UniOntBot: Semantic Natural Language Generation based API approach for Chatbot Communication

Lakindu Gunasekara
Informatics Institute of Technology
No 57, Ramakrishna Road,
Colombo 6, Sri Lanka
+94 (0) 77 1629563
lakindu1995@gmail.com

Kaneeka Vidanage
Informatics Institute of Technology
No 57, Ramakrishna Road,
Colombo 6, Sri Lanka
+94 (0) 76 8209746
kaneeka.v@iit.ac.lk

Abstract— **Natural Language Generation is a sub task of the natural language processing where the machine represents texts in human understandable language. Although there have been various researches carried out for Natural Language Generation since the 1970s, there are only few available work carried out with semantic technologies compared to linguistic surface oriented structures. This paper presents the available work which combine semantic technologies with Natural Language Generation, and it identifies opportunities and drawbacks of such systems. The research is carried out on how to use semantic natural language generation with chatbots with lower computational cost and on the ability to reuse it for similar domains with less coding on natural language generation component for small scale level domains. Additionally, a new architecture for chatbot using an ontology and a new domain ontology with natural language resource bind are contributed by the researchers.**

Keywords— **Natural Language Generation, Semantic Technologies, Ontologies, Knowledge Modelling, Chatbots, Application Programming Interfaces, Architecture**

I. INTRODUCTION

Apart from providing tertiary education, a university spends a significant amount of resources on gaining and maintaining their visibility among the target markets. In order to attract younger generation, the marketing and management strategies need to be precise. For a university website or social media page, there can be miscellaneous questions asked by pre-enrolled students or other stakeholders. In some scenarios, the replies for enquiries might get delayed due to time zone differences or the unavailability of human resources in the university.

The target market for higher education institutions is primarily on mobile devices. A study done in 2017 shows that millennial internet users spend, on average, 223 minutes per day immersed in their mobile devices [1], which is a gradual increase over the past years. Therefore, instant messaging is preferred to the use of emails in order to communicate. This is where chatbots can be used in order to fill the gap.

Most of the queries from universities tend to be of similar type. This can be taken as an opportunity to build a chatbot using the frequent queries. The chatbot should be able to clarify applicant queries even before connecting to a university agent.

There have been various approaches followed in order to create conversation agents in the development process. Artificial Intelligence, usage of rule-based chatbots are two main different approaches.

In the process of developing a chatbot with traditional techniques for a smaller domain such as a university, it takes time due to the need for training of massive amount of data, and there are several filtering methods required in order to understand users' questions. So in the case of a smaller domain chatbot, it takes time to develop it accurately due to the above issues [2]. Another issue is that websites with relevant results are existing, but they do not provide the specific information needed directly [3]. Generating such natural responses to the user is a tricky job because of the lack of information in the particular domain to train data. In order to develop a proper Natural Language Generation with traditional methods, it is required to have an accurate dataset for the specific university or the domain. In the traditional approach of the generation of a natural language, the machine does not understand the context with the generated text. In the following segment of the paper, the disadvantages of the traditional approach are given.

Traditionally human knowledge is presented in an informal and predominant manner in natural language. Natural language is highly expressive, and there is no need for an extra learning effort. When concerning the presenting of knowledge, the disadvantage is the possibility of ambiguity, vagueness and potential inconsistencies in natural language. To represent knowledge in computers, people use formal languages [4].

In order to present knowledge, there should be a bridge to map conceptual distance. Even though there are several text generation mechanisms available in semantic technologies to address the above problems, the usage of such technologies is limited to a particular tool or a platform. Such limitations of semantic technologies lead the developers to find alternative methods which are computationally not intelligent.

There is some work available with semantic technologies with natural language generation, but none of the approaches have used ontology based chatbots. Our goal is to propose a solution for chatbot communication by analysing the existing work and approaches.

In section II, we present the methodology on how the knowledge is gathered to analyse the solution, which is the initial phase of the research. In section III, we present an in-depth analysis of the chatbots, natural language generation and how semantic technology will provide more background information. Section IV analyses the existing libraries and existing work on three different aspects. In section V, a comparison of the existing work is carried out. A reflection of the research gap is discussed in Section VI.

978-1-7281-5569-2/19/$31.00 ©2019 IEEE

Authorized licensed use limited to: Leeds Beckett University. Downloaded on February 20,2023 at 14:29:19 UTC from IEEE Xplore. Restrictions apply.

Section VII will discuss the proposed solution along with each component of the project. Section VIII and IX cover the testing and evaluation of the project. Section X concludes the research and states the future work.

II. METHODOLOGY

To gain domain knowledge, we used literature surveys under several categories, i.e. ontology based chatbots, Natural Language Generation, Semantic Natural Language Generation, Knowledge Modelling and SPARQL, and we came across nearly 80 research papers related to our research. The significant work identified are discussed in the section IV of the paper.

With the knowledge gained, we constructed a theory for the research based on the use of semantic natural language generation for chatbot text generation based on the hypothesis, "if the semantic natural language can be used to generate texts based on the ontology, this technology can be used for chatbots to generate responses with a lower computational cost with an intelligent approach for the machine". Since we started with a theory, the research methodology is identified as a deductive approach. To start off the project we followed the PRINCE2 project management methodology since it is flexible with a research project. When designing the system, we used an object oriented design as the design methodology. After the system is designed we used Prototype model as the development methodology.

III. MODULAR DESCRIPTION

In this section, an in-depth analysis of each component of the chatbot is made, and this section also shows how the links are attached to chatbots with semantic natural language generation by analysing the strengths and weaknesses.

A. Chatbots

Chatbot can be a software which interacts with people using natural language by building a conversation. The chatbot interaction with the users can be textual or auditory depending upon the need [2].

Compared to commercial chatbot domains, the domain of a university/institute is smaller. But the number of queries directed to a university is not less than that of the queries directed to a commercial institution, and a similar set of queries would be asked. When a university grows, the methods of interaction with its stakeholders become more cumbersome. In such scenarios, the automation of the support system becomes a mandatory task.

Using traditional methods such as machine learning in chatbot development for smaller domains will not give successful results due to the lack of data in the domain and also due to lack of training relating to conversations. A paper published in ICICCT in 2018 [2] states that they have used TensorFlow to develop a chatbot system targeting small businesses. In the results section of the paper, it is stated that the machine learning approach to create a bot for a smaller domain does not give accurate results because the machine learning does not understand the meaning of the sentences. The chatbot only learns to respond to the users based on the previous experience [2]. Another paper which evaluates the current development challenges mentions that "Chatbot generates responses using deep learning techniques to train generative model which is used to input and generate the answer. Keep in view of this need, there are millions of examples which are needed to learn by Chatbot to deliver responses" [5]. According to research published on ontology based chatbot, it states that in order to develop traditional chatbots, the developer must learn specific languages such as AIML, high botmaster interreference and the use of non-matured technology [6]. A chatbot is a system that comprises of several key components such as, 1) Natural Language Understanding, 2) Dialogue Management, 3) Knowledge Base, 4) Natural Language Generation.

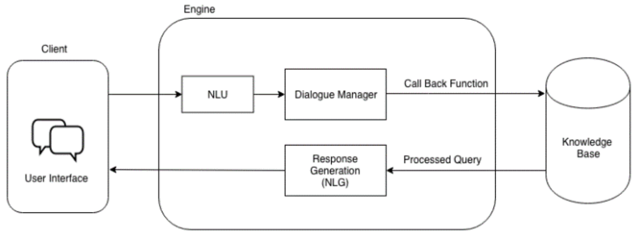

Fig. 1 Chatbot Architecture

Natural Language Understanding (NLU) is an integral part of a chatbot system which is used to obtain the context of the user utterances. This can be done using a simple keyword extraction by using the given keywords as values. The purpose of Natural Language Understanding is to interpret text.

Dialogue Management is the central component of the chatbot. It accepts the output of the Natural Language Understand component and produces the responses to be communicated to the users. This needs an external knowledge resource to create the responses and to control the flow [7]. The raw responses generated from the Dialogue Manager will be passed to text Natural Language Generation component to refine the text response.

Knowledge base component will arrange the data in a formal way in order to provide data seamlessly to the dialogue management component. The domain related information can be stored in this component.

Natural Language Generation (NLG) is a sub component of Natural Language Processing, which constructs the understandable text responses in natural language with a non-linguistic representation [8]. The Natural Language Generation component will be discussed broadly in the next few chapters since the project is based on NLG component of the system.

B. Natural Language Generation

Natural Language Generation is the way of generating suitable texts for human readability in a machine perspective [9]. The major work carried out for Natural Language Generation was from the 1970s onward, which are Goldman's work on the issue of working on lexicalizing the underlying conceptual materials. Another work carried out is providing description for a tic-tac-tie game by Davey in 1979. These are considered as some of the first contributions focused on NLG [10].

The output of the generation can be in a pre-structured or in a completely unstructured format.

Authorized licensed use limited to: Leeds Beckett University. Downloaded on February 20,2023 at 14:29:19 UTC from IEEE Xplore. Restrictions apply.

i) Static responses: This is the simplest way of generating a text. The text response is already pre-defined by the system, where the variables of the sentences are replaceable. This type of responses could be used as a template such as, “The weather of <location> is <temperature> C”.

ii) Dynamic responses: Another approach to generate text is by using resources. This can be a knowledge base along with a decision-making component which determines the score for the nearest generated response by matching up with the user query. This is mostly used with Q and A systems.

iii) Generated responses: In this approach, a huge chunk of conversation examples is taken into a deep learning technique to train the machine to generate responses. More data would be needed in order to provide more accurate responses, and it would also incur a higher computational cost. Sometimes the generated responses could be irrelevant, yet with more training and defining, more rules would give more accurate information [11].

Even though there have been many representations over the years for linguistic surface-oriented structures over semantic or conceptual representations, the most “natural” representation of Natural Language Generation is considered as semantic/conceptual representation. Because of the characteristics of the semantic web machine processable paradigm, it has been an interesting topic to NLG enthusiasts [12].

Even though the traditional NLG needs to be trained with higher computational cost, the component needs to have more data and more conversations in order to provide accurate output. But if the computer can understand the context of the stored information, there can be a hypothesis that the computer would perform better with less computational cost and less filtering algorithms.

C. Semantic Web and NLG

Semantic web can be identified as an extended version of the world wide web since the data are associated in a way that machines can easily process without the help of humans [13]. Semantic web converts the unstructured data into meaningful formatted data/information. A unique feature of semantic web is that it can be understood by both humans and machines.

It is built upon a Resource Description Framework (RDF) and the syntax is designed using the Uniform Resource Identifiers (URI). The goal of this technology is to search for information with less effort without using filtering techniques and with a lesser computational cost.

When a computer performs operations, there is a lack of understanding of each element which is being executed, which means computers “have no notion” on its operational elements. This is where the semantic technologies based on the hypothesis of machines will demonstrate more intelligent behaviour if the operational elements are equipped with formal descriptions [13].

In a scenario of a particular domain, such knowledge can be represented using separate ontologies and can be integrated later with OWL import mechanisms which provide a modularization of knowledge with limited access. The task of generating texts via an ontology is known as ‘ontology verbalisation’ [14]. According to the research paper, ‘Automating generation of textual class definition from OWL to English’, the architecture of the OWL verbalizer is as follows:

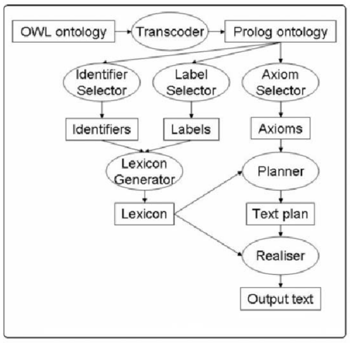

Fig. 2 OWL verbaliser architecture

There are different kinds of tools acting as the ontology verbalisers. For instance, some verbalisers are concerned only about ABox (individuals in OWL); or other verbalisers can produce separate sentences for each OWL axiom. There are several ontology authoring tools available such as Protégé, SWOOP and topBraid Composer.

IV. EXSITING SYSTEMS

The existing systems are discussed in the following section and the systems are divided into three categories, namely OWL verbalise libraries, OWL verbalise projects and Ontology based chatbots.

A. OWL Verbalisers Libraries

1) Attempto Controlled English:

Attempto Controlled English is a first-order logic controlled natural language for English language which is restricted to its subsets [4]. This verbaliser is good to use in generic domains since the lexicons are also done in an automatic manner. The verbaliser supports the ontology classes and properties as well as individuals and assertions for the individuals. Aggregation is not supported in the ACE’s OWL verbaliser. Since the verbaliser does not provide any functionality for a special use case or specified domain, this cannot provide accurate results which our research focuses on, and a third-party integration of the verbaliser is not mentioned in the papers.

2) NaturalOWL system:

NaturalOWL is a natural language generation system which is typically used to describe an ontology by the individuals and classes of the ontology in the OWL form. It is an open-source library which supports both English and Greek [17].

Because of the flexibility of creating NL resources, it is more suitable on smaller domains to create an NLG system. Since the user needs to define the NL resources, this kind of library is not suitable on large scale ontologies, where the author will be having extra work on generating the texts. Also, this library does not provide any support on integration with other systems clearly.

B. OWL Verbaliser Projects

1) BioASQ:

BioASQ is an exploitation and dissemination plan with the objective of building a platform for organizing the challenges for biomedical questions and answering [19].

Authorized licensed use limited to: Leeds Beckett University. Downloaded on February 20,2023 at 14:29:19 UTC from IEEE Xplore. Restrictions apply.

According to the above-mentioned paper, if an ontology is mapped and created by defining the classes and individuals, any ontology could be used to generate text with the NaturalOWL or any OWL verbaliser. But if an ontology is created from the beginning, the author must have a proper understanding of the type of responses expected to generate and on how the knowledge should be structured in an ontology using protégé or any other ontology editors. Creating only an ontology is not sufficient, but it is also necessary to create a Natural language resource which is relevant to the domain of the ontology.

2) ReadMe generation from OWL ontology for NLP

This paper is written on an ontology-based solution for generating the ReadMe files for software by using an NLP tool. [20]. Although in the above research, authors have not described the OWL verbalising library, the functions of generating text are mentioned. Before generating texts, it is important to structure the ontology in a meaningful manner. For the translation, another ontology is created to support the main ontology which acts as the Natural Language resource. But to get quality sentences, there should be an expression aggregation and a surface realisation apart from the above mentioned NLG steps. The approach taken to generate ReadMe texts gives a clear overview to get a start in our research.

C. API based NLG systems

1) SimpleNLG:

SimpleNLG is a Java API based library which generates sentences after defining the subject verb and objects. The library follows an API approach to construct sentences and the words need to be passed from the Java programme. A programming knowledge is required to setup the inputs to the library from the Java application. This is a good approach for the small-scale domains since the lexicons can be defined by the user even though the user needs to have a programming skill. However, for the ontologies written in OWL file, the creator states that it is not tested specifically on the OWL ontologies. Therefore, for our use case which is to generate an API for chatbots, using ontologies would be a challenging task.

D. Ontology based chatbots

1) OntBot: Ontology based Chatbot:

A chatbot is a computer programme which interacts with users using natural language generated from the machine. Ontologies facilitate the communication between the human and the machine [6].

Even though ontology is used for the database section of the chatbot, which will reduce the computational cost and will search the data effortlessly, the natural language generation component comprises of several advanced techniques which will still provide a higher computational cost within the NLG component.

2) E-commerce ontology chatbot:

The chatbot research done by the Vegesna, Jain and Porwal is an e-commerce chatbot which retrieves relevant data from E-Bay website and presents the details to users. In this paper, it is stated that "There are a variety of information extraction technologies, but it is difficult to find the 'real' information that is required directly. We propose an Ontology based chatbot approach to get the desired appropriate answer to the query." [3]. With the above statement, the advantage of using an ontology is highlighted, and furthermore, the suggested approach would retrieve the exact data without the effort of searching by the user. Since the ontology and e-commerce websites both have a structured body, this method is suggested by the author.

The response generation is done using natural language generation techniques but the paper does not reveal the text generation section broadly. But the answer generated will be sent back to the user using a webhook between the API.AI (TensorFlow) and the user interface. We can assume that TensorFlow is used to create the NLG by retrieving the relevant information.

V. EVALUATION OF EXISTING SYSTEMS

This section presents a comparison of the related work and the libraries discussed in the section IV.

A. OWL Verbaliser Comparison

TABLE I. OWL VERBALISER COMPARISON

Verbaliser	Coverage	Classes & Properties	Individuals & assertions	Lexicon	Domain	Suitable for small domains
ACE	OWL 2	Yes	Yes	Automatic	Generic	No
SWAT	OWL 2	Yes	Yes	Automatic	Generic	No
ROA	Not-specified	Yes	Yes	Automatic	Generic	No
SWOOP	OWL-DL	Yes	Not-specified	Automatic	Generic	No
MIAKT	RDF	No	Yes	Automatic	Generic	No
Liber	RDF	No	Yes	User-defined	Specific	Yes
NaturalOWL	OWL-DL	Yes	Yes	User-defined	Specific	Yes

In the selection of the OWL verbaliser for the project, the above details were considered mainly evaluating them for our case of smaller domain natural language generation development. The evaluation is conducted based on the topics shown in the above table. Verbaliser coverage is the type of file which is supported on each verbaliser in order to create the generation. Class and Property support known as the TBox is a basic element of an ontology. Ontology has classes and properties assigned with each element to describe the element well. Individuals and assertions known as ABox support will be helpful to define an element of an ontology. Lexicons indicate the source of the lexicon entries, which is the defining of the vocabulary. The domain can be indicated as the suitable domain which an OWL verbaliser can be used. Generic means that the OWL verbaliser can be suitable for a general purpose rather than for a specific domain, and finally, based on the

Authorized licensed use limited to: Leeds Beckett University. Downloaded on February 20,2023 at 14:29:19 UTC from IEEE Xplore. Restrictions apply.

domain, if the domain is specific, we can generate accurate results by defining the vocabulary.

None of the verbalisers have been used with a third-party integration instead of using a tool such as Protégé. Therefore, this is limited to relevant software.

B. *Ontology based Chatbot Comparison*

TABLE II. ONTOLOGY BASED CHATBOT COMPARISON

Chatbot	Domain	Use of ontology	Framework	NLG system
OntBot	General	Yes	Own system	JBoss DNA, MorphAdorner, Verbix
ClimeBot	Climate	Yes	TensorFlow	.Static response, Entailment module
Ecommerce OntBot	ECommerce	Yes	TensorFlow	Not specified
ChatBol	Football	Yes	Rasa	Wikidata

In the evaluation of the existing chatbots which use ontology, most of the bots are domain specific and have used various mechanisms to generate responses. By omitting the traditional chatbot development methods, the ontology-based approach was selected to share the knowledge for the users. Even though ontologies have been used, they have not been fully used in the system. If an ontology exists, the text generation can be done using the ontology itself by using OWL verbalisers.

VI. REFLECTION OF RESEARCH GAP

In the previous sections, we have discussed the problem of the domain and technology and the related contributions made by other researchers. Our focus is for developing a chatbot text generation system for small scale domains with minimal computational cost and with less amount of data to produce accurate information. The model of the system is reusable for similar domains to produce sentences, and moreover it can be achieved with minimal coding.

By reviewing the related products in the above chapters, we have discussed the related products in three different areas: OWL verbalisers, Products developed using OWL verbalisers and ontology based chatbots. When evaluating the above products, it is found that none of the verbalisers have focused on the use of OWL verbalisers on chatbots to generate responses and the ontology based chatbots have not used the semantic web approach to develop the response, instead most of the ontology based chatbots use a static response generation or multiple methods in one component making the architecture complex. Even though an ontology has been used already, the other researchers have not used the existing ontology to create responses within the semantic technologies. Instead, the ontology text generators are focused on generating ReadMe's or on developing a complaint answering system, which is still under research.

If the integration of OWL verbalisers with the ontology based chatbots is a success, the end goal of the research will give successful results and further research can be conducted on evaluating the performance and to prove of the hypothesis created.

VII. PROPOSED SOLUTION

This section discusses the proposed solution of the UniOntBot (University Ontology Chatbot), and the figure below shows the high-level architecture of the UniOntBot.

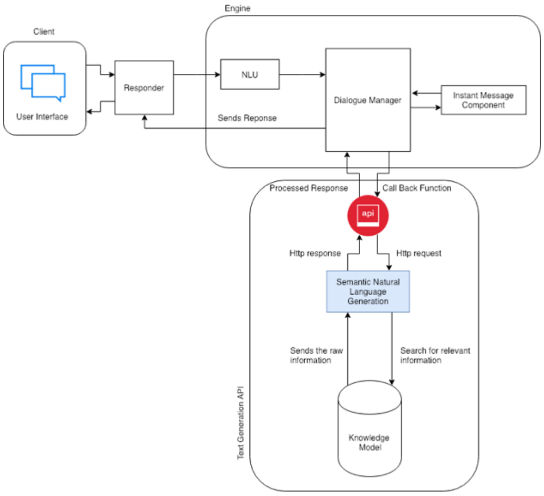

Fig. 3 High level Architecture of UniOntBot

The proposed system is a chatbot which contains a knowledge model along with an OWL verbaliser by exposing to the chatbot system through an API. The chatbot contains an NLU component and a dialogue management component in order to process the user inputs. After inputs are received to the NLU, it will identify the context of the user utterance and will pass the values to the dialogue management system. If the user query is not an informative query, an instant response will be sent back to the user. If the query is seeking for information, a request is called for the API with the values. The values will then be checked from the ontology, and the relevant raw information will be retrieved and passed to the OWL verbaliser (Semantic Natural Language Generation). From the OWL verbaliser, a complete sentence will be generated and an API response will be sent back to the dialogue management component in the chatbot. The response will then be sent to the user interface of the chatbot and to the user.

The uniqueness of the solution is that integration of an OWL verbaliser and the informative sentences are being generated with the ontology itself by the machine with an understanding.

Even though a chatbot starts with the NLU component, the research is carried out within the API, and the priority of the components is given in the reverse order of the chatbot.

A. *Knowledge Model*

With the use of protégé, an ontology is created by adding the details of the university/institute. A snap of the ontology created is added in the figure.

A knowledge representation can be interpreted in machines with the use of ontologies. This ontology contains the details of the degree information along with other user query related information. After interviewing the marketing executives in an institute, we have identified the frequent questions asked via emails or phone calls. The identified data are added to the data properties of the classes created.

Authorized licensed use limited to: Leeds Beckett University. Downloaded on February 20,2023 at 14:29:19 UTC from IEEE Xplore. Restrictions apply.

The yellow circle icon in the figure represents the classes and the purple diamond represents the individuals attached with each class. The format of the ontology is an OWL file. The identified data are categorised into three sections, i.e. the degree details, information about the university and how to apply to the university.

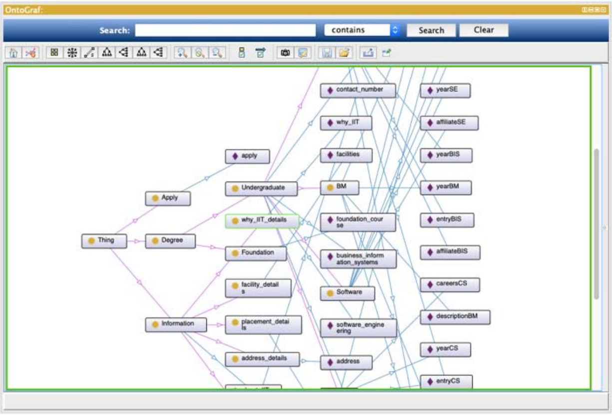

Fig. 4 Ontology Graph

B. OWL verbaliser (NLG component)

The OWL verbaliser is an integral component of the system where the sentences are generated from this component with the details from the ontology discussed. We picked NaturalOWL as the OWL verbaliser in this project. A detailed overview of NaturalOWL is discussed in the related libraries section.

For this project, we used the NaturalOWL plugin, which is created for protégé software. After adding the plugin, we created an NLResource for the ontology we created above. The plugin provides the features in the protégé, i.e. Lexicons, NL names, Sentence plans, sections and order user modelling and Text generation.

Since NaturalOWL sentences need to be defined by the user, it is easy for customisations according to the domain needs rather than generating texts automatically for smaller domains. The lexicons have to be defined by the user. Nouns, adjectives and verbs need to be defined or have the ability to import a pre-defined NL resource.

The NL Names will allow the definition of certain names as the user needs, and they can be added to sentence plans later on. Sentence plans play a major role in defining the sentences, and they create the structure of each sentence required to get the output by defining the nouns, verbs, strings, prepositions, concatenations, property fillers and owners. With the use of the following details, a sentence can be created. Sections and order provide the order of the multiple sentences.

Another special characteristic of NaturalOWL is the User modelling. The sentences can be generated as per user types. The generated texts can sound like a child or an adult. This can be defined by the user. And the text generation section provides features defining the graph distance and other configurations and the output of the sentence in the protégé.

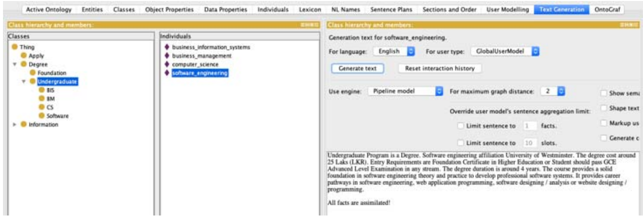

Fig. 5 NaturalOWL text generation

C. API

In order to expose the protégé generated sentences, we use an API based approach, which will be integrated with the core of the chatbot system. We created a Spring Boot application which is a Java based API by customising the provided example for java application for our own purpose. The Java API is set on getting the input from the API call from the chatbot, and it retrieves the surface realisation component which provides the generated sentence.

The relevant individual or class is searched within the ontology with the use of IRIs and process in the NL Engine and outputs the sentence as the surface realisation.

D. Dialog Manager

Dialog manager is the core of the chatbot system. In order to demonstrate the use of semantic NLG component, a chatbot is created. Rasa core is the dialogue management component of the system, which will decide on what response is needed to be sent. In order to decide on what responses to be sent, we need to fill the core with write stories, define domains and train the dialogue model.

E. Natural Language Understanding

The NLU component understands the user queries and extracts the context of user utterances. Intents and entities are defined in the chatbot system. As the NLU component, Rasa NLU library is used. Rasa NLU follows a modular design which relies on the existing NLP and Machine learning libraries such as scikit-learn, spacy, tensorflow, keras and sklearn-crfsuite. To improve accuracy, a list of questions annotated with intents and entities are prepared in a markdown file. Following are some of the example questions created for the UniOntBot project.

- Can you tell me about [software engineering](degree)?
- What is [Business Management](degree) course about?
- What's the [cost](degreeinfo) of [Software Engineering] (degree) degree?

There is a list of questions pre-defined in many ways in order to identify the user query context accurately.

F. Screenshots of the chatbot

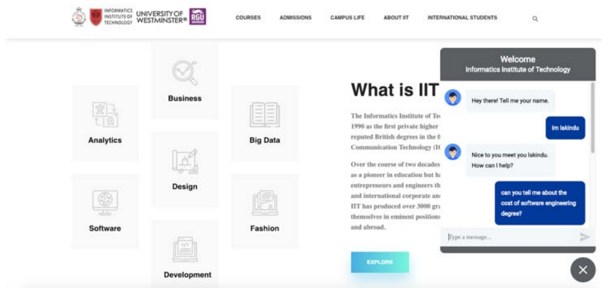

Fig. 6 Screenshot 1

Authorized licensed use limited to: Leeds Beckett University. Downloaded on February 20,2023 at 14:29:19 UTC from IEEE Xplore. Restrictions apply.

VIII. TESTING

A criterion is needed for testing in order to reduce the gap between expected and actual results.

A. Accuracy

To test the accuracy of the system, a manual data gathering is performed by taking 11 users and getting 10 queries per each person to construct a confusion matrix.

n = 110	Expected: No	Expected: Yes	
Actual: No	TN = 23	FP = 11	34
Actual: Yes	FN = 7	TP = 69	76
	30	80	

Fig. 7 Confusion matrix

Overall accuracy of the system,

• (TP+TN)/total = (69+23)/110 = 0.836

Recall rate of the system,

• TP / (TP+FN) = 69 / (69+11) = 0.907

Precision rate of the system,

• TP / (TP+FP) = 69 / (69+11) = 0.862

B. Performance

The performance of the system is tested locally by testing the overall application performance and API performance.

1) Overall Application Performance

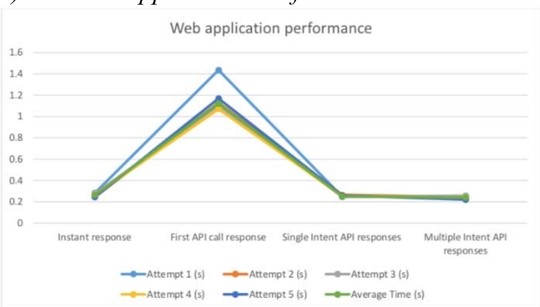

Fig. 8 Application performance

First an instant response is tested by the user, which would be replied from the chatbot itself. Secondly, an informative query is asked from the chatbot, which would need an API call to retrieve the information. As it is the first API call, it takes a little time to retrieve. The graph shows that it has taken more than 1 second to retrieve the information and present the response. After the first API call, the rest of the API calls are given under 0.5 seconds.

2) API Performance

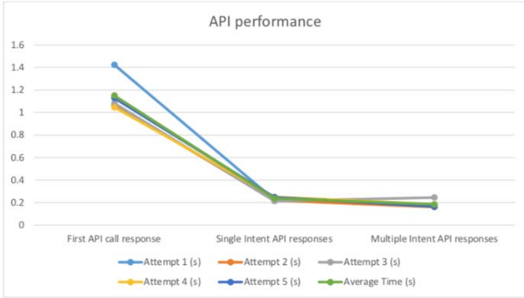

Fig. 9 API performance

The API performance is tested locally by using the postman application to determine the time. The Figure represents the results of the API performance. Even though the first API call takes time to retrieve information, later on, even multi intent input of an API response would maintain a time below 0.4 seconds.

IX. EVALUATION

A thematic analysis was conducted to get evaluation from industrial and domain experts.

TABLE III. SAMPLE EVALUATION

Designation	Research Associate, University of Cambridge, UK
"This seems quite interesting, and I am wondering how well did the NaturalOWL library integrate with the RASA framework, and what challenges you have encountered." "All in all, it looks like a nice chatbot, with some opportunities for interesting improvements." "It all sounds very interesting and I am glad to see you have managed to integrate NaturalOWL through RASA with such success. Thank you for sharing this additional video as well, as it makes NaturalOWL's usage much clearer."	

There are 13 evaluations gathered from different experts, and further information could be gathered by contacting the authors.

X. CONCLUSION

The work carried out so far have not been applied on the text generation of the ontology based chatbots, and mostly the text generation has been a static method in most of the chatbots. To make better use of the ontologies, our solution which is an ontology based natural language generation system would be an approach. But so far, the OWL verbalisers are available on the research level along with their experimental products such as BioASQ.

With the knowledge gained, we propose an API based mechanism which can be applied on chatbots to generate text response by OWL verbalisers. Since our approach is an API, it can be easily integrated with any system as the future work, and a framework can be developed to be used for any domain to plug and play with any chatbot or any other system.

REFERENCES

[1] Statista, "Millennials: daily mobile access 2017," *Statista*, 2018. [Online]. Available: https://www.statista.com/statistics/283138/millennials-daily-mobile-usage.

[2] R. Singh, H. Patel, M. Paste, and N. Mishra,

Authorized licensed use limited to: Leeds Beckett University. Downloaded on February 20,2023 at 14:29:19 UTC from IEEE Xplore. Restrictions apply.

"Chatbot using TensorFlow for small Businesses 1," *2018 Second Int. Conf. Inven. Commun. Comput. Technol.*, no. Icicct, pp. 1614–1619, 2018.

[3] A. Vegesna, P. Jain, and D. Porwal, "Ontology based Chatbot (For E-commerce Website)," *Int. J. Comput. Appl.*, vol. 179, no. 14, pp. 51–55, 2018.

[4] N. E. Fuchs, K. Kaljurand, and T. Kuhn, "Reasoning Web," vol. 3564, no. May 2014, 2005.

[5] A. Rahman, A. Al Mamun, and A. Islam, "Programming challenges of Chatbot : Current and Future Prospective," *IEEE Reg. 10 Humanit. Technol. Conf.*, no. March, pp. 19–23, 2018.

[6] H. Al-Zubaide and A. A. Issa, "Ontbot: {Ontology} based chatbot," *Innov. Inf. …*, pp. 7–12, 2011.

[7] "A SITUATION-BASED DIALOGUE MANAGEMENT USING DIALOGUE EXAMPLES Cheongjae Lee , Sangkeun Jung , Jihyun Eun , Minwoo Jeong , and Gary Geunbae Lee Department of Computer Science and Engineering Pohang University of Science and Technology , South Korea," *Strategy*, pp. 69–72, 2006.

[8] J. P. B. M. Aires, "Automatic Generation of Sports News," 2016.

[9] K. F. McCoy, "Natural language generation and assistive technologies," p. 1, 2012.

[10] R. Dale and B. Di, "Introduction to the Special Issue on Natural Language Generation," no. Goldman 1974, 1998.

[11] J. Couto, "Building a Chatbot: Analysis and Limitations of Modern Platforms," *DZone*, 2017. .

[12] B. Academy, O. F. Sciences, and I. Technologies, "Natural Language Generation and Semantic Technologies," vol. 14, no. 2, pp. 3–23, 2014.

[13] K. Staykova, "Natural language generation and semantic technologies," *Cybern. Inf. Technol.*, vol. 14, no. 2, pp. 3–23, 2014.

[14] R. Stevens, J. Malone, S. Williams, R. Power, and A. Third, "Automating generation of textual class definitions from OWL to English," *J. Biomed. Semantics*, vol. 2, no. 2, p. S5, 2011.

[15] A. Third, S. Williams, and R. Power, "OWL to English : a tool for generating organised easily-navigated hypertexts from ontologies," *Proc. 11th International Semant. Web Conf. ISWC2011*, pp. 5–8, 2011.

[16] B. Smith *et al.*, "The OBO Foundry : coordinated evolution of ontologies to support biomedical data integration," vol. 25, no. 11, pp. 1251–1255, 2007.

[17] I. Androutsopoulos, G. Lampouras, and D. Galanis, "Generating Natural Language Descriptions from OWL Ontologies : A Detailed Presentation of the NaturalOWL System," vol. 48, pp. 671–715, 2012.

[18] E. Reiter and R. Dale, "Building Natural Language Generation Systems."

[19] A. M. Alvers, G. Tsatsaronis, A. N. Ngomo, A. Kos-, and C. Armeniakou, "Exploitation and Dissemination Plan October 2014," vol. 4, no. October, 2014.

[20] D. Sadoun *et al.*, "Natural Language Generation and Semantic Technologies Kamenka Staykova," no. May, pp. 46–49, 2016.

[21] L. W. Nadjet Bouayad-Agha, Gerard Casamayor, "Natural Language Generation in the context of the Semantic Web," vol. 00, pp. 1–21, 2013.

[22] C. M. Keet and L. Khumalo, "Toward a knowledge-to-text controlled natural language of isiZulu," *Lang. Resour. Eval.*, 2016.

[23] D. Toniuc and A. Groza, "Climebot: An argumentative agent for climate change," *Proc. - 2017 IEEE 13th Int. Conf. Intell. Comput. Commun. Process. ICCP 2017*, no. September, pp. 63–70, 2017.

Authorized licensed use limited to: Leeds Beckett University. Downloaded on February 20,2023 at 14:29:19 UTC from IEEE Xplore. Restrictions apply.