**福州大学本科生毕业设计（论文）开题报告**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **姓名** | **李娟** | **学号** | **221200219** | **专业** | **软件工程** |
| **题目** | **基于MapReduce的网络级并行搜索引擎** | | | | |
| 一．研究背景、概况及意义1．研究背景 近年来，行业应用数据规模的爆炸性增长推动了大数据技术的迅猛发展。在常规大数据迅猛增长的同时，语义数据特别是RDF数据也以数亿甚至数十亿元组的规模大量涌现。2009年初,语义网是估计包含44亿三元组。一年之后,语义网的大小扩大三倍到130亿三元组，和当前的趋势表明,这种增长速度并没有改变。2012年3月，LOD项目所收集的RDF数据集已经包含了超过325亿条RDF三元组。而可伸缩的推理是语义网的一个至关重要的问题。而分布式推理具有良好的可伸缩性。利用并行计算技术解决大规模RDF数据相关问题已经成为学术界和工业界的普遍共识。  当前有Google提出的MapReduce并行计算模型以其高可扩展性和高易用性成为目前大数据处理最为成功的并行计算技术之一。Hadoop是Google MapReduce框架的一个开源实现，其提供了类似于GoogleGPS的分布式数据存储系统HDFS以及类似于Google分布式数据存储系统HDFS以及类似于Google BigTable的面向半结构化数据存储和管理系统HBase。 研究概况 当前世界上有很多的科研组织和大学团队在从事RDF方面的研究。DARPA(Defense Advanced Research Projects Agency，美国国防部高级计划研究署)支持开发的DAML，是语义Web研究的主要推动力，在RDF之上以描述逻辑为基础，成功地以机器可读的方式表示了语义关系。而推理一直是研究者关注的焦点问题之一。  目前国内对RDF及相关语义万维网技术的研究也已经在很多大学和机构展开。 现实意义 目前基于MapReduce的并行推理的研究工作是将传统的推理技术直接迁移到MapRedece框架下，这种方法的效率低下。为了实现MapReduce下大规模RDF数据的高效推理。本文在RDFS和OWL Horst等相关技术分析的基础上，通过采用合理的数据划分方案和优化的推理规则执行策略，能将推理计算分解为多个相互间没有依赖关系的独立的推理任务，解决传统推理算法的直接迁移带来的大量数据移动问题。同时，优化规则的执行次序，避免了迭代计算，提高了推理过程的执行效率。 | | | | | |
| 二．研究主要内容 随着语义网的快速发展，RDF语义数据大量涌现，大规模RDF语义数据推理的一个主要问题是计算量大、完成计算需要消耗很长的时间，显然，传统的单机语义推理引擎难以处理大规模的语义数据，另一方面，现有的基于MapReduce的大规模语义推理引擎，缺乏对算法在分布和并行计算环境下执行效率的优化，使得推理时间仍然较长。此外，现有的推理大量的语义网数据和它的快速增长带来了在执行效率和可伸缩推理上大量计算的挑战。基于MapReduce的并行化语义推理引擎，采用合理的数据划分模型和并行化算法，降低计算节点间的通信开销。同时优化推理规则的执行次序，提升推理的计算速度。通过实验结果表明，该算法有很好的优越性。  本文主要研究工作包括如下几个方面：  (1) 分析现有基于MapReduce的大规模语义推理引擎的现状。并分析分布式推理方法的挑战；  (2)简要介绍MapReduce编程模型；  (3) 简要介绍RDFS规则，提出RDFS推理的推理的好处以及分布式推理的挑战。并提出一系列技术实现基于MapReduce的RDFS规则集算法；  (4) 在第三部分的基础上，提出了OWLterHorst片段推理。由于OWL规则有多个实例三元组为先行词，所以主要讨论新的技术来实现OWL Horst推理的算法；  (5)扩展算法，提出增量更新算法；  (6)使用真实数据集和LUBM数据集对基于MapReduce的网络级并行搜索引擎进行评估，并验证我们的算法是有优越性的。 | | | | | |
| 三．研究步骤、方法及措施 软件开发生命周期可以包含三个阶段：分析、设计、实现。对于运用工程学的原理和方法来组织和管理软件的生产和维护的软件工程来说，软件开发的标准过程包括六个阶段[14]。结合项目的特点，下面给出具体的步骤、方法及措施。  1. 进行对基于MapReduce网络级并行搜索引擎的调研  通过网上查找与Hadoop分布式相关的资并借阅相关的图书馆书籍，对 Hadoop平台进行相关的了解，对MapReduce原型进行深入研究并掌握，同时对对大数据推理进行调研，分析当前在Hadoop分布式平台下对数据推理的现状。经过一些不同的途径调研，决定做基于MapReduce网络级并行搜索引擎优化算法研究。  2.讨论前期准备。  查找相当的文献，书籍，资料，对整个项目所涉及到的技术，有了初步认识。结合已掌握的编程技术，开始学习开发过程中要用到的新技术，做好开发过程中的知识储备和技术学习工作。  3.需求分析，平台搭建。  需求明确后，开始制定初步的计划。及时做好系统规格说明书、软件项目计划、软件需求规格说明书相关文档的记录工作，项目需求有调动及时调整。同时，将Hadoop开发平台搭建起来，并熟悉如何使用Hadoop开发平台。  4.项目正式开发，包括文档撰写和代码编写等  根据前期的详细设计，开始编码调试。实现系统核心算法代码，代码的编写有一个统一的风格，都将统一定制。在编码的过程，对每一个小的模块也必须进行初步的调试，最后整合起来。  5.后期项目测试和维护  当大部分编码工作完成后，接下来就是要进行大量测试，具体测试方法将参照软件工程课本上的多种测试方法，从多方面对系统进行测试，发现系统的Bug及时作出调整。 | | | | | |
| 四．研究进度计划 2016.02.17－2016.02.27 查阅相关文档、确定论文题目，并进行需求调研分析  2016.02.28－2016.04.28 撰写开题报告,掌握Hadoop平台及RDFS和OWL推理  2016.04.29－2016.05.02 撰写中期报告以及外文翻译、文献综述等文档  2016.05.03－2016.05.20 详细设计并进行开发，代码实现，并撰写相关文档  2016.05.21－2016.05.22 论文一稿  2016.05.23－2016.05.24 论文二稿  2016.05.25－2016.05.26 论文定稿，准备答辩材料 | | | | | |
| 五．参考文献 [1] Bio2RDF, <http://bio2rdf.org>, 2010.  [2] FactForge, <http://www.factforge.com>, 2010.  [3] Hadoop, <http://hadoop.apache.org>, 2010.  [4] Large triple stores wiki pagelarge triple stores wiki page, <http://esw.w3.org/topic/LargeTripleStores>, 2010.  [5] LarKC deliverable 5.2.2, <http://hadoop.apache.org>, 2010.  [6] Linked Data Semantic Repository (LDSR), <http://www.ontotext.com/ldsr/>,2010.  [7] Linked Life Data (LLD), <http://linkedlifedata.com>, 2010.  [8] WebPIE website, <http://www.cs.vu.nl/webpie>, 2010.  [9] D. Battré, A.H鰅ng, F. Heine, O. Kao. On triple dissemination, forward-chaining,and load balancing in DHT based RDF stores, in: Proceedings of the VLDB Workshop on Databases, Information Systems and Peer-to-Peer Computing (DBISP2P). 2006.  [10] M. Cai, M. Frank, RDFPeers: a scalable distributed RDF repository based on a structured peer-to-peer network, in: Proceedings of the International WorldWide Web Conference. 2004.  [11] J. Dean, S. Ghemawat, Mapreduce: simplified data processing on large clusters, in: Proceedings of the USENIX Symposium on Operating Systems Design & Implementation (OSDI), 2004, pp. 137–147.  [12] Q. Fang, Y. Zhao, G. Yang, W. Zheng, Scalable distributed ontology reasoning using DHT-based partitioning, in: Proceedings of the Asian Semantic Web Conference (ASWC), 2008.  [13] Y. Guo, Z. Pan, J. Heflin, LUBM: a benchmark for OWL knowledge base systems, Journal of Web Semantics 3 (2005) 158–182.  [14] P. Hayes (Ed.), RDF Semantics, W3C Recommendation, 2004.  [15] A. Hogan, A. Harth, A. Polleres, Scalable authoritative OWL reasoning for the  web, International Journal on Semantic Web and Information Systems 5 (2) (2009).  [16] A. Hogan, J. Pan, A. Polleres, S. Decker, Saor: Template rule optimisations for distributed reasoning over 1 billion linked data triples, The Semantic Web -ISWC 2010,2010,pp.337–353.  [17] A. Hogan, A. Polleres, A. Harth, Saor: authoritative reasoning for the web, in: Proceedings of the Asian Semantic Web Conference (ASWC), 2008.  [18] H.J. ter Horst, Completeness, decidability and complexity of entailment for RDF schema and a semantic extension involving the OWL vocabulary, Journal of Web Semantics 3 (2–3) (2005) 79–115.  [19] M.F. Husain, P. Doshi, L. Khan, B. Thuraisingham, Storage and retrieval of large rdf graph using hadoop and mapreduce, in: M.G. Jaatun, G. Zhao, C. Rong (Eds.), Cloud Computing, vol. 5931, Springer, Berlin, Heidelberg, 2009, pp. 680– 686 (Chapter 72).  [20] Z. Kaoudi, I. Miliaraki, M. Koubarakis, RDFS reasoning and query answering on top of DHTs, in: Proceedings of the International Semantic Web Conference (ISWC), 2008.  [21] V. Kolovski, Z. Wu, G. Eadon, Optimizing enterprise-scale OWL 2 RL reasoning in a relational database system, in: The Semantic Web – ISWC 2010, 2010, pp. 436–452.  [22] S. Kotoulas, E. Oren, F. van Harmelen, Mind the data skew: distributed inferencing by speeddating in elastic regions, in: Proceedings of the WWW, 2010.  [23] R. Mutharaju, F. Maier, P. Hitzler, A mapreduce algorithm for el+, in: Proceedings of the 23rd International Workshop on Description Logics (DL2010), Waterloo, Canada, 2010.  [24] A. Newman, Y. Li, J. Hunter, Scalable semantics the silver lining of cloud computing, in: Proceedings of the 4th IEEE International Conference on eScience, 2008.  [25] E. Oren, S. Kotoulas, G. Anadiotis, R. Siebes, et al., Marvin: distributed reasoning over large-scale semantic web data, Journal of Web Semantics 7 (4) (2009) 305–316.  [26] A. Schlicht, H. Stuckenschmidt, Peer-to-peer reasoning for interlinked ontologies, International Journal of Semantic Computing (2010) (Special Issue on Web Scale Reasoning).  [27] R. Soma, V. Prasanna, Parallel inferencing for OWL knowledge bases, in: International Conference on Parallel Processing, 2008, pp. 75–82.  [28] P. Stutz, A. Bernstein, W. Cohen, Signal/collect: Graph algorithms for the (semantic) web, in: Proceedings of the ISWC, Shanghai, China, 2010.  [29] J. Urbani, S. Kotoulas, J. Maassen, N. Drost, et al., Webpie: a web-scale parallel  inference engine, 2010 (1st prize at the 3rd IEEE SCALE challenge at CCGrid).  [30] J. Urbani, S. Kotoulas, J. Maassen, F. van Harmelen, et al., Owl reasoning with mapreduce: calculating the closure of 100 billion triples, in: Proceedings of the ESWC 2010.  [31] J. Urbani, S. Kotoulas, E. Oren, F. van Harmelen, Scalable distributed reasoning using mapreduce, in: Proceedings of the ISWC, 2009.  [32] J. Urbani, J. Maassen, H. Bal, Massive semantic web data compression with mapreduce, in: HPDC ’10: Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, ACM, New York, NY, USA, 2010, pp. 795–802.  [33] J. Weaver, J. Hendler, Parallel materialization of the finite rdfs closure for hundreds of millions of triples, in: 8th International Semantic Web Conference (ISWC2009), 2009. | | | | | |
| 学生签名：　　　　　　年　　月　　日 | | | | | |
| **指导教师意见（**对本课题的深度、广度及工作量的意见及开题是否通过**）：**  通过**□**完善后通过**□**未通过**□**  **指导教师签名：　　　　　　年　　月　　日** | | | | | |

注：开题报告用A4纸打印装订在毕业设计（论文）任务书后，学生可根据开题报告的长度加页。开题是否通过请指导教师在**□**内打“√”。