IMPACT: International Journal of Research in
Engineering & Technology (IMPACT: IJRET)
ISSN (E): 2321-8843; ISSN (P): 2347-4599
Vol. 4, Issue 5, May 2016, 1-12

EVALUATION AND VALIDATION OF ONTOLOGY USING PROTEGE TOOL

VISHAL JAIN[1] & S. V. A. V. PRASAD[2]

[1]Research Scholar, Computer Science and Engineering Department, Lingaya's University, Faridabad, India

[2]Professor, Electronics and Communication Department, Lingaya's University, Faridabad, India

ABSTRACT

Ontology is one of the most important constituent in semantic web layered architecture. Without ontology, it is impossible to maintain relationships among real world entities. Various operations can be performed on ontology like merging, mapping, evaluation and validation of ontologies. The paper classifies ontologies at various levels like lexical level, hierarchical level, syntactic level and design level. Besides this, a comparative study is also provided on different methods for evaluation and validation of ontologies like Prompt, OntoMetric, OntoClean and many more.

KEYWORDS: Semantic Web, Ontology, Evaluation, Protégé, Prompt

INTRODUCTION

The term semantic web was coined by Sir Tim Berners Lee in 1965. It came into existence with aim to abridge gap between humans and machines. Its architecture includes ontology that can be said as spine of semantic web. Ontologies are data models that represent meaning of semantics in expressive way. Ontologies are used to maintain relationship among real world entities belonging to particular domain. It has various definitions like philosophical, formal, explicit, specific, shared and many more. There are notions to express ontologies called as Ontology languages like RDF, OWL. They have predefined syntaxes and logic based semantics that performs reasoning and manipulations with the help of ontologies. Semantic web is treated as third generation of web (Web 3.0) that focuses on generation of metadata and its annotations are filled in machine understandable form. Difference between current web and semantic web can be illustrated by taking an example of library. An old library with full of books without catalogue is treated as current web while modern computerized library with catalogues is semantic web. Obviously, modern librarians work faster because they have to search catalogues directly rather than searching whole books. In catalogue, results are retrieved/found on basis of author's name, publisher's name, ISBN number etc. The most important feature in modern library is that record field values are ordered and their values are interpreted in international standards like MARC format. It uses vocabularies in form of concept hierarchy like Dewey Decimal Classification System (DDC) or Universal Decimal Classification System (UDC) . These standards are very vital for dissemination of information in libraries. Similarly in web, these standards are used as ontologies to capture values of records. Then, these values acts as metadata to maintain interoperability between standards. If there are multiple standards, then they are to be mapped first before sharing of information. Semantic web technologies like XML, RDF aims to create ontologies and metadata either from scratch or from existing ontologies. New ontology can be created from existing ontology by performing various ontology evaluation approaches like PROMPT, OntoMetric etc. Thus, it is concluded that semantic web is an application for generation of metadata and enhances the results of current web with the help of ontology. There are various problems associated with the development of semantic web. According to Kevin Kelly, it suffers from fax effect which means that development of semantic web is costly and its technologies have not been utilized fully. But, still most of researchers are trying their hands on this web technology to achieve machine human interaction.

Continuous efforts are being laid down by researchers in order to make information systems as intelligent systems that encompass human interaction with machines. It has led to focus towards semantic concepts of data that holds interpretation and relationship with other concepts . In recent years, various studies have been conducted by scientists, researchers in order to allow semantic web technologies to work in distributed environment and enables knowledge sharing of information in machine understandable format. It is necessary to validate and evaluate ontologies while building them because ontology building is a task that requires working from scratch of project. Ontology Evaluation is one of the key techniques for the area of semantic web. Ontology can be evaluated for a particular domain in which it exists but an independent evaluation of ontology is still hard to solve. They can be solved on basis of 3 levels

- Scope of ontology
- Taxonomy view (whole view, Isahierarchy)
- Adaptability of semantic web relations

Semantic Web

This section let readers think about few questions like Why would current web need any extension? Why there are irrelevant results produced while on current web? The reason that is common to both questions is lack of knowledge gap between user and machines. Current web does not offer mechanism to provide deeper understanding of information. Various knowledge management solutions and technologies are there in field of AI to deal with this while missing information can be accessed with the help of ontologies. Ontologies can be social as well as formal. They are formal in such a way that they maintain human-machine interaction to enable knowledge reasoning while social confines maintaining relationships between classes and properties of other ontologies. Semantic web aims to transform web documents to information. Meaningful data is called as information. It involves creation of common framework that leads to sharing of data and its reuse among various applications. Application of semantic technologies covers areas like data integration, knowledge discovery, and resource discovery, classification of data and designing of intelligent systems.

Figure 1: "Evaluation of Semantic Web"

Ontology

Ontology is treated as formal, explicit specification of shared conceptualization . Besides its formal nature, philosophical aspects, handling real world scenarios, it also acts as medium of linking between human and machines. Ontology in itself is a vast research area that includes mapping, merging, extraction, moving and evaluation of ontologies.

Ontology evaluation approaches are classified in following categories.

- On basis of comparing ontologies
- On basis of usage and application of ontologies
- On basis of set of documents related to domain ontology
- On basis of human evaluation in order to meet ontology requirements and compatibilities.

Components of ontology include classes, properties, instances, inheritance functions, slots, frame values and sub classes. Relationship between classes and subclasses is defined by super concept-sub-concept and is defined as isahierarchy. Example: There is class named Institute with its sub classes IIT, IIIT, NIT's. So, it is represented as IIT is a subclass of Institute. There are social ontologies that help to achieve interoperability among social web applications in order to move from social web to semantic web. Some of them include FOAF, SIOC, XFN, GoodRelations, RSS Feeds and many more.

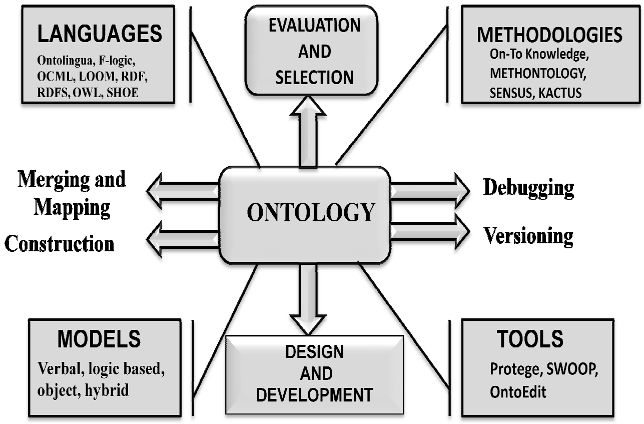

Figure 2: "Ontology and its Constituents"

CLASSIFICATION OF ONTOLOGY EVALUATION APPROACHES

Ontology evaluation approaches are classified in following categories.

- On basis of comparing ontologies
- On basis of usage and application of ontologies
- On basis of set of documents related to domain ontology
- On basis of human evaluation in order to meet ontology requirements and compatibilities.

CURRENT APPROACHES IN ONTOLOGY EVALUATION AND VALIDATION

A craze towards learning of semantic web and its technologies has led to development of huge number of ontologies that needs to be evaluated and validated based on following approaches are stated below:

Evolution Based

As the word suggests evolution specifies evolving from time to time. It is known that ontologies vary from time to time that leads to enhancement of knowledge and more precise results. This method tracks changes and improvements done in existing ontology when subjected to different versions. As per Noy, evolution in ontology is being caused due to three features mainly-variable domain, changes in specification and conceptualization.

Logical (Rule-based)

This approach makes use of rules for deducing inferences in given ontology. When object properties of two classes are different from each other, they are represented as (owl: different from) or disjoint from each other as (owl: disjoint with).

The rule for corresponding query follows some syntax. If query is "My aunt is my father's sister", then represent it with rule in corresponding family ontology. Rule is written as: Rule1_//_MyAunt_is_my_father's_sister.

Metric-Based (Feature-based)

This approach provides quantitative scenario to given ontology as it tracks variations of classes and properties of source ontology and target ontology. It also performs operations like union, intersection between two schemas of ontology that leads to distributed percentage of instances of given schema.

METHODS FOR EVALUATING ONTOLOGIES

OntoMetric

Why OntoMetric?

Choosing ontology for new project among various domains is one of the major problems that are being faced by knowledge engineers. Till now, ontologies are being chosen by little experience of researchers but it should be chosen by taking its schema into consideration.

What is OntoMetric?

This method consists of set of processes that user must select in order to determine compatibility and selection of ontologies. It is used for selection of optimal ontology among various domains and making it compatible as per standards of given project. OntoMetric is based on Analytic Hierarchy Process (AHP) that also provides methods for reusing of ontologies. AHP considers dimensions that need to be checked before using ontology. Following are the features of AHP:

- Content of ontology
- Ontology implementation language
- Steps required for building ontology

- Platform used
- Const incurred in building ontology

So, it is concluded that OntoMetric acts as quantitative measure for every candidate ontology by using dimensions.

OntoClean

Why OntoClean?

Finding meta-relations among concepts is not much easier task. It requires cleaning of ontologies.

What is OntoClean?

Following are properties of OntoClean:

- Rigidity- It defines the links between property and individuals. A property is said to be rigid iff it is vital for all its mentioned instances. A property is non rigid if it is not vital for some of its instances and it is anti-rigid iff it is not vital to all its instances.
- Unity- This property specifies that parts of schema are unified if they are found by joining instances to common relation R as represented by $\mathbf{< R \text{ € } I^1 \vee I^2 \vee \ldots I^n >}$

There are two building blocks that play vital role in implementation of OntoClean:

- Set of axioms that specifies given requirements and constraints of given ontology.
- A Meta ontology that is also called as Schema hierarchy. It includes object properties and data type properties.

OntoQA

It is implemented in form of java application that employs Sesame (open source framework). It acts as RDF repository in which various OntoQA components are used as:

- Ontology- It finds metric values of ontology. Ontology schema holds following elements viz. Classes (C), Properties (P), Instances (I) and Inheritance Functions (HC). Knowledge base of ontology holds following elements viz. Instances (I), Class Instantiation Function (CF) and Relationship Instantiation Function (RI)
- Ontology and Keywords- It uses Word Net to find synonyms of given terms used in given ontology. It also uses above calculated metric values to determine overall quality value of ontology.
- Keywords- OntoQA makes use of Swoogle-a crawler based meta search engine that finds RDF and OWL documents in context of entered keywords.

Prompt

It is one of plug-in and acts as tool for comparing ontologies. It is partial not complete algorithm for representing ontologies. With the help of this plug-in, user can perform various functions on given ontology like comparison, merging, mapping, extract features from source ontology and move it to target ontology. It holds various features that are used to provide suggestions and reduces conflicts between ontologies. These features cover:

- Classes and slots used for merging
- Hierarchy of both schemas (PROMPT will give better suggestion if two classes are similar because they are easier to merge)
- Attachment of slots with respect to classes
- Facets and their values (It is required to restrict range of classes while merging their slots)

Besides this, PROMPT also helps to identify conflicts that are among the following-

- Naming conflicts
- Null references
- Redundant classes

Working of PROMPT Algorithm:

Input- Source ontology and target ontology

Steps are:

- List of common classes is being created and matched
- Operation is performed by user on basis of PROMPT's suggestions
- PROMPT performs operations automatically and lists extra changes to do with ontology.
- Generates suggestion list on basis of structure of ontology
- Determine conflicts occurred while merging both ontologies and provide its solutions.

In our Protégé-based implementation, we use Protégé component-based architecture to allow the user to plug in any term-matching algorithm.

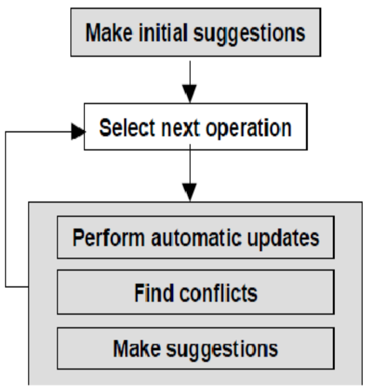

Figure 3: The Flow of Prompt Algorithm

The gray boxes show the action performed by the Prompt tool. And the white boxes show the actions performed by the user.

Modes of PROMPT Tool

The PROMPT tab allows you to manage multiple ontologies in Protégé-2000. Using PROMPT you can:

- Merge two ontologies into one;
- Extract a part of an ontology;
- Move frames from included to including project;
- Compare two versions of the same ontology and create a merged version

Except for move frames, these operations create a copy of the ontology in your working project and leave the original project intact. In the case of move frames, however, both the included and including projects are changed.

Merge Mode

It lets users to merge two existing ontologies and develops new ontology from them. The original ontologies are left untouched. In merge mode, the Prompt Tab has three sub tabs:

- The Suggestions Tab
- The Conflicts Tab
- The New Operations Tab

With the help of these tabs, merging and mapping processes can be done effectively.

Extract Mode

It lets users to extract/retrieve some subset features of knowledge base that contributes to part of source ontology. Retrieved ontology can either be saved as new .owl file or moved in existing project but source ontology left unchanged. In extract mode, the Prompt Tab has two sub tabs:

- The Suggestions Tab
- The New Operations Tab

The Conflicts tab does not appear in extract mode.

Via these tabs, PROMPT guides you through the extract process, making suggestions based on the frames you have already copied.

Moving Frames Mode

Moving frames mode allows us to move frames from an included to an including project. This is the only mode that alters the original ontology as well as the target ontology. It is a good idea to make copies of both the including and included ontologies before you enable moving frames mode.

There is inclusion mechanism in Protégé that led to reusing of ontologies and their frames from existing project. This mode allows moving of frames of existing ontology to included project.

In moving frames mode, the Prompt Tab has two sub tabs:

- The Conflicts Tab
- The New Operations Tab

The Suggestions tab does not appear in moving frames mode.

Via these tabs, PROMPT guides you through the moving frames process, identifying conflicts and proposing conflict-resolution strategies.

Compare Mode

This mode is only used for comparing two ontologies by finding frame changes in them. Like merge, compare uses a number of heuristics to make a best guess as to changes and correspondences. In this case, the heuristics behind the standard PROMPT merge can be modified. When the two projects to be merged are known to be different versions of the same ontology, Protégé can use stronger heuristics, suggesting, for example, that a single unmatched sibling of the same parent in the two different versions may be the same frame with a different name. By concentrating on the differences between two similar projects (rather than the similarities of two different projects as in merge), compare can give much better results for the type of reconciliation required in version control.

PROMPT Operations

PROMPT is used to copy, merge, move and extract information from source ontologies to target ontologies and vice versa. Except move operation, source ontology remains unchanged because in move operation, source ontology classes have been added into current project.

The operations available depend on our initial choice of how the mode for incorporating our source ontologies into the working project. The available operations for each type of PROMPT action are as follows:

Table 2: Modes of Operations in PROMPT

Mode	Available Operations
Merge Mode	Merge Classes, Merge Slots, Merge Instances, Copy Class, Copy Slot, Copy Instance, Remove Parent
Extract Mode	Copy Class, Copy Instance, Copy Slot
Moving Frames Mode	Move Class, Move All Instances Of Class, Move Instance, Move Slot
Compare Mode	*View Only; No Operations Available*

CONCLUSIONS & FUTURE SCOPE

Different kinds of mismatches that could happen in ontology integration and sketched the current solutions to reconcile different mismatches have been discussed in this paper and also argued that mappings are crucial components for many applications. Many works on ontology mapping have been done in the context of a particular application domain.

ACKNOWLEDGEMENTS

I, Vishal Jain, would to give my sincere thanks to Dr. M. N. Hoda, Director, Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi, for giving me opportunity to do Ph.D from Linagay's University, Faridabad.

REFERENCES

1. Nuno Silva and João Rocha : Ontology Mapping For Interoperability In Semantic Web. GECAD - Knowledge Engineering and Decision Support Research Group.
2. T. Berners-Lee. The semantic web. Scientific american, 284(5):35–43, 2001.
3. A.Valente, T. Russ, R. MacGrecor, and W. Swartout. Building and (re)using an ontology for air campaign planning. IEEE Intelligent Systems, 14(1):27–36, 1999.
4. T. R. Gruber. Translation approach to portable ontology specification Knowledge Acquisition, 5(2):199–220, 1993.
5. J. Heflin, J. Hendler, and S. Luke. Coping with changing ontologies in a distributed environment. In Proceeding of the AAAI workshop on ontology management, 1999.
6. N. F. Noy and M. A. Musen. Prompt: algorithm and tool for automated ontology merging and alignment. In Proceeding of Seventeenth National Conference on Artificial Intelligence (AAAI-2000), 2000.
7. Natalya F. Noy and Deborah L. McGuinness Stanford University, Stanford, CA, 94305 Ontology Development 101: A Guide to Creating Your First Ontology
8. H. Chalupsky. Ontomorph: A translation system for symbolic logic. In A. G. Cohn, F. Giunchiglia, and B. Selman, editors, KR2000: Principles of Knowledge Representation and Reasoning, pages 471–482, San Francisco, CA, 2000. Morgan Kaufmann.
9. H. Chalupsky. A translation system for symbolic knowledge. In Proceedings of the 7th International Conference on Principles of Knowledge Representation and Reasoning, 2000.
10. M. Klein. Combining and relating ontologies: an analysis of problems and solutions. In Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI-01), Workshop: Ontologies and Information Sharing, Seattle, USA, 2001.
11. P. Visser, D. M. Jones, T. Bench-Capon, , and M. Shave. An analysis of ontological mismatches: Heterogeneity versus interoperability. In AAAI 1997 Spring Symposium on Ontological Engineering, Stanford, USA, 1997.
12. P. Visser, D. M. Jones, T. Bench-Capon, and M. Shave. Assessing heterogeneity by classifying ontology mismatches. In Proceedings of the International Conference on Formal Ontology in Information Systems (FOIS98), Trento, Italy, 1998.
13. G. Wiederhold. An algebra for ontology composition. In Proceedings of 1994 Monterey Workshop on Formal Methods, pages 56–61, CA, USA, 1994.

14. P. Visser and T. Bench-Capon. On the reusablity of ontologies in knowledge-system design. In Proceeding of the seventh International Workshop on Database and Expert Systems Applications, pages 256– 261, 1996.

15. S. Melnik and S. Decker. A layered approach to information modeling and interoperability on the web. In Proceedings of the ECDL 2000 Workshop on the Semantic Web, Lisbon, Portugal, 2000.

16. J. Euzenat. Towards a principled approach to semantic interoperability. In A. Gomez-Perez, M. Gruninger, H. Stuckenschmidt, and M. Uschold, editors,Workshop on Ontologies and Information Sharing, IJCAI01, Seattle, USA, 2001.

17. V. Chaudhri, A. Farquhar, R. Fikes, P. Karp, and J. Rice. Okbc: A programmatic foundation for knowledge base interoperability. In Proceedings of AAAI-98, pages 600–607, 1998.

18. S. Bowers and L. Delcambre. Representing and transforming model-based information. In Proceedings of the FirstWorkshop on the Semantic Web at the Fourth European Conference on Digital Libraries, Lisbon, Portugal, 2000.

19. D. Calvanese, S. Castano, F. Guerra, D. Lembo, M. Melchiorri, G. Terracina, D. Ursino, and M. Vincini. Towards a comprehensive framework for semantic integration of highly heterogeneous data sources. In Proceedings of the 8th InternationalWorkshop on Knowledge Representation meets Databases (KRDB 2001), 2001.

20. D. Calvanese, D. G. Giuseppe, and M. Lenzerini. Ontology of integration and integration of ontologies. In Proceedings of the International Workshopon Description Logic (DL 2001), 2001.

21. B. Richardson, L.J. Mazlack. Approximate Ontology Merging For The Semantic Web. University of Cincinnati, Cincinnati, United States. Published in IEEE, 2004.

22. A.Alosoud, V. Haarslev, N. Shiri. Deptt. Of Computer Science and Software Engineering, Concordea Univesity, Montreal, Quebec, Canada. Published in Journal of Information Science XX(X) 2008 pp. 1-20. JIS-0759-v3.

23. A.Sheth and J. Larson. Federated database systems for managing distributed, heterogeneous, and autonomous databases. ACM Computer Survey, 22(3), 1990.

24. J. Madhavan, P. A. Bernstein, P. Domingos, and A. Halevy. Representing and reasoning about mappings between domain models. In Proceedings of the Eighteenth National Conference on Artificial Intelligence and Fourteenth Conference on Innovative Applications of Artificial Intelligence (AAAI 2002), pages 80–86, Edmonton, Alberta, Canada., 2002. AAAI Press.

25. A.Maedche, B. Motik, N. Silva, and R. Volz. Mafra - a mapping framework for distributed ontologies. In Proceedings of the 13th European Conference on Knowledge Engineering and Knowledge Management EKAW, Madrid, Spain, 2002.

26. D. McGuinness, R. Fikes, J. Rice, and S. Wilder. An environment for merging and testing large ontologies. In Proceedings of the 7th International Conference on Principles of Knowledge Representation and Reasoning, Colorado, USA, 2000.

27. S. Melnik, H. Garcia-Molina, and E. Rahm. Similarity flooding: A versatile graph matching algorithm and its

application to schema matching. In Proceedings of the International Conference on Data Engineering (ICDE), 2002

28. N. F. Noy and M. A. Musen. Prompt: algorithm and tool for automated ontology merging and alignment. In Proceeding of Seventeenth National Conference on Artificial Intelligence (AAAI-2000), 2000.
29. W. Litiwin, L. Mark, and N. Roussopoulos. Interoperability of multiple autonomous databases. ACM Computer Survey, 22(3):267–293,1990.
30. R. Hull. Managing semantic heterogeneity in databases: A theoretical perspective. Proceeding of the 16th ACM SIGACT SIGMOD SIGART Symposium on Principle of Database systems (PODS'97), pages 51–61, 1997
31. T. Catarci and M. Lenzerini. Representing and using interschema knowledge in cooperative information systems. Journal of Intelligent Cooperative Information Systems, 2(4):375–398, 1993.
32. C. Batini and M. Lenzerini. A comparative analysis of methodologies for database schema integration. ACM Computer Surveys.18(4), 1986.
33. E. Rahm and P. A. Bernstein. A survey of approaches to automatic schema matching. The VLDB Journal, 10:334–350, 2001.
34. D. McGuinness, R. Fikes, J. Rice, and S. Wilder. An environment for merging and testing large ontologies. In Proceedings of the 7th International Conference on Principles of Knowledge Representation and Reasoning, Colorado, USA, 2000.
35. N. F. Noy and M. A. Musen. Prompt: algorithm and tool for automated ontology merging and alignment. In Proceedings of Seventeenth National Conference on Artificial Intelligence (AAAI-2000), 2000.
36. G. Stumme and A. Maedche. FCA-Merge: Bottom-up merging of ontologies. In Proceedings of the International Joint Conference on Artificial Intelligence IJCAI 01., Seattle, USA, 2001.
37. D. Beneventano, S. Bergamaschi, F. Guerra, and M. Vincini. The momis approach to information integration. In ICEIS 2001, Proceedings of the 3rd International Conference on Enterprise Information Systems, Portugal, 2001.
38. D. Beneventano, S. Bergamaschi, I. Benetti, A. Corni, F. Guerra, and G. Malvezzi. Si-designer: A tool for intelligent integration of information. In 34th Annual Hawaii International Conference on System Sciences (HICSS-34). IEEE Computer Society, 2001.
39. N. F. Noy and M. A. Musen. Evaluating ontology-mapping tools: Requirements and experience. In n the Proceedings of the Workshop on Evaluation of Ontology Tools at EKAW'02 (EON2002), Siguenza, Spain, 2002.

AUTHORS

Vishal Jain has completed his M.Tech (CSE) from USIT, Guru Gobind Singh Indraprastha University, Delhi and doing P.hD from Computer Science and Engineering Department, Lingaya's University, Faridabad. Presently he is working as Assistant Professor in Bharati Vidyapeeth's Institute of Computer Applications and Management, (BVICAM), New Delhi. His research area includes Web Technology, Semantic Web and Information Retrieval. He is also associated with CSI, ISTE.

Dr. S. V. A. V. Prasad has completed M.Tech., Ph.D. Presently working as professor and Dean School of Electrical Sciences. Sir has actively participated and organized many refresher courses, seminars, workshops on ISO, ROHS, Component Technology, WEEE, Organizational Methods, Time Study, Productivity enhancement and product feasibility etc. Sir has developed various products like 15 MHz dual Oscilloscope, High Voltage Tester, VHF Wattmeter, Standard Signal, Generator with AM/FM Modulator, Wireless Becom, High power audio Amplifier, Wireless microphone and many more in the span of 25 years (1981 – 2007). Sir has awarded for excellence in R&D in year 1999, 2004 and National Quality Award during the Year 1999, 2000, 2004 and 2006.Sir is Fellow member of IEEE and life member of ISTE, IETE, and Society of Audio & Video System. Sir has published more than 90 research papers in various National & International conferences and journals. Sir research area includes Wireless Communication, Satellite Communication & Acoustic, Antenna, Neural Networks, and Artificial Intelligence.