International Journal of Future Generation Communication and Networking
Vol. 13, No. 4, (2020), pp. 2384–2401

Generating User Guides For Ontologies: A Domain And Schema Independent Verbalisation Approach

Kaneeka Vidanage[1], Noor Maizura Mohamad Noor[1], Rosmayati Mohemad[1] and Zuriana Abu Bakar[1]
Faculty of Ocean Engineering Technology and Informatics, University Malaysia Terengganu – UMT
E-Mail: kaneeka.online@gmail.com

Abstract

Ontology development is an incremental and iterative task. In that atmosphere, manual creation/maintenance of a user guide for an ontology will be difficult and require much effort for both ontologists and domain specialists. However, maintaining a user guide documentation throughout and after the ontology construction process has many advantages (discussed in the paper). Verbalisation (the process of converting technical encodings in an ontology to natural language) is the technical process that can be introduced to automate the user guide construction process. Existing verbalisation mechanisms have numerous shortcomings. In this research, a novel verbalisation mechanism is introduced, addressing existing shortcomings to facilitate the iterative and incremental growth of the user guide along with the ontology's evaluation

Keywords: *AliceBot, Ontology, User-Guide, Verbalisation*

1. Introduction

Ontology construction is not a one-time task. It`s an iterative and incremental operation. Both domain specialists and ontologists need to work with the mutual realization to achieve the goal of an error-free ontology [1-2; 3-5]. If an ontology can be accompanied by a user guide, there are many advantages associated with it. One such usage is the ability to utilize the user guide for knowledge verification. Ontology creation is an iterative and incremental task. Therefore, at the end of each iteration, an ontology increment will be produced. Each ontology increment will contain a specific set of axioms associated with the competency questions derived from the domain of concern
[6-7].

Suggested practice in collaborative ontology engineering principals is to cross-reference and verify the accuracy of the residing axioms of the ontology increment with the mutual involvement of the domain specialists and the ontologies. Domain specialists, being non-computing specialists, mostly will not be in a position to comprehend the RDF or the OWL versions of the ontology increments [1-2; 3-5]. Hence, if there is a mechanism available to convert the technical semantics residing inside the ontology increment to a comprehensible English language in the form of a user guide, domain specialists can examine the contents of the user guide and approve the accuracy of the knowledge embeddings of the ontology increment. Or else they can reject it and request for further refinements. This is a vital benefit which can be obtained from a user guide [6-7].

Subsequently, if the latest user guide for the ontology increment version is available, ontologists can examine through it to determine the competency scope of it. This decision will be useful in determining the appropriateness of the current ontology increment for a differential purpose [6-7].
Likewise, there are several critical advantages available, if accurate user guide is available for the latest version of the ontology increment. But as aforementioned, ontology construction being an iterative and incremental operation, it`s not realistic to modify the user guide, manually to

ISSN: 2233-7857 IJFGCN

synch with the latest version of the ontology increment. This will result in version mismatches of the latest ontology increment and the user guide. This can be pointed out as a critical problem, as it will hinder the derivation of the aforementioned advantages from the user guides.

In this research novel algorithm is proposed to generate the user guides of the ontology increments in a fully automated manner with no need of any domain or schema associated prior configurations. This proposed algorithm ensures the generation of the user guide in human understandable English, despite the domain and schemata. The proposed algorithm is tested quantitatively and qualitatively for its accuracy

2. Literature Review

Verbalization is the process of converting technical encodings into understandable natural language (i.e. English). Verbaliser is the piece of software with that capability. After examining the existing verbalisers for the auto-generation task of the user guide, multiple shortcomings were found. Extensive configuration effort, statically mapping the verbaliser to the ontology schemata, and the need for semantic annotations for concept realisation are some of the critical bottlenecks associated with the existing verbalisers [14-16].

a) Shortcomings of the existing verbalisation process

Technically, verbalisation is defined as the process of translating axioms defined in ontology to natural language [10-11]. Most of the existing verbalisation systems rely on the complex Natural Language Generation (NLG) pipeline to convert axioms into the natural language

[17]. This is a complex technological pipeline where all the phases need to be accurately fulfilled to obtain an understandable natural language output. Namely, those steps are defined as content selection, discourse planning, lexicalization, aggregation, generation of referring expressions, and finally linguistic realisation [14].

Among those steps, the discourse planning step is vital to achieve coherent verbalisation output. Discourse planning utilises the 'Rhetoric Structure Theory (RST)' for the coherent organisation of the text [18]. RST is based on the two main conceptions of nucleus and satellite. Nucleus represents the significant axioms associated with the considered domain, and satellite represents the associative properties linked with the nucleus that are required to elaborate the nucleus [18]. Therefore, if the identification of the nucleus and satellite did not occur in a domain specific manner, it will adversely affect the clarity of the verbalised contents [17, 19]. For that reason, there is a manual phase with the domain specialist and the ontology engineer to properly assign weights to the axioms defined in the domain considered. Afterwards, with the help of the pre-defined rule sets, it will automate the RST, assuring appropriate discourse planning, leading towards accurate and coherent verbalisation.

The problem that arises is that the same verbaliser cannot be used for any other domain without doing the above defined prior configurations, which is referred to as a portal configuration. This makes a verbalisation-ready framework always become domain-dependent [14].

The next restriction is the necessity of annotations to enrich the semantic realisation of the concepts in the knowledge model. Again, this requires additional effort from the ontology engineers, and in most cases relevant foundational de-facto standard meta-models are (i.e. Dublin Core, FOAF) needed to be incorporated in the knowledge model. This is because the existing verbalisation frameworks are configured to link with only the pre-defined annotated endpoints of those de-facto standard meta-models. This poses an additional overload as well as acts as a modelling restrictor [15-16].

As a final disadvantage, it can be pointed out that most of the existing verbalisers produce Control Natural Language (CNL) that resembles the assembly language and is not the colloquial English understood by laymen. Therefore, another Natural Language Processing (NLP) layer has to be introduced to overcome the barrier of converting technical English constructs to its colloquial format. One of the main causes for this is that the existing evolution of verbalisation has evolved to the level of Attempt to Control English (ACE), which is a form of Control Natural Language (CNL). In CNL, verbalisers attempt to extract the triple formulations in the ontology and convert them exactly into English, where the contextualised connectivity and colloquialism will be lost [20-22].

That is the main reason for the verbalised output to look very primitive and the flow to seem inconvenient to interpret by the end-user. The most critical deficiency associated with verbalisers is that they are schema and domain specific [23-25]. Hence, it can be easily concluded that through the existing verbalisation mechanisms, the aforementioned research gaps of domain-dependence, excessive human involvement associated with configurations and CNL less colloquialisms are not properly resolved.

b) Existing Systems

i. NaturalOWL [17]:

NaturalOWL is one of the sophisticated Ontology Web Language (OWL) based verbalisation systems currently available. NaturalOWL is capable of working with any domain. But the most important prerequisite is, before using the NaturalOWL on a particular domain, it has to be configured. This is referred to as portal ontology configuration. The portal configuration of NaturalOWL needs to be conducted by a domain specialist via using a tool like Protégé. Therefore, there is an excessive skills bottleneck, hence the expert should be a domain specialist who is literate on semantic concepts as well. That is unless at least two human resources (i.e. ontologists and domain specialists) are utilised. The purpose of this portal ontology configuration is to make the NaturalOWL familiar with the domain specific natural language resources. This is not an easy process; hence the domain specialist needs to assign an importance score for the potential domain specific natural language resources.

At the time of the verbalisation, NaturalOWL then uses appropriate lexicons to efficiently carry out the verbalisation task, unless the verbalisations generated are excessively complex, thus producing incomprehensible text.

NaturalOWL uses a NLG architecture as its backbone. It is a pipeline architecture with the series of processing steps of document-planning, micro-planning, and surfacerealisation. The requirement of the properly configured portal ontology becomes essential to perform each of these tasks in the pipeline effectively and to produce high quality verbalised English output.

In literature, NaturalOWL is defined as a sophisticated verbaliser. But the configuration effort is very high, which acts as a trade-off decision in selecting NaturalOWL for the verbalisation tasks. The necessity of the current research problem is for a verbaliser capable of working in a domain and schema independent manner, with no human configuration effort. But as already elaborated, it can be concluded that NaturalOWL does not cater to the expectation of this research.

ii. LODE [26]

Live OWL Documentation Environment (LODE), created [26] is another popular tool capable of verbalising an ontology. More than a tool, this is a live web service. Therefore, if the end-user plans to use LODE for verbalising his/her ontology, that ontology must have been published and available on the web. This is because the input requested by LODE is a Cool-URI (Berners-Lee et al., 2015) of the ontology to be verbalised as introduced by [26]. The concept of Cool- URI [27] expects several criteria to be fulfilled by web resources. Tim-Bernes-Lee has declared that the resource must be available on the web; the URI used to encode the resource must be simple; it has to be stable, even ten years passed; and URI must be in a manageable structure [27].

The pseudo-URL structure expected by LODE is in the form of https://w3id.org/lode/optionalparameters/published_ontology_url. Therefore, this could be pointed out as one restriction in using LODE, because the requirement of this research emphasises ontologies under development. It is a waste to upload an intermediate increment of the ontology to the web, as it is not the latest and accurate version. Additionally, it is irresponsible as it contributes to the untidiness on the web [28]. As another restriction, [26] point out the annotations such rdfs:comment, rdfs:label, owl:versionIRI, dc:publisher and other possible descriptive annotations must be filled with useful information in the semantic script of the OWL ontology.

This is because LODE is programmed to extract information from annotation tags and represent it as the verbalised content using XSLT technology. Consequently, if the semantic script of the ontology does not contain annotation tags and explanations, it will adversely affect the verbalisation process of the LODE. Therefore, this acts as another restriction in deriving the potential use of LODE for the current research. Similarly, there are several other tools developed on top of the LODE such as OnToology [29] and Widoco [30] which also have the same set of issues.

Need for the ontology resource to be available in the web and the necessity of additional annotations for semantic realisations can be regarded as glitches associated with LODE.

iii. MIKAT [31]

Medical Imaging and Advanced Knowledge Technologies (MIKAT) is another verbaliser which can function only on the breast cancer domain, because this verbaliser is statically attached to the breast cancer domain ontology. Therefore, this verbaliser cannot have the facility to operate as a domain-independent verbaliser. Similarly, there are several other verbalisers such as [32] and [33] which are also statically attached to a fixed domain ontology and cannot function in a domain and schema in an independent manner.

c) Reflection

Through the literature analysis conducted above, it is apparent that existing verbalisation mechanisms and tools are domain and schema dependent and they cannot be used repeatedly to generate the synchronised versions of the user guides of the iterative and incremental expansions conducted on the ontology under construction.

Therefore, it is decided to experiment on the potential of using a chatbot to function as a verbaliser because chatbots are the ideal technology for colloquial statement generation which also coincide with the necessity of user guide generation for the ontologies.
As already conversed, existing verbalization mechanisms based on RST and NLG principals have resulted in numerous complexities. Those shortcomings directed to seek for another new dimension which is appropriate to the layman comprehensible text generation to be included in the user guide. This paved the path to investigate more on chatbot technologies. Therefore,

ISSN: 2233-7857 IJFGCN

initially multiple chatbot technologies are reviewed (refer Table 1) to find the best choice suited to the purpose of this research.

As visible in Table 1 (below), among the technologies reviewed, most require a domain-specified training phase. This attribute makes those technologies tightly coupled with the relevant domain and extensive manual training effort also can be regarded as a trade-off. .IBM Watson has a steep learning curve; premium functionalities are costly and external integrations are challenging. Artificial Intelligent Markup Language (AIML) is the only technology that contains attributes in favor of the current research's requirements. AIML lacks a data specific training phase. AIML's Artificial Linguistic Internet Computer Entity (ALICE) bot derives its knowledge from AIML scripts [39]. If AIML scripts can be auto-generated, AIML's ALICE bot can function as a domain and schema independent resolution. Additionally, AIML is easily expandable and external integrations are also very well supported. Considering all those outcomes, it is decided that AIML technology will be used as a chatbot mechanism for this research. An algorithm has to be introduced to automate the AIML scripts' creation and alter its functionality to work as a verbaliser instead of a chatbot. Those aspects will be discussed in detailed in the forthcoming sections of this paper.

Technology	Remark
✓ DialogFlow [34,35]	✓ Training requirement is a dire constraint that needs human involvement for different contexts (done via manually typing potential phrases for the training) ✓ Basis is on Machine Learning & domain specific training → *Makes it a domain-dependent solution, as a dataset will be domain specific* ✓ Regular expression for pattern matching support is not available ✓ Advanced features are not freely available ✓ Can integrate with only one web hook – interaction with external knowledge bases will not be an easy task.
✓ RAZA[34]	✓ Write domain specific stories – Manual training task. →*Makes it a domain-dependent solution, as a dataset will be domain specific* ✓ Training the dialogue model – Expose the domain specific user stories for training purposes ✓ Complex and learning curve for the usage ✓ Memory hungry technology – which will slow the machine performance
✓ IBM Watson [36]	✓ Integration with third party resources is difficult ✓ Steep learning curve ✓ Costly
AIML / ALICE [37, 38]	✓ No datasets required for training purposes ✓ Intelligence is extracted from the knowledge scripts – can be

	auto-generated via axioms extraction from the ontology ✓ Manual integrations and expansions are also supported ALICE contains a robust collection of AIML scripts to make the bot more intelligent ✓ Freely available ✓ Easy to use ✓ A lot of potential for external integrations ✓ Stimulus response model can be used to organize knowledge ✓ Regular expression for pattern matching support is available

Table 1 Chatbot technology comparison

3. Methodology

Design Science Research Methodology is used govern the flow of activities conducted throughout this research. The work-flow associated with the Design Science Research Methodology is depicted in figure 1.

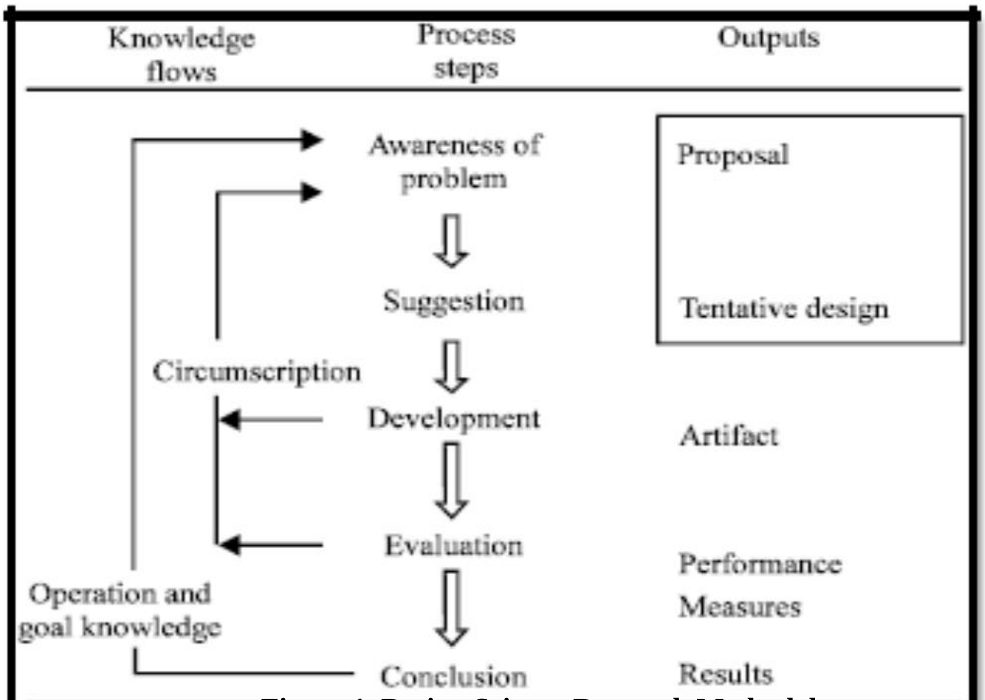


Figure 1. Design Science Research Methodology

As per the phases of the Design Science Research Methodology, initial step is to establish the awareness about the problem of concern. Through a systematic literature review conducted, problem of concern is justified. Existing systems and their shortcomings were reviewed. Potential technologies appropriate for the development of a solution is reviewed as depicted above in table 1. Ultimately, as justified in table 1 Artificial Intelligent Markup Language (AIML) is chosen as the main medium for the development, amidst the other alternatives available. The high level work-flow of the proposed algorithm can be depicted as in figure 2.

Figure 2. High Level flow of the proposed algorithm

The pseudocodes associated with the operations executed in each phase of the algorithm can be elaborated as mentioned below.

Phase – I [Knowledge Extraction]

Start
Upload relevant RDF / OWL file of the ontology increment to be tested.
Conduct format verification either as a RDF / OWL file.
According to the verified format trigger the relevant RDF / OWL extraction logic.
While [Until EOF]
    Extract class info from the ontology increment file.
    [Inheritance classes info / Disjoint classes info / Equivalent classes info and etc.]
    Extract property associated information from the ontology increment file
    [Data properties / Object properties]
    IF [Individuals Exists]
    Extract Individual specific properties [Data / Object properties]
    End IF
    Store extracted information in the relevant relations of the DB schema without violating the mapping sequences.
End While
End

The first segment of the algorithm is responsible for the knowledge extraction. RDF or the OWL format of the stipulated ontology increment will be uploaded to the system. Henceforth, the first phase of the algorithm will execute and it will trigger the format specific extraction logic. REGEX (Regular Expressions) based extraction rules are written to extract each of the important semantic elements residing inside the ontology increment. The sematic elements could be the classes, inheritance relationship mappings and variety of class relationship mappings, data and object properties, individual specific properties and etc. Afterwards the extracted semantic elements are stored in the relevant relations of the database schema proposed to store the extracted information. A snapshot of the proposed database schemata is included in the figure 3 below.

ISSN: 2233-7857 IJFGCN

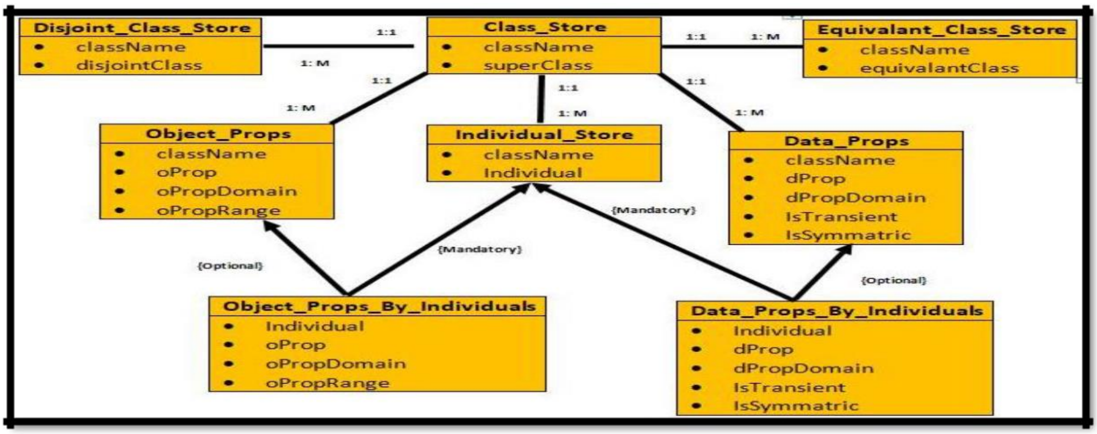


Figure 3. Database Schema

Phase – II [Auto generate AIML template]

Start

Extract stored facts from the DB schema.

Stow them separately in different array list objects.

[One array list for inheritance mappings, another for disjoint mappings, another for data properties, another for object properties and likewise…..]

Initialize and write the general structure of the AIML file

<xml>

<aiml>

Start appending extracted semantic elements as separate categories to the initialized AIML file.

ArrayList<String> inheritenceMappingsArr=new ArrayList<> () // For example extracting inheritance mappings

I=0

While [I>=inheritenceMappingsArr.size ()] // Begin appending

<category>

<pattern>Inheritance Class Infor </pattern>

<template> inheritenceMappingsArr.get (i) </template>

</category>

I++

End While

Repeat the same process for all other semantic elements as well and append those as separate categories to the AIML file.

Once writing of all semantic elements are completed in the AIML file:

Close the AIML file

</aiml>

</xml>

End

Second phase of the algorithm is responsible for the creation of the AIML template file associated with the ontology increment of concern. In accomplishing this requirement, the

Google`s AliceBot engine`s code is customized. Originally Google`s AliceBot is functioning as a chatterbot based on the intelligence provided to it from the AIML template. Here, in this research, AliceBot`s AIML template is auto generated based on the contents extracted from the ontology increment. As already stated, originally AliceBot is a chatter bot. Therefore, semantic element labels residing inside the ontology increment are submitted to the AliceBot engine as requests one after another. Then the AliceBot engine will extract the required intelligence from the relevant categories residing in the auto generated AIML template and the responses are generated. These generated responses are considered as the inputs for the verbalization process.

Phase – III [Verbalization]

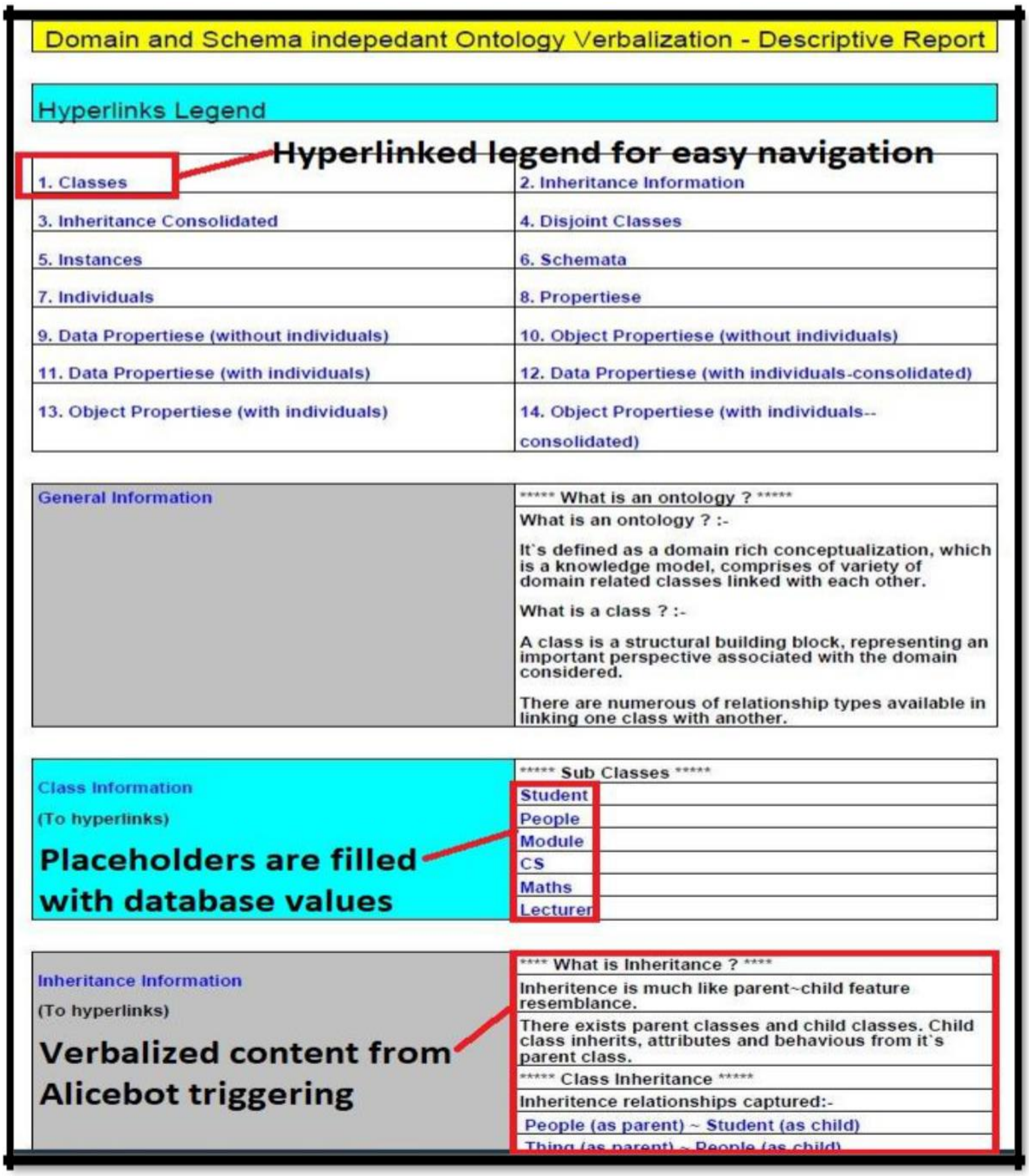
```
Start
Load AliceBot Engine
ArrayList<String> verbaizedStoreArr=new ArrayList<> ()
Supply Semantic Element types as a request to the AliceBot Engine.
While [Semantic Category Name == Semantic Element Request]
        Traverse through the generated AIML template file.
         VerbalizedStoreArr. add (Locate and extract semantic category specific AIML template
contents – Flag value to denote the semantic element type)
End While
Verbalization of all the semantic elements are completed.
I=0
While [I > = verbaizedStoreArr.size ()]
        Iterate through the verbaizedStoreArr Array List
        String arrayVal [] =verbaizedStoreArr.get (I). split ('-').
        iText PDF Report Plugging (arrayVal [0] )
        I++
End While
Generate the PDF version of the verbalized report via the help of the iText plugging.
End.
```

Final part of the proposed algorithm is responsible for the verbalization task. As specified in the pseudocode, when the residing semantic elements inside the ontology increment are submitted as label requests to the AliceBot engine, it will query the AIML template`s relevant categories. Consequently, information or literal explanations stored in between the template tags will be extracted as the responses and programmatically it will be stored inside a specially defined array list with a flag value associated. This flag value is important for conditional verifications as necessary. Henceforth, using the split command the delimiter and the literal value or the response can be stored inside an array defined. Eventually, the extracted literal or the string response is fed as an argument to the iText PDF Report plugging for the verbalization report generation.

This process will iteratively execute for all semantic elements residing inside the ontology increment. The final outcome of it will be the verbalization report which elaborates the axiomatic contents residing inside the ontology increment. The generated verbalization report should be similar to a report as depicted in figure 4.

ISSN: 2233-7857 IJFGCN

Figure 4. Verbalisation Report



For the testing and evaluation process of the proposed algorithm, it was exercised in three different domains. The selected domains for the experiment were psychotherapy, law and marine biology. For each domain a pool of three people were introduced. They were two domain specialists and one ontologist.
As the first step, all the teams were instructed to select a niche area of the associated domain and to brainstorm. Henceforth, group working in the psychotherapy domain selected the cognitive behavioral therapy niche, and the law group selected the labor law niche and the marine biology group selected the corrals niche. Once these niche areas are confirmed, a brainstorming sessions were carried out and several competency questions were recognized. Eventually after multiple discussions within the team mates, first pre-conceptualized version of the ontology increment was derived with the involvement of the ontologists in each group of concern.

Subsequently, the RDF / OWL version of the ontology increment was fed to the proposed prototype. Aforementioned algorithm was executed on the specific ontology increment and the technical encodings residing inside the ontology increment was converted into a human understandable user documentation.

Henceforth, this user document was given to the two domain specialists whom were functioning in each of the groups for the verification of the knowledge embeddings. Practically, there could be instances, where the ideas conveyed by the domains specialists could have been misinterpreted by the ontologists. This could lead into glitches of the ontology increment constructed by the ontologist.

But with this approach of deriving the user guide document associated with the respective ontology increment will allow a chance for the domain specialists to carefully verify, whether it`s the same notion which has been modeled by the ontologists. If the rationales proposed by the domain specialists and the ontologists constructions are tallying with each other, domain specialists can provide the sign off for the continuation to the next iteration. Unless, the located glitches need to be re-discussed with the ontologists and the ontology increment needs to be re modified accordingly. This is a significant practical advantage provided for the domain specialists, for the verification of the accuracy of the knowledge embeddings residing inside the ontology increment under concern.

4. Results and Discussion

As elaborated in the methodology section, experiment was designed and carried out for the three domains identified. Experiment was continued for three iterations to ensure the validity of the results obtained. Work flow in figure 5 depicts the overall process followed in the experiment.

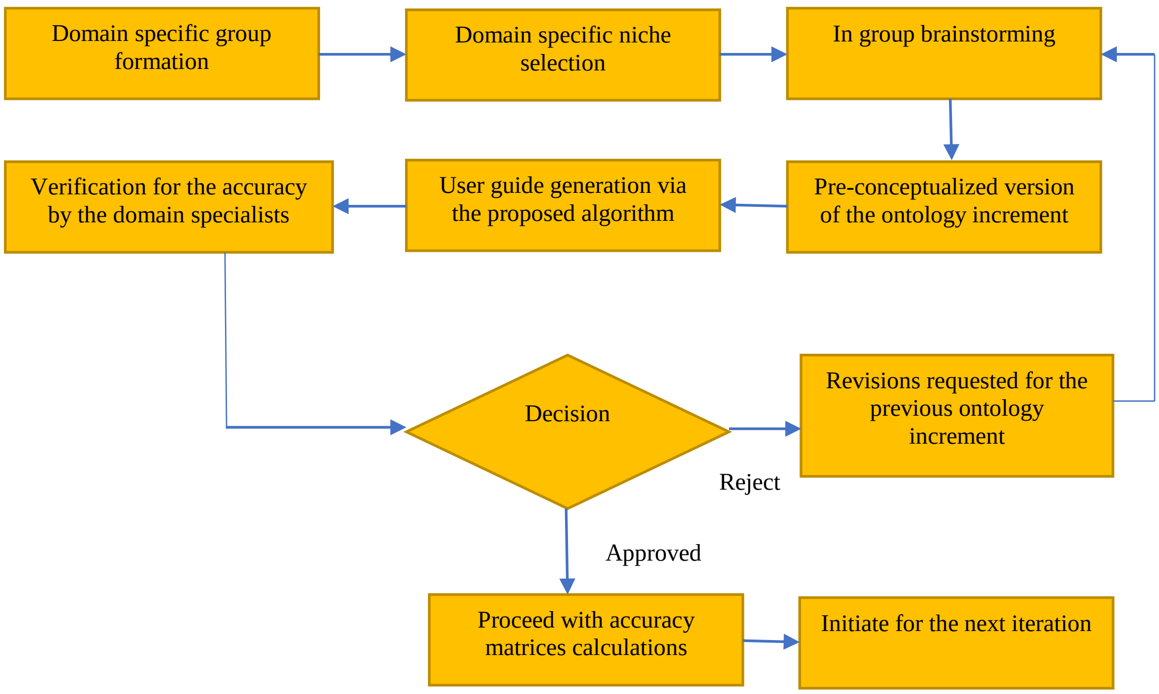
In each iteration of the experiment, once the domain specialists approved the proposed structure, precision, recall and accuracy matrices were calculated. Rationale of true positives and false positives were used for this calculation.

If the contents in the generated user guide matches with the ideologies suggested by the domain specialists, those instances were considered as true positives. Unless those are recognized as false positives.
Commencing from the initial pre-conceptualized version of the ontology increment, aforementioned accuracy matrices were calculated for all three domains under the investigations. Hence, those information were logged in the table 2. In each iteration, some new axiomatic knowledge elements were added to the previous version of the ontology increment. This will ensure, how the proposed taxonomical structures of the ontology increments were evolving as a one single consolidated unit.

As depicted in the table 2, it was visible on the aspect of gradual escalation of the accuracy matrices. This confirms the evolution of appropriate gelling of the axiomatic embeddings introduced via proper taxonomic architectures.
Without stopping from the quantitative analysis, as the next phase of the evaluation qualitative interviewing procedure was used. The interview session was conducted as a questionnaire guided operation. Foundations of the CCP framework was utilized in deriving the questions of the questionnaire used to govern the interview session. CCP framework is a widely used mechanism for opinion segmentation. It allows assessors to look at a particular problem in multiple perspectives.

Below table 3 depicts the mapping structure utilized in deriving the questions in association with the CCP framework guidelines. Controlled interview sessions were conducted on the three domain specific groups. Questions mapped to the segments of the CCP framework were used to govern the flow the interview.

The responses provided by the teams were also assessed, repetitive fragments were removed, and unique aspects were documents in table 4 below.



ISSN: 2233-7857 IJFGCN

Figure 5. Experimental work-flow

Domain	Iteration – 01		Iteration – 02		Iteration – 03		Averaged	
Psychotherapy	Precision	0.84	Precision	0.88	Precision	0.93	Precision	0.88
	Recall	0.78	Recall	0.80	Recall	0.85	Recall	0.81
	Accuracy	0.85	Accuracy	0.88	Accuracy	0.91	Accuracy	0.88
	F-Measure	0.81	F-Measure	0.83	F-Measure	0.88	F-Measure	0.84
Law	Precision	0.77	Precision	0.81	Precision	0.85	Precision	0.81
	Recall	0.71	Recall	0.78	Recall	0.83	Recall	0.77
	Accuracy	0.78	Accuracy	0.80	Accuracy	0.82	Accuracy	0.80
	F-Measure	0.73	F-Measure	0.79	F-Measure	0.84	F-Measure	0.79
Marine Biology	Precision	0.80	Precision	0.82	Precision	0.85	Precision	0.82
	Recall	0.79	Recall	0.83	Recall	0.89	Recall	0.84
	Accuracy	0.83	Accuracy	0.85	Accuracy	0.88	Accuracy	0.85
	F-Measure	0.79	F-Measure	0.82	F-Measure	0.87	F-Measure	0.83

Table 2. Accuracy matrices

CCP Framework Perspectives	‘Wh’ questionnaire term	Mapping question
Content	{What}	What is your opinion about the results provided by the proposed algorithm / prototype? What are the pros / cons you experienced whilst using this algorithm / prototype? What do you think about the inner workings of this prototype? [Technical – For ontologists] What is novel about this approach?
Context	{Why}	Why do you think this type of a prototype is useful / not useful? Why do you think the provided results are accurate / not accurate?
	{Who}	Who would be benefited from this suggested approach?
Process	{How}	How could this prototype / process assist ontology construction? How could this prototype / process assists ontology verifications?

Table 3. Questionnaire mapping

CCP Framework Perspective s	‘Wh’ questionnaire term	Questio n Number	
Content	{What}	1.	Concepts residing in the ontology were accurately verbalized. Mapping sequences are also correctly and textually depicted. Meaning of the concepts were verbalized in layman terms.

ISSN: 2233-7857 IJFGCN

			Hyperlinked structures available to easily traverse between concepts and verbalized elaborations. OWL / RDF specific technical encodings are accurately verbalized to natural language
		2.	Can use with any domain, with no configurations. No training phases, before verbalization. Legible verbalized output. Layman friendlier verbalized output Execution time is quite high for larger ontologies. At the prototype level verbalized outputs are satisfactory, but there is more room for further enhancement
		3.	Comprising off with only 03 main modules, such as extraction, template generation and verbalization. Semantic element names are supplied to the AliceBot engine as requests. Generated response is configured to be produced as a PDF report Rather than developing a complete new module, AliceBot engine is used to trigger the verbalized response, which enforces re-usability of available technological resources.
		4.	Algorithm to auto generate the AIML template, as per the semantic contents in the ontology increment. Domain and Schema independent fully automated verbalization
Context	{Why}	5.	As a user guide to ontology increments. To determine the scope of the ontology, for a differential usage. To verify the accuracy of the knowledge embeddings included to the ontology increment.
		6.	After comparing the contents in the generated user guide against the domain specialist's ideologies, resembling relations are identified.
	{Who}	7.	Domain specialists will get the opportunity to verify the knowledge embeddings included by the ontologists. Ontologists will be benefited from the feedbacks provided by the domain specialists via cross referencing the contents in the verbalized report. Ontologists who are seeking for potential ontologies for differential purposes will be benefited
Process	{How}	8.	Knowledge sharing As a brainstorming platform Collective ideology formation
		9.	Knowledge embeddings verification in ontology increments. Collective ideology formation

Table 4. Response clusters

For the integration of the qualitative and quantitative outcomes, iterative framework was used. Iterative framework is very useful to methodically determine the efficacy of the conducted research. Below tabular structure denotes (i.e. table 5) the application of the iterative framework to this research.

ISSN: 2233-7857 IJFGCN

Iterative Framework Step	Justification Elaborations
1 – What are the data telling me?	In quantitative experiment conducted by domain specialists, there was an average precision of 0.92, average recall of 0.86 and average F-measure of 0.88 → *Application facet* Additionally, the qualitative feedback provided (i.e. summary is documented in table 3) by 03 ontologists also depicts positive attributes about the entire framework, and it`s workarounds. → *Technological facet.*
2- What do I want to know?	How effective is the proposed prototype and algorithm in terms of user guide generation?
3 – Is there a dialectical relationship in step 01 and step 02?	Yes. Both qualitative and quantitative results yielded has depicted the efficacy of the proposed prototype and algorithm, in terms of the research objective to be addressed. Therefore, it can be concluded the proposed solution is satisfactory at its current state.

Table 5. Application of iterative framework

5. Conclusion

Introduction of a user guide to an ontology has bilateral advantages. Technical contribution is the novel algorithmic process introduced for the verbalisation, which can function with no extensive human involvements. The specialty of the proposed algorithm is, it can work in a domain and schema independent manner without the need of special manual configurations. Hence, it will greatly facilitate the workload of both the domain specialists and the ontologists.

Other contribution is the application level impact elicited by the fully automated construction of the user guide. This user guide is ontology increment specific and verbalized in human readable natural language (i.e. English). This user guide can be used to cross-validate the knowledge embedded inside the ontology increment under construction. This will facilitate the role of the domain specialists and ontologists via timely synchronizing both the parties in a collaborative ontology construction environment.

Though it was attempted to compare the proposed algorithm with existing approaches, it was not feasible to locate an existing domain and schema independent verbalisation algorithm.

Most of the existing verbalisation algorithms had the weaknesses of domain and schema dependency. Additionally, majority requires extensive human involvement for the configurations. In that sort of an atmosphere, the proposed algorithm can be emphasized as a significant novel contribution.

The intension of this research is not to merely verbalise existing ontologies. But to effectively blend, both domain specialists and ontologists to improve the efficacy of the collaborative ontology construction process.

References

[1] Saha, S., Usman, Z., Li, W., Jones, S., & Shah, N. (2019). Core domain ontology for joining processes to consolidate welding standards. Robotics and Computer-Integrated Manufacturing, 59, 417-430. doi:10.1016/j.rcim.2019.05.010

[2] Elgammal, A., &Turetken, O. (2015). Lifecycle Business Process Compliance Management: A Semantically-Enabled Framework. 2015 International Conference on Cloud Computing (ICCC). doi:10.1109/cloudcomp.2015.7149646

[3] De Nicola, A., & Missikoff, M. (2016). A lightweight methodology for rapid ontology engineering. Communications of the ACM, 59(3), 79-86. doi:10.1145/2818359 doi:10.1093/bib/6.3.239

[4] Ingram, J., & Gaskell, P. (2019). Searching for meaning: Co-constructing ontologies with stakeholders for smarter search engines in agriculture. NJAS - Wageningen Journal of Life Sciences, 100300. doi:10.1016/j.njas.2019.04.006\

[5] Strohmaier, M., Walk, S., PPschko, J., Lamprecht, D., Tudorache, T., Nyulas, C., …Noy, N. F. (2013). How Ontologies are Made: Studying the Hidden Social Dynamics Behind Collaborative Ontology Engineering Projects. SSRN Electronic Journal. doi:10.2139/ssrn.319903

[6] Mishra, S., Jain, S., Rai, C., & Gandhi, N. (2018). Security Challenges in Semantic Web of Things. IBICA 2018.

[7] Bianchini, D., Antonellis, V.D., & Melchiori, M. (2015). A Web-Based Application for Semantic-Driven Food Recommendation with Reference Prescriptions. WISE.

[8] Emmanuel, K.S., Mathuram, C., Priyadarshi, A.R., George, R.A., & Anitha, J. (2019). A Beginners Guide to Procedural Terrain Modelling Techniques. 2019 2nd International Conference on Signal Processing and Communication (ICSPC), 212-217.

[9] Ueno, M., Morishita, M., & Isahara, H. (2017). Artificial Curation for Creating Learners Manual based on Data Semantics and User Personality. DCAI.

[10] Cheng, Y., He, F., Lu, X., &Cai, W. (2019). On the role of generating textual description for design intent communication in feature-based 3D collaborative design. Advanced Engineering Informatics, 39. Retrieved from https://doi.org/10.1016/j.aei.2019.02.003

[11] Venugopal, & Kumar. (2015). Ontology Verbalization using Semantic-Refinement. IOS Press. doi:0000-0000/15/$00.00

[12] MacArthur, J.A., Bowler, E., Cerezo, M., Gil, L., Hall, P., Hastings, E., Junkins, H., McMahon, A., Milano, A., Morales, J., Pendlington, Z.M., Welter, D., Burdett, T., Hindorff, L., Flicek, P., Cunningham, F., & Parkinson, H.E. (2017). The new NHGRI-EBI Catalog of published genome-wide association studies (GWAS Catalog). Nucleic Acids Research.

[13] Poveda-Villalón, M., García-Castro, R., & Gómez-Pérez, A. (2014). Building an ontology catalogue for smart cities.

[14] Bouayad-Agha, N., Casamayor, G., &Wanner, L. (2014). Natural Language Generation in the context of the Semantic Web. Semantic Web, 5, 493-513.

[15] Dai, Y., Zhang, S., Chen, J., Chen, T., and Zhang, W. Semantic Network Language Generation Based on a Semantic Networks Serialization Grammar. World Wide Web 13, 3(2010), 307–341.

[16] Demir, S., Carberry, S., and Mccoy, K. A Discourseaware Graph-based Content Selection Framework. In Proceedings of the 6th International Natural Language Generation Conference (INLG) (2010), pp. 17–27.

[17] Androutsopoulos, I., Lampouras, G., &Galanis, D. (2013). Generating Natural Language Descriptions from OWL Ontologies: the NaturalOWL System. Journal of Artificial Intelligence Research, 48, 671-715. doi:10.1613/jair.4017

ISSN: 2233-7857 IJFGCN

[18] Mann, W.C., & Thompson, S.A. (1988). Rhetorical Structure Theory: Toward a functional theory of text organization. Text & Talk, 8, 243 - 281.

[19] Sportelli, F., &Franconi, E. (2016). Formalisation of ORM Derivation Rules and Their Mapping into OWL. On the Move to Meaningful Internet Systems: OTM 2016 Conferences, 827-843. doi:10.1007/978-3-319-48472-3_52

[20] Bojars, U., Liepins, R., Gruzitis, N., Cerans, K., &Celms, E. (2016). Extending OWL Ontology Visualizations with Interactive Contextual Verbalization. VOILA@ISWC.

[21] Keet, C. M., &Khumalo, L. (2016). On the verbalization patterns of part-whole relations in isiZulu. Proceedings of the 9th International Natural Language Generation conference. doi:10.18653/v1/w16-6629

[22] KaarelKaljurand and Norbert E. Fuchs. 2007. Verbalizing owl in attempt to controlled English. In Proceedings of Third International Workshop on OWL: Experiences and Directions, Innsbruck, Austria (6th–7th June 2007), volume 258

[23] Guimarães, M. A., Zisman, R. P., & Renzi, A. B. (2019). Pharmaceutical Online Store Project: Usability, Affordances and Expectations. Advances in Intelligent Systems and Computing, 523-534. doi:10.1007/978-3-030-20040-4_47

[24] Sun, Mi, Olsson, Paulsson, &Harrie. (2019). Utilizing BIM and GIS for Representation and Visualization of 3D Cadastre. ISPRS International Journal of Geo-Information, 8(11), 503. doi:10.3390/ijgi8110503

[25] Dorothy, B., & S. Ramesh Kumar, B. (2018). DORBRI: An Architecture for the DoD Security Breaches Through Quantum IoT. International Conference on Computer Networks and Communication Technologies, 491-496. doi:10.1007/978-981-10-8681-6_44

[26] Motta, E., Mulholland, P., Peroni, S., d'Aquin, M., Gomez-Perez, J.M., Mendez, V., Zablith, F.: A novel approach to visualizing and navigating ontologies. In: Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N., Blomqvist, E. (eds.) The Semantic Web { ISWC 2011: 10th International Semantic Web Conference, Bonn, Germany, October 23-27, 2011

[27] Cool URIs don't change, Tim Berners-Lee, 1998. http://www.w3.org/Provider/Style/URI.

[28] Beek, W., Rietveld, L., Bazoobandi, H. R., Wielemaker, J., & Schlobach, S. (2014). LOD Laundromat: A Uniform Way of Publishing Other People’s Dirty Data. The Semantic Web – ISWC 2014, 213-228. doi:10.1007/978-3-319-11964-9_14

[29] Alobaid, A., Garijo, D., Poveda-Villalón, M., Santana-Perez, I., Fernández-Izquierdo, A., & Corcho, O. (2019). Automating ontology engineering support activities with OnToology. Journal of Web Semantics, 57, 100472. doi:10.1016/j.websem.2018.09.003

[30] Garijo, D. (2017). WIDOCO: A Wizard for Documenting Ontologies. Lecture Notes in Computer Science, 94-102. doi:10.1007/978-3-319-68204-4_9

ISSN: 2233-7857 IJFGCN

[31] Bontcheva, K., & Wilks, Y. (2004). Automatic Report Generation from Ontologies: The MIAKT Approach. Natural Language Processing and Information Systems, 324-335. doi:10.1007/978-3540-27779-8_28

[32] Bao, J., Cao, Y., Tavanapong, W., & Honavar, V. (2004). Integration of Domain-Specific and DomainIndependent Ontologies for Colonoscopy Video Database Annotation. Artificial Intelligence Research LaboratoryIowa State University

[33] Poulovassilis, A., Selmer, P., & Wood, P. T. (2016). Approximation and Relaxation of Semantic Web Path Queries. SSRN ElectronicJournal. doi:10.2139/ssrn.3199265

[34] Singh, A., Ramasubramanian, K., & Shivam, S. (2019). Introduction to Microsoft Bot, RASA, and Google Dialogflow.

[35] Janarthanam, S. (2017). Hands-On Chatbots and Conversational UI Development: Build chatbots and voice user interfaces with Chatfuel, Dialogflow, Microsoft Bot Framework, Twilio, and Alexa Skills.

[36] Ralston, K. (2019). Personalizing Chatbot Conversations with IBM Watson.

[37] Wallace, The Elements of AIML Style, ALICE A. I. Foundation, Inc. March 28, 2003

[38] Wallace, R. S. (2014). AIML 2.0 Working Draft, https://docs.google.com/document/d/1wNT25hJRyup cG5 1aO89UcQEiG - HkXRXusukADpFnDs4/pub

[39] Madhumitha, Keerthana, &Hemalatha. (2019). Interactive Chatbot Using AIML. Advanced Networking & Applications, 217-223

[40] McCarthy, J. (1980). "Circumscription—A Form of Non-Monotonic Reasoning." Artificial Intelligence 13(1–2): 27–39

[41] Whittemore R. 2005. Combining evidence in nursing research: methods and implications. Nursing Research 54(1):56-62

[42] Srivastava, P., & Hopwood, N. (2009). A Practical Iterative Framework for Qualitative Data Analysis

[43] Caldarola, E.G., Rinaldi, A.M.: An approach to ontology integration for ontology reuse. In:2016 IEEE 17th International Conferenceon Information Reuse and Integration(IRI) (2016). https://doi.org/10.1109/iri.2016.58

ISSN: 2233-7857 IJFGCN