Volume 9, Issue 7, July – 2024 International Journal of Innovative Science and Research Technology
ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24JUL1103

An Enhanced Technology for Ontology Mapping

Abhishek Patil
Department of Computer Engineering Pillai
College of Engineering, New Panvel, India

Abstract:- The process of defining content from different ontologies is time-consuming, tedious and error-prone. To solve these problems, new methods for ontology comparison have been developed. This process focuses on the integration of ontologies for various applications, but also requires maintaining the integrity of the integrated ontologies. The concept ontology associated with integration is designed to be more efficient, accurate and useful. You can combine these two and compete together to increase accuracy. Comparing this approach with existing methods should provide greater accuracy and efficiency in ontology comparison.

Keywords:- *Automated Negotiation, Open Environment, Heterogeneity Problem, Ontology Mapping***.**

I. INTRODUCTION

In today's global digital communications, effective integration and data exchange between different systems and data is essential. Ontology mapping involves creating a dialogue between elements and relationships in different ontologies and plays an important role in ensuring interoperability. However, ontology systems often face challenges such as scalability, accuracy, and adaptability to dynamic environments. Provide valuable on-the-job training. Using advanced technologies in artificial intelligence, machine learning and knowledge representation, our technology promises to revolutionize the ontology mapping method. Principles, methods and practices. We will show how it surpasses existing methods in terms of efficiency, accuracy and versatility, paving the way for seamless information integration across different domains and platforms. Real data will be presented: the effectiveness of our global technology and its ability to drive innovation across industries, including e-commerce, health, wellness, money and more. Additionally, we will discuss the implications of our findings for education and industry and suggest avenues for research and future developments in the field of ontology mapping. Advanced ontology mapping provides unprecedented ability to enable interoperability across digital environments.

II. EXISTING SYSTEM

There are several methods developed and used for ontology matching over the period of time, One of those are mentioned above include SAMBO, ICOMA, ICOMA++ ,YAM, ILOMPTI with novel ontology mapping algorithm and semantic heterogeneity matching:

- Step 1. Identify the boundaries and limitations of the ontology.
- Step 2. Consider reusing existing ontologies
- Step 3. Enumerate important terms in the ontology Step 4. Define the categories and the hierarchy of categories.
- Step 5. Define the properties of classes—slots Step 6. Define the facets of the slots
- Step 7. Create instances

The approach you describe is very close to the process of knowledge engineering, which involves creating the ontology or knowledge representation for a given object. In the context of ontology mapping, this tutorial provides a way to create ontologies as the basis for conceptual mapping and the relationship between different information.

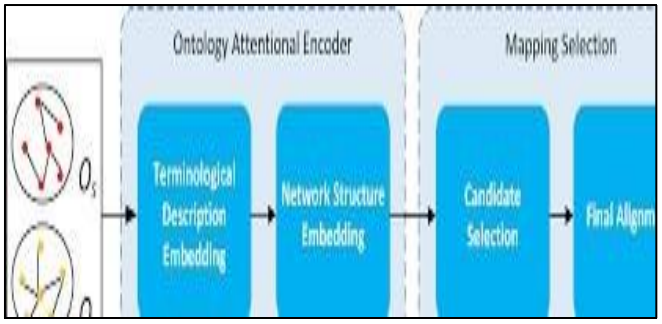

Ontology Attentional Encoder
Mapping Selection
O_s
Terminological Description Embedding
Network Structure Embedding
Candidate Selection
Final Alignm

Fig 1: Existing Model

The existing model depicted in the figure represents a framework for ontology mapping, which is a crucial task in integrating and aligning information across different knowledge bases or ontologies.

III. LITERATUE SURVEY

Patrick Lambrix and His Tan presented the paper SAMBO - A Framework for Shared and Interoperable Biomedical Ontologies in 2006. In recent years, many biomedical ontologies have been created, many of which contain overlapping knowledge. In order to use multiple ontologies, they need to be optimized or integrated. This paper presents a framework for the integration and integration of ontologies. They also developed a unified and integrated biomedical ontology (SAMBO) based on this framework. The framework is also a first step towards a framework that can be used for comparative analysis of strategic partnerships and their linkages. This paper evaluates different strategies and their combinations in terms of performance and runtime and compares SAMBO with two other systems [8]. al proposed COMA, a system for the combination of comparative models.

Many database applications require parallel structures, such as integration of Web data sources, data warehouse loading, and XML thread mapping. An automated process is needed that integrates various combinations to reduce the user's work as much as possible. Although such combinations have attracted great attention recently, the question of how to best combine different algorithms still needs further study. Therefore, this article develops the COMA model integration as a platform for the integration of multiple candidates. The authors present various matchers, specifically a new way to use the results of previous matcher applications, and several ideas for combining the results of matcher applications.

They use COMA as a framework to evaluate the performance and connections of different candidates on real models. They use COMA to evaluate different types of matchmaking, i.e. collection of matcher-specific results, matching direction, matched candidates and compatible ladies, as well as different matcher uses, i.e. one matcher versus multiple matchers. Matcher is assembled, unused and reusable. Other combinations and combinations should be added to increase similarity [9].

In this paper, the previous COMA model is extended using a hybrid approach to combine different algorithms. COMA++ implements significant improvements and provides a general framework for solving large-scale real-world problems. Using the representation of data, COMA++ supports schemas and ontologies in the same way as the powerful standard language W3C XML Schema and OWL. COMA++ includes new methods for ontology comparison, particularly the use of shared taxonomies. COMA++ can not only be used to solve the matching problem, but also to compare and evaluate the performance of different algorithms and different strategies [10].

This article describes YAM (Yet Another Matcher), a standard matcher factory. It can actually create custom matchers for a specific matching scenario based on user input. This approach is based on machine learning as adversarial patterns can be viewed as a classification. Many tests on matchers generated by YAM and the combination tool show that our method can create the best matcher for a given situation. In the pre-competitive stage, it creates candidates for the model according to the situation with the help of machine learning algorithms [11].

Scalable ontology matching algorithm. Due to the holistic nature and scale of real-world domain ontologies, matching the ontology with good efficiency and scalability is a challenge. This article introduces LOMPT, Large Ontology Matching Using Partitioning Technology. It has a model- based partitioning algorithm that divides a large ontology into smaller partitions. Then, the distribution of ontology pairs is discovered based on the distribution of anchor terms, where anchor terms are defined by the proposition of the matching string.

This paper presents a classification-based ontology matching algorithm that can be effective for growing ontologies. A new neighbor-based proximity measure has been developed to reduce the time spent, especially since the proximity model is calculated only between neighbors. The efficiency of the LOMPT ontology mapper is very high because the proposed partitioning algorithm can be scaled linearly. In the future, the time spent searching for anchors should be reduced, for example, by using more lightweight sequence matchers from randomly selected organizations [17].

IV. PROPOSED SYSTEM

We offer correct and effective solutions for combining two storage facilities with good techniques. We use ontology- based top-k global model design to solve the problem of integrating models into the system. This framework uses ontologies as an integration model to build better global systems with less user involvement; because ontologies can provide context and detailed information and improved methods for storing hierarchies and eliminating duplicates. One of the goals of the proposed framework is to achieve a high level of automation, with the result that only the user selects the unified ontology. The following is the conceptual model of the initial data converted into ontology. The matching process is done between ontology and nature.

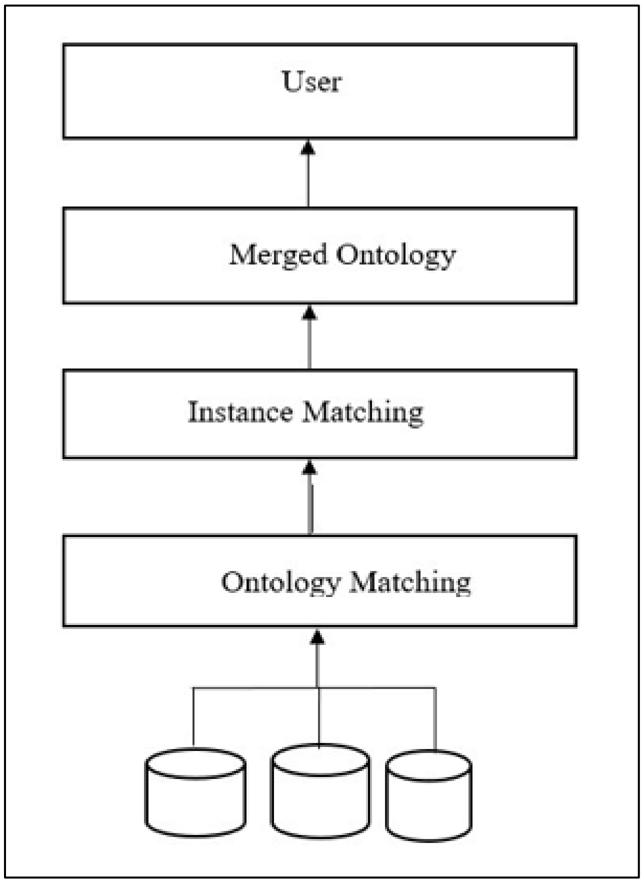


Fig 2: Conceptual Model

When the integration process is completed, we have a unified ontology. Similar. If the similarity between two attributes is more than the threshold, the similarity between them is set to 1, otherwise it is set to degree 0 and use the similarity value to consider the top-k merged ontology. (3) Third, if there is a hierarchical structure in the local ontology, the similarity between the parent category and the child category is recalculated to rearrange the inheritance relationship in the top-k merged ontology.

V. ALGORITHM

In this section, we introduce algorithm for Top-K merged ontologies which takes two ontologies and K as an input where K is the number of top merged ontologies. This will produce output of top-K merged ontology [5].

- Algorithm: Compute_Top-K-Merged-Ontologies( ) Input: local ontology O1, local ontology O2, K
- Output: topKMergedOntologies
- ✓ Step1: simArray := NULL, disArray := NULL, assignArray := NULL, diffArray := NULL;
- ✓ Step 2: foreach class Ci ∈ O1 Step 3: foreach class Cj ∈ O2
- ✓ Step 4: if there exists matched properties between Ci and Cj
- ✓ Step 5: {
- ✓ Step 6: Compute simArrayij using Equation (2); Step 7: disArrayij := 1 - simArrayij;
- ✓ Step 8: if simArrayij > disArrayij Step 9: { assignArrayij := 1; } Step 10 else
- ✓ Step 11: { assignArrayij := 0; }
- ✓ Step 12: diffArrayij := |simArrayij - disArrayij|; Step 13: }
- ✓ Step 14: end foreach Step 15: end foreach
- ✓ Step 16: foreach assignArrayij ∈assignArray
- ✓ Step 17: if assignArrayij = 1
- ✓ Step 18: { Mij := Merge class Ci ∈ O1 and Cj ∈ O2; } Step 19: end foreach
- ✓ Step 20: TopKMergedOntologies[1] := Mij + not merged classes in O1 and O2;
- ✓ Step 21: Sort diffArray in descending order; Step 22: for t := 1 to K-1
- ✓ Step 23: flippedArray := assignArray;
- ✓ Step 24: if diffArrayij has the t-th smallest value Step 25: { flippedArrayij := 0; }
- ✓ Step 26: else if ΣdiffArrayij has the t-th smallest value Step 27: { set all the corresponding flippedArrayij := 0; }
- ✓ Step 28: foreach flippedArrayij ∈flippedArray
- ✓ Step 29: if flippedArrayij = 1
- ✓ Step 30: { Mij : = Merge class Ci ∈ O1 and Cj ∈ O2; } Step 31: end foreach
- ✓ Step 32: TopKMergedOntologies[t+1] := Mij + not merged classes in O1 and O2;
- ✓ Step 33: end for
- ✓ Step 34: return TopKMergedOntologies;

VI. SYSTEM ARCHITECTURE

➢ *The System Architecture for our Proposed Work is as Follows:*

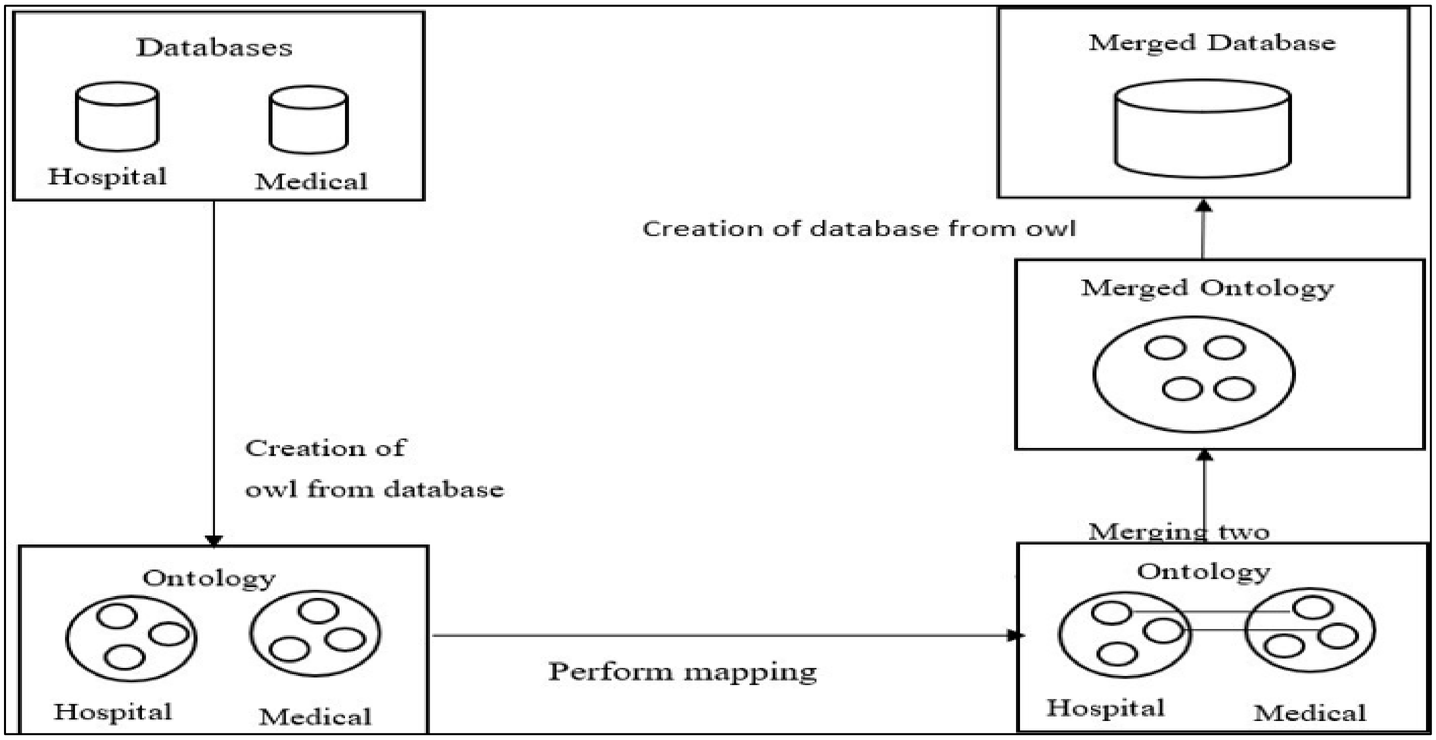


Fig 3: Architecture of a Combined Approach

In this section we will describes all modules shown in Fig.4.1 (Architecture)that are require to implement proposed scheme.

A. Reading Source Database

First, we created two databases to compare the ontology: hospital and medical. These databases serve as input algorithms. We also need other data to store the results together between these two databases. Similarly we create some tables in each database instance, hospital database has doctor, patient and medical tables. In medical records there are words of doctors, medicines and patients. By combining two databases, we will obtain four tables: doctor, patient, medicine and treatment pain. After creating the database, we need to access it while running the algorithm. So in this mode we are reading a database of hospitals and treatments.

B. Creation of Owl From Database

In this mode, since the root engine supports the owl file, we convert the file to an owl file. Initially, our data source is in the data warehouse, where we cannot follow the recommended process because it is used for drawing ontologies. We use protégé API for this. Protege-OWL API is an open source Java library for Web Ontology Language (OWL) and RDF(S). The API provides classes and methods for loading and saving OWL archives, querying and manipulating OWL data structures, and making inferences based on descriptive logic engines. Additionally, the API is optimized for graphical user interface use. Protege is a simple, configurable platform for building on-demand model-driven applications and products. Protege has an open architecture that allows programmers to integrate plugins that can be displayed as separate tabs, custom user interface elements (widgets), or perform other functions on the fly. Protege- OWL editor provides many editing and search tools for OWL models and therefore serves as a good starting point for rapid application development. When we convert data into ontology, the tables in the data are converted into models, the rows in each table are converted into attributes, and the data in the tables are converted into resources.

C. Algorithm to Merge Two Owl

We talked about 3 steps to find the consistency of the group during the planning process. Among them, we use the Hausdorff method. The modified Hausdorff distance has two strengths: first, it is simple, and second, its value increases with the difference between the two groups. After calculating the similarity between two different ontologies according to the given equation, the upper merged ontology is calculated using the top-k algorithm.

VII. RESULT

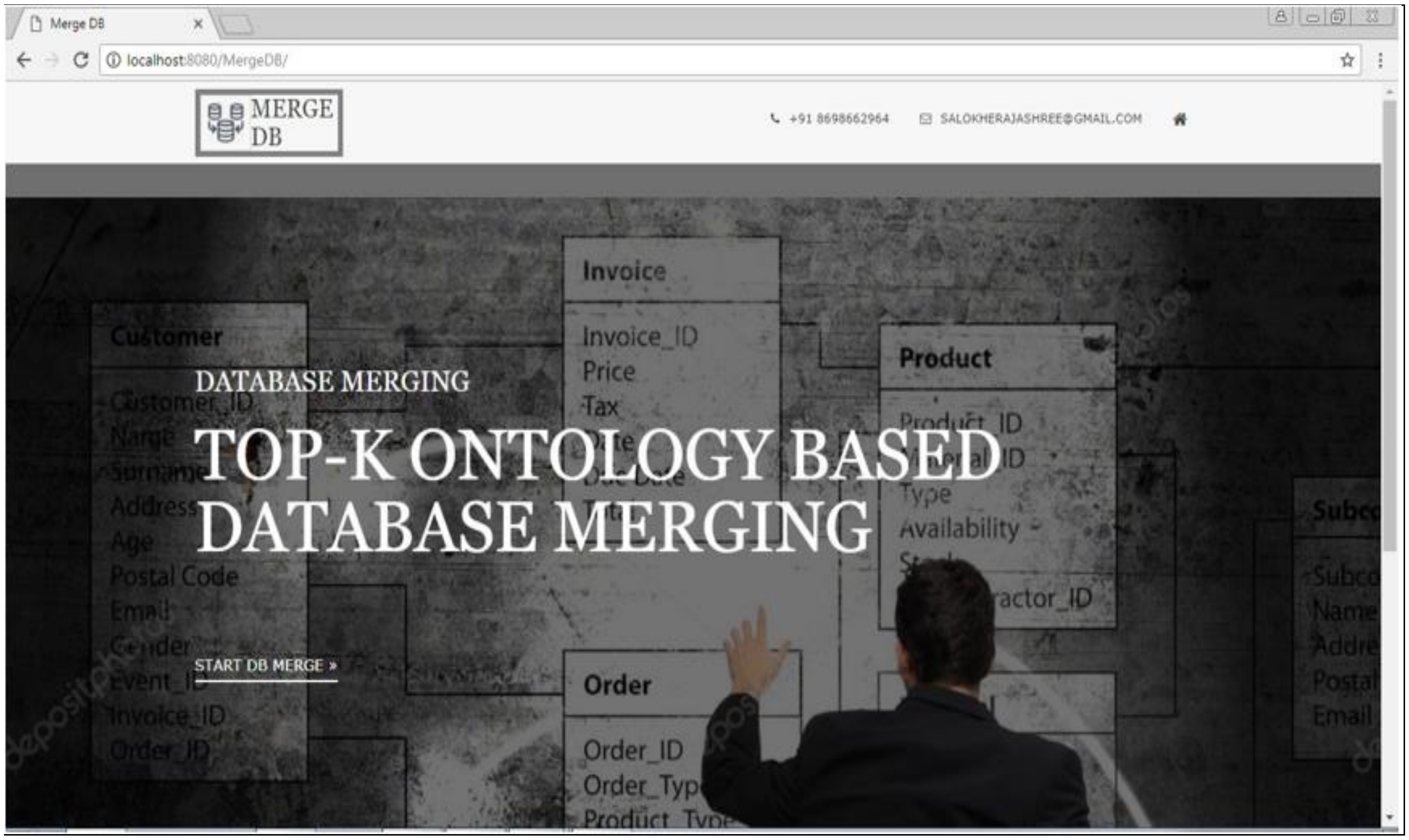


Fig 4: Home Page

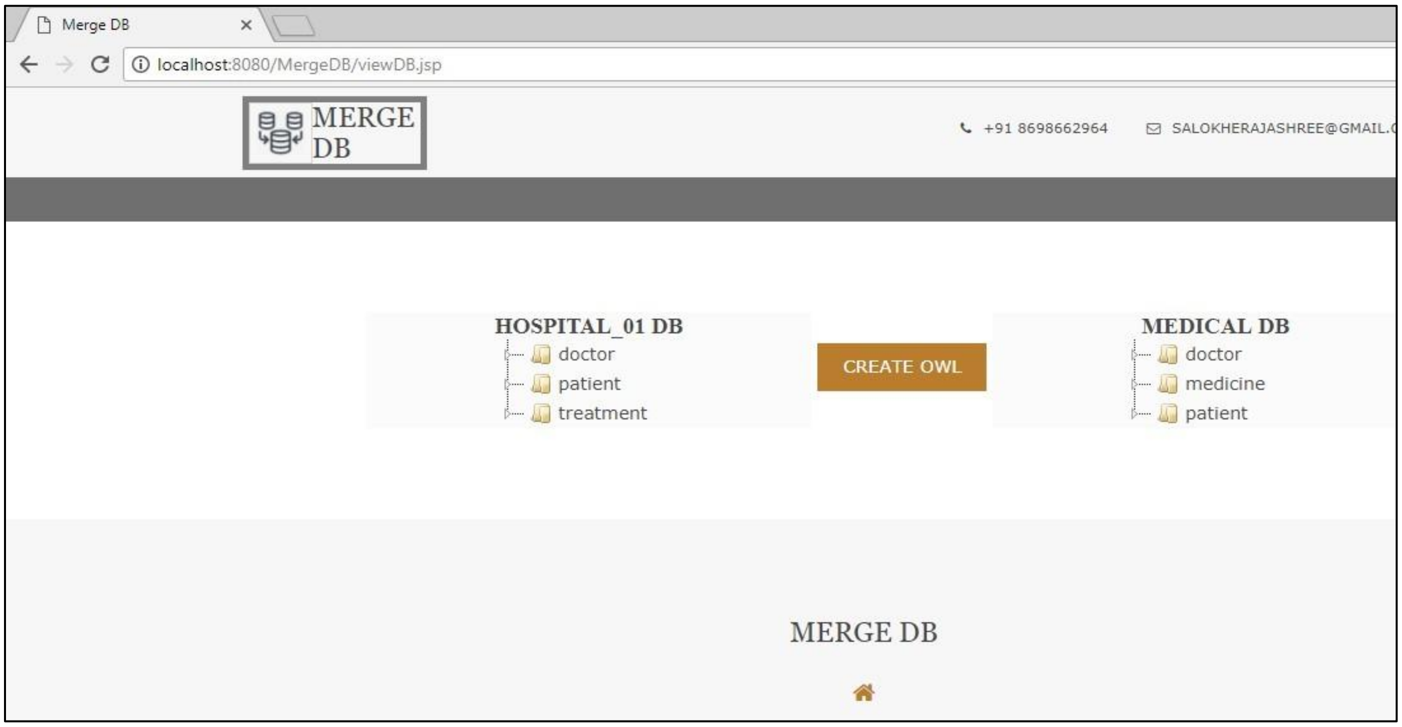
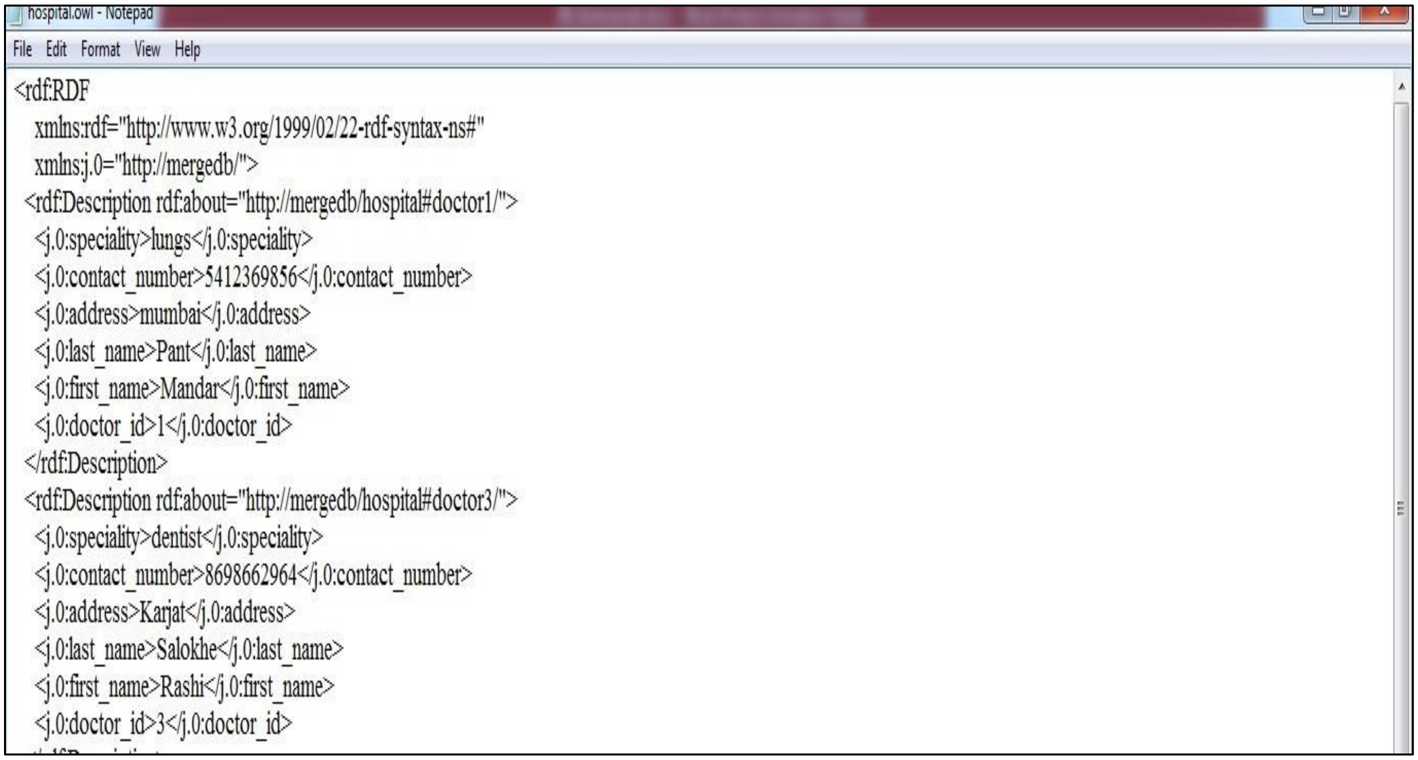
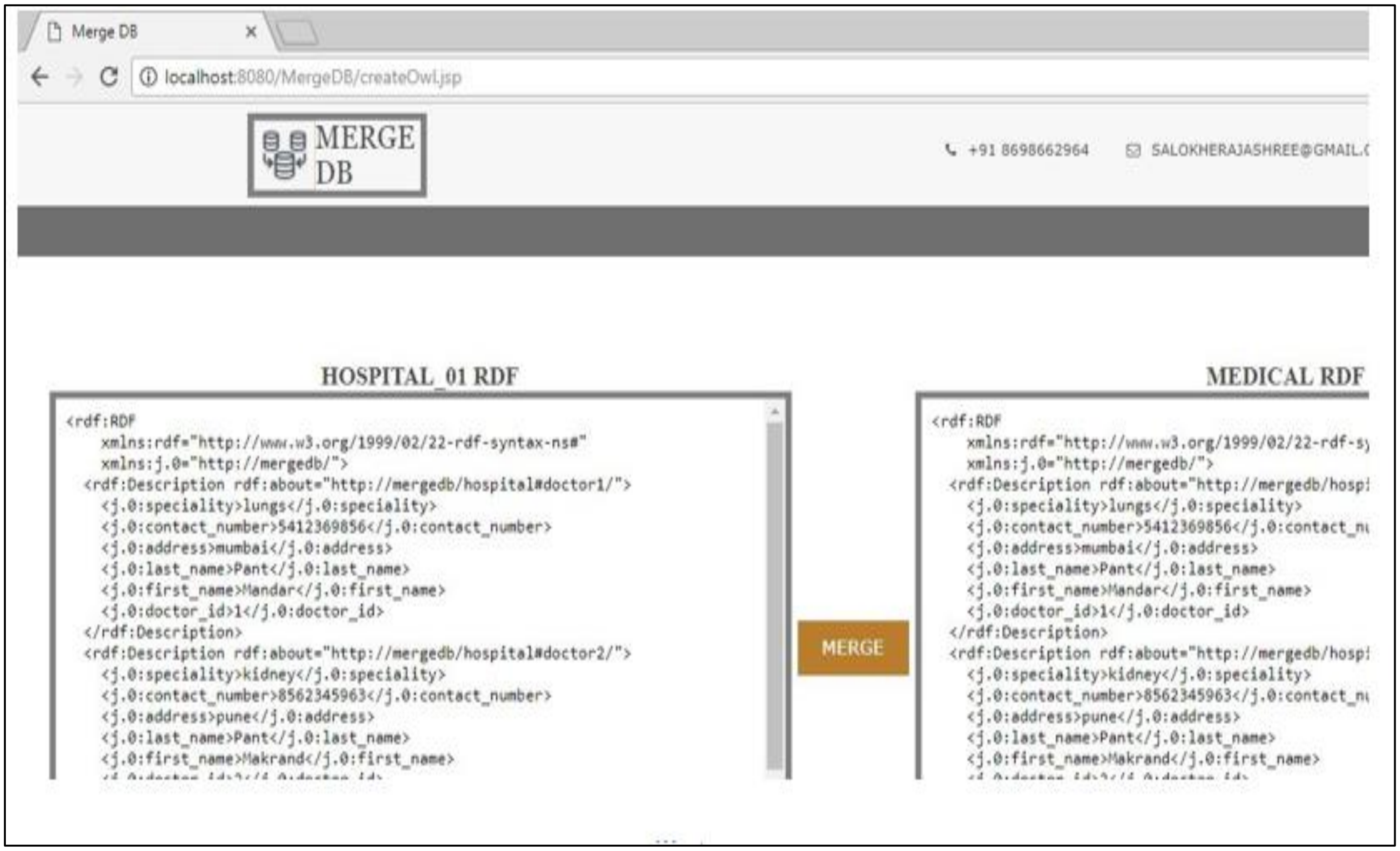


Fig 5: Reading Source Database



```
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:j.0="http://mergedb/">
  <rdf:Description rdf:about="http://mergedb/hospital#doctor1/">
    <j.0:speciality>lungs</j.0:speciality>
    <j.0:contact_number>5412369856</j.0:contact_number>
    <j.0:address>mumbai</j.0:address>
    <j.0:last_name>Pant</j.0:last_name>
    <j.0:first_name>Mandar</j.0:first_name>
    <j.0:doctor_id>1</j.0:doctor_id>
  </rdf:Description>
  <rdf:Description rdf:about="http://mergedb/hospital#doctor3/">
    <j.0:speciality>dentist</j.0:speciality>
    <j.0:contact_number>8698662964</j.0:contact_number>
    <j.0:address>Karjat</j.0:address>
    <j.0:last_name>Salokhe</j.0:last_name>
    <j.0:first_name>Rashi</j.0:first_name>
    <j.0:doctor_id>3</j.0:doctor_id>
```

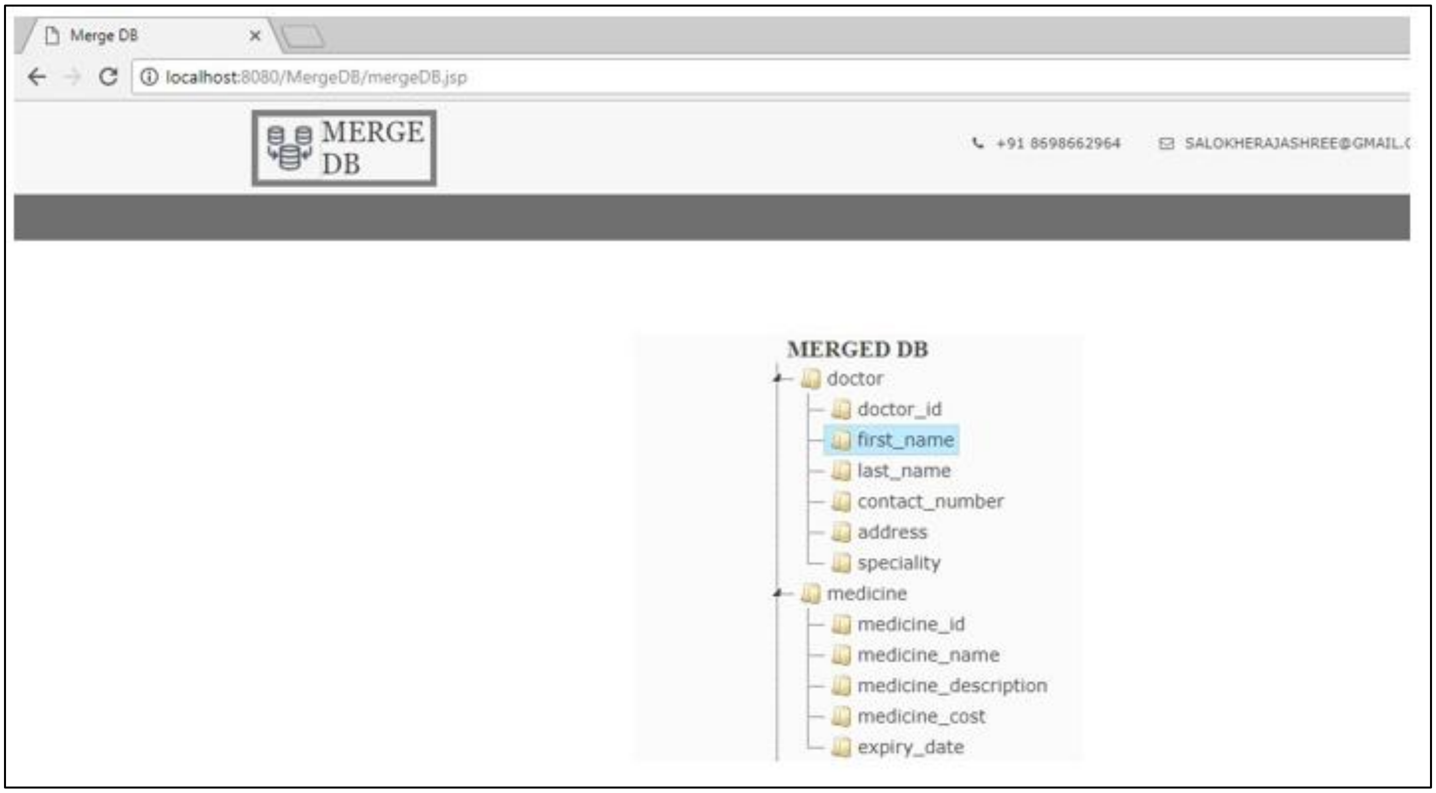
Fig 6: Hospital.owl File



```
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:j.0="http://mergedb/">
  <rdf:Description rdf:about="http://mergedb/hospital#doctor1/">
    <j.0:speciality>lungs</j.0:speciality>
    <j.0:contact_number>5412369856</j.0:contact_number>
    <j.0:address>mumbai</j.0:address>
    <j.0:last_name>Pant</j.0:last_name>
    <j.0:first_name>Mandar</j.0:first_name>
    <j.0:doctor_id>1</j.0:doctor_id>
  </rdf:Description>
  <rdf:Description rdf:about="http://mergedb/hospital#doctor2/">
    <j.0:speciality>kidney</j.0:speciality>
    <j.0:contact_number>8562345963</j.0:contact_number>
    <j.0:address>pune</j.0:address>
    <j.0:last_name>Pant</j.0:last_name>
    <j.0:first_name>Makrand</j.0:first_name>
```

```
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-sy
   xmlns:j.0="http://mergedb/">
  <rdf:Description rdf:about="http://mergedb/hospi
    <j.0:speciality>lungs</j.0:speciality>
    <j.0:contact_number>5412369856</j.0:contact_nu
    <j.0:address>mumbai</j.0:address>
    <j.0:last_name>Pant</j.0:last_name>
    <j.0:first_name>Mandar</j.0:first_name>
    <j.0:doctor_id>1</j.0:doctor_id>
  </rdf:Description>
  <rdf:Description rdf:about="http://mergedb/hospi
    <j.0:speciality>kidney</j.0:speciality>
    <j.0:contact_number>8562345963</j.0:contact_nu
    <j.0:address>pune</j.0:address>
    <j.0:last_name>Pant</j.0:last_name>
    <j.0:first_name>Makrand</j.0:first_name>
```

Fig 7: Merging Two Ontologies



MERGED DB
- doctor
  - doctor_id
  - first_name
  - last_name
  - contact_number
  - address
  - speciality
- medicine
  - medicine_id
  - medicine_name
  - medicine_description
  - medicine_cost
  - expiry_date

Fig 8: Resulted Merged Schema

VIII. FUTURE SCOPE

Several extensions of this work can be elucidated, such as searching for relational spaces with different cardinality such as 1:n, m:1, and m:n for ontology mapping. This mapping implementation matches character by character, but cannot match semantic strings. In the future, we hope to do the matching through semantic mapping. Also, the accuracy of mapping various ontologies can be improved. Some important incentives for further research and development are:

- Addressing Semantic Heterogeneity: As data continues to proliferate across diverse domains and sources, the challenge of semantic heterogeneity becomes more pronounced. Enhanced ontology mapping technology offers a Solve this problem by providing good solutions standardized framework for reconciling semantic differences and integrating heterogeneous data sources.
- Enabling Interoperability in Emerging Technologies: With the emergence of new technologies such as the Internet of Things (IoT), blockchain, and augmented reality, there is a growing need for interoperability between different systems and devices. Ontology mapping can play a crucial role in enabling seamless communication and data exchange in these interconnected environments, fostering innovation and unlocking new possibilities.
- Advancing Artificial Intelligence and Machine Learning: Ontology mapping technology can enhance the capabilities of artificial intelligence (AI) and machine learning systems by providing them with structured, semantically enriched data. This can lead to more accurate and contextually relevant AI applications, including natural language understanding, recommendation systems, and autonomous decision-making.
- Facilitating Knowledge Discovery and Innovation: By enabling the integration and analysis of diverse knowledge sources, ontology mapping technology facilitates knowledge discovery and innovation across various domains. It empowers researchers, scientists, and domain experts to uncover hidden patterns, relationships, and insights, driving progress and discovery in fields such as healthcare, finance, and scientific research.

IX. CONCLUSION

The development and use of ontology mapping technology has broad prospects for improving knowledge representation, knowledge integration, and semantic interaction. It uses advanced algorithms, machine learning, and rigorous testing to resolve the semantic heterogeneity of different documents. Technology supports seamless communication, data integration and knowledge discovery, allowing organizations to realize their potential in many areas. Due to the hierarchical type structure and the weak position of the local type, redundant information will be produced in the global type. In practice, we have learned that we can match many sequences from different data sets using the over-K merge algorithm. This algorithm gives us the accuracy of each performance measure. We have worked on various ontology matching and ontology map security to obtain accurate and efficient results. The generated data will take into account the character for the text and the characters of the character. The binary values of these symbols are mapped by calculating them. If all the mappings are done automatically and inferences are made on it, the erroneous results will reduce the value of the entire mapping process. The mathematical complexity will increase if the data shows a certain semantics. The problem of "getting more profit". Also, research will benefit more from the constraints specified in the ontology by the attributes, relations and constraints of the ontology for searching and understanding the content for different library users. Also, it has been mentioned for different library users. Useful tools in this way we treat the ontology problem as a binary problem. Experimental results show that our method is effective on the data. The system can also compare samples from two databases. A good system maintains hierarchical structure and eliminates unnecessary data.

REFERENCES

[1]. Ahmed K. Elmagarmid, Panagiotis G. Ipeirotis, and Vassilios S. Verykios, “Duplicate Record Detection: A Survey”, IEEE Computer Society, 2007.
[2]. Monica Scannapieco, Ilya Figotin, Ahmed Elmagarmid ,“Privacy Preserving Schema and Data Matching”, ACM Sigmod Internationl Conference on management of data, 2007.
[3]. Zhe Lim and Benjamin I. P. Rubinstein, “Sub-Merge: Diving all the way down to the attribute-cost degree in Statistical Schema Matching”, Association for the Advancement of Artificial Intelligence, 2014.
[4]. Binbin Gu, Zhixu Li Et. al., “The Interaction between Schema matching and record matching in data integration, a topic discussed in the IEEE Transactions on Knowledge and Data Engineering, was presented in 2016”.
[5]. Longzhuang Li,Yuzhe Wei Et.al., “A Framework for Ontology-Based Top-K Global Schema Generation”, Springer, 2017.
[6]. Pavel Shvaiko, Jerome Euzenat, “Ontology matching: nation of the art and future demanding situations”, IEEE Transactions on Knowledge and Data Engineering, 2013.
[7]. Prasenjit Mitra, Peng Liu, “Privacy-preserving Ontology Matching”, American Association for Artificial Intelligence, 2005.
[8]. Patrick Lambrix and He Tan, “SAMBO - A System for Aligning and Merging Biomedical Ontologies”, Web Semantics 4(3):196-206, 2006.
[9]. Hong-Hai Do Erhard Rahm, “COMA - A system for flexible combination of schema matching approaches”, VLDB Conference, 2008.
[10]. David Aumueller, Hong-Hai Do, Et.al., “Schema and Ontology Matching with COMA++”, SIGMOD, 2005.
[11]. Fabien Duchateau Et.al., “(Not) Yet Another Matcher”, CIKM’09, Hong Kong, China, 2009.
[12]. Sergio Cerón, Itzamá López, “Instance-Based Ontology Matching For Open and Distance Learning Materials”, global evaluation of research in Open and allotted Learning Volume 18, Number 1, 2017.
[13]. Keke Gai ,Meikang Qiu, Saravanan Jayaraman, Lixin Tao, “Ontology-Based Knowledge Representation for Secure Self-Diagnosis in Patient-Centered Teleheath with Cloud Systems”, IEEE 2nd International Conference on Cyber Security and Cloud Computing, 2015.
[14]. Toshiyuki Amagasa, Fan Zhang, “A Scheme for Privacy-Preserving Ontology Mapping”, ACM, 2014.
[15]. Saira Gillani, Muhammad Naeem, Et.al, “Semantic Schema Matching Using DBpedia”, I.J. Intelligent Systems and Applications, 2013.
[16]. Xingsi Xue, Jeng-Shyang Pan, “A segment-based approach for large-scale ontology matching”, Knowl Inf Syst., 2016.
[17]. K.Saruladha, “LOMPT: An efficient and scalable ontology matching algorithm”, VLDB Conference, 2009.
[18]. Xingsi Xue,Jeng-Shyang Pan, “A segment-based approach for large-scale ontology Matching”, Springer-Verlag London, 2016.
[19]. Mohammad A. Al-Ramahi and Suleiman H. Mustafa, “N-Gram Methods for Arabic Text Document Mapping; Case Study: Academic Authorization ”, Vol. 21, No. 1, 2012, pages 85-105, 2012.
[20]. Abdulla Ali, "Text Similarity", IMM-BSc, 2011.
[21]. Zhe Lim and Benjamin I. P. Rubinstein, "Sub-Merge: Diving all the way down to the attribute-price level in Statistical Schema Mapping", Association for the Development of Artificial Intelligence, 2014.
[22]. Binbin Gu, God Li Et. al., "Collaboration between Schema Mapping and Record Mapping in Data Integration", IEEE transaction in Knowledge and Data Engineering, 2016.
[23]. Longzhuang Li, Yuzhe Wei Et.al., "Ontology-based Top-K Global Schema Generation Framework", Springer, 2017.

[24]. Paulo Carvalho, Solange Rito Lima, Luis Alvarez Sabucedo,João Marco Cardoso Silva “Towards a holistic semantic support for context-aware network monitoring: An ontology-based approach”,2020.
[25]. Salvatore Flavio Pileggi “Knowledge interoperability and again in Empathy Mapping: An Ontological Approach”,2020.
[26]. Nikolay Maksimov, Alexander Lebedev “Knowledge ontology system”,2021.
[27]. Mo’men ElsayedNermeen ElkashefYasser F.Hassan “Mapping UML Sequence Diagram into the Web Ontology Language OWL”,2020.