Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 07 Issue 02
2024
Research Article
https://doi.org/10.56979/702/2024

Development of OWL Structure for Recommending Database Management Systems (DBMS)

Salahuddin[1*], Abdul Manan Razzaq[1], Syed Shahid Abbas[1], Mohsin Ikhlaq[1], Prince Hamza Shafique[1], and Inzimam Shahzad[2]

[1]Department of Computer Science, NFC Institute of Engineering and Technology, Multan, Pakistan.
[2]School of Computer Science and School of Cyberspace Science, Xiangtan University, Xiangtan, Hunan, China.
[*]Corresponding Author: Salahuddin. Email: msalahuddin8612@gmail.com

Received: April 21, 2024 **Accepted:** August 21, 2024 **Published:** September 01, 2024

Abstract: This research focuses on the development of an OWL (Web Ontology Language) structure designed specifically for recommending Database Management Systems (DBMS). The proliferation of various types of DBMSs and their diverse features pose a challenge for users seeking optimal choices based on specific requirements. OWL provides a standardized framework for representing knowledge and semantics, making it suitable for modeling the complex relationships and characteristics of DBMSs. The methodology involves defining OWL classes and properties to capture essential attributes such as data model, scalability, performance, security features, and compatibility with different operating systems and programming languages. Additionally, the ontology incorporates user preferences and requirements as input to refine the recommendations. The implementation includes the creation of an OWL ontology populated with information about existing DBMSs, their capabilities, and user reviews. Reasoning mechanisms are employed to infer relationships and derive recommendations based on the user's specified criteria. Evaluation of the OWL structure involves testing its effectiveness in accurately recommending suitable DBMSs compared to traditional methods. Metrics such as recommendation accuracy, coverage of DBMS features, and user satisfaction will be used to assess the performance of the ontology-based recommendation system. The outcomes of this research are expected to provide valuable insights into the application of semantic technologies, specifically OWL, in enhancing the selection process of DBMSs. By leveraging structured knowledge representation, the developed OWL structure aims to facilitate informed decision-making and improve the efficiency of DBMS selection for diverse applications and user requirements.

Keywords: OWL Structure; Database; Data Base Management System (DBMS); Web Development.

1. Introduction

The Semantic Web marks a significant development in web technology by creating a dataset that machines can better understand and use. The central element of this framework is ontology, which is important for the development and improvement of knowledge in a specific domain. According to [1], ontologies aim to capture and represent knowledge comprehensively, providing common understanding and reusability across multiple applications. It has become important by emphasizing the role of taxonomies in the development of modern literature. Unlike previous practices, which were often limited to small groups with limited access to experts, today's ontology development requires collaboration. These involve contributions from many experts, each bringing their own expertise to the table. First. This process is important to facilitate effective collaboration and ensure consistency and relevance of the ontology model across the team. Data and Ontology engineering methods require a way to develop ontologies, which is important for the development of knowledge in the Semantic Web. As described in [9], ontology engineering encompasses a wide range of activities, including design, lifecycle management, and the methods, tools, and languages used to create ontologies. , methods, and tools are specifically designed to help connected groups or interest groups. This process usually begins with an in-depth exploration of the resource that the ontology will address. Once you have a good understanding of the name, you can create a conceptual model using familiar expressions such as OBO or OWL. Ability to adapt to a changing environment. As the ontology continues to evolve through parallel development, tools to resolve conflicts between participants become important.

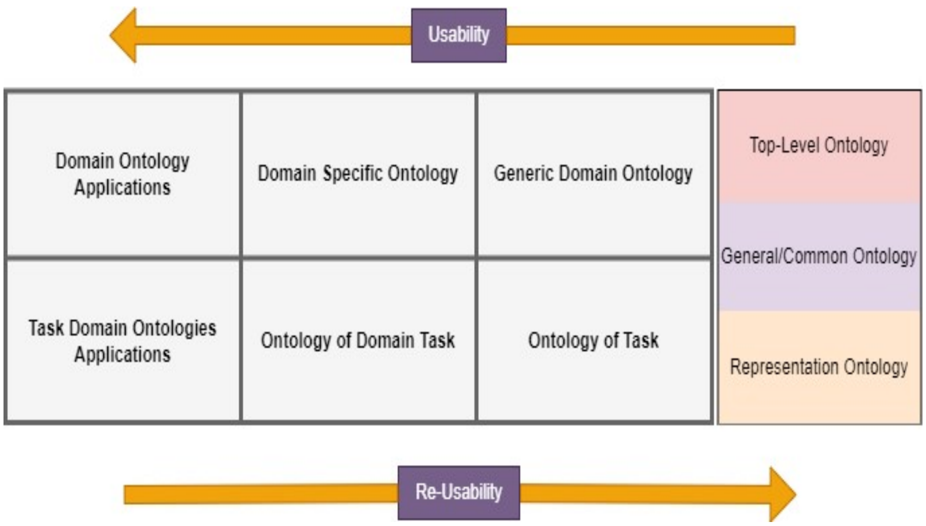

Figure 1. Ontological Classification according to their Reusability

The main stages of ontology engineering include documentation and evaluation.

- **Documentation:** This stage involves creating human-readable documentation to help users understand the OWL or RDF data generated during the process. This information is important to ensure that the ontology is accessible and understandable to both developers and end users. Validation: Ensure that the ontology meets all requirements.

Since ontologies and terminology are intended for reuse, they are often published on the web along with their documentation to encourage greater adoption. This provides a deeper understanding of the real world and the words used to describe them. Overcoming standards and usability challenges is critical to encouraging broad and valid community participation.

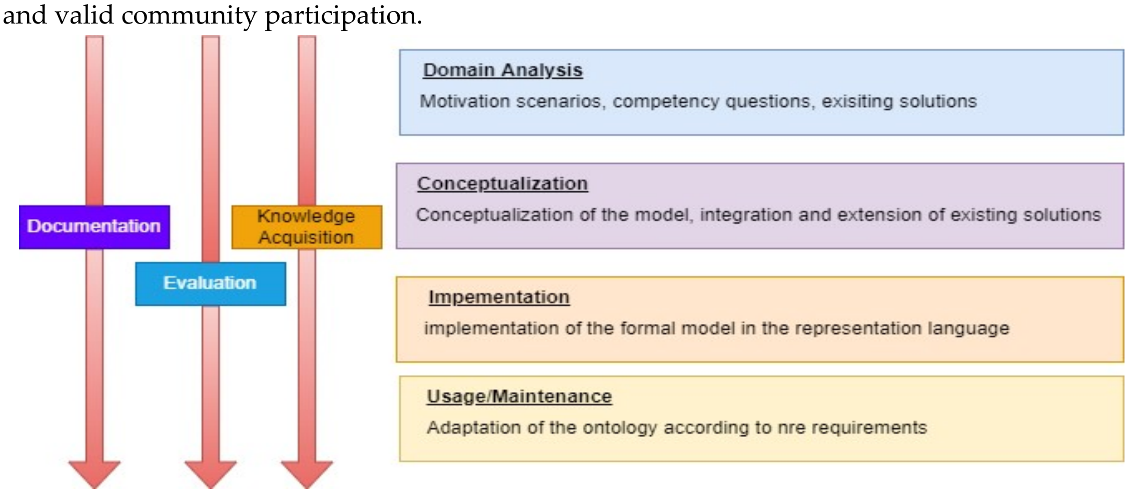

Figure 2. The Process of Ontology Engineering

Drawing parallels with software development processes, ontology engineering borrows methodologies such as those outlined in Methodology [11]. By integrating established software engineering practices, such as requirement formulation, planning, testing, and deployment, ontology development can be standardized and streamlined.

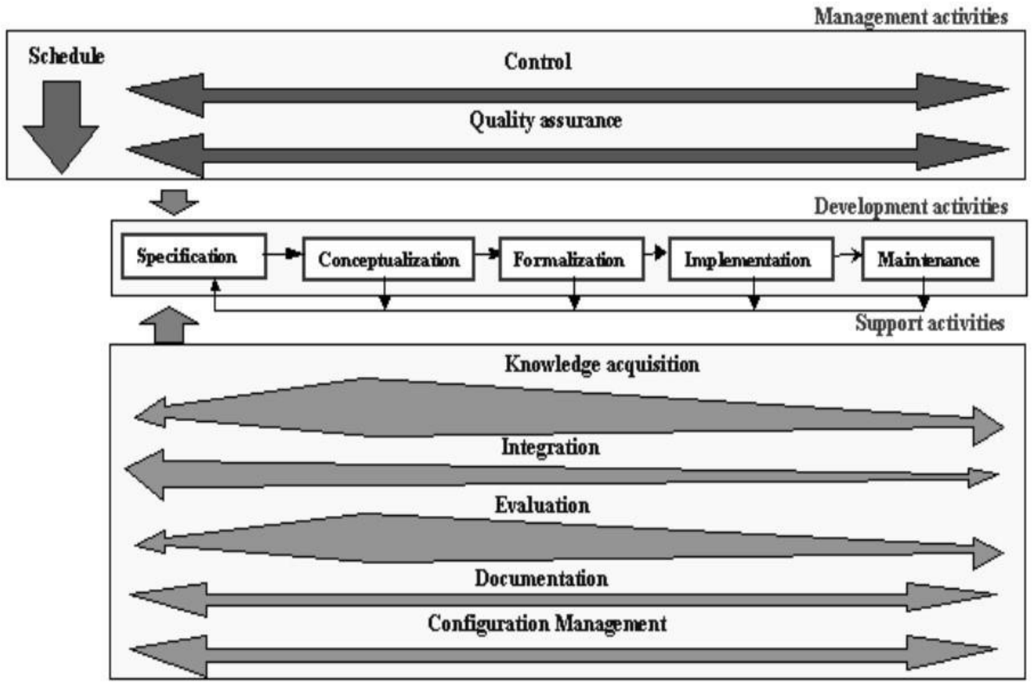

Figure 3. Ontology development life cycle

Two main categories of software methodologies applicable to ontology engineering are traditional and agile methodologies. Traditional methodologies, while rigorous, may struggle with the dynamic and unpredictable nature of ontology development. Agile methodologies like Scrum emphasize flexibility, iterative development cycles (sprints), and continuous adaptation to changing requirements. Agile methodologies promote collaboration, functional demonstrations over documentation, and ongoing customer engagement, all of which are beneficial for ontology development.

Integrating agile principles into ontology engineering can increase operational efficiency and operational flexibility. Examples such as Rapid OWL [14] and XP [15] demonstrate the effectiveness of agile methods by focusing on the iterative process while adapting to specific areas of the engineering knowledge phase. Combine a challenging process with a simple model to manage complexity, enable collaboration, and revolutionize the accuracy and usability of the ontology model.

2. Results

Protégé OWL is a versatile and powerful software package designed to create ontologies specific to the Semantic Web. As a continuous integration with the Protégé ontology editor, it provides a high-level framework necessary for ontology development and smart building implementation. Graphical User Interface (GUI) and API: Users can use Protégé's intuitive GUI to edit classes, objects (slots), and instances in the OWL ontology. The GUI is built on the Protégé framework-based information architecture, providing a familiar and usable environment. Protégé OWL also provides APIs for integration into custom applications. Graphical editor for logical OWL expressions: Protégé OWL includes a user-friendly language editor that simplifies the creation of logical expressions using keyboard input or mouse interaction. It supports graphical representation of preset and custom groups, making it easy to cut, copy, and paste. Wizards to simplify a difficult task: Protégé OWL includes wizards that make ontology engineering work for the useful. These sprites support functions such as grouping, ungrouping, setting multiple properties, and assigning values simultaneously.

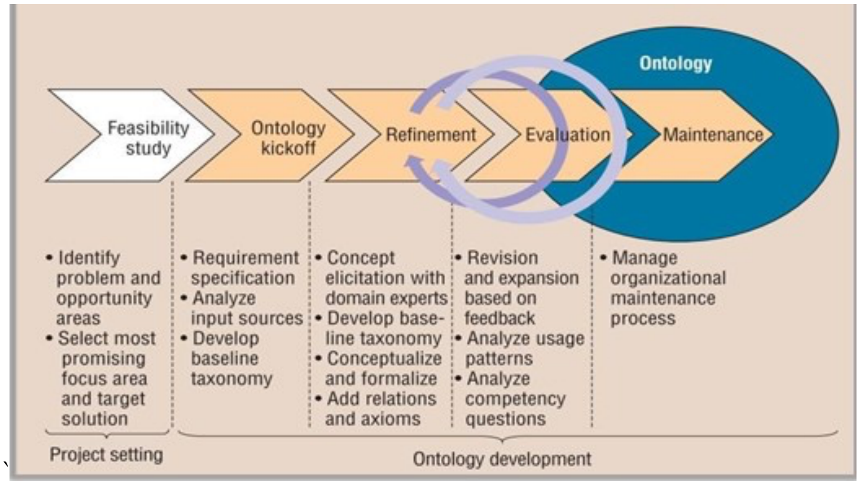

Figure 4. On-To-Knowledge Methodology

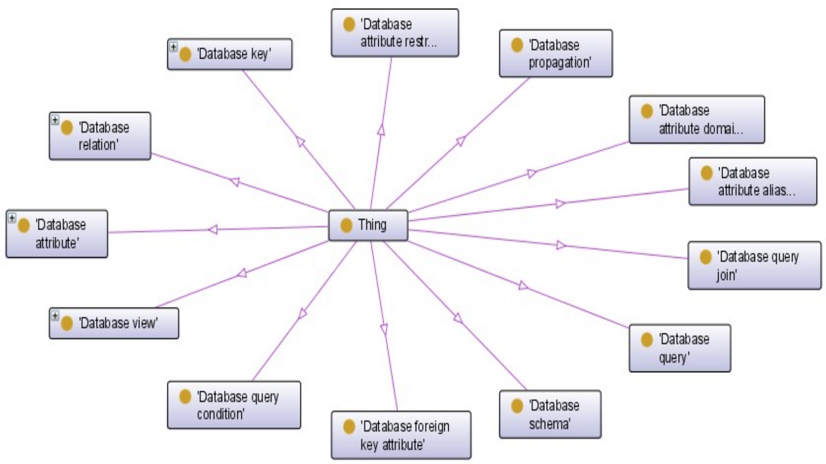

Figure 5. the SKEM ontology development process

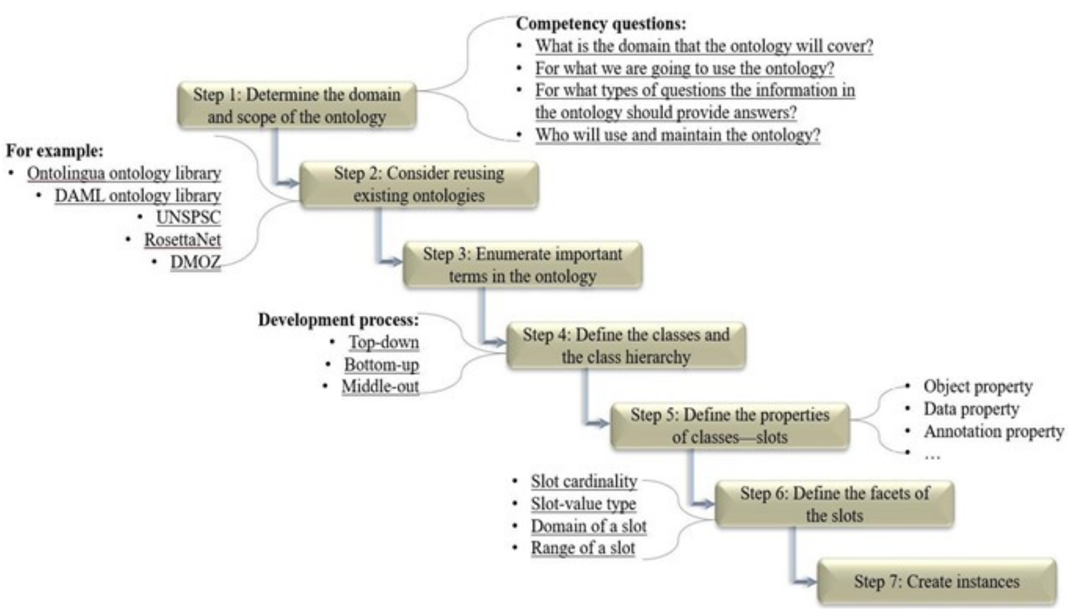

Figure 6. Onto Graph Representation

Direct Access to Reasoners: The software provides seamless connectivity to powerful reasoners like Racer, enabling advanced reasoning capabilities directly within the UI. Users can perform tasks such as consistency checking, subsumption testing, and instance classification efficiently.

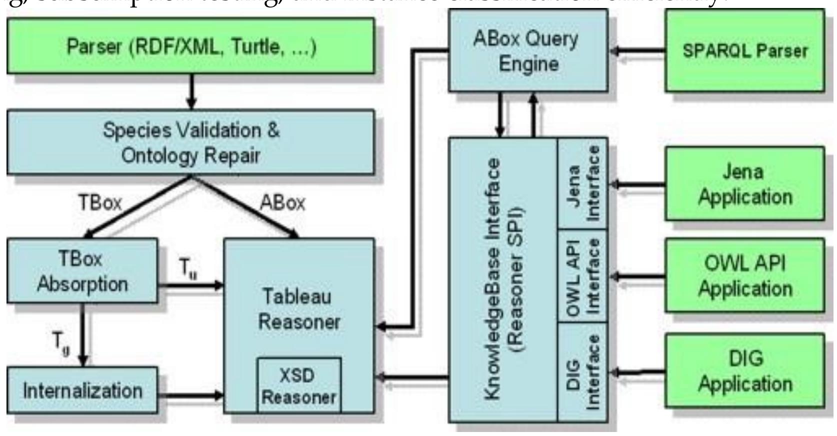

Figure 7. Main components of the Pellet Reasoner

In essence, Protégé OWL is well-suited for developing OWL ontologies and integrating them into intelligent applications due to its comprehensive functionality. It supports both graphical and API-driven ontology editing, simplifies complex tasks with wizards, and facilitates advanced reasoning tasks through direct access to powerful classifiers. These features make Protégé OWL a preferred choice for researchers and developers working in ontology engineering and Semantic Web applications.

$$Precision = \frac{valid\ \ number\ of\ T\ extracted}{Total\ number\ of\ T\ extracted}$$

Our suggested ontology's accuracy values are laid forth in simple fashion.

Table 1. Precision and Recall of Extracted Vocabulary

Vocabulary	Evaluation Measures Precision
Concepts	0.98
Data properties	0.96
Object properties	0.96
Constraints	0.93
Average Results	0.9575

3. Discussion

The complexities of ontology creation highlight the need to select appropriate languages, tools, and methods that fit specific domains and goals. This section is based on the discussion of knowledge discovery terms and design tools to examine different approaches to ontology design, each addressing different taxonomy design needs.

Cyc development methods leverage the experience of building the Cyc Knowledge Base (KB) [52], focusing on building a general knowledge base for reuse [53]. Cyc KB has access to over a million users and provides a large repository of common-sense knowledge that facilitates interaction between different agents and platforms, thus treating it as an ontology [54]. The Cyc method is divided into three stages:

Manual Coding: The basic knowledge involves manual coding of tacit and implicit knowledge, which is a labor-intensive process without automation due to the difference in knowledge quality.

Added bonus: New information continues to be added using existing data to expand the scope and depth of Cyc KB. Automation: As time goes by, computing devices are increasingly used, while human intervention focuses on specific information. Abstractions such as concepts, methods, and related objects serve as symbolic representations of general information. Micro theories in Cyc KB manage the differences between originals, theories, and theories by resolving the confusion caused by specific methods such as micro theories of human activity and organization. Inviting difficulties such as maintaining the consistency of different thoughts and feelings. Micro theories play an important role in balancing and simplifying thought processes suitable for different perspectives, and in providing explanations of the situation.

Uschold and King's process [55, 56], first used in the ontology industry in 1995, suggests a method for ontology development:

Objective Outcome: Knowing exactly what is behind the development of the ontology, why it is used, and the participants who benefit. Ideas and relationships. Feasibility within the framework. Create an introduction to ontology types and functions for ontology descriptions. It focuses specifically on the ontology of data and aims to cover different time periods and domains.

Table 2. Ontology detail with methodology used

	Name of the ontology	Author	Year	Methodology used
1	TOronto Virtual Enterprise ontology (TOVE)	(Gruninger, M., and Fox, 1995)	(1995)	Gruninger & Fox Methodology
2	The Reference Ontology	(Arpírez, Gómez-Pérez, Lozano-Tello, & Pinto, 2000)	(1998)	Methontology
3	Knowledge Acquisition Ontology	Blázquez, J. Fernandez, M. García-Pinar, J. & Gómez-Pérez, A.	(1998)	Methontology
4	Chemical ontology	Fernadez-Lopez, M., et. al	(1999)	Methontology
5	Environmental pollutants ontology Rojas-Amaya, 1999)	(Asunción Gómez-Pérez and Dolores	(1999)	Methontology
6	Legal Ontology	(CORCHO, O., 2002)	(2002)	Methontology
7	Pizza ontology	(Drummond, N., Horridge, M., Stevens,	(2005)	No clue on
8	Wine Ontology	(Graca, J., Mourao, M., Anunciacao, O., Monteiro, P., Pinto, H.S. and Loureiro, 2005)	(2005)	Enterprise Ontology+ Methontology

9	Information Science Ontology	Sawsaa, A. & Lu, J	(2010)	Methontology
10	Beer Ontology	(Heflin, 2012)	(2012)	No clue on methodology
11	Quran ontology for Juz' Amma	(Iqbal, Mustapha, & Yusoff, 2013)	(2013)	ontology merging approach
12	Textile chemical ontology	(Ferrero & Lloret, 2014)	(2014)	Methontology
13	Food Ontology	(Dutta, Chatterjee, & Madalli, 2015)	(2015)	YAMO

This process involves different strategies for ontology creation, each providing unique benefits that are suited to different organizations and information needs. The Cyc approach emphasizes knowledge of the construction and management of content nuances, while the Uschold and King approach focuses on developing the ontology process based on the application environment and the available resources that are most impacted.

4. Conclusion

Many of the ontology topics discussed above are of limited scientific scope, indicating areas where further research could yield important insights and understandings. This section explores gaps in current knowledge and suggests directions for future research, and concludes with a summary. Focus on identifying key concepts and relationships. Future research can be expanded to explore more complex aspects of ontology, such as capturing interactions among concepts, including properties, axioms, and constraints. It is also possible to use data generated from various sources (such as databases) other than data collection to improve the automation and understanding of the ontology creation process. It remains overshadowed by many elements. Current approaches to interpreting, comparing, and integrating ontologies often focus on individual concepts and relationships. Future research could extend this resource to include various ontology components, such as attributes, axioms, and entire ontology models. Improvements in tools for finding and analyzing ontologies for reuse, including taxonomy and top-down methods, could facilitate more efficient and broader ontology reuse. They are still largely language-specific and lack vendor ontology processes. Efforts to create easy-to-use ontology components face challenges related to understanding progress and identifying moving elements. Exploring similarities between ontology models and software models (such as analysis models) reveals opportunities to create ontology models that can guide ontology development and facilitate repository organization. > Ontology development suggests integration with software development, as suggested by Vladan Davidic [101]. Coordination of these activities can be very useful, especially in areas such as development, structure, and material-based construction. This interdisciplinary communication is important to develop the ontology model, thus improving ontology reuse and automation, and making ontology development accessible to experts outside the ontology specialization discipline. A small ontology scale implementation designed to simplify the ontology development process for non-experts. An approach emphasizing the understanding or use of technology can contribute to ontology reuse and increase the efficiency of the ontology development lifecycle. Future research directions should integrate knowledge from software engineering and other disciplines to facilitate the creation of ontologies consistent with the design principles and standards designed for ontology databases.

References

1. S. M. Cahn, editor. Classics of Western Philosophy. Hackett Publishing Company, 6 edition, 2002.
2. Khan, S.U.R.; Raza, A.;Waqas, M.; Zia, M.A.R. Efficient and Accurate Image Classification Via Spatial Pyramid Matching and SURF Sparse Coding. Lahore Garrison Univ. Res. J. Comput. Sci. Inf. Technol. 2023, 7, 10–23.
3. D. Fensel. Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce. Springer, 2001.
4. Khan, S.U.R.; Zhao, M.; Asif, S.; Chen, X. Hybrid-NET: A fusion of DenseNet169 and advanced machine learning classifiers for enhanced brain tumor diagnosis. Int. J. Imaging Syst. Technol. 2024, 34, e22975.
5. Khan, S.U.R.; Zhao, M.; Asif, S.; Chen, X.; Zhu, Y. GLNET: Global–local CNN's-based informed model for detection of breast cancer categories from histopathological slides. J. Supercomput. 2023, 80, 7316–7348.
6. Khan, S.U.R.; Asif, S.; Bilal, O.; Ali, S. Deep hybrid model for Mpox disease diagnosis from skin lesion images. Int. J. Imaging Syst. Technol. 2024, 34, e23044.
7. Raza, A.; Meeran, M.T.; Bilhaj, U. Enhancing Breast Cancer Detection through Thermal Imaging and Customized 2D CNN Classifiers. VFAST Trans. Softw. Eng. 2023, 11, 80–92.
8. Dai, Q., Ishfaque, M., Khan, S. U. R., Luo, Y. L., Lei, Y., Zhang, B., & Zhou, W. (2024). Image classification for sub-surface crack identification in concrete dam based on borehole CCTV images using deep dense hybrid model. Stochastic Environmental Research and Risk Assessment, 1-18.
9. Simone Braun et al. "Ontology Maturing: a Collaborative Web 2.0 Approach to Ontology Engineering." In: Ckc 273 (2007).
10. Khan, S. U. R., & Asif, S. (2024). Oral cancer detection using feature-level fusion and novel self-attention mechanisms. Biomedical Signal Processing and Control, 95, 106437.
11. Kent Beck. Extreme programming explained: embrace change. addisonwesley professional, 2000.
12. Khan, U. S., & Khan, S. U. R. (2024). Boost diagnostic performance in retinal disease classification utilizing deep ensemble classifiers based on OCT. Multimedia Tools and Applications, 1-21. Horrocks. Ontologies and the semantic web. Communications of the ACM, 51(12):58{ 67, 2008.
13. Asif, S., Khan, S. U. R., Amjad, K., & Awais, M. (2024). SKINC-NET: an efficient Lightweight Deep Learning Model for Multiclass skin lesion classification in dermoscopic images. Multimedia Tools and Applications, 1-27.
14. Farooq, M. U., Khan, S. U. R., & Beg, M. O. (2019, November). Melta: A method level energy estimation technique for android development. In 2019 International Conference on Innovative Computing (ICIC) (pp. 1-10). IEEE.
15. N. Guarino and P. Giaretta. Ontologies and knowledge bases: towards a terminological clarification, volume Toward Very Large Knowledge Bases, pages 25–32. IOS Press, 1995.
16. Shahzad, I., Khan, S. U. R., Waseem, A., Abideen, Z. U., & Liu, J. (2024). Enhancing ASD classification through hybrid attention-based learning of facial features. Signal, Image and Video Processing, 1-14.
17. Y. Wand and R. Weber. Information Systems and Conceptual Modelling: A Research Agenda. Information Systems Research, 13(4):363–376, 2002.
18. D. L. McGuinness. Ontologies Come of Age. In Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential. MIT Press, 2002.
19. S. L. Reed and D. B. Lenat. Mapping Ontologies into Cyc (White Paper). http://www.cyc.com/doc/white papers/mapping-ontologies-into-cyc v31.pdf, 2002.
20. D. Beimel and M. Peleg. Using owl and swrl to represent and reason with situation-based access control policies. Data & Knowledge Engineering, 70(6):596{615, 2011.
21. D. L. McGuinness, F. Van Harmelen, et al. Owl web ontology language overview. W3C recommendation, 10(2004-03):10, 2004.
22. B. Parsia and E. Sirin. Pellet: An owl dl reasoner. In Third International Semantic Web Conference-Poster, page 18, 2004.
23. N. F. Noy and D. L. McGuinness. Ontology development 101: A guide to creating your first ontology. 2001.
24. M. Fernandez-L ´ opez and A. G ´ omez-P ´ erez. Overview and analysis of methodologies ´ for building ontologies. Knowledge Engineering Review, 17(2):129–156, 2002.
25. OntoWeb European Project. Technical roadmap (Deliverable D.1.1.2 OntoWeb EUIST2000-29243), 2002
26. OntoWeb European Project. A survey of ontology learning methods and techniques (Deliverable D1.5 OntoWeb EU-IST-2000-29243), 2003.

27. M. Fernandez, A. G ´ omez-P ´ erez, and N. Juristo. Methontology: From ontological art ´ towards ontological engineering. In Proceedings of the AAAI1997 Spring Symposium on Ontological Engineering, pages 33–40, 1997.
28. Waqas, M., Khan, Z., Ahmed, S. U., & Raza, A. (2023, November). MIL-Mixer: a robust bag encoding strategy for Multiple Instance Learning (mil) using MLP-Mixer. In 2023 18th International Conference on Emerging Technologies (ICET) (pp. 22-26). IEEE.
29. Mahmood, F., Abbas, K., Raza, A., Khan, M. A., & Khan, P. W. (2019). Three dimensional agricultural land modeling using unmanned aerial system (UAS). International Journal of Advanced Computer Science and Applications, 10(1).
30. Yang Cai and Julio Abascal. Ambient Intelligence in Everyday Life: Foreword by Emile Aarts, volume 3864. Springer, 2006.
31. Mariano Fernandez-L ´ opez and Asunci ´ on G ´ omez-P ´ erez. Overview and analysis of methodologies for build- ´ ing ontologies. The Knowledge Engineering Review, 17(2):129–156, 2002.
32. Abdelghany, A. S., Darwish, N. R. and Hefni, H. A. (2019) 'An agile methodology for ontology development', International Journal of Intelligent Engineering and Systems, 12(2), pp. 170– 181. doi: 10.22266/IJIES2019.0430.17.
33. Fernández-López, M. and Gómez-Pérez, A. (2002) 'Overview and analysis of methodologies for building ontologies', Knowledge Engineering Review, 17(2), pp. 129– 156. doi: 10.1017/S0269888902000462.
34. Jones, D., Bench-Capon, T. and Visser, P. (2007) 'Methodologies for Ontology Development', in Proc. IT&KNOWS Conference of the 15th IFIP World Computer Congress.
35. Abdelghany, A. S., Darwish, N. R. and Hefni, H. A. (2019) 'An agile methodology for ontology development', International Journal of Intelligent Engineering and Systems, 12(2), pp. 170– 181. doi: 10.22266/IJIES2019.0430.17.
36. Corcho, O. et al. (2005) 'Building Legal Ontologies with METHONTOLOGY and WebODE', in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer, Berlin, Heidelberg, pp. 142–157. doi: 10.1007/978-3-540-32253-5_9. \
37. Szturcová, D. and Rapant, P. (2013) 'Enhanced methodology for ontology development', Computing and Informatics, 32(5), pp. 1038–1054
38. Staab, S., Studer, R. and Gmbh, O. (2001) 'Knowledge Processes and Ontologies', Intelligent Systems, IEEE. IEEE, 16(1), pp. 26–34.
39. Bravo, M., Reyes, L. F. H. and Reyes Ortiz, J. A. (2019) 'Methodology for ontology design and construction', Contaduria y Administracion, 64(4), pp. 1–24. doi: 10.22201/FCA.24488410E.2020.2368.
40. Andryani, R., Negara, E. S. and Darma, U. B. (2015) 'Survey on Development Method of Ontology', in The 4th ICIBA 2015, International Conference, pp. 20–21.
41. Approach", in Proc. 8th Banff Knowledge Acquisition for Knowledge-based Systems Workshop, Banff, January 30th-Febraury 4th, 4.1-4.15.
42. Peroni, S. (2017) 'A simplified agile methodology for ontology development', Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10161 LNCS, pp. 55–69. doi: 10.1007/978-3-31954627-8_5.
43. GUARINO, N., CARRARA, M. and GIARETTA, P. (1994) "An Ontology of MetaLevel Categories", in J. Doyle, E. Sandewall and P. Torasso (eds.) Principles of Knowledge Representation and Reasoning: Proceedings of the 4th International Conference, Morgan Kaufmann, San Mateo, CA.
44. Khan, M. A., Khan, S. U. R., Haider, S. Z. Q., Khan, S. A., & Bilal, O. (2024). Evolving knowledge representation learning with the dynamic asymmetric embedding model. Evolving Systems, 1-16.
45. Jiang, X., & Tan, A. -H. (2005). Mining ontological knowledge from domain-specific text documents. In Proceedings of the fifth IEEE international conference on data mining ICDM '05 (pp. 665–668).
46. Wajid, M., Abid, M. K., Raza, A. A., Haroon, M., & Mudasar, A. Q. (2024). Flood Prediction System Using IOT & Artificial Neural Network. VFAST Transactions on Software Engineering, 12(1), 210-224.
47. Raza, A., Soomro, M. H., Shahzad, I., & Batool, S. (2024). Abstractive Text Summarization for Urdu Language. Journal of Computing & Biomedical Informatics, 7(02).