Imperial Journal of Interdisciplinary Research (IJIR)
Vol-3, Issue-9, 2017
ISSN: 2454-1362, http://www.onlinejournal.in

An Advanced Inference for Scalable Distributed Reasoning Using MapReduce

Tejashree Shinde[1] & Prof. G. S. Deokate[2]
[1]Student, ME Computer, SPCOE, Department Of Computer Engineering, Otur
[2]Assistant Professor, SPCOE, Department Of Computer Engineering, Otur

Abstract: *Reasoning on a web scale becomes increasingly challenging because of the large volume of data involved and the complexity of the task by means of ontology mapping. Ontology mapping processes users queries that can provide more correct results when the mapping process can deal with the uncertainty effect that is caused by the incomplete and inconsistent information used and produced by the mapping process. Here, an IDIM concept is used to deal with large-scale incremental RDF datasets. Resource Description Framework (RDF) is an important data, presenting standard of the semantic web to process the increasing RDF data. MapReduce is a widely-used parallel programming model that can be used to represent uncertain similarities created by both syntactic and semantic similarity algorithms. The proposed One-Class Clustering Tree (OCCT) characterizes the entities that should be linked together. The construction of TIF and EAT significantly reduces the re-computation time for the incremental inference as well as the storage for RDF triples. Therefore, users can execute their query more efficiently without computing and searching over the entire RDF closure used in the prior work. The final results are evaluated by comparing it against benchmark models in web information gathering.*

Keywords: *Ontology; Resource Description Framework (RDF); One-Class Clustering Tree (OCCT); MapReduce.*

1. Introduction

Semantic reasoning of data on a Web scale becomes increasingly challenging because of the large volume of data involved that raises the complexity of the task. Ontology mapping in the context of Question Answering can provide more correct results if the mapping process can deal with unreliability that is caused by the incomplete and inconsistent information used and produced by the mapping process.
In the year of 2009, the semantic web [2] contain 4.4 billion triples and has now reached over 20 billion triples. Its growth rate is still increasing. As it has evolved into a global knowledge-based framework to promise a kind of machine intelligence, supporting knowledge searching over such a big and increasing dataset has become an important issue.
Resource Description Framework (RDF) is an important data representation standard used to describe knowledge in the semantic web. Deriving inferences in the large-scale RDF [1] files, referred to as large-scale reasoning, poses challenges in three aspects:

1. Distributed data on the web make it difficult to acquire appropriate triples for appropriate inferences.
2. The growing amount of information requires scalable computation capabilities for large datasets.
3. Fast processing for inferences is required to satisfy the requirements of online query.

Due to the performance limitation of a centralized architecture executed on a single machine or local server when dealing with large datasets, distributed reasoning approaches executed on multiple computing nodes have thus emerged to improve the scalability and speed of inferences. But this consumes too much of time and space for reasoning. The concept of an incremental and distributed inference method (IDIM) for large-scale RDF datasets via MapReduce overcomes these issues.
MapReduce can provide a solution for large scale RDF data processing which is a widely-used parallel programming model. It presents a novel approach can be used to represent uncertain similarities created by both syntactic and semantic similarity algorithms. The choice of MapReduce is motivated by the fact that it can limit data exchange and alleviate load balancing problems by dynamically scheduling jobs on computing nodes [5]. In order to store the incremental RDF triples more efficiently, two novel concepts, transfer inference forest (TIF) and effective assertional triples (EAT) are used. Their use can largely reduce the storage and simplify the reasoning process. Based on TIF/EAT, we need not compute and store RDF closure and the reasoning time, so significantly decreases that a user's online query can be answered timely, which is more efficient than existing methods to our best knowledge. More importantly, the update of TIF/EAT needs only minimum computation since the relationship between new triples and existing ones is fully used.

Objectives:

1. It can limit data exchange and alleviate load balancing problems.
2. It can well leverage the old and new data to minimize the updating time and reduce the time when facing big RDF datasets

2. Related Work

Generally a large-scale image search system consists of two key components, an effective image feature representation and an efficient search mechanism. It is well known that the quality of search results relies heavily on the representation power of image features. The latter, an efficient search mechanism, is critical since existing image featctivesures are mostly of high dimensions and current image databases are huge, on top of which exhaustively comparing a query with every database sample is computationally prohibitive. we represent images using the popular bag-of-visual- words (BoW) framework , where local invariant image descriptors (e.g., SIFT [3]) are extracted and quantized based on a set of visual words. The BoW features are then embedded into compact hash codes for efficient search. For this, we consider state-of-the-art techniques including semi-supervised hashing and semantic hashing with deep belief networks. Hashing is preferable over tree-based indexing structures (e.g., kd-tree as it generally requires greatly reduced memory and also works better for high-dimensional samples) with the hash codes, image similarity can be efficiently measured. In hamming space by hamming distance, an integer value obtained by counting the number of bits at which the binary values are different.

The sheer amount of Web pages and the exponential growth of the Web suggest that users are becoming more and more dependent on the search engines' ranking methods to discover information relevant to their needs. Typically, users expect to find such information in the top-ranked results, and more often than not they only look at the document snippets in the first few result pages and then they give up or reformulate the query. This can introduce a significant bias to their information finding process and calls for ranking methods that take into account not only the overall page quality and relevance to the query, but also the match with the users' real search intent when they formulate the query. It generally requires greatly reduced memory and also works better for high-dimensional samples. With the hash codes, image similarity can be efficiently measured (using logical XOR operations) in Hamming space by Hamming distance, an integer value obtained by counting the number of bits at which the binary values are different. It have become the cause of precision rate decrease on simple matching of tags to a given query. A common practice to improve search performance is to re-rank the visual documents returned from a search engine using a larger and richer set of features. The ultimate goal is to seek consensus from various features for reordering the documents and large scale applications, the dimension of Hamming space is usually set as a small number (e.g., less than a hundred) to reduce memory cost and avoid low recall. Most photo images stored on the Web have lots of tags added with user's subjective judgments not by the importance of them. So, in tagged web image retrieval, they boosting the retrieval precision. In previous works for image search Re-ranking suffers from the unreliability of the assumptions under which the initial text-based image search result is employed in the Re-ranking Process.

The main contributions of this desertation are summarized as follows.

1) We propose a novel representation method TIF/EAT to support incremental inference over large-scale RDF datasets which can efficiently reduce the storage requirement and simplify the reasoning process.

2) An efficient and scalable reasoning method called IDIM is presented based on TIF/EAT, and the corresponding searching strategy is given to satisfy end-users' online query needs.

3) We have implemented a prototype by using the Hadoop platform. It allows one to perform experiments of different methods on billion triples challenge (BTC) benchmark data. A real-world application on healthcare domain is also presented to validate the effectiveness of our method.

Existing System:

In Existing system, the proposed concept of an incremental and distributed inference method for large-scale ontologies by using MapReduce realizes high-performance reasoning and runtime searching, especially for incremental knowledge base [8]. By constructing, using novel concepts of transfer inference forest and effective assertional triples, the storage is largely reduced and the reasoning process is simplified and accelerated to satisfy end-users' online query needs. The processing was made via MapReduce, which is motivated by the fact that it can limit data exchange and alleviate load balancing problems by dynamically scheduling jobs on computing nodes.

Drawbacks of Existing System are as follows,

1. The Query time for IDIM is affected when the incremental triples affect the structure of the inference forests. If an RDF dataset has few ontological triples, the size of constructing dataset TIF is also small [7].

2. The changes in the structure of TIF affect the performance improvement with ontological triples. The advantages of TIF/EAT cannot be exploited well, if the size of the tree is small.

3. Proposed Work

In order to overcome the existing drawbacks, the data clustering method is used in this paper that makes the processing of data more efficiently by means of linking the data sets.

A. One-Class Clustering Tree (OCCT)

A clustering tree is a tree in which each of the leaves contains a cluster instead of a single classification. Each cluster is generalized by a set of rules that is stored in the appropriate leaf. This data linkage method aimed at performing one-to-many linkage [4],[5]. The data linkage is performed among entities of different types.

For example, in a student database, we might want to link a student record with the courses she should take. It is done according to different features which describe the student and features describing the courses.

The OCCT [9],[11] was evaluated using datasets from three different domains. They are,

Data leakage prevention
Recommender systems
Fraud detection.

In the data leakage prevention domain, the goal is to detect abnormal access to database records that might indicate a potential data leakage or data misuse. The goal is to match an action, performed by a user within a specific context, with records that can be legitimately retrieved within that context. In the recommender systems domain the proposed method is used for matching new users of the system with the items that they are expected to like based on their demographic attributes.

In the fraud detection domain, the goal is to identify online purchase transactions that are executed by a fraudulent user and not the legitimate user.

The results show that the OCCT performs well in different linkage scenarios. In addition, it performs at least as accurate as the well known as decision tree data-linkage model, while incorporating the advantages of a one class solution. Additionally, the OCCT is preferable over the decision tree because it can easily be translated to linkage rules.

System Flow:

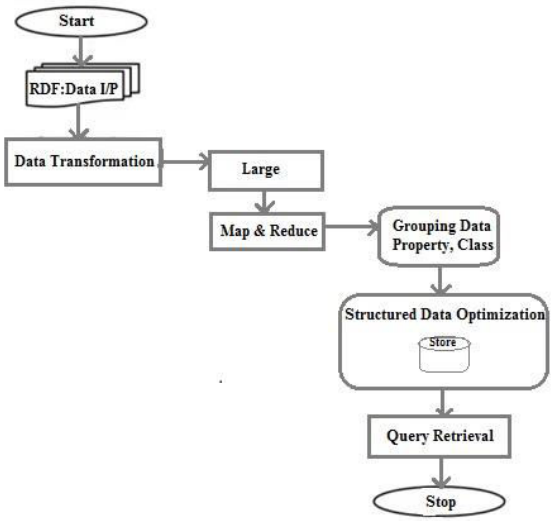

Figure 1. System Flow

B. Algorithm: Maximum Likelihood Estimation

Maximum Likelihood Estimation (MLE) splitting criterion used in order to choose the attribute that is most appropriate to serve as the next splitting attribute. Each candidate attributes from the set of attributes splits the node data set into subsets according to its possible values. For each of the subsets, a set of probabilistic models is created, one for each attribute of second dataset. Each probabilistic model is built to describe the probability given. In order to create the probabilistic models decision tree are used. Each of these trees represents the probability of its class attribute values given the values of all other attributes [6].

Once the set of models has been induced, the probability of each record given these models is calculated. A subset's score is calculated as the sum of all scores of the records belonging to it. The attribute's final score is determined by the sum of the subset's individual scores. The goal is to choose the split that achieves the maximal likelihood and therefore we choose the attribute with the highest likelihood score as the next splitting attribute in the tree. The computational complexity of building a decision model using the MLE method is dependent on the complexity of building a statistical model and the time it takes to calculate the likelihood.

System Architecture:

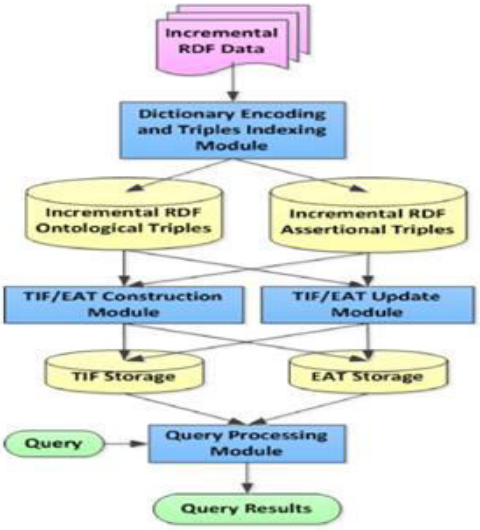

Figure 2. System Architecture

Advantages of proposed system:

1. The OCCT model is better generalized and avoids over-fitting by means of pruning the data.
2. Fraud detection is used to obtain the genuine matching data for legitimate users to access [10].
3. Maximum Likelihood Estimation can handle multiple ways of splitting the data entities.
4. It is easy and quick method to compare the datasets by obtaining the matching entities [12].

4. System Specification and System Requirements

A. HARDWARE REQUIREMENTS:

- Processor - Pentium -IV
- RAM - 256 MB(min)
- Hard Disk - 20 GB
- Input Device: Standard Keyboard and Mouse.
- Output Device: VGA and High Resolution Monitor

B. SOFTWARE REQUIREMENTS:

- Operating System: Windows95/98/2000/XP/7.
- Software : Eclipse
- Platform : Hadoop, Java, MySQL.

5. Conclusion

With the upcoming data deluge of semantic data, the fast growth of ontology bases has brought significant challenges in performing efficient and scalable reasoning. Mapping process can deal with the uncertainty effect that is caused by the incomplete and inconsistent information used and produced by it for processing users' queries that can provide more correct results. MapReduce represents uncertain similarities created by both syntactic and semantic similarity algorithms. OCCT characterizes the entities that should be linked together using the splitting criterion of MLE. TIF and EAT construction significantly reduces the re-computation time for the incremental inference as well as the storage for RDF triples. Therefore, users can execute their query more efficiently without computing and searching over the entire RDF closure.

6. Future Enhancement:

In the future, the methods can validate for more datasets, such as other benchmarks and other types of datasets and also can be done in other ontology languages [9] that make the processing of data to the user's request in a highly efficient manner.

7. Acknowledgement

I express my sincere thanks to my project guide Prof. G. S. Deokate who always being with presence & constant, constructive criticism to made this paper. I would also like to thank all the staff of computer department for their valuable guidance, suggestion and support through the paper work, who has given co-operation for the project with personal attention. Above all I express our deepest gratitude to all of them for their kind-hearted support which helped us a lot during paper work.

8. References

[1] N M. S. Marshall *et al*., "Emerging practices for mapping and linking life sciences data using RDF", A case series,‖ *J. Web Semantics*, vol. 14, pp. 2–13, 2012.

[2] J. Guo, L. Xu, Z. Gong, C.-P.Che and S. S. Chaudhry, "Semantic inference on heterogeneous e-marketplace activities", *IEEE Trans. Syst.,Man, Cybern. A, Syst., Humans*, vol. 42, no. 2, pp. 316–330, Mar. 2012.

[3] J. Urbani, S. Kotoulas, J. Maassen, F. V. Harmelen and H. Bal, "WebPIE: A web-scale parallel inference engine using mapreduce", *J. Web semantics*, vol. 10, pp. 59–75, Jan 2012.

[4] J. Urbani, S. Kotoulas, E. Oren, and F. Harmelen, "Scalable distributed reasoning using mapreduce", in *Proc. 8th Int. Semantic Web Conf.,* Chantilly, VA, USA, pp. 634–649, Oct. 2009.

[5] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters", *Commun. ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[6] C. Anagnostopoulos and S. Hadjiefthymiades, "Advanced inference in situation-aware computing", *IEEE Trans. Syst., Man, Cybern. A, Syst.,Humans*, vol. 39, no. 5, pp. 1108–1115, Sept. 2009.

[7] H. Paulheim and C. Bizer, "Type inference on noisy RDF data", in *Proc. ISWC*, Sydney, NSW, Australia, pp. 510–525, 2013.

[8] G. Antoniou and A. Bikakis, "DR-Prolog: A system for defeasible reasoning with rules and ontologies on the Semantic Web", *IEEE Trans.Knowl. Data Eng.,* vol. 19, no. 2, pp. 233–245, Feb. 2007.

[9] D. Lopez, J. M. Sempere, and P. García, "Inference of reversible tree languages", *IEEE Trans. Syst.,Man, Cybern. B, Cybern.,*vol. 34, no. 4, pp. 1658–1665, Aug. 2004.

[10] A. Schlicht and H. Stuckenschmidt, "MapResolve", in *Proc. 5th Int. Conf. RR,* Galway, Ireland, pp. 294–299, Aug. 2011.

[11] Ma'ayan Dror and Asaf Shabtai, "OCCT: A One-Class Clustering Tree for One-to-many Data linkage", *IEEE trans. on knowledge and data engineering*, tkde-2011-09-0577, 2013.

[12] B. C. Grau, C. Halaschek-Wiener and Y. Kazakov, "History matters: Incremental ontology reasoning using modules", in *Proc. ISWC/ASWC,* Busan, Korea, pp. 183–196, 2007.