An Incremental and Distributed Inference Method for Large-Scale Ontologies Based on MapReduce Paradigm

ABSTRACT:

Incremental and distributed inference method for large-scale RDF datasets using MapReduce, which realizes high-performance reasoning and runtime searching, especially for incremental knowledge base. Constructing transfer inference forest and effective assertional triples, the storage is largely reduced and the reasoning process is simplified and accelerated.

INTRODUCTION

Resource description framework is a basic representation of ontologies used to describe the knowledge in the Semantic Web. A large volume of Semantic Web data, the fast growth of ontology bases has brought significant challenges in performing efficient and scalable reasoning. Centralized reasoning methods are not sufficient to process large ontologies. Distributed reasoning methods are thus required to improve the scalability and performance of inferences. 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. 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.

EXISTING SYSTEM

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.
DISADVANTAGE OF EXISTING SYSTEM

- The data volume of RDF closure is ordinarily larger than original RDF data.
- The storage of RDF closure is thus not a small amount and the query on it takes nontrivial time.
- The data volume increases and the ontology base is updated, these methods require the recomputation of the entire RDF closure every time when new data arrive.
- Which takes much time (usually several hours or even days for large) and space (generally the ontology size is more than original data size).

PROPOSED SYSTEM

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. In order to store the incremental RDF triples more efficiently, we present two novel concepts, transfer inference forest and effective assertional triples. Their use can largely reduce the storage and simplify the reasoning process.

ADVANTAGE OF PROPOSED SYSTEM

- Linear scalability, automatic failover support, and convenient backup of MapReduce jobs
- Distributed data on the web make it difficult to acquire appropriate triples for appropriate inferences
- Which can well leverage the old and new data to minimize the updating time and reduce the reasoning time when facing big RDF datasets.
- To speed up the updating process with newly-arrived data and fulfill the requirements of end-users for online queries
ARCHITECTURE:
HARDWARE REQUIREMENTS:

- System: Pentium IV 2.4 GHz.
- Hard Disk: 40 GB.
- Floppy Drive: 44 Mb.
- Monitor: 15 VGA Colour.

SOFTWARE REQUIREMENTS:

- Coding Language: Java 1.7, Hadoop 0.8.1
- Database: MySql 5
- IDE: Eclipse