Smart Virtual Expert System to Assist Psychiatrists (SVESTAP)

Udara Srimath S. Samaratunge Arachchillage

Faculty of Computing, Department of Software Engineering, Sri Lanka Institute of Information
Technology (SLIIT), Malabe, Sri Lanka,
E-mail: samaratunge@gmail.com

Abstract—Psychological issues in the world are exponentially growing and the treatment gap is also comparatively high. The main reason would be the shortage of expertise and time-consuming in conventional diagnose process. The main objective of this research is to lower the mental issues treatment gap of professionals or apprentices in the field by creating a virtual expert system to assist psychiatrists. This system diagnoses most common mental disorders such as Depression Disorder, Anxiety Disorder, and Dementia. The proposed expert system can communicate with patients, to identify the current state of the illness. During the conversation, a standard questionnaire is given for the disease verification purpose. The experienced mental health professionals can use this expert system to assist in diagnosing process and the apprentices of the psychology can use this expert system as a training asset.

Index Terms—Psychiatrists, Expert System, Knowledge base, Ontology, Natural Language Understanding (NLU), Natural Language Generation (NLG), Anxiety, Dementia.

I. INTRODUCTION

Health is considered as one of the most vital factors to measure the development of the society, and mental health is significant among them. According to the current statistics, the healthcare industry is one of the top most profitable industries in the world [1]. With the evolution of this scientific knowledge, the healthcare field emerged in areas of medical drugs, treatment methods, and medical equipment and involved with performing research and development activities globally. Even though the knowledge in medicine has been expanding with the time, the patients diagnose process remains as it is since the last couple of years. When diagnosing most mental diseases of patients, the experts would provide a key contribution to determine the exact disease [2]. Based on the gained experience, experts can diagnose mental disorder conditions precisely before proceeding to treatments.

The one major problem in this field is the comparatively low availability of those professional experts and scarce finding them from hospitals in rural areas. Therefore, it would be a challenging task to determine the exact mental disorder for psychiatrists who are newly appointed for the field. Furthermore, erroneously diagnose illnesses of the patient and proceed with treatments would detrimentally impact for the patient and eventually, the whole effort would be in vain. Hence, it is impelled to develop virtual expert system in purpose of assisting for the psychiatrist is key concern. Consequently, construct knowledgebase specific to the psychological diseases can be utilized to pinpoint the exact mental illness.

II. RELATED WORKS

There have been many types of researches and experiments were conducted to develop decision support systems as an attempt to enhance the productivity and efficacy during disease diagnose. Since healthcare domain is a vast area considering immense clinical terms it comprises, and many symptoms and diseases need to be identified, analyzed, and categorized before moving towards treatments. Mainly virtual experts use fed rules and factors to analyze the situation and make decisions but, the accuracy of the decision is the crucial factor to be considered. When analyze the existing expert systems their behavior, and implemented focus is categorically different.

A. MYCIN expert system

MYCIN was an early expert system that used artificial intelligence to identify severe infections, such as bacteremia and meningitis, and to recommend antibiotics, with the dosage adjusted for patient's body weight [3]. Furthermore, MYCIN operated using a simple inference engine, and the knowledge base embedded with rules [4]. MYCIN is one of the well-known programs that embody intelligence and provide data to behave in an intelligent manner. Compared to other artificial intelligent (AI) programs, it was slow in progress and not always in forwarding direction [3].

B. DXplain decision support system

Apart from that, DXplain is a decision support system that developed at the laboratory of computer science at the Massachusetts General Hospital, which has the characteristics of both electronic medical textbook and a

medical reference system [5]. DXplain can provide a list of clinical manifestations such as signs, symptoms, and laboratory examinations for approximately 2000 diseases [5]. Moreover, it provides justifications for each of these diseases and suggests what further clinical information should be collected pertaining to each disease.

C. CADUCEUS medical expert system

In addition to that, CADUCEUS was a medical expert system completed by Harry Pople in University of Pittsburgh in 1980, which took around ten years to build the knowledge base having ability to diagnose up to 1000 different diseases [6]. Both expert systems (CADUCEUS and MYCIN) were developed using the concept of inference engine and incorporate abductive reasoning to deal with additional complexity of internal diseases [6,7].

It is hard to find the expert systems which diagnose most mental illnesses and the above-mentioned expert systems (CADUCEUS, Mycin, DXplain) are not strong enough to recognize patient's symptoms and determine the disease exactly.

D. Mental Health Diagnostic Expert System (MeHDES)

This is AI rule-based reasoning (fuzzy based reasoning) system developed using techniques such as fuzzy logic and fuzzy genetic algorithm [8], [9], [10]. This fuzzy based reasoning is one approach to construct domain knowledge. It is extended version of semantic web rule and it does not create any inconsistencies for the knowledgebase. Nevertheless, describe the ontology in fuzzy knowledgebase has limited use and fuzzy inference is limited to rules only [11]. As well this MeHDES does not support question and answering approach and system did not act as a separate human counterpart to identify the situation in a conversational manner.

Therefore, the system needs to be more dynamic and it should be further enhanced to response dynamic questions. Hence it is obvious, to develop virtual expert system rule-based reasoning is not only sufficient, whereas understanding user's question and generate suitable answer is another essential part of it. Therefore, the proposed system fulfills the research goal by performing the role of the virtual human counterpart.

Furthermore, it uses the conversational approach to diagnose the patient and its behavior is highly dynamic when making the decision based on the patient's response.

III. Research Methodology

How this proposed system fulfills the key idea of the research goal is significantly considered during the system design stage. Designing smart virtual expert system is the main research problem and this can be decomposed into four substantial research questions. Refer the following Table 1 represents the defined approach that addresses each research question.

Table 1. Research based question map with defined approach

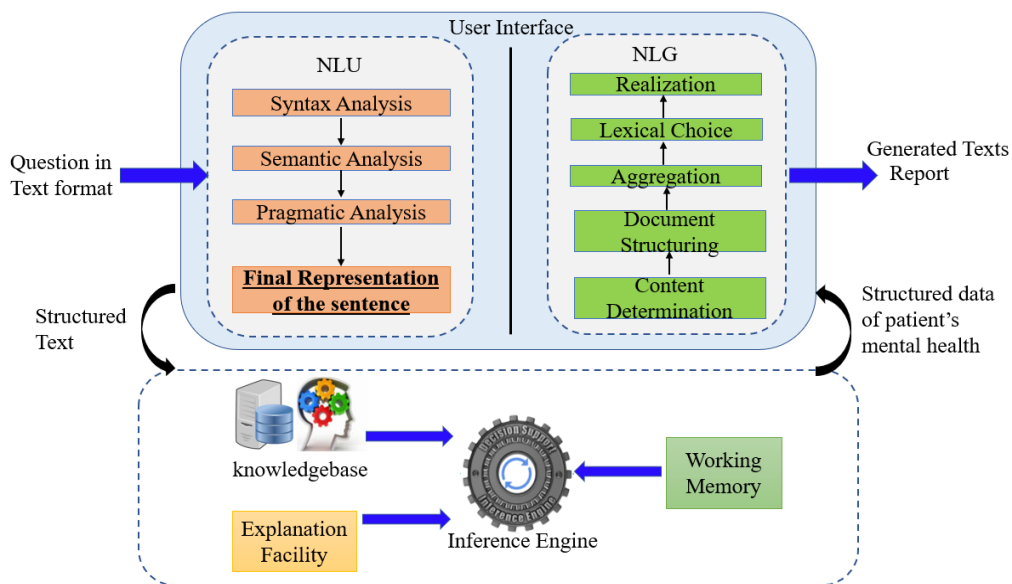| Research Question	Defined approach
1) How to construct the knowledge in expert?	a) Design Knowledgebase
2) How to diagnose patients?	b) Design question and answering approach to diagnose patients
3) How to query the knowledge in expert?	c) Design and implement SPARQL queries
4) How to generate the response?	d) Implement Natural Language Generation (NLG) Technique

Fig.1. High-level diagram of the system

According to the high-level diagram depicted in Fig. 1 diagnostic process can be described as follows.

1. The knowledge base comprises necessary knowledge to diagnose the illness, and this is acquired by a domain expert.

2. During this diagnose process, the proposed expert system asks questions from the patient and analyze

the patient's response using natural language understanding (NLU) technique.

3. This NLU technique analyzes those responses (language: English) and extracts keywords to query the knowledge base.

4. SPARQL queries are used to query the knowledgebase and extracts information based on the given keywords in step 3. Then find most probable disease and send those results to the response formation process.

5. Natural language generation (NLG) technique is used to form the response in human understandable manner.

The Fig. 2 represents interactions of each component in this proposed system.
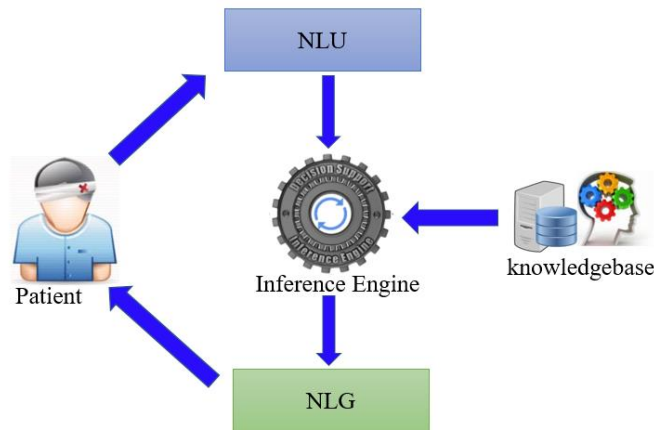

Fig. 2. Interactions of each component in SVESTAP

A. Knowledgebase Construction for SVESTAP

The knowledgebase is the most important part in this SVESTAP to diagnose illnesses. The knowledgebase is constructed by mapping domain experts' knowledge into an ontology [12]. For this, knowledge acquisition session is conducted with real human experts. Furthermore, to accomplish this knowledge acquisition process professional psychiatrist or panel of psychiatrists need to be interviewed through a questionnaire. This questionnaire should be meticulously planned to fully elicit the knowledge and experience of these experts. Especially, this diagnose process and judgment of diseases based on the symptoms, are performed during the discussion with patients. According to the symptoms of the patient, inference engine infers new knowledge using the embedded knowledge in the knowledge base.

Ontology comprises the logical structure of data in Extensible Markup Language (XML) format and all rules pertaining to that domain can be stored as Semantic Web Rule Language (SWRL) rules [13]. During this construction process classes, object/data properties, instances (individuals) and rules should be designed [14]. Standard books referred by psychiatrists (Eg. Diagnostic and Statistical Manual of Mental Disorder/ Mental State Examination) are used to map those details to Ontology. Fig. 5 illustrates, how classes are linked with properties to construct relationships in the knowledgebase. To implement the ontology Protégé IDE is used and Fig. 3 depicts class hierarchy related to three mental diseases. It can be visualized Onto Graph view as represented in Fig. 4.
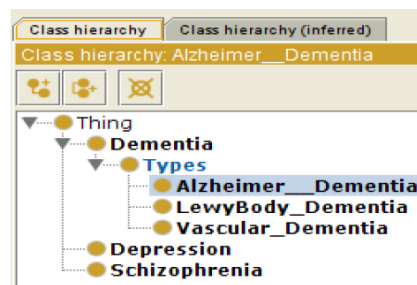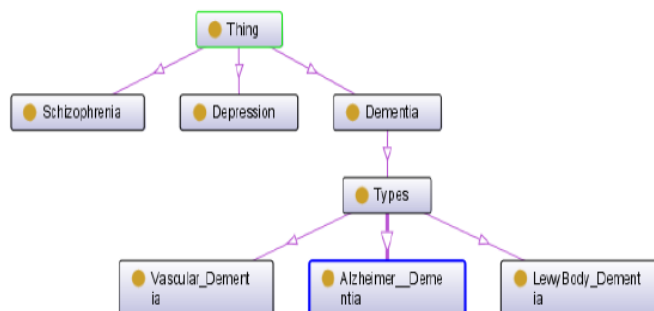

Fig.3. Class hiarachy in domain knowledge
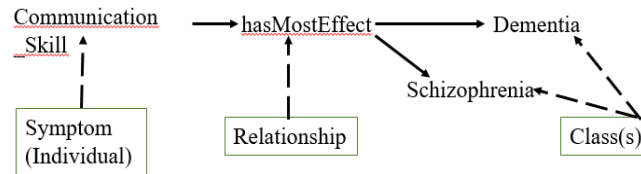

Fig.4. Class hiarachy in Onto Graph view

Fig.5. Ontology relationships among classes and properties

B. Understand the Query Using NLU Technique

NLU technique is used to understand the answer given by the user as depicted in Fig. 6. This can be further divided into sub-processes such as (Morphology, Syntax analysis, Semantic analysis, Pragmatics and Discourse analysis) [15, 16]
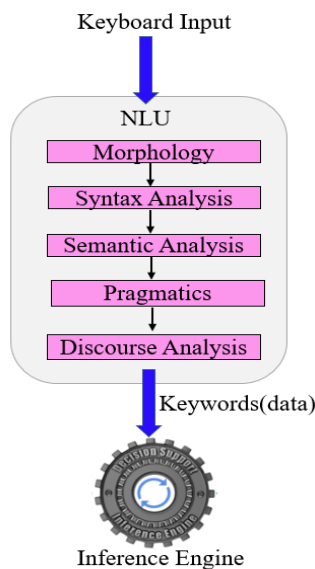

Fig.6. NLU Process

1. **Morphology**: Initially, the system breaks the long natural language sentence into sub-sentences to derive the meaning of it.
2. **Syntax analysis**: In this process, it realizes the relationships in between sentences through syntaxes. And mainly consider how words are put together to form the correct sentences and what structural role each word has.
3. **Semantics analysis**: In this step, consider how these words are combined to derive the meaning of the sentence. Hence, the system can highlight which keyword is put through the knowledgebase.
4. **Pragmatics:** Sometimes the sentence gives different meaning based on the context. As a result, in this step system considers how sentences are used in different situations and how it affects the interpretation of the sentence.
5. **Discourse:** In here, it considers how current sentence affects to the interpretation of the next sentence. Therefore, the system can keep track of the next sentence to match ordinary keywords which are coming from the user's response.

NLU sub processes perform the following tasks to elicit keywords from the sentence.

1. Split the sentence into words (Use the delimiter of space to split the sentence into words).
2. Identify the tag of the words (e.g. noun phrases, nouns, verb phrases, verbs, adjective, etc.).
3. Develop a method to identify keywords of the sentence and negation of the sentence.
4. Develop a method to identify the synonyms of keyword related to the answer (when keywords are not matched with the given context).
5. Develop a method to check whether any spellings or grammatical mistakes in the sentence.
6. Send extracted keywords of the sentence to knowledgebase querying process.

C. Query the Knowledgebase and Diagnosing

Keywords are extracted from above section B stage (Understand the user's question using NLU technique) and it is used to query the constructed domain knowledge to derive the correct disease. Query answering is important in semantic web and several query languages were designed for this purpose such as Resource Description Framework Data Query Language (RDQL), Second generation Resource Description Framework Query Language (SeRQL), and most recently used SPARQL [17]. SPARQL queries can be used across diverse data sources, to verify whether the data is stored natively as RDF or viewed as RDF via middleware [18]. SPARQL contains capabilities for querying required and optional graph patterns along with their conjunctions and disjunctions. The output of SPARQL queries can be results-sets or RDF graphs [18, 19].

E.g. Data in turtle format can be represented as below.

```
@prefix svestap: <http://xmlns.com/svestap /0.1/> .
_:a svestap:symptom "confusion and disorientation" .
_:a svestap:symptom "memory loss" .
_:b svestap:disease "Dimentia" .
_:b svestap:symptom "palpitation" .
_:c svestap:symptom "sweating".
_:b svestap:disease "Anxiety" .
```

Use following query to extract data

```
PREFIX svestap: <http://xmlns.com/svestap/0.1/>
SELECT ? disease ?symptom
WHERE
{
        ?x svestap:disease ?disease
        ?x svestap:symptom ?symptom
}
```

SPARQL uses "?" to define variables and SELECT clause define returning variables as "?symptom" and "?disease" and WHERE clause define matching criteria. Fig. 7 illustrates sample query that retrieves disease by giving the set of symptoms. In here, there are four results sets fulfilling above criteria and following datasets are given as result set in Table 2.

Table 2. SPARQL Results set of matching criteria

disease	symptom
"Dimentia"	confusion and disorientation
"Dimentia"	memory loss
"Anxiety"	palpitation
"Anxiety"	sweating

```
"PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>\n" +
"PREFIX owl: <http://www.w3.org/2002/07/owl#>\n" +
"PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>\n" +
"PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>\n" +
"PREFIX s: <http://www.semanticweb.org/sachintha/ontologies/2016/4/u
"\n" +
"\n" +
"SELECT DISTINCT (STRAFTER(str(?b),\"#\") AS ?new) WHERE {\n" +
    "{\n" +
        "SELECT  ?b  WHERE { s:"+symptom1+" rdf:type  ?z .\n" +
    "?z ?a ?b.} \n" +
"}\n" +
"\n" +
    "{\n" +
        "SELECT  ?b  WHERE { s:"+symptom2+" rdf:type  ?z .\n" +
    "?z ?a ?b.} \n" +
"}\n" +
"\n" +
    "{\n" +
    "SELECT  ?b  WHERE { s:"+symptom3+" rdf:type  ?z .\n" +
    "?z ?a ?b.} \n" +
    "}\n" +
"\n" +
    "{\n" +
    "SELECT  ?b  WHERE { s:"+symptom4+" rdf:type  ?z .\n" +
    "?z ?a ?b.} \n" +
    "}\n" +
"} ORDER BY (?b)";
```

Fig.7. SPARQL query to select disease from symptoms

During the conversation, the proposed expert system asks few questions from the patient and analyze patient's responses to extract symptoms. From the NLU technique, symptoms were extracted and proceed to the next step of identifying disease category those symptoms belong to. Identify the exact disease based on the given symptoms is another key concern. This goal is accomplished in two stages as follows.

1. First execute SPARQL queries separately for each symptom and fetch all probable diseases for given symptom and categorize them as depicted in Fig. 8.
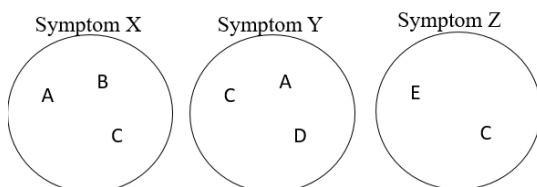
2. Then find intersection of disease sets to find most probable disease. Refer the Fig. 9 for interaction of diseases.

Symptom X ∩ Symptom Y = {Disease C},
Symptom X ∩ Symptom Z = {Disease C, Disease E},
Symptom Y ∩ Symptom Z = {Disease C},
Symptom X ∩ Symptom Y ∩ Symptom Z = {Disease C}.

Fig.9. Interaction of Diseases

Fig.8. Sets of Diseases

E.g. Diseases ∈ {A, B, C, D} and Symptoms ∈ {X, Y, Z}, then diseases for symptoms are as follows,
Symptom X ∈ {Disease A, Disease B, Disease C},
Symptom Y ∈ {Disease C, Disease A, Disease D},
Symptom Z ∈ {Disease E, Disease C}.
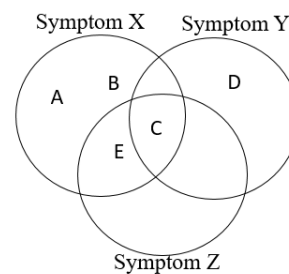
If there is no disease which has all given symptoms, then proposed system get the disease with maximum number of matching diseases. Then the result will be passed to the NLG process to generate the response familiar to user.

D. Response Formation Using NLG Technique

Response formation is the final stage in the proposed system, after querying the knowledgebase (using SPARQL queries) the new inferred knowledge is given as set of keywords. To comprehend the final results, these keywords must be organized in a human-readable manner [20]. As depicted in Fig. 10, NLG process achieves this by converting keywords into human readable natural language sentences.
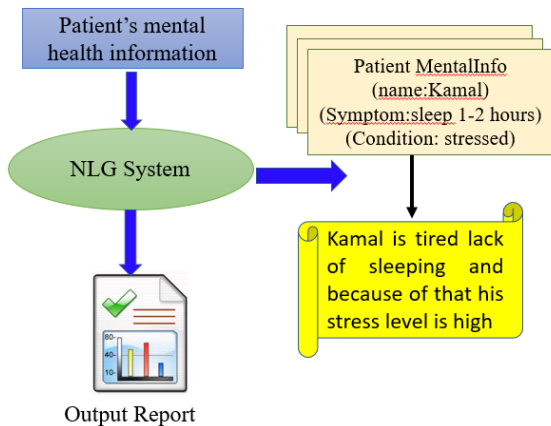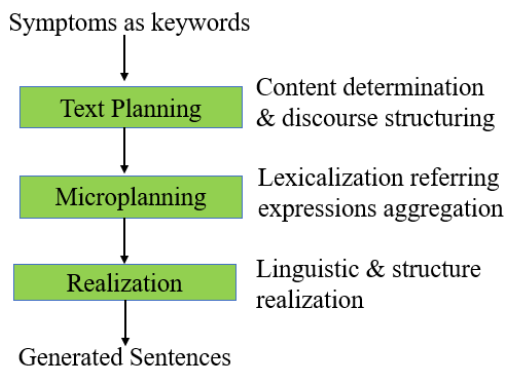

Fig.10. NLG Overview

Fig.11. Subprocess of NLG

NLG technique is implemented using Java language and separate library called "SimpleNLG" needs to be imported. This comprises inbuilt functions to accomplish the tasks depicted in Fig. 11 (Text planning, microplanning and realization) [22]. Sentence generation can be summarized as per the below three steps:

1. Develop a method to retrieve symptom keywords from the knowledgebase.
2. Develop a method to include required words and form meaningful phrases.
3. Develop a method to generate full sentence based on small sentence-phrases using previously mentioned method.

To generate symptoms in question format for the user, separate xml document has been used to refer them as lexicons. The words in this xml document contains nouns, verbs, and complements (adjectives) refer Fig. 12. Then

keys are extracted from xml file using Xpath statements and "SimpleNLG" technique assists you to write a program which generates grammatically correct English sentences as illustrated in Fig. 13.

```xml
<root>
        <!-- Question1 -->
  - <Question>
        <keyWord>Memory</keyWord>
        <noun>Your Memory</noun>
        <verb>be</verb>
        <compliment1>good</compliment1>
        <compliment2>average</compliment2>
        <compliment3>poor</compliment3>
        <adjective>for recent activites</adjective>
    </Question>
        <!-- Question2 -->
  - <Question>
        <keyWord>awake</keyWord>
        <noun>You</noun>
        <verb>awake</verb>
        <compliment1>sometimes</compliment1>
        <adjective>your family members at midnight</adjective>
    </Question>
        <!-- Question3 -->
  - <Question>
        <keyWord>BP</keyWord>
        <noun>Your Blood Pressure</noun>
        <verb>be</verb>
        <compliment1>low</compliment1>
        <compliment2>normal</compliment2>
        <compliment3>high</compliment3>
    </Question>
        <!-- Question4 -->
  - <Question>
        <keyWord>VH</keyWord>
        <noun>You</noun>
        <verb>see</verb>
        <compliment1>rarely</compliment1>
        <compliment2>sometimes</compliment2>
        <compliment3>always</compliment3>
        <adjective>visual hallucinations</adjective>
    </Question>
```

Fig.12. Lexicon keywords xml

```java
public String genSentence(String noun, String verb, Stri

    Lexicon lexicon = Lexicon.getDefaultLexicon();
    NLGFactory nlgFactory = new NLGFactory(lexicon);
    Realiser realiser = new Realiser(lexicon);

    SPhraseSpec p1 = nlgFactory.createClause();
    p1.setSubject(noun);
    p1.setVerb(verb);
    p1.setComplement(com);

    if (tense == 1 && negative == 1)
        p1.setFeature(Feature.NEGATED, Tense.PAST);
    else if(tense == 1 && negative == 0)
        p1.setFeature(Feature.TENSE, Tense.PAST);
    else if(tense == 0 && negative == 1)
        p1.setFeature(Feature.NEGATED, Tense.PRESENT);
    else
        p1.setFeature(Feature.TENSE, Tense.PRESENT);

    String output = realiser.realiseSentence(p1);
    return output;
```

Fig.13. Sentence Generation

After diagnosing the patient, the proposed system conducts a separate patient evaluation process to determine the intensity of the detected disease. This evaluation is performed by giving a Hypomania Checklist (HCL-32) standard questionnaire [21]. The intensity of the disease can be determined based on the

responses given for the questionnaire. In here, system acts as a separate human-counterpart and provides responses for the user's questions.

IV. RESULTS AND OUTCOME

This proposed system gives ten questions for the patient to answer and patient can answer them by using natural language sentences as shown in Fig. 14. The given response will be processed by the system and it elicits all symptoms from the response of patient to diagnose the disease. Then based on the given symptoms, it identifies the exact disease that patient is suffering from and formulate it as the response. The Fig. 15 represents, how it generates the response by inferring new knowledge.

1) Question: How many hours do you sleep on a day?
    1.   Answer: I sleep around about 6 hours per day
2) Question: How about your memory about pass week or so?
    2.   Answer: I remember something
3) Question: How do you feel these days?
    3.   Answer: I feel good
4) Question: What about your communication skills
    4.   Answer: Good
5) Question: How old are you?
    5.   Answer: I am 34 years old
6) Question: Do you awake the members of your family at midnight
    6.   Answer: Yes I do
7) Question: How about your blood pressure
    7.   Answer: It is normal
8) Question: Do you see visual hallucination in sometimes?
    8.   Answer: I see rarely
9) Question: Do you hear some unusual voices?
    9.   Answer: I here always
10) Question: How about your energy?
    10.   Answer: I feel weak

Fig.14. Sample Q&A process with SVESTAP and Patient
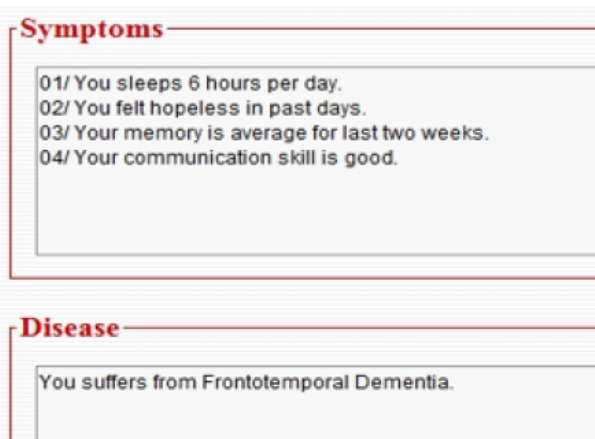

Fig.15. Infer the disease based on symptoms

SVESTAP system was evaluated considering the factors such as understanding of the application, features included, usage of application, and willingness to use the application. From this evaluation positive and negative feedbacks were taken as illustrated in Table 3.

Table 3. User feedback summary

	Positive	Negative
Understanding of the application to use	80%	20%
Features included	85%	15%
Usage of the application	90%	10%
Willingness to use the application	95%	5%

According to the chart depicted in Fig. 16 more than 80% of feedbacks were postive regarding the system usability and willingness to use the application is comparatively high.
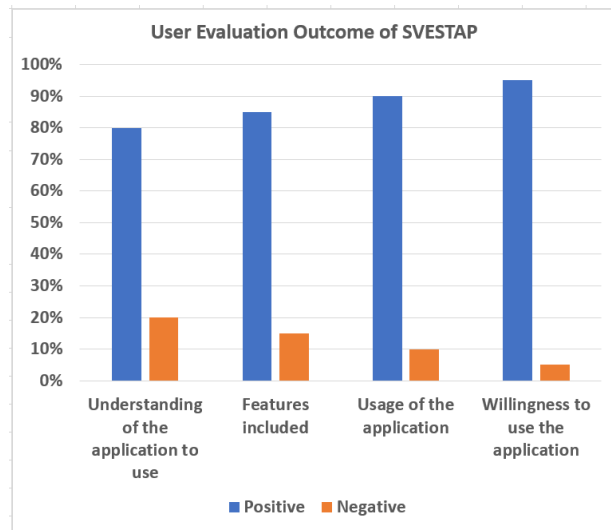

Fig.16. User evaluation outcome of SVESTAP application

V. SUMMARY/CONCLUSION

Nowadays the availability of experts to provide services symultaneously for multiple users would be a challenging task. To fullfill this problem, experimental analysis are conducted in many fields in developing similar virtual expert systems. Development of virtual experts would be an ideal solution but, validity of the made decision is questionable before applying them to real environment. Specially in medical domain, accuracy is the most decisive factor to be considered. As a result, development of expert system to assist real human expert would reduce upcoming risk in the medical domain. As the initial step, I would like to suggest a system that works with a real human expert, which maximizes accuracy factor of the made decision. Therefore, the decisions of disease judgement should always be aligned with actual psychiastrist and the end system response should be frequently evaluated. This system can be utilized for apprentices in the medical field to validate made disease judgement and to identify the root cause before proceeding for treatment.

REFERENCES

[1] H. P. Selker and A. J. J. Wood, "Industry Influence on Comparative-Effectiveness Research Funded through Health Care Reform," *New England Journal of Medicine*,

vol. 361, pp. 2595-2597, Dec 2009. doi: 10.1056/NEJMp0910747.

[2] A. Demirhan, "Pattern analysis for the neuroimaging based diagnosis of schizophrenia," *2017 25th Signal Processing and Communications Applications Conference (SIU)*, Antalya, Turkey, 2017, pp. 1-4.

[3] D. E. Heckerman, E. H. Shortliffe, "From certainty factors to belief networks," *Artificial Intelligentce in Medicine,* vol. 4, no. 1, pp. 35-52, Feb 1992.

[4] C. WenBin, L. XiaoLing, L. YiJun and F. Yu, "A machine learning algorithm for expert system based on MYCIN model," *2010 2nd International Conference on Computer Engineering and Technology*, Chengdu, 2010, pp. V2-262-V2-265.

[5] G. Barnett, J.J. Cimino, J.A. Hupp and E. P. Hoffer, "Dxplain: An evolving diagnostic decision-support system," *Journal of the American Medical Association (JAMA)*, vol. 258, no. 1, pp. 67-74 1987.

[6] O. J. Ayangbekun, A. I. Olatunde and F. O. Bankole, "An Expert System for Diagnosis of Blood Disorder," *International Journal of Computer Applications (IJCA)*, vol. 100, no. 7, pp. 37-40, Aug 2014.

[7] C. V. D. Schatz and F. K. Schneider, "Intelligent and Expert Systems in Medicine – A Review," *XVIII Congreso Argentino de Bioingenier ú SABI 2011 - VII Jornadas de Ingenier ú Cl ńica Mar del PlataSara*, pp. 28 – 30, September, 2011, pp. 326-331.

[8] A. T. Owoseni, I. O. Ogundahunsi,"Mobile-Based Fuzzy Expert System for Diagnosing Malaria (MFES)", *International Journal of Information Engineering and Electronic Business(IJIEEB)*, Vol.8, No.2, pp.14-22, 2016. DOI: 10.5815/ijieeb.2016.02.02.

[9] V. Jain, S. Raheja, "Improving the Prediction Rate of Diabetes using Fuzzy Expert System", *International Journal of Information Technology and Computer Science (IJITCS)*, no. 10, pp. 84 - 91, 2015. DOI: 10.5815 / ijitcs. 2015.10.10.

[10] T. W. Wlodarczyk, M.O'Connor, C. Rong, M. Musen, "SWRL-F - A Fuzzy Logic Extension of the Semantic Web Rule Language"Available:http://c4i.gmu.edu/ursw/2010/papers/URSW2010_P1_WlodarczykEtAl.pdf.

[11] T. W. Wlodarczyk, M. O'Connor, C. Rong, M. Musen, "SWRL-F - A Fuzzy Logic Extension of the Semantic Web Rule Language" Available: http://c4i.gmu.edu/ursw/2010/papers/URSW2010_P1_WlodarczykEtAl.pdf.

[12] K. Balachandran and S. Ranathunga, "Domain-Specific Term Extraction for Concept Identification in Ontology Construction," *2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, Omaha, NE, 2016, pp. 34-41.

[13] B. Jafarpour, S. R. Abidi and S. S. R. Abidi, "Exploiting Semantic Web Technologies to Develop OWL-Based Clinical Practice Guideline Execution Engines," in *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 1, pp. 388-398, Jan. 2016. doi: 10.1109/JBHI.2014.2383840

[14] S. Sitthithanasakul and N. Choosri, "Application of software requirement engineering for ontology construction," *2017 International Conference on Digital Arts, Media and Technology (ICDAMT)*, Chiang Mai, 2017, pp. 447-453.

[15] M. Bates, "Models of natural language understanding," *Proceedings of the National Academy of Sciences*, vol. 92, pp. 9977-9982, Oct 1995.

[16] W. G. Lehnert, M. H. Ringle, "Strategies for natural language processing," New York, Psychology Press, 2014.

[17] S. Im, M. Sohn, S. Jeong and H. J. Lee, "Keyword-Based SPARQL Query Generation System to Improve Semantic Tractability on LOD Cloud," *2014 Eighth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, Birmingham, 2014, pp. 102-109. doi: 10.1109/IMIS.2014.95

[18] I. Abdelaziz; M. R. Al-Harbi; S. Salihoglu; P. Kalnis, "Combining Vertex-centric Graph Processing with SPARQL for Large-scale RDF Data Analytics," in *IEEE Transactions on Parallel and Distributed Systems*, vol. PP, no. 99, pp. 1-1.

[19] A.Merazi, M. Malki, "SQUIREL: Semantic Querying Interlinked OWLS traveling Process Models", *International Journal of Information Technology and Computer Science (IJITCS)*, No. 12, pp. 30 – 39, 2015. DOI: 10.5815 / ijitcs. 2015.12.04.

[20] S. Puls, D. Lemcke and H. Worn, "Context-sensitive natural language generation for human readable event logs based on situation awareness in human-robot cooperation," *2014 11th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, Kuala Lumpur, 2014, pp. 350-355.

[21] J. Angst et al, "The HCL-32: Towards a self-assessment tool for hypomanic symptoms in outpatients," *Journal of Affective Disorders,* vol. 88, no. 2, pp. 217-233, Oct 2005.

[22] N. Dethlefs, "Domain Transfer for Deep Natural Language Generation from Abstract Meaning Representations," in *IEEE Computational Intelligence Magazine*, vol. 12, no. 3, pp. 18-28, Aug. 2017. doi: 10.1109/MCI.2017.2708558.

Authors' Profiles

Udara Srimath S. Samaratunge Arachchillage received the M.Sc. degree in Enterprise Application Development from Sheffield Hallam University (SHU) in Sheffield, UK and won the best performer award in 2014. He received his B.Sc. Special honour degree in field of Information Technology (IT) from Sri Lanka Institute of Information Technology (SLIIT) in Sri Lanka in 2010 and B.Sc. degree in IT from Curtin University of Technology, Bentley, Perth, Western Australia in 2009.

He is a Researcher, Lecturer and Software Engineer and currently works as Research Engineer in Department of Software Engineering, Institute of Cybernetics, National Research Tomsk Polytechnic University, Russia and Lecturer in Faculty of Computing, Department of Software Engineering, Sri Lanka Institute of Information Technology (SLIIT) in Sri Lanka. His research interests are Semantic Web, Expert System, Knowledgebase construction, Natural Language Processing, Software Engineering, Human Computer Interaction, Wireless Communication.

He is a member of Computing Society of Sri Lanka (CSSL) and manuscript reviewer in IEEE.