Fully Automated Ontology Increment's User Guide Generation

Google AliceBot and AIML based Algorithmic Approach

Kaneeka Vidanage[1], Rosmayati Mohemad[2], Noor Maizura Mohamad Noor[3], Zuriana Abu Bakar[4]

Faculty of Ocean Engineering Technology & Informatics, University Malaysia Terengganu (UMT), Malaysia[1, 2, 3, 4]

Faculty of Computing, Department of Data Science, NSBM Green University, Sri Lanka[1]

Abstract—This research focuses on a domain and schema independent user-guide generation for ontology increments. Having a user guide or a catalogue/manual is vital for quick and effective knowledge dissemination. If a user guide can be generated for an ontology as well, there could be ample advantages. Stakeholders can scan across the user guide of the ontology and verify the eligibility of it, against the intended purposes. Additionally, this could be useful in ontology`s version management requisites and knowledge verification requirements as well. Even though, ontology construction being iterative and incremental operational, there will be several intermediate versions before it reaches to the fine-tuned final version. Therefore, manual user guide creation will be a tedious and impossible operation. Consequently, this research focuses on a novel algorithmic approach to domain and schema independent ontology verbalization. A special algorithm is created to alter the functionality of Google's AliceBot to work as a verbalizer, instead of a chatterbot. Artificial Intelligent Modelling Language (AIML) technology is utilized to create the templates for the ontology specific knowledge embeddings. This entire process is fully automated via the proposed novel algorithm, which is a key contribution of this research. Eventually, the generated user guide generation tool is evaluated against three different domains with the involvement of fifteen stakeholders and 82% of averaged acceptance has been yielded.

Keywords—AliceBot; artificial intelligent modelling language; ontologist; verbalizing

I. INTRODUCTION

Ontologies are recognized as domain rich conceptualizations [1] which are both machine and human-readable [2-3]. Further, as of its enriched ability on conceptualization, it`s ideal to encode specialized human knowledge [4]. Subsequently, encoded knowledge will be machine-readable accomplishing endless domain-specific reasoning and knowledge representation necessities. Because of those unique features, the popularity of ontology-based applications escalated drastically. Therefore, presently, there`re thousands of applied ontologies developed in numerous domains such as biology, agriculture, bioinformatics, law, management, etc. [5-6]. Applied ontologies are used to overcome issues coming from non-computing domains and by using the aforementioned benefits of the ontologies, most of those issues can be effectively resolved [30]. Construction of an accurately defined applied ontology is a complex process, which requires both ontologists and domain specialists to work hand-in-hand with mutual understanding throughout the entire process. As a methodical workflow to fulfil the effective bridging of domain specialists and ontologists, "collaborative ontology engineering" has been emerged out as a separate niche under the umbrella of ontology engineering [7-9].

One of the crucial necessities to be fulfilled in collaborative ontology engineering is the proper glueing in-between domain specialists and ontologists [10]. Unless effective participation and collective contribution reaching towards an error-free applied ontology will not be realistic. Researchers have mentioned, ontologists need to have reasonable insight on the domain to be modelled and vice-versa domain specialists should have a sound understanding of the essential basics associated with semantic web and knowledge modelling. Once this state is achieved only, collective, and effective participation of both parties can be expected, leading towards the construction of an error-free applied ontology schemata [11-12].

Even though, there is a critical bottleneck caused due to shortage or illiteracy of comprehension, on semantic concepts, experienced by non-computing domain specialists such as lawyers, medical doctors, agricultural specialists, etc. [13-15].

Ontology construction, being a complex, iterative and incremental task, it`s expected at the end of each iteration, domain specialists should cross-reference and verify, that the knowledge provided by them, are accurately and consistently modelled, and embedded to the ontology by the ontologists [16-17]. Then the errors located can be corrected then and there, without waiting until it reaches to complex conceptual flaws resulting in an erroneous schema. In accomplishing this requirement, lack of or no literacy in semantic concepts is a strong hindrance. Because to properly understand an ontological taxonomy defined, the user should have reasonable knowledge associated with basic object-oriented concepts such as inheritance and semantic concepts such as triple concept, data properties, object properties, disjoint classes, symmetric classes and, the concept of individuals, etc. Though the person understands those aspects, the next step is writing an appropriate SPARQL or SQWRL query to verify the accuracy of the knowledge embeddings. Properly understanding the schemata of the ontology and along with the use of accurate syntaxes, forming up an accurate SPARQL query is a challenging task, even for a computer scientist at once. Therefore, obviously, it`s an unrealistic goal to be expected from a non-computing domain specialist like a medical doctor, business manager or lawyer. Even if the SPARQL query is

written and executed, results will be returned as a triple with URLs and pre-processing of those are required to derive the answers in plain English, where this is also could not be a feasible task based on the competency level of a non-computing domain specialist [13-15].

All these obstacles are constraining the domain specialists` involvement, in accomplishing knowledge verification necessities, which is a crucial step in collaborative ontology engineering [9,18]. On the other hand, the creation of a catalogue (i.e., user guide) for the ontology`s structure is very important for its latter maintenance and knowledge diffusion requirements [21-22]. However, manually fulfilling this task could escalate the workload of the ontologists. Further, lately, if any alteration occurred in the schemata, the entire catalogue needs to be re-written or updated accordingly. This is going to be a highly effort-consuming and tedious task on the shoulders of the ontologists [21-22].

This research is focusing on a fully automated mechanism to verbalize (i.e., output the knowledge encoded in the ontology, in its natural language form) the entire ontology, despite its domain or the schemata [19-20]. This will resolve the technological challenges non-computing domain specialists need to face and it will revoke the burden of manual user guide creation efforts from the ontologists' workload as well. Therefore, both domain specialists and ontologists are benefited from this novel contribution. The key contributions of the proposed technique against the existing mechanisms are: -

- The proposed technique can work with any domain → Domain independent.

- The proposed technique can work with any schema → Schema independent.

- No external / manual configuration requirements.

- Converts semantic contents in the ontology increments into layman understandable English.

II. LITERATURE REVIEW

In the process of collaborative ontology engineering, the collective opinion of both the ontologists and domain specialists are very vital. There needs to be a proper workflow to accomplish the collective opinion derivation requirements [23-24]. Otherwise, in a collaborative environment, multiple people will raise multiple viewpoints and try to stick to their perspectives. This will ultimately lead to the issue of "Tragedies of Commons" [25-26]. Hence, in Shneiderman`s "Information Seeking Mantra" concept, Shneiderman has pointed a proper workflow to methodically integrate the dispersed viewpoints of the stakeholders to reach towards an overall collective insight, at the end [27]. Unfortunately, the idea of Shneiderman`s "Information Seeking Mantra" requires a specialist tool support to fulfil its workflow steps. Information Seeking Mantra concept first requests the stakeholders to get an overall idea about the problem of concern. In accomplishing this request, Shneiderman suggests the use of both visualization and verbalization tool support. Next, zoom towards the required information only. Both verbalization and specially defined visualization techniques can fulfil the second step's requirements as well. The third and fourth steps are focusing on, filtering unnecessary information and look for information on demand, respectively [27-29]. As the outcome of this research, though a special prototype is proposed to address all requirements of Shneiderman`s "Information Seeking Mantra", this paper`s scope is constrained to discuss its verbalization feature only, to manage the scope of this paper.

Technically, verbalization is defined as the process of translating axioms defined in ontology to natural language [19-20]. Most of the existing verbalization systems rely on the complex Natural Language Generation (NLG) pipeline to convert axioms into natural language [31]. This is a complex, technological pipeline where all the phases need to be accurately fulfilled, to get an understandable natural language output. Namely, those steps are defined as content selection step, discourse planning step, lexicalization step, aggregation step, generation of referring expressions and finally linguistic realization step [32].

Among all those steps, the discourse planning step is very vital to achieve coherent verbalization output. The discourse planning step utilizes the 'Rhetoric Structure Theory (RST)' for the coherent organization of the text [65]. RST is based on two main conceptions as nucleus and satellite. The nucleus represents the significant axioms associated with the considered domain and the satellite represents the associative properties linked with the nucleus which are required to elaborate the nucleus [65]. Therefore, if the identification of the nucleus and satellite didn`t occur in a domain-specific manner, it will adversely affect the clarity of the verbalized contents [31,33]. 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 verbalization.

The problem that arises here is the inability of using the same verbalizer for any other domain. Complex prior configurations, which is referred to as portal configuration is a must. This makes a verbalization ready framework to become domain-dependent always [32]. This is a key limitation associated with the existing verbalizing techniques.

The next restriction is the necessity of annotations to enrich the semantic realization of the concepts in the ontology. Again, this request additional effort from the ontology engineers and in most cases eligible foundational de-facto standard meta-models are (i.e. Dublin Core, FOAF) needed to be incorporated into the ontology. Because the majority of the existing verbalization frameworks are configured to link with the pre-defined annotated endpoints of those de-facto standard meta-models only. This poses an additional overload to the ontologists, and it acts as a modelling restrictor also [34-35]. Therefore, the free will of the ontologists and domain specialists are restrained, as they need to plan everything in a manner to suit up with the de-facto standard meta-models.

As the final disadvantage, it can be pointed, that most of the existing verbalizers produce Control Natural Language (CNL) which is resembling to assembly language and it`s not colloquial English that can be understood by laymen.

Therefore, another Natural Language Processing (NLP) layer must be introduced to overcome the barrier of converting technical English constructs to its colloquial format, which will be another processing overhead. One of the main causes for this is, the existing evolution of verbalizers have evolved up to the level of Attempt to Control English (ACE), which is a form of Control Natural Language (CNL). In CNL, verbalizers attempt to extract the triple formulations in the ontology and to exactly covert them into the English language, where the contextualized connectivity and colloquialism will be lost [36-38].

TABLE I.   EXISTING SYSTEM ANALYSIS

Tool	Restrictions / Limitations
NaturalOWL [31, 52]	Extensive domain-specific configuration efforts.
LODE [53]	Ontology increment must be published on the web and the accessibility url must be according to the cool_URI format. XSLT script of the LODE is configured to work with only standardized metamodels like FOAF, Dublin-Core, etc. This acts as a modelling restrictor.
SWAT Tools [54]	Extensive redundancies in the HTML link sequences generated for the ontology. Split information problem with lots of dispersed information here and there. Make the role of the domain specialist very difficult.
MIKAT [55]	Verbalizer is statically attached to the breast cancer domain. It cannot work with any other domain. Fully domain-dependent.

TABLE II.   EXISTING ALGORITHM ANALYSIS ACCORDING

Algorithm	Deficiencies
CNL - ER Algorithm [56]	Works only for entity relationship diagram`s verbalizations. Does not work for ontologies.
Contextual Verbalization Algorithm [36]	On-demand verbalization only. Generated results cannot be saved for later use. Additionally, the verbalized outputs are primitive and restricted to the level of ACE.
Semantic Expressions Algorithm [57]	The output is not in the natural language. It`s in System Verilog format. This doesn`t cater to the requirement to be fulfilled.
FST Verbalizer Algorithm [58]	Incapable to verbalize an ontology
Variable Verbalization Algorithm [59]	Based on domain-specific training of a machine learning model. Doesn`t address the domain independence requirement. It`s not possible to organize a training dataset, without fixing the domain.
Semantic Refinement Algorithm [60]	Extensive domain-specific configuration effort.

That`s the main reason for the verbalized output to look very primitive and the flow seems inconvenient to interpret by the end-users. Most of the deficiencies associated with verbalizers are schema and domain-specific [39-41]. Among those, some of the general deficiencies are reviewed above. Hence, it can be easily concluded, through the existing verbalization mechanisms, the afore-mentioned research gaps of domain-dependence, excessive human involvement associated with configurations and CNL based less colloquialism are not properly resolved. Hence, the emphasis of this research is to propose a novel approach to overcome the aforesaid shortages.

Table I contains a comparison of famous existing verbalization tools, their limitations and why they cannot resolve the issue of domain and schema independent verbalization by producing a colloquial user guide.

Further, several existing verbalization algorithms are also reviewed to recognize their deficiencies. Table II contain the details of verbalizer algorithmic analysis.

Therefore, according to the discussion conducted in the literature review section, it`s apparent there is a research gap to be resolved. The following sections of the paper discuss the steps followed to fulfil the recognized research gap.

III. METHODOLOGY

After completion of an intense and extensive systematic literature assessment [42], it's concluded that the aforesaid research gaps are still not been resolved properly.

Subsequently, the blend of the think-a-loud protocol [43]and systems thinking [44] notions are used to collectively brainstorm on the problem of concern and ultimately AIML (Artificial Intelligence-based modelling/mark-up language) is selected to implement the proposed solution, as AIML is recognized as an ideal technology for creating natural language software agents with an XML dialectic [44-48].

Table III contains the analysis results of the technology review conducted.

As per the collective brainstorming results logged in Table III, it was determined AIML is the ideal technology platform to be utilized to resolve the research problem. Because it has broad external integration support and no domain-related training datasets are required to train a domain-specific model. Hence, AIML ideally matches up with the domain and schema independence requirement.

Design science research methodology is selected as of the investigative nature of this research [47]. Implemented version of the prototype is quantitatively and qualitatively assessed on its functionality Fig. 1 exposes the application workflow of design science research methodology`s operation for this research.

The importance of the research problem being investigated and its timely relevance was already justified via the literature review results studied. Additionally, a comprehensive literature analysis was conducted again on existing verbalization tools and algorithms to explicitly justify the deficiencies unresolved. Consequently, it was recognized:

a) Domain and schema independent ontology verbalization tool with.

b) Zero configuration effort.

c) For the fully automated user guide generation of the iterative and incremental ontology increments.

as the consolidated research, objective to be accomplished from this research.

Henceforth, via an adequate amount of collective brainstorming, the following algorithm was designed. The proposed algorithm comprises three main operational phases as depicted in Fig. 2.

TABLE III. TECHNOLOGY SELECTION

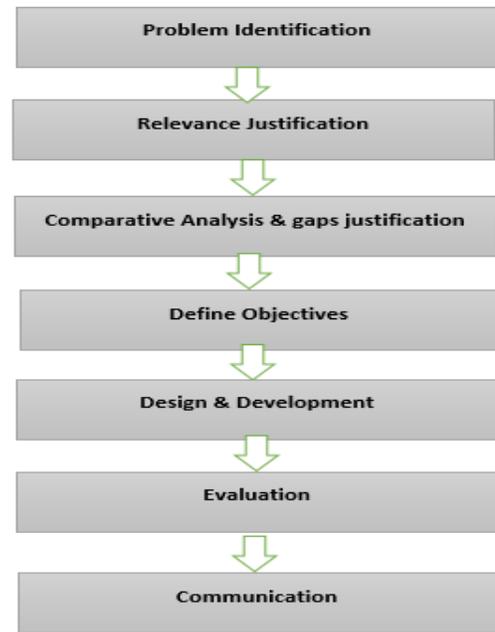| Technology	Remark
DialogFlow [61]	-The training requirement is a dire constraint that needs human involvement for different contexts (done via manually typing potential phrases for the training) -The basis is on Machine Learning & domain-specific training → Makes *it a domain-dependent solution, as a dataset will be domain-specific* -A regular expression for pattern matching support is not available -Advanced features are not freely available -Can integrate with only one webhook – interaction where external knowledge bases are restricted.
RAZA[63]	-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 a challenging learning curve before the usage -Memory hungry technology – which will slow the entire system performance - External integration assistance is very minimal. -Overfitting and underfitting issues.
IBM Watson [62]	-Integration with third party resources is difficult -Steep learning curve -Costly
AIML / ALICE [45-48]	-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 -A regular expression for pattern matching support is also available

Fig. 1. Research Process.

Fig. 2. Algorithm Phases.

Phase-I - [Knowledge Extraction]

Start
Upload RDF / OWL version of the ontology increment to be verbalized.

Check for the format as RDF or OWL.
Trigger format-specific knowledge extraction logic.
While [Until EOF == TRUE]
    Extract class information
    Extract data properties
    Extract object properties
    Extract class-specific individuals (if existing)
    Stow them appropriately in different relations of the
RDBMS.
End While

Ontology increment to be verbalized can be directly uploaded to phase-I of the algorithm. Phase-I of the algorithm contains code snippets for knowledge extraction from both RDF(Resource Description Framework) and OWL (Web Ontology Language) formats. Henceforth, the extracted knowledge elements will be separately stowed in the database relations as classes, data properties, object properties etc. Implementation of phase-I of the algorithm can be depicted as in Fig. 3 code snippet.

```
if (fileType.equals("rdf"))
{
String namespace=modelLoader(path);// Derive main namespace of the Knowldge Model
Document doc=XMLDocuementLoader(path);// Initialize XML dom parser
individualExtractorRDF(doc,path,namespace);// Extract Individual By Individual
}
else if (fileType.equals("owl")) // If the file extension is OWL
{
    OWLOntologyManager manager = OWLManager.createOWLOntologyManager();
    File file = new File(path);
    OWLOntology ontology = manager.loadOntologyFromOntologyDocument(file);
    individualExtractorOWL(ontology,path);
}
```

Fig. 3.   Knowledge Extraction Code Snippet.

Phase-II - [Auto-generation of the ontology specific AIML template]

The main purpose of this phase is ontology increment specific AIML template generation. This can be identified as a critical contribution to this research. Reasons for the choice of AIML technology is already elaborated in Table III. The proposed pseudo-code for the phase-II operation can be listed as mentioned below:

Start
Sequentially extract stored semantic elements from the database relations.
Stow them in individually created ArrayLists
Initiate a blank baseline AIML template (Fig. 4)
<xml>
  <aiml>
    <category>
      Extract elements from one specific ArrayList (i.e., classes).
      <pattern>
      Derive the name of the semantic element type stowed in the selected ArrayList
       <template>
       I=0
      While (I >= ArrayList. Size ())
      Extract specified semantic element contents from the selected ArrayList
      Continue the appending process until reaching the end of the ArrayList size
      I++
      End While
</template>
</pattern>
</category>
Repeat the same process for all semantic elements residing inside all the ArrayLists.
</aiml>
</xml>
End

```
<?xml version="1.0" encoding="UTF-8"?>
<aiml>
<!-- -->
<category>
    <pattern>GENERAL</pattern>
        <template>What is an ontology ? :-

        It's defined as a domain rich conceptualisation, which is a knowledge model, comprises of variety of domain related classes

        What is a class ? :-

        A class is a structural building block, representing an important perspective associated with the domain considered.

        There are numerous of relationship types available in linking one class with another.</template>
</category>

<category>
    <pattern>CLASS</pattern>
        <template>$placeholder_1$</template>
</category>

<category>
    <pattern>INHERITENCE_GENERAL</pattern>
        <template>Inheritance is much like parent-child feature resemblance.
There exists parent classes and child classes. Child class inherits, attributes and behaviour from it's parent class.</template>
</category>
```

Fig. 4.   Snapshot of the Baseline AIML Template.

As depicted in Fig. 4, the generalistic baseline AIML template structure's placeholder contents will be filled by the information extracted from phase-II of the algorithm. Subsequently, the generalized baseline AIML template will be ontology increment specific. This will be done in a fully automated manner by phase-II of the algorithm.

Phase-III - [Verbalization Process]

The main purpose of phase -III is for the generation of the verbalized user guide. The pseudocode operation of the phase-III is as visible below.

Start
Load AliceBot Engine (Fig. 5)
Submit a class-specific semantic structure name (i.e. Data Properties of the Student Class) as a request object to the AliceBot Engine.
AliceBot Engine will traverse across the autogenerated AIML template and seek the most matching category
Locate for class-specific template contents
Derive the class specific<template> contents residing inside the matching category.
Cross-references the extracted contents against the values of the class-specific attributes via executing a parameterized database query.
Stow the extracted AIML pattern sequences and class-specific verbalized contents in an ArrayList.
Repeat the same process and stow all statements to be verbalized inside verbalizedreposiArr ArrayList.
K=0
While [K >= verbalizedreposiArr.size()]
        Iterate across the ArrayList contents.
        iText PDF Report Pluggin (verbalizedreposiArr.get(K).toString())
        K++
End While
Generate PDF version of the verbalized user guide for the specific ontology increment
End

```
public Chat commonInitialization()
{
    String resourcesPath = getResourcesPath();
        System.out.println(resourcesPath);
        MagicBooleans.trace_mode = TRACE_MODE;
        Bot bot = new Bot("ALICEBOT", resourcesPath);
         bot.writeAIMLFiles();
        Chat chatSession = new Chat(bot);
        bot.brain.nodeStats();
        return chatSession;
}
```

Fig. 5.   Load AliceBot Engine.

Fig. 6.   Code Snippet for the Phase-III Execution.

Fig. 7.   Verbalized Report Sample.

The initial step of phase-III is to load the AliceBot engine. Generally, AliceBot is a free chatbot engine provided by Google. However, through this algorithm, the behaviour of the AliceBot is altered from a general chatterbot to a verbalization engine. That's a significant contribution in phase-III of this algorithm.

Loading of the AliceBot engine can be depicted as in Fig. 5.

Henceforth, contents to be verbalized will be supplied as a request to the AliceBot Engine. Afterwards, AliceBot will traverse through the customized baseline AIML template generated by phase-II of the algorithm (i.e., Fig. 4) and conduct the verbalization process as per the information placed in the AIML template and by filling the placeholder values from the contents extracted from the database. This process can be depicted as in the code snippet in Fig. 6.

IV. RESULTS AND DISCUSSIONS

A sample sketch of the portion of the verbalized report generated for a criminal law ontology increment is depicted in Fig. 7. The entire verbalized user guide is hyperlinked to facilitate easier navigation across the document. To prevent information overloading and cluttering, a segment-wise approach is followed to efficiently layout the verbalized contents, facilitating readability.

Fig. 8.   Evaluation Workflow.

AliceBot is a Google chatterbot, developed on the foundations of AIML technology. With the help of the aforementioned algorithm designed, the operation of the chatterbot is altered to an automated verbalizer, which is a significant research contribution.

As a means of enhancing the validity of the experiment conducted, the same verbalized user guide generation process is conducted for three different domains. Those are on COVID-19 ontology increment, aquaculture ontology increment and the criminal law ontology increment. Snapshots of the taxonomical structures of those ontology increments are included in the appendix of this paper.

The proposed evaluation workflow is depicted in Fig. 8. The proposed evaluation framework utilized in this research can be pointed out as another contribution to this research. Both quantitative and qualitative emphasis were blended in this proposed framework.

First, the operationalization step was carried out. In there several open-ended questions were compiled against the research`s objectives. This mapping of the questions in the questionnaire with the research`s objective is the process of operationalization [64]. This will make sure the responses collected via the questions in the questionnaire are very much relevant and matching across with the requirements of the research. The list of open-ended questions defined for this evaluation is as mentioned below:

1) Have you been notified about the existing verbalization mechanisms?

2) In contrast to those, have you identified any positive capabilities of the proposed mechanism?

3) Do you think it will facilitate the role of ontology increment verification?

4) Can you elaborate on how it will facilitate the role of the inspectors?

5) What are the deficiencies you identified in the proposed verbalization mechanism?

In the pre-warm-up setup, all stakeholders were exposed to a specially created synoptic video clip about the research conducted. This phase acts as a retrospect and summarizes the significant aspects of the research conducted to all the stakeholders involved in the evaluation. This was done, before the official commencement of the evaluation process as it will resolve all unclear areas associated with this experimental setup.

In the controlled interview sessions, a face-to-face interview series were conducted with nine domain specialists belonging to the criminal law, COVID-19 and aquaculture domains and six ontologists. The above listed five questions were the main basis for the controlling of the interview sessions with the fifteen stakeholders. All controlled interview sessions were video recorded to facilitate later interpreting requirements. The recording was done, via getting prior approval and consent from all the involved participants and it was utilized only for research purposes and not for any other personnel gains.

10	20	30	40	50	60	70	80	90	100
Very Poor	Fairly OK – Major Revisions				Good & Acceptable – Minor Revisions			Exceptional	

Fig. 9. Close-Ended Response Grid.

During the thematic extraction phase, all recorded interviews were transcribed into a textual format. Henceforth, those were iteratively examined for several turns by the involved research staff. All information gathered through the repetitive analysis were segregated into a few generalistic themes. At the point of the analysis, initially at a drastic rate new themes started to emerge out, by the time of reaching the ninth transcription, a saturation of the themes were noted, where the same themes commenced repeating again and again. This characteristic was recognized as reaching the saturation state of the interview findings.

The theme extraction allowed to recognize of mostly insightful areas of the research. It was not feasible to collect all important opinions only via quantitative terms. Therefore, the qualitative phase enforced through controlled interview sessions created the opportunity to recognize significant user insights.

Henceforth, a close-ended question series was compiled to gather more insights on the located themes. This allows us to derive attention to details on the specific aspects with a numerical emphasis. A special rating grid was utilized to extract stakeholder opinions as depicted in Fig. 9.

Following five questions were provided in a close-ended format and requested to rate the opinions:

1) Natural English verbalization of the technical semantics of the ontology increment is accurate.

2) Acts as a manual / user guide for the ontology increment, enforcing offline usage as well.

3) Boosts comprehension when concepts are verbalized in layman terms.

4) Domain and schema independent, configuration free operation.

5) How would you rate the verbalization assistance provided by the tool support?

Following Table IV summarizes the averaged response scores derived via all 15 stakeholders involved in this experiment.

TABLE IV. RESPONSE SCORES

Verbalized User Guide [VU]	Law	85%
	COVID-19	80%
	Aquaculture	83%
	Averaged	**82%**

A summarized collection of the qualitative interpretations and insights gathered through the controlled interview session were depicted in Table V.

TABLE V. QUALITATIVE RESPONSE OVERVIEW

Verbalized User Guide [VU]	1. Surface level enlightenment for the domain specialists about the conceptualizations of the ontology increments 2. Boosts blended comprehension when visualization and verbalizations are integrated during analysis. 3. Offline information reviewing facility is important, as doesn't need to stick to a computer screen all the time. 4. Acts as a user guide/manual for each version of the ontology increment. 5. Fully automated, configuration free, domain and schema independent operation. 6. Natural language (i.e., English) representation of the semantic concepts in a domain specialist friendly manner.

As the final phase of the evaluation process, the iterative framework was applied to reflective asses on the research objective accomplishment. Iterative framework [49-51] is an established framework to evaluate the efficacy of research objective accomplishment logically. The operation of the iterative framework is governed via three different but interconnected questions. Reflective evidence must be provided for each section in place. The discussion associated with the iterative framework steps were logged in Table VI.

TABLE VI. ITERATIVE FRAMEWORK

Steps in Iterative Framework	Reflective Evidence
01 → What are the data telling me?	Quantitative Metrics:- As depicted in Table IV, multiple domain-specific qualitative opinion scores were utilized to validate the effectiveness of the constructed verbalization prototype and its operational effectiveness. Qualitative Assessment:- With the involvement of the stakeholders contributed for the ontology increment constructions, empirical assessment of the verbalization prototype was done, in terms of the results returned, accuracy, usability, technical aid provided & etc. Further, as visible in Table V, the stakeholders' reflective opinion themes were logged As a collective reconciliation, it can be concluded both the quantitative and qualitative experimental phases have yielded satisfactory results.
02 → What do I want to know?	The overall operational efficacy of the verbalization algorithm/prototype developed in terms of domain/schema independence knowledge verbalization.

03 → Is there a dialectical relationship between step 01 & 02?	In the quantitative phase of evaluation, the verbalization prototype is exposed to multiple ontology increments in three different domains. In all those, experiments quantitative matrices are calculated to determine the overall efficacy of the verbalization prototype and it`s apparent the overall operation has yielded successful results. In the qualitative assessment phase, stakeholders' opinions were thematically assessed, and refined outcomes were tabulated in Table V. Both, quantitative and qualitative evaluation phases conducted on the criteria of the domain/schema independence verbalization, have collectively yielded successful outcomes. Therefore, as the overall final reflection, as per the iterative framework rationale, it can be concluded as, there is a positive and satisfactory link between step-01 and step-02, which reflects the overall efficacy of the verbalization prototype/algorithm resulted from this research.

V. CONCLUSION

Effective synchronization between ontologists and domain specialists are a must to accomplish the effective operation of the collaborative ontology engineering goals. To fulfil this purpose, domain specialists should closely involve in the verification process of knowledge embeddings present in each ontology increment. Because error-free, domain-oriented applied ontology construction is an iterative and incremental operation. Hence, it's an extremely effective practice to expose the ontology increment for the cross-validations of the knowledge embeddings, at the end of each iteration.

Verbalization is recognized as a very effective procedure for fulfilling this necessity, as it's a natural language representation mechanism of the internal knowledge embeddings of the ontology of concern.

Usually, lack of technical literacy on semantic concepts, querying skill sets will act as a barrier for the non-computing domain specialists (i.e. lawyers, medical doctors, business consultants, bankers, etc. …), to complete their role of cross-validation in collaborative ontology engineering workflow.

But as conversed in this paper, almost, all of the existing verbalizers have lots of limitations, confiding their operation to a statically attached one domain or complex configuration phases or technically complex verbalized results that are not easy to interpret. Therefore, this research focused on addressing those shortcomings of the existing verbalizers and uplifting the operational efficacy of the verbalizers to the next level, by making them operate in a domain and schema independent manner.

In this research, AIML based Alice bot's operational flow is transferred into a form of a verbalizer, by the introduction of

a fully automated newly defined algorithm, which is a significant technical contribution resulting out from this research. Henceforth, its operational accuracy is quantitatively and qualitatively evaluated, where both mechanisms have yielded an overall of 82% acceptance.

Domain and schema independent ontology verbalization with no manual configurations and fully automated user guide construction for the ontology of concern are two critical application-level contributions yielding out from this research.

But as one limitation, it can be concluded that this verbalizer will only work for ontologies with a lexicon-based schematic structure, as the backbone of this prototype is developed on top of a chatter bot's architecture.

REFERENCES

[1] Spasic, I., Ananiadou, S., McNaught, J., & Kumar, A. (2005). Text mining and ontologies in biomedicine: Making sense of raw text.Briefings in Bioinformatics, 6(3), 239-251.

[2] Kashyap V (2008) Ontologies and Schemas. The Semantic Web, 79-135. doi:10.1007/978-3-540 764526_5

[3] Abeysiriwardana, P. C., & Kodituwakku, S. R. (2012). Ontology-Based Information Extraction for Disease Intelligence. International Journal of Research in Computer Science, 2(6), 7-19. doi:10.7815/ijorcs.26.2012.051

[4] Katifori, A., Vassilakis, C., & Dix, A. (2010). Ontologies and the brain: Using spreading activation through ontologies to support personal interaction. Cognitive Systems Research, 11(1), 25-41. doi:10.1016/j.cogsys.2009.02.001

[5] Hendler, J., & Berners-Lee, T. (2010). From the Semantic Web to social machines: A research challenge for AI on the World Wide Web. Artificial Intelligence, 174(2), 156-161. doi:10.1016/j.artint.2009.11.010

[6] Ramanathan V. Guha (2013). "Light at the End of the Tunnel". International Semantic Web Conference 2013 Keynote.

[7] 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

[8] 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\

[9] 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.3199036

[10] Narock, T., & Fox, P. (2015). The Semantic Web in Earth and Space Science. Current Status and Future Directions. Amsterdam, Netherlands: IOS Press.

[11] McDaniel, M., &Storey, V. C. (2019). Evaluating Domain Ontologies. ACM Computing Surveys, 52(4), 1-44. doi:10.1145/3329124

[12] Façanha, R. L., Cavalcanti, M. C., & Campos, M. L. (2019). A Systematic Approach to Review Legacy Schemas Based on Ontological Analysis. Metadata and Semantic Research, 63-75. doi:10.1007/978-3-030-14401-2_6

[13] Trokanas, N., &Cecelja, F. (2016). Ontology evaluation for reuse in the domain of Process Systems Engineering. Computers & Chemical Engineering, 85, 177-187. doi:10.1016/j.compchemeng.2015.12.003

[14] Munir, K., & SherazAnjum, M. (2018). The use of ontologies for effective knowledge modelling and information retrieval. Applied Computing and Informatics, 14(2), 116-126. doi:10.1016/j.aci.2017.07.003

[15] Elve, A. T., & Preisig, H. A. (2018). From ontology to executable program code. Computers & Chemical Engineering. doi:10.1016/j.compchemeng.2018.09.004

[16] 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

[17] 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

[18] Simperl, E., &Luczak-Rösch, M. (2013). Collaborative ontology engineering: a survey. The Knowledge Engineering Review, 29(1), 101-131. doi:10.1017/s0269888913000192

[19] 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

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

[21] 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.

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

[23] Bonacin, R., Nabuco, O., &Pierozzi, I. (2016). Ontology models of the impacts of agriculture and climate changes on water resources: Scenarios on interoperability and information recovery. Future Generation Comp. Syst., 54, 423-434.

[24] 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.3199036

[25] Hardin, G. (1968). The Tragedy of the Commons. Science, 162(3859), 1243-1248. Retrieved January 29, 2020, from www.jstor.org/stable/1724745

[26] Nagle, Frank, The Digital Commons: Tragedy or Opportunity? A Reflection on the 50th Anniversary of Hardin's Tragedy of the Commons (December 13, 2018). Harvard Business School Strategy Unit Working Paper No. 19-060. Available at SSRN: https://ssrn.com/abstract=3301005 or http://dx.doi.org/10.2139/ssrn.3301005

[27] May, P. Hanrahan, D. A. Keim, B. Shneiderman, and S. Card, "The state of visual analytics: Views on what visual analytics is and where it is going," in 2010 IEEE Symposium on Visual Analytics Science and Technology, 2010, pp. 257–259.

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

[29] Bonacin, R., Nabuco, O., &Pierozzi, I. (2016). Ontology models of the impacts of agriculture and climate changes on water resources: Scenarios on interoperability and information recovery. Future Generation Comp. Syst., 54, 423-434.

[30] 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.3199036

[31] 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

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

[33] 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

[34] 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.

[35] 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.

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

[37] 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

[38] 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

[39] 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

[40] 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

[41] 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

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

[43] Renzi, A.B., Chammas, A., Agner, L., Greenshpan, J.: Startup Rio: user experience and startups. In: Design, User Experience, and Usability: Design Discourse, Los Angeles, CA, pp. 339–347. Spring (2015)

[44] Arnold, R.D., & Wade, J. (2015). A Definition of Systems Thinking: A Systems Approach

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

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

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

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

[49] Stockdale, R., & Standing, C. (2006). An interpretive approach to evaluating information systems: A content, context, process framework. European Journal of Operational Research, 173, 1090-1102.

[50] Boyatzis, R. (1998). Transforming Qualitative Information: Thematic Analysis and Code Development.

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

[52] Androutsopoulos, I., Lampouras, G., & Galanis, D. (2013). Generating Natural Language Descriptions from OWL Ontologies: the NaturalOWL System. J. Artif. Intell. Res., 48, 671-715.

[53] Gonano, C.M., Tomasi, F., Mambelli, F., Vitali, F., & Peroni, S. (2014). Zeri e LODE. Extracting the Zeri photo archive to linked open data: formalizing the conceptual model. IEEE/ACM Joint Conference on Digital Libraries, 289-298.

[54] Third, A., Williams, S., & Power, R. (2011). OWL to English: a tool for generating organised easily-navigated hypertexts from ontologies.

[55] 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

[56] Hossain, B., Rajan, G., & Schwitter, R. (2019). CNL-ER: A Controlled Natural Language for Specifying and Verbalising Entity Relationship Models. ALTA.

[57] Krishnamurthy, R., & Hsiao, M. (2020). Transforming Natural Language Specifications to Logical Forms for Hardware Verification. 2020 IEEE 38th International Conference on Computer Design (ICCD), 393-396.

[58] Sak, H., Beaufays, F., Nakajima, K., & Allauzen, C. (2013). Language model verbalization for automatic speech recognition. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, 8262-8266.

[59] Perera, V., Selvaraj, S.P., Rosenthal, S., & Veloso, M. (2016). Dynamic generation and refinement of robot verbalization. 2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), 212-218.

[60] Venugopal, V.E. (2018). Ontology Verbalization using Semantic-Refinement.

[61] 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.

[62] Ralston, K., Chen, Y., Isah, H., & Zulkernine, F. (2019). A Voice Interactive Multilingual Student Support System using IBM Watson. 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA), 1924-1929.

[63] Bocklisch, T., Faulkner, J., Pawlowski, N., & Nichol, A. (2017). Rasa: Open Source Language Understanding and Dialogue Management. ArXiv, abs/1712.05181.

[64] Jespersen, L.N., Michelsen, S.I., Holstein, B., Tjørnhøj-Thomsen, T., & Due, P. (2018). Conceptualization, operationalization, and content validity of the EQOL-questionnaire measuring quality of life and participation for persons with disabilities. Health and Quality of Life Outcomes, 16.

[65] Fonseca, c. (2016). A new theory for the study of texts: theory of rhetoric structure - Rethorical structure theory.
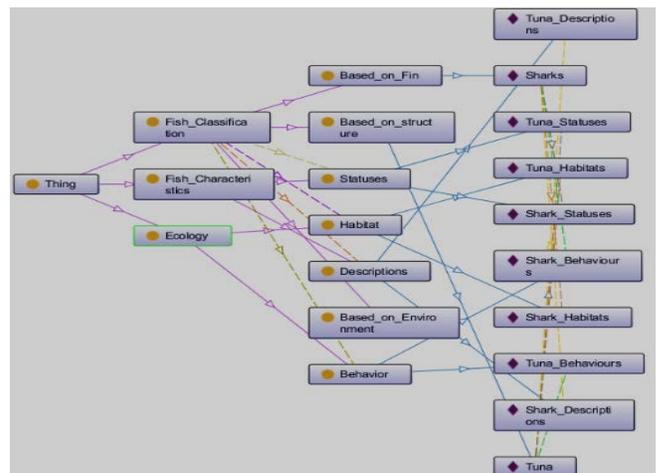
APPENDIX

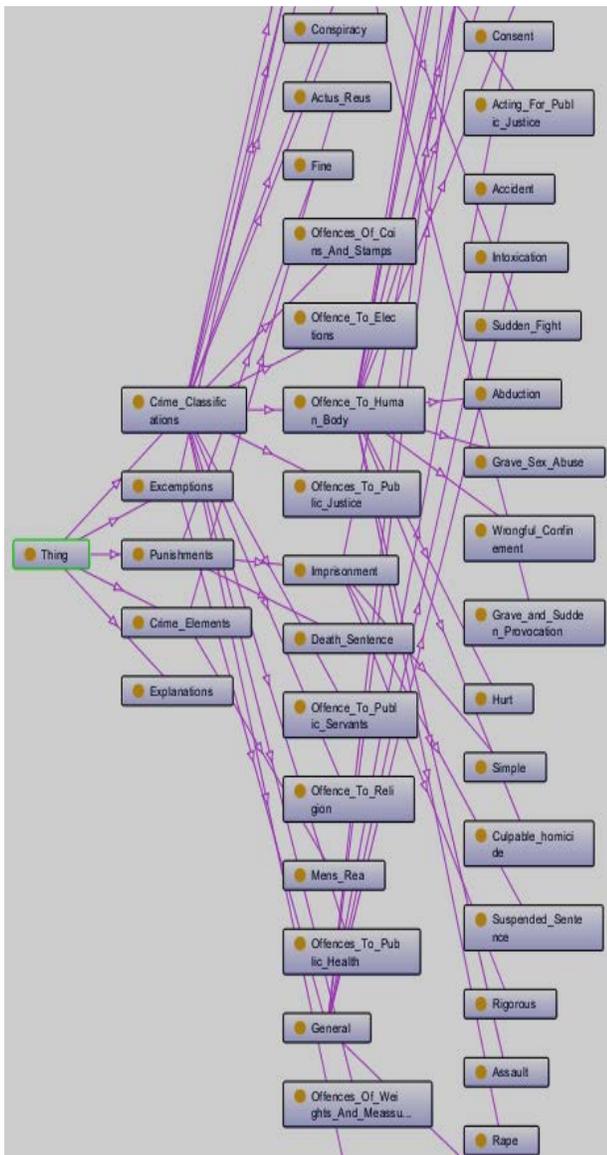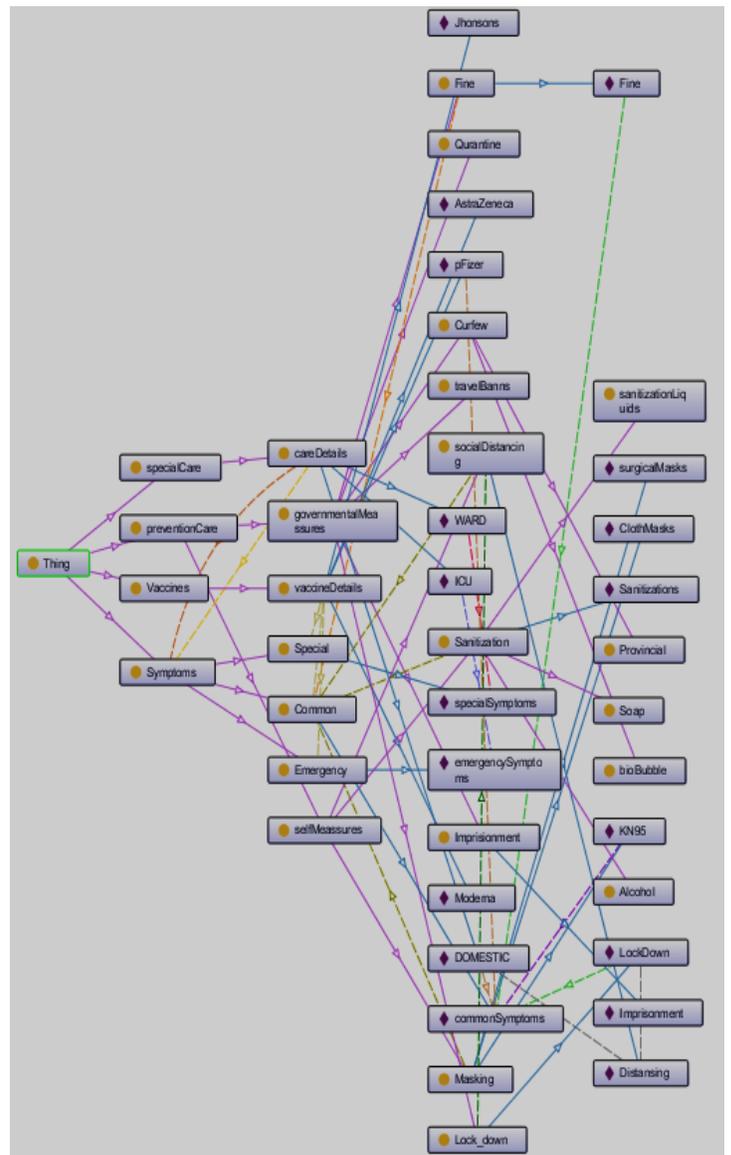Fig. 10. Ontology Increment for Aquaculture Domain.

Fig. 11. Ontology Increment for Criminal Law.

Fig. 12. Ontology Increment for COVID-19.