Semantic Technology *(IJWesT)* Vol.1, Num.3, July 2010 DOI : 10.5121/ijwest.2010.1301 1

[25] Ast, M., Glas, M., Roehm, T., & Luftfahrt, V. B. (2014). *Creating an ontology for aircraft design*. Deutsche Gesellschaft für Luft-und Raumfahrt-Lilienthal-Oberth eV

[26] Munir, K., & Anjum, M. S. (2018). The use of ontologies for effective knowledge Modeling and information retrieval. *Applied Computing and Informatics*, *14*(2), 116-126

[27] Bechhofer, S. (2009) "OWL: Web ontology language." *Encyclopedia of database systems*. Springer, Boston, MA, 2009. 2008-2009

[28] Richard F., Patrick H., & Ian H. (2004). OWL-QL—a language for deductive query answering on the Semantic Web. Web Semantics: Science, Services and Agents on the World Wide Web 2 19–29

[29] Cardoso, J., & Pinto, A. M. (2015). The Web Ontology Language (OWL) and its Applications. In *Encyclopedia of Information Science and Technology, Third Edition* (pp. 7662-7673). IGI Global

[30] Sengupta, K., & Hitzler, P. (2014). Web ontology language (OWL). *Encyclopedia of Social Network Analysis and Mining*, 2374-2378.

[31] Arenas, M., Grau, B. C., Kharlamov, E., Marciuška, Š., & Zheleznyakov, D. (2016). Faceted search over RDF-based knowledge graphs. *Journal of Web Semantics*, *37*, 55-74

[32] Calbimonte, J. P., Mora, J., & Corcho, O. (2016, May). Query rewriting in RDF stream processing. In *European Semantic Web Conference* (pp. 486-502). Springer, Cham

[33] Jeen B., Michel K., Stefan D., Dieter F., Frankvan H., & Ian H. (2002). Enabling knowledge representation on the Web by extending RDF Schema. *Computer Networks* 39 609–634

[34] Khan, J. A., & Kumar, S. (2014, October). Deep analysis for development of RDF, RDFS and OWL ontologies with protege. In *Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization* (pp. 1-6). IEEE

[35] Schenner, G., Bischof, S., Polleres, A., & Steyskal, S. (2014, September). Integrating Distributed Configurations With RDFS and SPARQL. In *Configuration Workshop* (Vol.1220, pp. 9-15)

[36] Rodríguez, N. D., Cuéllar, M. P., Lilius, J., & Calvo-Flores, M. D. (2014). A survey on ontologies for human behavior recognition. *ACM Computing Surveys (CSUR)*, *46*(4), 43

[37] Slimani, T. (2015). Ontology development: A comparing study on tools, languages and formalisms. *Indian Journal of Science and Technology*, *8*(24), 1-12

[38] Aminu, E. F. & Adewale, O. S. (2015) 'A Mechanism for Detecting Dead URLs in XTM-Based Ontology Repository' *International Journal of Computer Applications* (0975 – 8887) Volume 111 – No 12

[39] Carroll, J., Herman, I., & Patel-Schneider, P. F. (2015). OWL 2 web ontology language RDF-based semantics. *W3C Recommendation (October 27, 2009)*

[40] Hacherouf, M., Bahloul, S. N., & Cruz, C. (2015). Transforming XML documents to OWL ontologies: A survey. *Journal of Information Science*, *41*(2), 242-259

[41] John, H. G., Mark, A. M., Ray, W. F., Williams, E. G., Monica, C., Henrik, E., Natalya, F. N. & Samson W. T. (2003). The evolution of Protégé: an environment for knowledge-based systems development. *International Journal of Human-Computer Studies*, Vol. 58, Issue1, Pages 89 – 123

[42] Chujai, P., Kerdprasop, N., & Kerdprasop, K. (2014). On transforming the ER model to ontology using protégé OWL tool. *International Journal of Computer Theory and Engineering*, *6*(6), 484

[43] Wohlgenannt, G., Sabou, M., & Hanika, F. (2016). Crowd-based ontology engineering with the uComp Protégé plugin. *Semantic Web*, *7*(4), 379-398

[44] Panita, Y., Dussadee, T., Thanapat, S., Asanee, K., Sachit, R., Margherita, S., & Johannes K. (2008). The AGROVOC Concept Server Workbench: A Collaborative Tool for Managing Multilingual Knowledge. *World Conference on Agricultural Information and IT*

[45] John D., Midori A. H., Melissa Haendel & Suzanna L. (2007). OBO-Edit—an ontology editor for biologists. *Bioinformatics*, Volume 23, Issue 16, 15, Pages 2198–2200

[46] Aditya, K., Bijan, P., Evren, S., Bernardo, C. C. & James, H. (2006) 'Swoop: A Web Ontology Editing Browser' *Web Semantics: Science, Services and Agents on the World Wide Web* Vol. Issue 2, 144–153

[47] Hogan, W. R., Wagner, M. M., Brochhausen, M., Levander, J., Brown, S. T., Millett, N., & Hanna, J. (2016). The Apollo Structured Vocabulary: an OWL2 ontology of Phenomena in infectious disease epidemiology and population biology for use in epidemic simulation. *Journal of biomedical semantics*, *7*(1), 50.

[48] Anikin, A., Litovkin, D., Kultsova, M., Sarkisova, E., & Petrova, T. (2017, September). Ontology visualization: Approaches and software tools for visual representation of large ontologies in learning. In Conference on Creativity in Intelligent Technologies and Data Science (pp. 133-149). Springer, Cham

[49] Alatrish, E. S. (2013) "Comparison some of ontology." *Journal of Management Information Systems* 8.2 (2013): 018-024

[50] García-Peñalvo, F. J., Ordónez de Pablos, P., García, J., & Therón, R. (2014). Using OWL-VisMod through a decision-making process for reusing OWL ontologies. *Behaviour & Information Technology*, *33*(5), 426-442.

[51] Corcho, O., Fernández-López, M., & Gómez-Pérez, A. (2003). Methodologies, tools and languages for building ontologies. Where is their meeting point?. *Data & knowledge engineering*, *46*(1), 41-64.

[52] Nanda, J., Simpson, T. W., Kumara, S. R., & Shooter, S. B. (2006). A methodology for product family ontology development using formal concept analysis and web ontology language. Journal of computing and information science in engineering, 6(2), 103-113.

[53] Gavrilova, T. & Gladkova, M. (2014). Big data structuring: the role of visual models and ontologies. *Procedia Computer Science* 31, 336 – 343

[54] Vigo, M., Bail, S., Jay, C., & Stevens, R. (2014). Overcoming the pitfalls of ontology authoring: Strategies and implications for tool design. *International Journal of Human*-Computer Studies, 72(12), 835-845.

[55] Lenat, D. B., & Guha, R. V. (1990). Building Large Knowledge-based Systems: Representation and Inference in the Cyc Project. Boston, MA: Addison-Wesley.

[56] Dean, J., Trevor, B. C. & Pepijn V. (1998). Methodologies for Ontology Development. 62-75

[57] Fernandez-Lopez, M. Gomez-Perez, A., Pazos-Sierra, A., & Pazos-Sierra, J. (1999) Building a chemical ontology using METHONTOLOGY and the ontology design environment', *IEEE Intelligent Systems & their applications* 4 (1)37–46

[58] Oscar C., Mariano F., and Asuncion G. (2003) 'Methodologies, tools and languages for building ontologies. Where is their meeting point?' *Data & Knowledge Engineering* 46, 41–64

[59] Rizwan, I., & Aida, M. (2013). An experience of developing Quran ontology with contextual information support. *Multicultural Education & Technology Journal* Vol. 7 No. 4, pp. 333-343.

[60] Tan, H., Ismail, M., Tarasov, V., Adlemo, A., & Johansson, M. (2016). Development and evaluation of a software requirements ontology. In *7th International Workshop on Software Knowledge-SKY 2016 in conjunction with the 9th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management-IC3K 2016, November 9-10, 2016, in Porto, Portugal* (pp. 11-18). SciTePress.

[61] Zhou, Z., Goh, Y. M., & Shen, L. (2016). Overview and analysis of ontology studies supporting development of the construction industry. *Journal of Computing in Civil Engineering*, *30*(6), 04016026.Soner, K., Ozgur, A., Orkunt, S., Samet, A., Nihan, K. C. & Ferda, N. A. (2012) 'Ontology-Based Retrieval System using Semantic Indexing' *Information Systems* 37, 294–305

[62] Zhanjun L., Victor R. & Karthik, R. (2007). A Methodology of Engineering Ontology Development for Information Retrieval. *International Conference on Engineering Design, Iced* 07 28 - 31, Paris, France

[63] Obrst, L., Chase, P., & Markeloff, R. (2012, October). Developing an Ontology of the Cyber Security Domain. In *STIDS* (pp. 49-56).

[64] Simperl, E., & Luczak-Rösch, M. (2014). Collaborative ontology engineering: a survey. *The Knowledge Engineering Review*, *29*(1), 101-131.

[65] Stojadinović, S. M., & Majstorović, V. D. (2014). Developing engineering ontology for domain coordinate metrology. *FME Transactions*, *42*(3), 249-255.

[66] Godspower, O. E. and Esingbemi, P. E. (2016). Ontology for Alleviating Poverty among Farmers in Nigeria. *INFOS '16*, May 09-11, 2016, Giza, Egypt

[67] Gruninger, M., & Fox, M. S. (1995). Methodology for the Design and Evaluation of Ontologies. Workshop on Basic Ontological Issues in Knowledge Sharing, Montreal.

[68] Hitzler, P., Gangemi, A., & Janowicz, K. (Eds.). (2016). *Ontology Engineering with Ontology Design Patterns: Foundations and Applications* (Vol. 25). IOS Press.

[69] Kim, H. M., & Laskowski, M. (2018). Toward an ontology-driven blockchain design for supply-chain provenance. *Intelligent Systems in Accounting, Finance and Management*, *25*(1), 18-27.

[70] Hadjar, K. (2016). University Ontology: A Case Study at Ahlia University. In *Semantic Web* (pp. 173-183). Springer, Cham.

[71] Bilgin, G., Dikmen, I., & Birgonul, M. T. (2018). An ontology-based approach for delay analysis in construction. *KSCE Journal of Civil Engineering*, *22*(2), 384-398

[72] El-Diraby, T. E. & Osman, H. (2011). A domain ontology for construction concepts in urban infrastructure products. *Automation in Construction* 20, 1120–1132

[73] Gómez-Pérez, A., Fernández-López, M., & Corcho, O. (2004). Methodologies and methods for building ontologies. *Ontological Engineering: with examples from the áreas of Knowledge Management, e-Commerce and the Semantic Web*, 107-197

[74] Hafedh, N., Mohamed, F., Imed R. F. & Basel S. (2014). A Critical Analysis of lifecycles and Methods for Ontology Construction and Evaluation' *1st International Conference on Advanced Technologies for Signal and Image Processing* - ATSIP'2014

[75] Haghighi, P. D., Burstein, F., Zaslavsky, A., & Arbon, P. (2013). Development and evaluation of ontology for intelligent decision support in medical emergency management for mass gatherings. *Decision Support Systems*, *54*(2), 1192-1204.

[76] Janowicz, K. (2012). Observation-driven geo-ontology engineering. *Transactions in GIS*, *16*(3), 351-374.

[77] Jinsoo, P., Kimoon, S., & Sewon, M. (2008). Developing Graduation Screen Ontology based on the METHONTOLOGY Approach. *Fourth International Conference on Networked Computing and Advanced Information Management*.

[78] Guo, B. H., & Goh, Y. M. (2017). Ontology for design of active fall protection systems. *Automation in Construction*, *82*, 138-153.

[79] Noman, I., Abu, Z. A. and Zubair, A. S. (2010) 'Semantic Web: Choosing the Right Methodologies, Tools and Standards', *Information and Emerging Technologies (ICIET), 2010 International Conference on*. IEEE, 2010.

[80] Rizwan, I., Masrah A. A. M., Aida, M. & Nurfadhlina, M. S. (2013). An Analysis of Ontology Engineering Methodologies: A Literature Review. *Research Journal of Applied Sciences, Engineering and Technology* 6(16): 2993-3000

[81] Gavrilova, T. A., & Leshcheva, I. A. (2015). Ontology design and individual cognitive peculiarities: A pilot study. *Expert systems with Applications*, *42*(8), 3883-3892.

[82] Ibrahim, A. A., Abdul, A. A. G., Wan, N. W. R. & Rodziah, A. (2014). A Comparative Study on Ontology Development Methodologies towards Building Semantic Conflicts Detection Ontology for Heterogeneous Web Services. *Research Journal of Applied Sciences, Engineering and Technology* 7(13): 2674-2679

[83] De Nicola, A., M., Missikoff & Navigli, R. (2009) 'A Software Engineering Approach to Ontology Building' *Inform. Syst.*, 34(2): 258-275.

[84] del Águila, I. M., Palma, J., & Túnez, S. (2014). Milestones in software engineering and knowledge engineering history: A comparative review. *The Scientific World Journal*, *2014*.

[85] Jain, V., & Singh, M. (2013). Ontology development and query retrieval using protégé tool. *International Journal of Intelligent Systems and Applications*, *9*, 67-75.

[86] Rao, L., Mansingh, G., & Osei-Bryson, K. M. (2012). Building ontology based knowledge maps to assist business process re-engineering. *Decision Support Systems*, *52*(3), 577-589.

[87] Serna, E., & Serna, A. (2014). Ontology for knowledge management in software maintenance. *International Journal of Information Management*, *34*(5), 704-710.

[88] Tiffani J. B., Furuya E. Y., Gilad J. K., James J. C., & Suzanne B. (2012). Development and evaluation of an ontology for guiding appropriate antibiotic prescribing', *Journal of Biomedical Informatics* 45 120–128

[89] Arp R. (2009) 'Practical steps in building a domain ontology, models and simulations 3: emergence, computation, and reality' Virginia: Charlottesville.

[90] Chen-Huei C., Fatemeh M. Z., & Huimin Z. (2011). Ontology for Developing Web Sites for Natural Disaster Management: Methodology and Implementation. *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, Vol. 41, No. 1

[91] Uschold, M & Grüninger, M, (1996). Ontologies: principles methods and applications', *Knowledge Engineering Review*11(2)93–137.

[92] Ceccaroni, L, Cortés, U & Sánchez-Marré, M, (2000) 'WaWO – an ontology embedded into an environmental decision-support system for wastewater treatment plant management' *Workshop on Applications of Ontologies and Problem solving Methods.14th European Conference on ArtificialIntelligence*(ECAI'00)2–1–2–9

[93] Bonanci, R., Nabuco, O. F. & Junior, I. P. (2016). Ontology models of the impacts of agriculture and climate changes on water resources: Scenarios on interoperability and information recovery' *Future Generation Computer Systems*. 54 (2016) 423–434

[94] Aree, T., Asanee, K., Supamard, P. & Uamporn, V. (2009) 'Ontology Development: A Case Study for Thai Rice' *Kasetsart J.* (Nat. Sci.) 43 : 594 – 604

[95] Niu, J., & Issa, R. R. (2015). Developing taxonomy for the domain ontology of construction contractual semantics: A case study on the AIA A201 document. *Advanced Engineering Informatics*, *29*(3), 472-482.

[96] Gregor, D., Toral, S., Ariza, T., Barrero, F., Gregor, R., Rodas, J., & Arzamendia, M. (2016). A methodology for structured ontology construction applied to intelligent transportation systems. *Computer Standards & Interfaces*, *47*, 108-119.

[97] Dutta, B., Chatterjee, U., & Madalli, D. P. (2015). YAMO: yet another methodology for large-scale faceted ontology construction. *Journal of Knowledge Management*, *19*(1), 6-24.

[98] Zeb, J., Froese, T., & Vanier, D. (2015). An ontology-supported asset information integrator system in infrastructure management. *Built Environment Project and Asset Management*, *5*(4), 380-397.

[99] Mezghani, E., Exposito, E., & Drira, K. (2016). A collaborative methodology for tacit knowledge management: Application to scientific research. *Future Generation Computer Systems*, *54*, 450-455.

[100] Keet, C. M., Ławrynowicz, A., d'Amato, C., Kalousis, A., Nguyen, P., Palma, R., ... & Hilario, M. (2015). The data mining OPtimization ontology. *Journal of web semantics*, *32*, 43-53.

[101] Rayyaan, R., Wang, Y., & Kennon, R. (2014). Ontology-based interoperability solutions for textile supply chain. *Advances in Manufacturing*, *2*(2), 97-105.

[102] Tebes, G., Olsina, L., Peppino, D., & Becker, P. (2021). Specifying and Analyzing a Software Testing Ontology at the Top-Domain Ontological Level. *Journal of Computer Science & Technology*, *21*.

[103] Howell, S., Beach, T., & Rezgui, Y. (2021). Robust requirements gathering for ontologies in smart water systems. *Requirements Engineering*, *26*(1), 97-114.

[104] John, S., Shah, N., Stewart, C. D., & Samlov, L. (2017, November). Software Centric Innovative Methodology for Ontology Development. In *KEOD* (pp. 139-146).

[105] Smirnov, A., Levashova, T., Ponomarev, A., & Shilov, N. (2021). Methodology for Multi-Aspect Ontology Development: Ontology for Decision Support Based on Human-Machine Collective Intelligence. *IEEE Access*, *9*, 135167-135185

[106] Sulaeman, M. M., & Harsono, M. (2021). Supply Chain Ontology: Model Overview and Synthesis. *Jurnal Mantik*, *5*(2), 790-799.

[107] Yang, L., Cormican, K., & Yu, M. (2020). Ontology learning for systems engineering body of knowledge. *IEEE Transactions on Industrial Informatics*, *17*(2), 1039-1047.

[108] Dalal, A. (2020). Modular ontology modeling meets upper ontologies: the upper ontology alignment tool.

[109] Lu, Y., Wang, H., & Xu, X. (2019). ManuService ontology: a product data model for service-oriented business interactions in a cloud manufacturing environment. *Journal of Intelligent Manufacturing*, *30*(1), 317-334.

[110] Palmirani, M., Martoni, M., Rossi, A., Bartolini, C., & Robaldo, L. (2019). Pronto: Privacy ontology for legal reasoning. In *International Conference on Electronic Government and the Information Systems Perspective* (pp. 139-152). Springer, Cham.

[111] Yunianta, A., Basori, A. H., Prabuwono, A. S., Bramantoro, A., Syamsuddin, I., Yusof, N., ... & Alsubhi, K. (2019). OntoDI: The methodology for ontology development on data integration. *Int. J. Adv. Comput. Sci. Appl.*, *10*(1), 160-168.

[112] Chen, J., Hu, P., Jimenez-Ruiz, E., Holter, O. M., Antonyrajah, D., & Horrocks, I. (2021). Owl2vec*: Embedding of owl ontologies. *Machine Learning*, *110*(7), 1813-1845.

[113] Smaili, F. Z., Gao, X., & Hoehndorf, R. (2018). Onto2vec: joint vector-based representation of biological entities and their ontology-based annotations. *Bioinformatics, 34*(13), i52–i60.

[114] Smaili, F. Z., Gao, X., & Hoehndorf, R. (2018). OPA2Vec: combining formal and informal content of biomedical ontologies to improve similarity-based prediction. *Bioinformatics, 35*(12).

[115] Wang, Y., & Wang, Y. (2018). Citrus ontology development based on the eight-point charter of agriculture. *Computers and Electronics in Agriculture*, *155*(October), 359–370. https://doi.org/10.1016/j.compag.2018.10.034

[116] Jaskó, S., Skrop, A., Holczinger, T., Chován, T., & Abonyi, J. (2020). Development of manufacturing execution systems in accordance with Industry 4.0 requirements: A review of standard-and ontology-based methodologies and tools. *Computers in industry*, *123*, 103300.

[117] Sicilia, M. A., Rodriguez, D., Garcia-Barriocanal, E., Sanchez-Alonso, S., 2012. Empirical findings on ontology metrics. Expert Syst. Applicat. 39 (8), 6706–6711