International Journal of Advanced Research in

Computer Science Engineering and Information Technology

Volume: 1 Issue: 3 08-Jan-2014,ISSN_NO: 2321-3337

Ontological Ranked Search with Dynamic Data Support

J.S.Beulah[1], Dr.Mary Metilda[2]

Assistant Professor, D.G.Vaishnav college[1]

Arumbakkam , Chennai-106

Asst Professor, Queen Mary's College[2]

Chennai-4

ABSTRACT— *Ranked search is one of the upcoming and prominent research areas for effective handling of documents and filtration/ranking of documents towards data search engine world. Ranked search associated with conceptual ontological data storage and novel document updating mechanisms helps in providing the best tool for document maintenance. This paper discusses about the exiting tools and techniques used for ontological storage, faster search and retrieval of documents and also about the tools used for the same.*

Index Terms: *Ontological search, ranked search, conceptual data storage, storage services, memory handling.*

1, Introduction:

Distributed storage involves huge task that includes proper maintenance of data, faster retrieval and documents must also be up to date. Hence the data is stored in a knowledge repository where in the data and its relationships are deposited to form ontology based storage such that it describes the domain of benefits and represents an agreed-upon impression of the domain's settings. These data must be encrypted to make it more secure and robust against threats and other phishing activities as the flexible distributed storage requires integrity and auditing mechanism for secure and dependable storage.

Querying mechanisms are usually associated with a scoring mechanism and ambiguity resolver to identify the fastness and accuracy of query retrieval. Keyword search for querying basically proves to be an efficient mechanism for text based searches as it eases the users by providing with faster,

accurate and relevant search results in spite of no prior domain knowledge. Data are dynamic in nature and hence the design techniques must support secure and efficient dynamic operations on the distributed storage, including block modification, data deletion, and data addition. Efficient mechanisms must be in place to avoid data loss with dynamic updation techniques and also to achieve quality and integrity. This paper discusses the tools and techniques proposed earlier for ontological storage, maintenance of documents, quick document retrieval and document update mechanisms.

2, Ontological Based Storage – Tools and Techniques

2.1. Service Oriented Definition for Enterprise Based Systemic Ontology

2.1.1 Work On Focus:

The enterprise ontology methodology of handling of heterogeneities holistic view provider's intent the notion of ―Service‖, which specifies the occurring between elements that shall interoperate by resolving ontology terminologies. The purpose of the Service specification Business Process Execution Language was specifying enterprise services to continue work on the business process language published in the organizations to quickly connect their information systems and the notion of orchestration.

In addition to the theoretical results and deployment scenarios a number of service-orientation have been developed based on Ψ theory, as proposed by Linda and Antonia [1] and are currently under development for a mortgage based Enterprise Ontology. This theory consists of an operation axiom, to identify various actions performed in an organization, followed by the transaction axiom to identify the co-ordination various actions. The distinction axiom focuses on the capabilities of the human performing the above actions and finally the organization theorem provides an ontological model for the enterprise with the help of outcomes from the above specified axioms. This way a basis for the notion of service is formed.

Then the generic service specific framework is built upon the axioms to concentrate on three areas such as service executor, service production, and service coordination to arrive at enterprise ontology. Hence an execution environment for dynamic matchmaking, selection, and invocation of Systemic Ontology eradicates the Standard transaction pattern of an enterprise as may rightly be called as its ontological model.

2.1.2 Merits:

An ontological act was designed to provide the maximum expressiveness possible while retaining computational completeness & decidability. To enthuse ontological transaction (B-transaction) practitioners about the potential users, the Semantic Web and ontology web technologies provides a match of supply and demand for service, between the requesting parties.

To outline the benefits of applying ontology-based approach to service specification technologies in contexts distinguish between communication acts and production acts on the data logical, info logical and ontological level as described in the organization Theorem.

To encourage the collaboration between the Ontological human service and the Info logical human service, the service specification framework is introduced.

2.1.3 Demerits:

The Generic Service Specification Framework on ontology-service doesn't independently applying this service as a part of a transaction rather than a whole transaction. The platforms that they will ultimately be deployed on, will never recall the service definition.

Specifying them and enabling generation of the software artifacts specific to those platforms of generic service specification framework wont supporting platform selection. Coordination facts enabling transformation of system specifications into specifications for particular deployment will never focus on the production act to be performed by the executor.

2.2. Clustering of Ontological Proposals

2.2.1 Work On Focus:

Ontology is viewed a storage space for concepts and its relationships from a specific domain. The Ontology-Based Text-Mining (OTM) describes integrated framework in collaboration with statistical method and optimization models, to learn the target ontology from the text documents of peer clusters. Research proves that it is rich information based text mining framework.

Current methods use text mining to learn the target ontology from prerequisite knowledge sources, which are agreed-upon conceptualization of the domain's ―real-world‖ setting. Structured optimization and statistical analysis of research project selection retrieves ontology containing the projects using a sorting algorithm in a straightforward way.

The Ontology-Based Text-Mining revolves around four phases. The first phase concentrates on building a tree structure for different existing disciplines, in the form of research ontology. The

second phase focuses on creating new proposals for the sorted disciplines identified in the previous phase. Then the research proposals are clustered using text mining techniques such as document collection, preprocessing, encoding, dimension reduction and vector clustering at the third phase. Finally the clusters are separated into sub groups based on the disciplines' characteristics as proposed by Jian, Wei, Yong-hong, Efraim, Shouyang and Ou [2].

It has been able to show that indeed the hybrid method for grouping improves text clustering efficiency and effectiveness of the research project selection process. In addition, we could show that the ontology based approaches further improve the results achieved by proposed Ontology-Based Text-Mining.

2.2.2 Merits:

To gain advantages of the ontological mechanism, the proposed approach of integrated ontology learning based on their similarities and text mining framework is in place. The feature extraction components are carefully designed to allow flexible connections to different ontology containing projects that are capable to perform aforementioned problems.

For this purpose, it employs text mining techniques such as term clustering and domain ontology lexico-syntactic patterns as well as other resources like text document encoding, vector dimension reduction, and text vector clustering.

2.2.3 Demerits:

The proposed text mining methodology does not exhibit sufficient descriptive content to enable the self-organized mapping (SOM) algorithm which will not uniquely determine the entity referred to the global context of data mining. To date, the work on integrating semantic background knowledge into text categorization affiliations in a group is quite scattered and it was fairly diversified. The experiments reported here are not implied as a particular proposals and reviewers combined with the term-concept of text-mining classification.

Empirical comparison does not provide the results of manual classification to text-mining classification reviewers which will latter being expanded to help in finding anonymous similarities and then to balance them automatically.

2.3. Conceptual Web Service Ontology

2.3.1 Work On Focus:

Ontology is a conceptual model for relevant aspects related to Web services discovery Services. It provides ontology based bootstrapping framework, which supports the deployment and interoperability of service-oriented relationship modeling. This paper outlines some of the main issues related to the semantic modeling of definition of ontologies and provides an overview of the services interfaces as presented by Aviv and Quan [3].

The design principles of this bootstrapping ontology are highlighted and a short description of the top-level elements to validate the concepts using the service free text descriptor results against existing ontologies. The model starts with token extraction for each service from Web Service Description Language (WSDL) document, followed by the Term Frequency/Inverse Document Frequency (TF/IDF) identification, a methodology for Information Retrieval (IR) for the keywords in the identified token. Then, the web context extraction process is performed to build a context descriptor for each web service, such that it consists of words, phrase or any term from the identified web service.

The web page clustering algorithm is utilized to arrive at the above context. These contexts are analyzed and ranked using context recognition algorithm. Context evocation and ontology evolution follows then to filter and refine the selected contexts to build a new concept and its relationships and relationship types. This conceptual model summarized in this paper represents the foundation for metadata of services interfaces from the viewpoint of the Web Service Modeling Ontology Working Group.

2.3.2 Merits:

The objective of Ontological bootstrapping is to enable distributed computation over the Internet by automated and dynamic discovery, composition, and execution of services, thus providing a new technology for web-based system engineering. More specifically, through the use of exhaustive semantic description frameworks WSDL descriptor will support the provision of intelligent mechanisms for the execution of existing ontology Services.

This specification of bootstrapping ontology extends this framework, and develops a formal ontology and language, consists of four different main elements for describing semantic web search queries.

Further standardization in the area of ontology evolution works toward a common architecture and platform for ontology concepts.

2.3.3 Demerits:

Ontology-Based - Ontologies are not efficient enough to use the data model throughout Free Text Description Verification. The interchanged service usage of ontologies fails to claims at a widely accepted state-of-the-art knowledge representation enabling technology for the ontology construction.

The extensive usage of ontologies will not always allows semantically enhanced information processing as well as support for interoperability. This approach proclaims to fail to facilitate the automatic building of an ontology, which recognizes the importance of successful deployment of Web Services by making component of the framework of ontology. In order to provide a concise for semantic descriptions, the latter is concerned with the support of existing and emerging execution technologies for the assistance to the ontology construction process.

2.4. The Eight Challenges of Ontology Matching

2.4.1 Work On Focus:

Historically, the need for ontology is viewed in detail with the advent of semantic heterogeneity arose to integrate heterogeneous databases, by addressing important challenges for ontology matching and thus each having their own data ontology matching. Ontology schema matching has generally been developed to operate on database schema for data translation taxonomies and this alignment is a set of correspondences entities belonging to the matched ontologies. Formally, the results of recent ontology are quadruple basics of ontology schema matching, which specifies the similarity degree of subsumption ontology alignment mapping as discussed by Pavel and Jerome [4].

The focal problem of managing heterogeneity in terms of Ontology matching has unique experiences and thus the different in emphasis on the ontology merging will obscure entities of the ontology structures. The ontology matching process is challenged with various options like Large-Scale Matching Evaluation, Efficiency Of Matching Techniques, Matching With Background Knowledge, Matching With Background Knowledge, User Involvement, Explanation Of Matching Results, Social And Collaborative Matching, and Alignment Management: Infrastructure And Support. The above processes are dependent on each other and help in accelerating the work progress of ontology matching.

2.4.2 Merits:

Increasingly, there is a recognized need for secure information sharing, and the implementation of ontology alignment information between diverse organizations and privacy preserving interoperation systems was well defined. In this scenario, the privacy of the metadata ontologies of information sources or the schema of databases must be preserved from semantic web and artificial intelligence as well as from databases in order to ensure interpreting an alignment of ontology matching.

The interoperation system does not assume a trusted mediator. But ideally, the State of the Art and Future Challenges of ontology wants the mediator to gain minimal information about the data and the metadata stored in its information sources. Several ontology matching algorithms use dictionaries / thesauri of documents to identify matching concepts.

2.4.3 Demerits:

Instance-based matching algorithms that are opaque work without any modifications of ontology semantics doesn't have the statistical properties, like distribution, frequency, entropy, mutual information of the instance ontology alignment.

It assumes that the process of ontology matching can be totally automated rather than mentioning the process outlined above of the ontology matcher which has access only to encrypted ontologies and cannot decrypt the proposed ontologies.

In cases, where the diverse solutions for matching do not want to share their ontologies even with the organization that they generally tends to share possibilities of information with such arrangement, which is not acceptable.

However, to the best of our knowledge, there exists no prior research that shows how privacy-preserving ontology matching can be enabled, since our interoperation architectures have not been influenced to the core of ontology alignment schema match.

2.5. Object Model Interface for Ontological Database

2.5.1 Work On Focus:

This paper discusses on an ontology-Driven Information System reconfiguration that uses ontological knowledge of relational database for the purpose of intervention procedures without failure/structure intervention of ontology implementation.

The research presented in this paper by Petr and Zdenek is a follow-up of the prior work involving the development of semantic web ontologies to support designers at the conceptual methodology for designing ontology-backed software applications [5]. A tire design result of ontology evolution is presented to demonstrate the proposed approach. The ontology-based method described in this paper can help retrieve and save the complex relations 0f incompatible domain-dependent user scenarios of driven ontology system.

In order to locate the proper information and query, the data from the databases, and the relationship between the ontology and databases, are built to offer users to implement Web Ontology Language at a multiple design results of system specific requirements. This web ontology builds an interface between ontology and the application in such a way that the object model of the interface contains the contract information and the platform-dependent control logic information. The above interface is tested against compile-time, run-time and reasoning time constraints to verify that it contains contract stability and maintainability, full length info about the interface and perfect validation to ensure non-violation of application consistency.

2.5.2 Merits:

In this study, an ontology-based system is proposed to solve problems raised in the ―proof-of-concept‖ application of ontology by expanding traditional development activity. A prototype system of original assumptions is developed to verify the proposed approach, using the semantic Web Ontology Language.

For seamless sharing domain of specific knowledge, the ontological multidisciplinary and intelligent technique that solves the problems identified in the OWL ontologies was depicted spontaneously.

In order to locate the proper information and ontology enrichment results the cause of relation in composition of optimizing the ontologies knowledge engineering was dealt with integrity constraints that are necessary to understand our proposal for a persistence layer.

2.5.3 Demerits:

Our configuration agent always minimizes the overheads of the current reconfiguration process by automating it and infers facts about the main-stream technology for data modeling from the ontological knowledge model and then decides whether the current environment can support the given manufacturing requirements.

The ontology-based method described in this paper cannot help to retrieve and save the complex relations by supporting the reasoning & integrate heterogeneous data resources, which will not offer users more accurate, proper and comprehensive data of ontology-based information system (OIS) design. A framework for such a knowledge-based optimization support system is not properly implemented and the practical approaches to access knowledge contained in OWL ontology programmatically have been introduced so far as a good overview of current approaches.

3, Secure Ranked Keyword Search – Tools and Techniques

3.1. Keyword Match Clustering on XML Documents

3.1.1 Work On Focus:

Searching documents with keywords returns faster results. This methodology, which has been recently introduced, returns a group of matching results from the related XML documents.

Two different approaches are discussed in this paper. The first method fetches the related documents based on the search query and then clusters the results. The second method performs clustering as and when the results are fetched. These clusters are finally arranged hierarchically based on different dimensions and then presented to the user for analysis of the result.

The above methods basically returns based on conceptual relationships among the entity nodes as the nodes are generally very informative in an XML. Then the results are clustered using ―Keyword Matching Pattern‖ (KMP). Then an active clustering algorithm is implemented for inferring and evaluating the KMPs. Furthermore, Xiping, Changxuan and Lei propose a ―Relaxation-based clustering algorithm‖ is followed to further cluster the KMPs as the result fetched from the active clustering is still huge enough for the input query [6]. Thus when evaluated experimentally, the clustering method proves to be an effective plan for ambiguous queries on large datasets.

3.1.2 Merits:

Clustered keyword search provides a conceptual view of the XML Documents in terms of semantic relationship between the XML nodes and hence very accurate results are returned.

Clustering the results helps the end user to easily identify their individual interest of their search query. Also this clustering methodology performs pattern matching with keywords that helps in active clustering.

Hierarchical arrangement of the results helps the end user to easily identify and analyze his general / in-specific view of the required data.

3.1.3 Demerits:

Understanding and identifying the relationship between nodes, while searching, becomes tedious when the number of nodes increases. Hence this implies that more and more formal analysis of XML documents is required to be performed using ER model.

Also for each XML document, the candidate entities must be generated separately. But due to space constraint, a set of sample candidate entities is used for all XML documents. Also many irrelevant results are returned in the keyword pattern matching search. Hence the clustering process becomes a tedious task as the number of result is high.

3.2 Index-Based Fuzzy Search with SQL

3.2.1 Work On Focus:

Search as you type is an active methodology that fetches results as and when the user types the character of a word. This method proposes the use of auxiliary indexes stored in the form of tables to assist in faster result retrieval.

Fuzzy novel search methodology is implemented with auxiliary indexes to perform single keyword and multi keyword search using SQL. The proposed methodology, incremental-computation technique, is tested with large and real datasets to enable database management system to retrieve results as you type. It helps to avoid session storage in applications, as the data is stored in database and helps is future search refinements.

Guoliang, Jianhua and Chen discuss [7] on the prefix search as an index-based method that contains inverted-index table and prefix table for storing indexes and prefixes of the keywords. The prefix of the search query keyword is matched using character-level incremental computation and word-level

incremental computation to return first N results to the user. The performance is enhanced by the usage of indexes at the same time providing incremental updates as well to the data. The support for fuzzy search is extended from single-keyword queries to multi-keyword queries and is experimentally evaluated with large real time datasets.

3.2.2 Merits:

The proposed method proves to leverage existing functionalities with high performance. Hence query execution proves to be efficient. Upon testing the technique with large database, better and faster results are returned.

This way it outperforms the existing methods available in any database tool. Also since the search is initiated as and when the user types, it helps in better system interaction. This intellisense methodology thus retrieves faster results and proves to be efficient and effective.

3.2.3 Demerits:

There is no proper support for ranking queries. It works perfectly for searching with a single table, but not more efficient with multiple tables. Also the indexing techniques work well with limited data and require much less work on index inversions. As and when the data size increases, the complexity and the processing time as well increases.

Keyword Search with Relational Databases Tuple

3.3.1 Work On Focus:

Keyword search is usually applied on a single table in a database or on multiple tables. Our problem definition is that the keyword search is to be implemented on a connected database tuples.

Our proposed approach traverses in defining a single-keyword-based structure-aware-index and keyword-pair based structure-aware index on multiple related tuples to capture its relationships [8]. The tuple units are usually related using Primary Key – Foreign Key relationships that are represented using Boolean Adjacency Matrix (BAM) graphs. This way it builds a meaningful structure that aids in answering keyword queries.

The proposed tuple-unit-based methodology also implements structure aware keyword-pair based ranking process to arrive at high performance and high efficiency. Also score expectation is arrived at for a keyword with respect to the tuple unit.

3.3.2 Merits:

The integration multiple related tuples of database helps achieving a really efficient output on keyword search. The storage of structural relationship of the database tuples and also the usage of indexes provides faster and accurate search results. The ranking that is implemented on account of tuple's relational structure along with textual data proves to be an effective mechanism.

3.3.3 Demerits:

Even though the mechanism is efficient, defining indexes, identifying relevancy between keywords and score expectation table generation are considered to be time consuming and costly operations.

4, Novel Methodology of Document Updation – Tools and Techniques

4.1 Public Auditing and Secure Data Updation

4.1.1 Work On Focus:

Data storage on a common server accessed by all users needs frequent auditing for maintaining data consistency. Delegating the maintenance work to TPA reduces the burden of data owner. In addition public auditability or dynamic data operations also helps ensuring remote data integrity.

Qian, Cong, Kui, Wenjing and Jin [9] analyzed and identified that first the data needs to be analyzed for potential security problems and an effective verification scheme is proposed as part of the Merkle Hash Tree construction for block tag authentication. It is an efficient authentication scheme to check the data's strength, that is, to verify whether the data is not modified or corrupted.

The technique of bilinear aggregate signature is used to extend the main result into a multiuser setting.

4.1.2 Merits:

The operation of public auditing is fully a dynamic data operation and hence the task is completed faster and scalable.

This helps multiple users performing different task on the same document, to perform an efficient batch processing to maintain data integrity.

The signature based scheme ensures that each user's update on the document is provided to Third Party Auditor and hence there is a high level of data integrity and data security.

Hence the public auditability, dynamic data operations and blockless verification proves to be highly efficient and provably secure.

4.1.3 Demerits:

Even though the public auditing supports dynamic data operation, the process of document maintenance becomes time consuming as more and more users have access and each user is allowed to do modifications on the same block of data at the same time.

During dynamic operations considerable number of original data blocks should be retrieved to ensure a reasonable detection probability, which again could result in a large communication overhead and greatly affects system efficiency.

4.2. Automated Principled Approach for Dependable Storage Solutions

4.2.1 Work On Focus:

Data loss in business environment is prevented by using various combinations of techniques such as mirroring, backups etc. Choosing the right solution can only end up with a cost efficient way.

Shravan, Kimberly, Arif and William [10] proposed a principled approach to provide data storage management solution to various shared and distributed applications. The proposed mechanism, principled automatic approach, helps in efficient data storage, better communications and high level of cost reduction. The approach involves two steps, design solver for choosing the best design graph for an application and the configuration solver process to propose the best configuration parameters for the chosen design to arrive at an optimal solution.

Initially the design solver process follows a greedy best fit algorithm to build a storage solution for assigning workloads to applications, followed by the refit algorithm to find the best node in the design graph. Then the reconfiguration algorithm is initiated to check the quality of the design graph and identify the most participating applications and to reconfigure their design graph. The configuration solver process now comes into picture to identify the data loss time and the recovery time and hence provides the ideal re-configuration parameters to optimize the design solver process.

4.2.2 Merits:

The proposed approach provides a combination of data protection and recovery techniques for each application. Also the quantitative configuration parameters associated with each data protection technique proves to be effective.

The device resources supports normal and recovery operation and the mapping of primary and secondary data copies onto the provisioned resource instances are achieved quickly as per this qualitative parameter decisions. Experimental evaluation of the proposed approach shows it covers a wide range of decision parameters to arrive at the best solution for creating an effective storage system at an expected cost.

4.2.3 Demerits:

The characteristics of the storage design problem make it difficult to apply simulation optimization techniques. Very little literature exists to optimize systems with qualitative decision variables, as the model being optimized is fairly complicated.

Hence the optimization strategy and evaluation algorithms must be tailored for the class of problems addressed in this paper.

CONCLUSION

As per the above analysis we arrive upon a conclusion that ontological data storage is an efficient data storage mechanism and Ranked Search provides a perfect searching mechanism. The keywords associated with the ranking mechanism helps in quick document retrieval process while providing dynamic data support through smart verification and updation techniques.

The dynamic data model comes with benefits such as fast localization support, lightweight computation cost and storage accuracy. Hence an Integrated solution for document search engines in Ontological Perspective is Cost Effective with Efficient Results and High Level of Security.

REFERENCES

[1] An Enterprise Ontology-Based Approach to Service Specification Linda I. Terlouw and Antonia Albani(IEEE Transactions On Services Computing, VOL. 6, NO. 1, JANUARY-MARCH 2013)

[2] An Ontology-Based Text-Mining Method to Cluster Proposals for Research Project Selection

Jian Ma, Wei Xu, Yong-hong Sun, Efraim Turban, Shouyang Wang, and Ou Liu(IEEE Transactions On Systems, Man, And Cybernetics—Part A: Systems And Humans, VOL. 42, NO. 3, MAY 2012)

[3] Bootstrapping Ontologies for Web Services Aviv Segev and Quan Z. Sheng(IEEE Transactions On Services Computing, VOL. 5, NO. 1, JANUARY-MARCH 2012)

[4] Ontology Matching: State of the Art and Future Challenges Pavel Shvaiko and Je´ roˆme Euzenat(IEEE Transactions On Knowledge And Data Engineering, VOL. 25, NO. 1, JANUARY 2013)

[5] Ontology-Driven Information System Design, Petr Kˇremen and Zdenˇek Kouba(IEEE Transactions On Systems, Man, And Cybernetics—Part C: Applications And Reviews, VOL. 42, NO. 3, MAY 2012)

[6] Returning Clustered Results for Keyword Search on XML Documents Xiping Liu, Changxuan Wan and Lei Chen(IEEE Transactions On Knowledge And Data Engineering, VOL. 23, NO. 12, DECEMBER 2011)

[7] Supporting Search-As-You-Type Using SQL in Databases Guoliang Li, Jianhua Feng and Chen Li (IEEE Transactions On Knowledge And Data Engineering, VOL. 25, NO. 2, FEBRUARY 2013)

[8] Finding Top-k Answers in Keyword Search over Relational Databases Using Tuple Units Jianhua Feng, Guoliang Li and Jianyong Wang IEEE Transactions On Knowledge And Data Engineering, VOL. 23, NO. 12, DECEMBER 2011

[9] Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing Qian Wang, Cong Wang, Kui Ren, Wenjing Lou and Jin Li(IEEE Transactions On Parallel And Distributed Systems, VOL. 22, NO. 5, MAY 2011)

[10] Designing Dependable Storage Solutions for Shared Application Environments Shravan Gaonkar, Kimberly Keeton, Arif Merchant and William H. Sanders (IEEE Transactions On Dependable And Secure Computing, VOL. 7, NO. 4, OCTOBER- DECEMBER 2010)

BIOGRAPHY

J.S.Beulah ,Assistant Professor from D.G.Vaishnav college, Arumbakkam, Chennai-106 and

Dr.Mary Metilda Asst Professor from Queen Mary's College, Chennai-4