Significant Big Data Interpretation using Map Reduce Paradigm

Lavanya Kakkirala
PG Scholar
Dept. of Computer
Science & Engg.
CMRCET, Hyderabad
Telangana,India

K .Venkateswara Rao
Associate Professor
Dept. Of Computer
Science & Engg
CMRCET,Hyderabad
Telangana,India

ABSTRACT
The development of ontologies involves continuous but relatively small modifications. Even after a number of changes, ontology and its previous versions usually share most of their axioms. For large and complex ontologies this may require a few minutes, or even a few hours. Cognitive on a Web scale becomes increasingly stimulating because of the large volume of data involved and the complexity of the task. Full re-reasoning over the entire dataset at every update is too time-consuming to be practical. Semantic information has been reduced by using Hadoop framework with simple machine learning algorithm. Each level of mapping and reducing is based on k-means clustering technique. Large set of information can be constructing or modified with the help of simple pattern based grouping. Dynamically grouping dependencies can be made based on attributes. Clustered values have got modifications like addition. At the end user query has been retrieved with the help of grouped items. The system has been assessed on the BTC benchmark and the results show that this method outperforms related ones in nearly all aspects.

General Terms
RDF (Resource Description Framework), RDFS (RDF schema), OWL (Web Ontology Language), SD (Structured Design), IDI (Incremental and Distributed Inference)

Keywords
Ontology, Hadoop, Semantic, Cognitive, Pattern, machine learning.

1. INTRODUCTION
The Semantic Web, as originally envisioned, is a system that enables machines to "understand" and respond to complex human requests based on their meaning. Such an "understanding" requires that the relevant information sources be semantically structured. By encouraging the inclusion of semantic content in web pages, the Semantic Web aims at converting the current web, dominated by unstructured and semi-structured documents into a "web of data". The main purpose of the Semantic Web is in driving the evolution of the current Web by enabling users to find, share, and combine information more easily. The semantic web is a vision of information that can be readily interpreted by machines, so machines can perform more of the tedious work involved in finding, combining, and acting upon information on the web. It uses metadata as well.

This model is inspired by the map and reduce functions commonly used in functional programming, although their purpose in the Map-Reduce framework is not the same as in their original forms. The key contributions of the Map-Reduce framework are not the actual map and reduce functions, but the scalability and fault-tolerance achieved for a variety of applications by optimizing the execution engine. As such, a single-threaded implementation of Map-Reduce (such as Mongo DB) will usually not be faster than a traditional (non-Map-Reduce) implementation; any gains are usually only seen with multi-threaded implementations. Only when the optimized distributed shuffle operation (which reduces network communication cost) and fault tolerance features of the Map-Reduce framework come into play, is the use of this model beneficial. Optimizing the communication cost is essential to a good Map-Reduce algorithm.

2. RELATED WORK

2.1 DR-Prolog: A System for Defeasible Reasoning with Rules and Ontologies on the Semantic Web
This presents an implemented defeasible reasoning system (DR-Prolog), which has been tested, evaluated and compared with existing similar implementations. Through the description of the system, process shows how user can combine the expressive power of a non-monotonic logic (defeasible logic) with the Semantic Web technologies (RDF(S), OWL, Rule-ML) to build applications for the logic and proof layers of the Semantic Web entirely describes reason for conflicts among rules that arise naturally on the Semantic Web. To address this problem, we proposed defeasible reasoning from the area of knowledge representation. The proposed system is Prolog-based, supports Rule-ML syntax, and can reason with monotonic and non-monotonic rules, RDF facts and RDFS and OWL ontologies.

Disadvantages

- DR-DEVICE uses the logic meta-program as a guiding principle, but there is no formal proof of the correctness of the implementation.
- To provide information on web process, assumed players will not be able to interfere due to communication problems and privacy or security concerns.
- A skeptical approach is sensible because it does not allow for contradictory conclusions to be drawn.
- It did not implement load/upload functionality in conjunction with an RDF repository.

2.2. Incremental Ontology Reasoning Using Modules

This is a technique proposed for incremental ontology reasoning—that is, reasoning that reuses the results obtained from previous computations. This is based on the notion of a module and can be applied to arbitrary queries against ontologies expressed in OWL DL. Here, it mainly focuses on a particular kind of modules that exhibit a set of compelling properties and apply the method to incremental classification of OWL DL ontologies. It did not depend on a particular reasoning method. For ontology development, it is desirable to re-classify the ontology after a small number of changes. In this scenario, the results are very promising. Incremental classification using modules is nearly real-time for almost all ontologies and therefore the reasoned could be working transparently to the user in the background without slowing down the editing of ontology.

Disadvantages

- Current reasoners did not reuse old results obtained which would lead to increase in the process time.
- Assuming axioms cannot follow some ontology principles.
- In contrast, this test is not applicable for the subsumption and cannot be directly used in high way ontological reasoners which are not tableaux-based.
- For complex ontologies such as the Wine ontology, the modules can be large.
- Large classification fragments might be more expensive than classifying the whole ontology.

2.3. Type Inference on Noisy RDF Data

An RDF knowledge base consists of an A-box, i.e., the definition of instances and the relations that hold between them, and a T-box, i.e., a schema or ontology. The SD-Type approach proposed exploits links between instances to infer their types using weighted voting. Assuming that certain relations occur only with particular types, they can heuristically assume that an instance should have certain types if it is connected to other instances through certain relations. For each property in a dataset, there is a characteristic distribution of types for both the subject and the object. Unlike traditional reasoning, this approach was capable of dealing with noisy data as well as faulty schemas or unforeseen usage of schemas. This process could be applied to virtually any cross-domain dataset.

Disadvantages

- It cannot be used for predicting missing types.
- It is often not feasible to manually assign types to all instances in a large knowledge base.
- Assumptions are not realistic for large and open knowledge bases.

2.4. Inference of Reversible Tree Languages

The inference of tree languages is related to the inference of context-free string languages using a structural sample, but the development of specific tree language learning algorithms should open new possibilities for the characterization of sub-classes of the context-free languages. The two classes of tree languages are characterized, some properties concerning these classes are proven, and they are also studied in relation to other well-known tree language classes.

Disadvantages

- Merging those states that do not fulfill the reversibility conditions.
- The characterization of new tree languages will not offer a way to learn new subclasses of context-free string languages.
- The development of tree language inference algorithms did not allow, in pattern recognition tasks,
- It did not give representation primitives to model the different classes of classification problems.

2.5. Parallel Materialization of the Finite RDFS Closure for Hundreds of Millions of Triples

This approach presents modern parallel computation techniques to compute the finite RDFS closure of large data sets. Previous work has used approximation to achieve higher scalability while other work focuses on minimizing dependencies in partitioning the work load. An ontological triple is a triple used in describing ontology and from which significant inferences can be derived. We have defined a scheme box partitioning and they have classes of rules which can be used to perform complete parallel inference on box partitions. Now, however the graphs have been partitioned into smaller graphs, and there is no guarantee that two blank nodes with the same label in different graphs are actually the same node and that all of the finite RDFS rules are box partitioning safe and have derived an embarrassingly parallel algorithm for producing the finite RDFS closure.

Disadvantages

- This is not a disk-based process, it is parallel and gives more load to user.
- Rule body will not match assertion triples.
- Production did not contribute to the inferencing of new triples.

3. PRELIMINARIES

3.1. RDF

Distributed reasoning methods focused on computing RDF closure for reasoning, which has taken too much time than normal and space. Web semantic work differentiates newly-arrived RDF triples and old ones but fails to reflect the relations between them at the end, resulting in a vast number of replicated triples throughout the reasoning thereby hindering its enactment. After that, two fuzzy implication engines were proposed based on the knowledge-representation model to enhance the context inference and classification for the well-specified information in Semantic Web. Some progress introduced a novel rule constrains approach that consisted of a concept parting policy and a semantic implication engine on a multiphase forward-chaining algorithm to solve the semantic inference problem in heterogeneous e-marketplace events. The problem of inference on deafening data and presented the SD-Type method based on numerical distribution of types in RDF datasets to deal with noisy data. Some process presented a temporal extension of the web ontology language (OWL) for expressing time-dependent information. A classical proposing of a distributed reasoning method for computing the closure of an RDF graph based on MapReduce and implemented it on top of Hadoop which highlighted the main draw-back of the Map-Reduce-based reasoning and then introduced Map-resolve method for more expressive logics. When the data volume increases and the ontology bases were updated, solving

methods required the re-calculation of the entire RDF conclusion each time when new data attained. To avoid such time-consuming process, reasoning methods need to be improved. A scalable similar implication method had calculated the RDF closure for RDF dataset. It also modified the procedures to process the declarations rendering to the status as incremental perceptive, but the performance of incremental updates was highly dependent on input data.

3.2. MapReduce

The key contributions of the Map-Reduce framework are not the actual map and reduce functions, but the scalability and fault-tolerance achieved for a variety of applications by optimizing the execution engine. As such, a single-threaded implementation of Map-Reduce (such as Mongo DB) will usually not be faster than a traditional (non-Map-Reduce) implementation; any gains are usually only seen with multi-threaded implementations. Only when the optimized distributed shuffle operation (which reduces network communication cost) and fault tolerance features of the Map-Reduce framework come into play, is the use of this model beneficial. Optimizing the communication cost is essential to a good Map-Reduce algorithm.

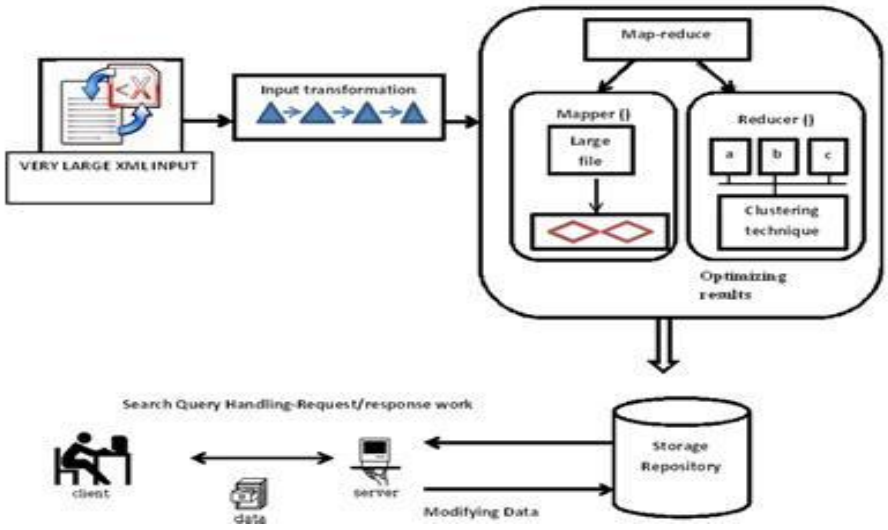

Fig1: Architecture of the proposed system

4. IDI METHOD OVER LARGE SCALE RDF

The ontological information has been gathered from the original Resource Description Framework data. The dictionary encoding and triples indexing module encodes all the triplicates into an exclusive and small identifier to decrease the physical size of contributed data. To efficiently compress a large amount of RDF data in parallel, we run a Map-Reduce algorithm on input datasets to reduce the key basis on k means clustering algorithm. A simple pattern can be made with optimizing the classified data on database engine. Construction and query retrieval can be allocated finally for fetching the interior set of data. Delivering queries based on mapping-reducing function performance. Finally, construction of recall and precision is done for gathering false positive and false negative performance evaluation.

Algorithm:

Input: Large data file

Output: client requested information

Function map (String name, String document):

// name: document name

// document: document contents

For each word w **in** document:

Emit (w, 1)

Function cache_m(k cluster of words)

Function reduce (String word, Iterator partialCounts):

// word: a word

// partialCounts: a list of aggregated partial counts

sum = 0

For each pc **in** partialCounts:

sum += ParseInt(pc)

emit (word, sum)

Sample Input:

10577 ANTH 211 F01 Introduction to Anthropology 1.0 Brightman M-W 03:10PM 04:30 ELIOT 414

20573 ANTH 344 S01 Sex and Gender 1.0 Makley T-Th 10:30AM 11:50 VOLLUM 120

10624 BIOL 431 F01 Field Biology of Amphibians 0.5 Kaplan T 06:10PM 08:00 PHYSIC 240A

20626 BIOL 431 S04 Seminar in Biology 0.5 Yezerinac Th 06:10PM 08:00 BIOL 200A

11:00AM 11:50 VOLLUM VLH

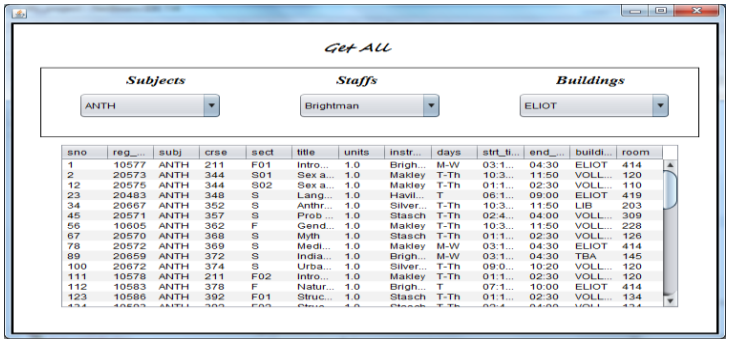

Fig 2: Sample Output

The activity diagram describes the activities carried out throughout the process which include data tranformation, mapping values, Clustering information, Updating data into the optimizer, Query request and finally Query response..

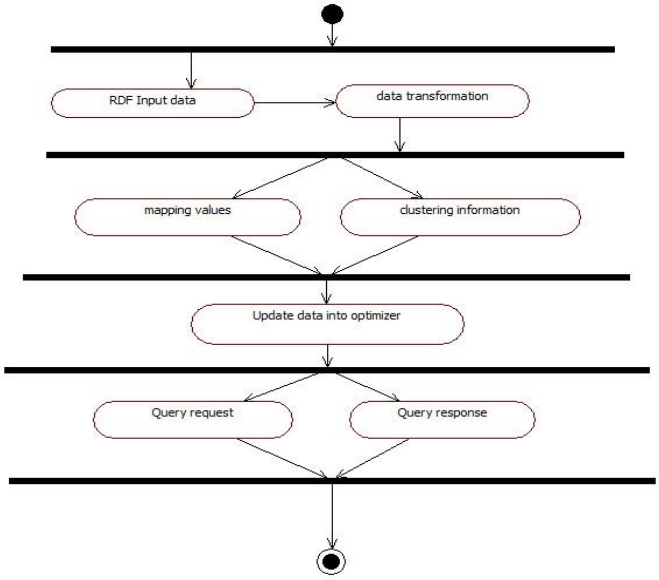

Fig 3: Activity Diagram

5. CONCLUSION

In this big data era, it becomes a very complex and challenging task for reasoning on the Web scale because of large volume of data present on the Web. The application of this method is to facilitate the needs of real-world healthcare data. Collaborating with all data to facilitate the searching of online purchased data-values based on one million jabong data, medical ontology by using semantic knowledge on web is gathered. On the whole map reduce concept with clustering work has been mainly aimed to penetrate data as well as modify the same on specified position in medical records. Thus the data construction can be easily processed with the help of Hadoop platform.

In the future, the proposed method would be validated on more datasets and extend IDIM to other ontology languages.

6. ACKNOWLEDGEMENTS

With great pleasure I would like to take this opportunity to express my heartfelt gratitude to all the people who helped in making this project a grand success. I also express a deep sense of gratitude to my guide Mr.K.Venkateswara Rao, Associate Professor, for his cordial support, valuable information and guidance, which helped me in completing this task through various stages.

7. REFERENCES

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

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

[3] 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, Sep. 2009.

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

[5] 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.

[6] V. Milea, F. Frasincar, and U. Kaymak, "tOWL: A temporal web ontol-ogy language,"IEEE Trans. Syst., Man, Cybern. B, Cybern. vol. 42, no. 1, pp. 268–281, Feb. 2012.

[7] 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.

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

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

[10] RDF Semantics [Online]. Available: http://www.w3.org/TR/rdf-mt/ [21] RDF Schema[Online]. Available: http://en.wikipedia.org/wiki/RDFS

[11] SPARQL 1.1 Overview[Online]. Available:http://www.w3.org/TR/sparql11-overview/

[12] Hadoop [Online]. Available: http://hadoop.apache.org/

[13] HBase [Online]. Available: http://hbase.apache.org/

[14] Billion Triples Challenge 2012 Dataset [Online]. Available: http://km.aifb.kit.edu/projects/btc-2012/

[15] Y. Guo, Z. Pan, and J. Heflin, "LUBM: A benchmark for OWL knowl-edge base systems,"J. Web Semantics, vol. 3, nos. 2–3, pp. 158–182, Oct. 2005.

[16] Bio2RDF [Online]. Available: http://bio2rdf.org