超大型中文叙词表本体的检索与推理研究

曾新红[1，2]　黄华军[2]　林伟明[1]

[1]（深圳大学图书馆，深圳 518060）

[2]（深圳大学计算机与软件学院，深圳 518060）

[摘要]本文研究了超大型中文叙词表本体检索和推理的网络化实现方法，并运用其成功地实现了“中国分类主题词表本体”（CCT1_OntoThesaurus）的共建共享系统（CCT1_OTCSS），时效性已达到实用要求。文中提出了基于 RDF 三元组思想的 Lucene 索引结构构造方法， 以 CCT1_OntoThesaurus （基于 CCT 一版的纯文本电子版建立）为例，构建了 Lucene 索引，在此基础上实现了高效的本体检索和一致性检测推理，并进一步实现了 CCT1_OTCSS 的各项功能。该解决方案对于我国现有的几部应用最为广泛的超大型中文叙词表快速实现本体化升级、网络化共建共享和动态完善具有良好的应用前景，对于国内外其他采用 XML、RDF 或 OWL 表示的大型知识组织系统（叙词表、本体等）实现网络化检索与推理也具有参考价值。

[关键词] Lucene，超大型中文叙词表本体，本体检索，推理，索引

[分类号] TP18　G254

Research on Retrieval and Reasoning of Ultra-Large-Scale OntoThesaurus

Zeng Xinhong[1,2]　Huang Huajun[2]　Lin Weiming[1]

[1](Shenzhen University Library, Shenzhen 518060, China)

[2](College of Computer and Software, Shenzhen University，Shenzhen 518060, China)

Abstract　This paper makes a research on an implementation of the network-based retrieval and reasoning about Ultra-Large-Scale OntoThesaurus, and the solution proposed has successfully applied to the realization of the CCT1_OTCSS, a co-construction and sharing system of an ultra-large-scale ontology named CCT1_OntoThesaurus. This paper proposes the structure of Lucene index based on the idea of triple "subject, predicate, object" of the RDF, and validates the feasibility of implementing efficient retrieval, terminology services and reasoning based on the Lucene index of ultra-large-scale OntoThesaurus. The solution can be reused for several ultra-large-scale Chinese thesauri most widely used in China at present, implementing quickly ontology-oriented upgrading, networked co-construction, sharing and dynamic updating for them, and also has a reference value for other large-scale knowledge organization systems (thesauri, ontology, etc.) in the form of XML, RDF or OWL at home and abroad.

Key words　Lucene, ultra-large-scale OntoThesaurus, ontology retrieval, reason, index

1 引言

中文叙词表本体共建共享系统(OntoThesaurus Co-construction and Sharing System，OTCSS)[1]是国家社科基金项目（编号：05CTQ001）的研究成果，原采用 Jena[2](惠普公司的一个针对语义网应用的 Java 开源工具包)和 SPARQL[3]（W3C 推荐的 RDF 标准本体查询语言）实现网络化检索和推理，可以有效解决目前国内已有的一般大型及以下规模（即 130 余部中的 120 余部）中文叙词表转换成本体后的检索和推理问题，其时效性已达到实用要求[1，4-6]。

但对于少数几部超大型中文叙词表（5 万叙词款目以上），原来采用的技术还远不能使其时效性达到通过网络实时使用的要求，需要寻求一种低投入、高效率的解决方案来解决这些 OWL 大型本体文件（60M 以上）的检索与推理问题，使我国现有的几部应用最为广泛的超大型中文叙词表可以快速地实现本体化升级、网络化共建共享

本文系国家社科基金项目“基于本体和知识集成实现中文叙词表的升级、共享和动态完善”（05CTQ001）的后续研究成果。

和动态完善。

2 国内外研究现状

近年来，国内已有许多针对 XML 文档的搜索及索引的研究[7-11]，但其查询速度慢且又不适合大数据量的要求。

HP 公司的 Jena 开发包可以实现对中小型和一般大型中文叙词表本体的检索与推理，但缺乏对超大型本体动态完善的支持。根据实验结果发现[5-6, 12-13]：在推理要求不高且数据量不大的情况下，Jena 就是一个不错的选择，但是对于超大型的本体，Jena 的支持还远远不够。

美国斯坦福大学的 Protégé-OWL 在做本体推理时一般使用 Jena 或 Jess 来实现。Jess 作为前向推理系统，推理时用空间换时间，会产生大量的中间数据，空间效率很低，同时由于 Jess 是通用推理引擎，不可能提供针对各种具体领域的效率优化能力[13]。Racer、FaCT、Pellet 等推理机和 Jena 一样，是针对具体本体语言的推理机，针对性较强，效率相对较高，但是都没有解决超大型本体检索与推理的案例。通过相关调查，也没有找到进一步实现美国国家癌症研究所(NCI)2003 年发布的叙词表 OWL 版本的检索和推理问题的信息。

Oracle 和 SQL Server 数据库都提供了高性能的 XML 存储和检索，并提供管理 XML 数据的基础架构，也可以满足大数据量的要求，但它们非开源软件，使用成本较高，且不支持本体推理，因此未成为本文的选择方案。

鉴于当前开源的全文检索引擎 Lucene[14]在实现检索方面的出色表现，虽然还没有发现使用 Lucene 来解决 OWL 本体大文件推理问题的相关研究和报道，05CTQ001 课题组仍决定尝试采用 Lucene 来解决这一难题。通过对 Lucene 全文检索引擎的深入研究，以及对现有的中文叙词表 OWL 本体大文件构建索引、定制查询与推理进行的初步实验，发现其在时间效率方面与已有的方法相比有很大的提高，而且能够满足大数据量的要求，最终 Lucene 成为本文的选择方案。

3 超大型中文叙词表本体(OntoThesaurus)的检索和推理解决方案

3.1 超大型OntoThesaurus的存储问题

本体的检索建立在它的存储机制上，目前 RDF 本体存储机制主要有数据库、内存、磁盘等载体。通过相关研究与实验可知，这三种存储方式分别具有以下特点：

采用关系数据库来存储本体三元组数据源是一种比较常见的存储方式。优点是数据库管理系统提供了标准的查询接口，使用方便。但存在第 2 节中提到的高成本问题和对本体推理支持不足的问题。参考文献[15]具体描述了目前使用数据库存储本体三元组的几种可能的形式，分别是：水平数据库（模型简单，但浪费存储空间，使数据库成为一张稀疏表），垂直数据库（Jena 等众多 RDF 引擎采用。缺点是每一次检索都需要遍历整个表，处理联合查询时，查询引擎很难获得高效率），以及按属性来建表的存储方式（造成表多而不易维护，且查询效率不高）。

基于内存的 RDF 存储与数据库相比，安装和配置较为简单，而且在内存中处理 OntoThesaurus 的三元组的插入、查询和推理操作的速度很快。但该机制的缺点也很明显，由于内存容量的限制，只适合小规模的数据量，而且存储是非持久性的[15]，内存还需处理其他很多进程易造成工作效率低下等。

如果将 OntoThesaurus 的信息以 OWL 文件的形式直接存储在磁盘，这样虽然可以存储海量数据，但是对查询引擎提出了很高的要求，需要具备解析和处理 OWL 大文件的能力，与其他方式相比，既缺乏查询支持，又不具备效率的优势[15]。

因此，考虑到既要存储大量数据，又要实现高效查询，本文选择为超大型 OntoThesaurus 建立存储在磁盘的 Lucene 倒排索引来实现本体检索与推理。

3.2 基于RDF三元组思想的Lucene索引存储结构

本文提出了基于 RDF 三元组思想的 Lucene 索引存储结构。

简单而言，本文需要索引的信息来自一个 RDF/XML 本体文件的 ABOX 部分，它包含多个资源描述，而一个资源描述是由多个语句（声明）构成，每一个语句是由资源（主体）、属性（谓词）、属性值（客体）构成的三元组，即表示资源具有的一个属性。

Lucene 的数据结构是虚拟文档，即 Document，OWL 本体大文件是 Lucene 需要索引的数据源，如何将这些数据源定制成 Lucene 的虚拟文档 Document 呢？RDF 的三元组思想为我们提供了参考，即本体文件中的一个三元组对应 Lucene 的一个虚拟文档 Document，然后将所有的虚拟文档交给 Lucene 做索引，通过倒排索引的形式存储在磁盘。

中文叙词表本体 OWL 文件是 RDF/XML 格式的文件，其 ABOX 部分由许多个 XML 结点组成，其中每个结点表示一个完整的中文叙词款目。我们可以这样认为：一个概念（叙词）表示一个资源，它由多个语句组成。例如，从 CCT1_OntoThesaurus 文件中抽取出来的一个叙词概念“焙烤食品”由两个语句表示：概念的中图法分类号是“TS219”，它的族首词是“食品”，如图 1 所示：

```
<Concept rdf:ID="焙烤食品">
  <CLCCode rdf:datatype="http://www.w3.org/2001/XMLSchema#string">TS219</CLCCode>
  <TopConcept rdf:resource="#食品"/>
</Concept>
```

图 1　OWL 本体大文件中的一个完整叙词款目

从这些信息我们可以知晓，叙词“焙烤食品”有两个属性，一个为 CLCCode，另一个为 TopConcept，其值分别为“TS219”和“食品”，用 RDF 三元组的有向图来表示，如图 2 所示。

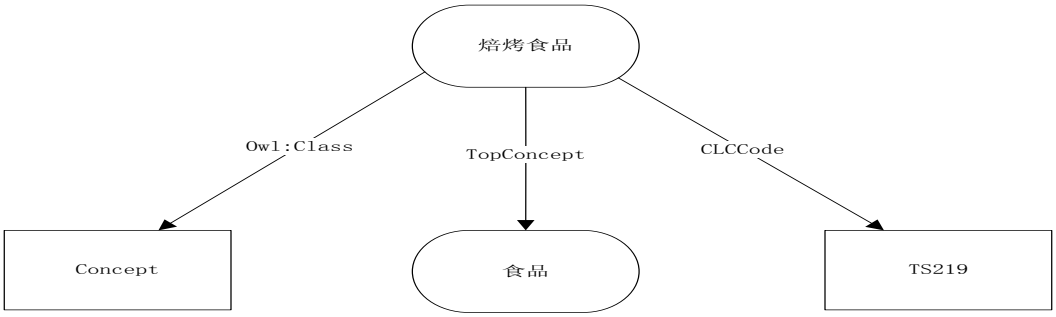

图 2　RDF 三元组有向图表示的叙词款目信息

根据 RDF 三元组的有向图，该叙词的相关信息在做索引的时候，可以构建两个 Lucene 虚拟文档，其表示方法如图 3 所示。

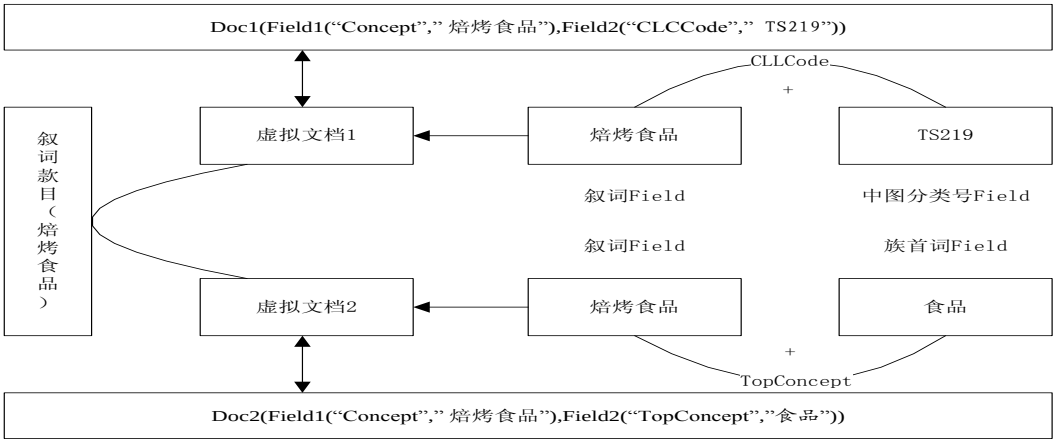

图 3　Lucene 虚拟文档表示的叙词款目信息

叙词的每个 Lucene 虚拟文档包含有两个数据字段(Field)，即代表一个三元组，其中概念名称如“焙烤食品”为

主体，属性名称如 CLCCode 或 TopConcept 为谓词，属性值如”TS219”或”食品”为客体。该叙词款目的存储结构设计成以 RDF 三元组思想为基础的 Lucene 虚拟文档后，生成 Lucene 索引的过程如图 4 所示。

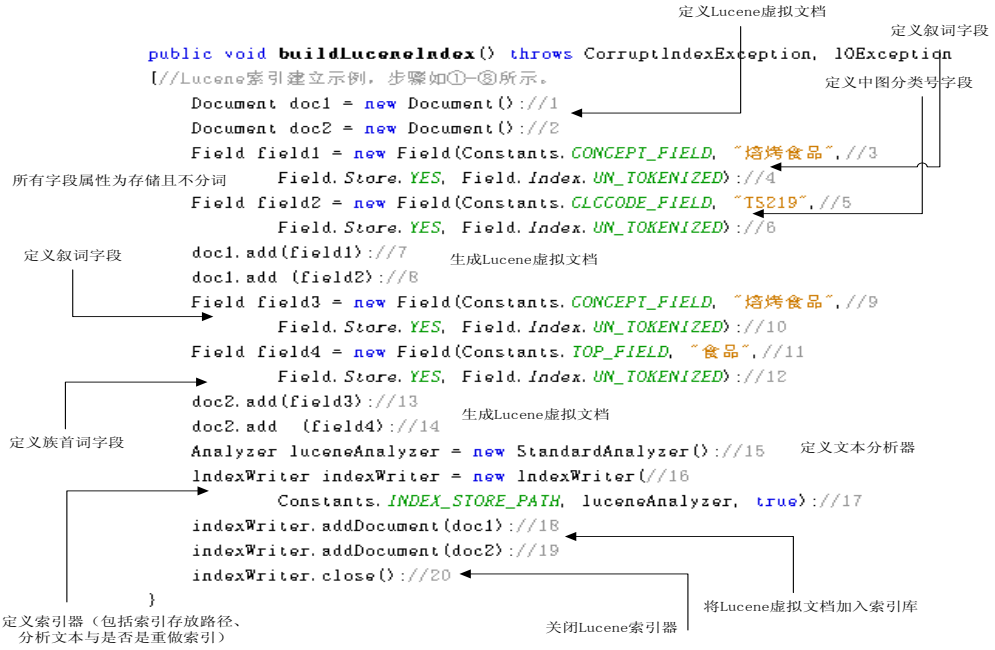

图 4　OntoThesaurus 的 Lucene 索引过程

3.3 超大型OntoThesaurus检索和推理解决方案的可行性

OntoThesaurus 的查询机制与它的语法和存储息息相关。在 OTCSS 中我们可以使用 Lucene 来定制各种查询，如构造基于定义域和属性域的词条查询和三元组的查询、布尔或、布尔与和布尔否查询等各种组合查询。倒排索引赋予 Lucene 高效的检索速度，功能强大且使用方便，这样 OntoThesaurus 的检索问题将迎刃而解。我们对现有的中文叙词表本体大文件进行了实验，用 Lucene 做索引后构造各种查询，其检索时间都保持在几百毫秒之内，具体的查询实现详见第 4 节。

将 OntoThesaurus 以 RDF 三元组的逻辑结构构造成 Lucene 虚拟文档后，中文叙词表本体转换成索引数据，它可以看作是若干个 Lucene 虚拟文档 Document 即三元组构成的一个集合，根据 RDF(S)推理规则，OntoThesaurus 的推理可以在已有的三元组上推出新的三元组，也可以在已有的三元组上推出某些矛盾的三元组，甚至在已有的三元组上推出多余的三元组等信息。

OntoThesaurus 中的每一个三元组由两个节点和连接两个节点的有向边组成，边的起始点为 OntoThesaurus 中的概念实例，边的终结点为该概念实例的某个属性值，有向边由该属性标记。因此 OntoThesaurus 为一个大型的三元组有向图，它由若干个推理子图组成，如图 5 所示。设节点为 OntoThesaurus 的叙词(C)，边为 OntoThesaurus 的谓词(Narrower)，我们可以从三元组 T(1, N, 4)、T(4, N, 3)和 T(1, N, 3)以及 T(8, B, 9)、T(9, B, 10) 、T(10, B, 6)和 T(8, B, 6)推出下位词传递关系 T(1, N, 3)和上位词传递关系 T(8, B, 6)越界。

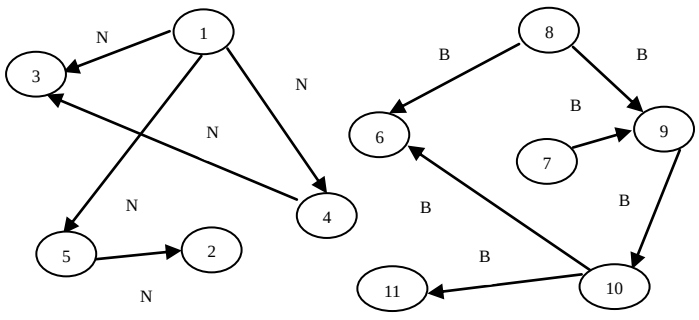

图 5　OntoThesaurus 中的推理有向图

OntoThesaurus 的推理是在相应的每个推理子图中进行的，这样可以节省大量的时间开销，避免每次推理都在整个 OntoThesaurus 中进行。对于每个推理图 G = (C, {T})，其中每条有向边对应一个三元组 t(S, P, O)∈T，推理图中的三元组为推理条件，对于每种推理结果，推理后对推理图进行更新，因此更新后的图还是一个推理图。

根据本文构造的 Lucene 索引及其逻辑结构，可以分析 Lucene 版的本体推理机在时间和空间上的复杂度，确定该本体推理机是否能够解决网络化的在线推理难题。设超大型的 OntoThesaurus 的三元组个数为 $n(n = n_p + n_c + n_h + n_t + n_n + n_b + n_r$，下标为 OntoThesaurus 中相应属性名称的首字母[1]），OntoThesaurus 中关系属性种类数目为 p，结合文献[6]为中文叙词表本体定义的一致性检测机制，本文构造了相应的推理算法来实现（详见 4.3 节），并且估算出其中一些推理的复杂度如表 4-1 所示。

表 4-1 Lucene 版本体推理机部分推理复杂度分析表

复杂度 / 全局检查	时间复杂度	空间复杂度
值域不一致	$O(n_h)$	$O(n_h)$
二元关系冲突	O(n*p)	O(n/p)
入口词多次出现	$O(n_h)$	$O(n_h)$
传递关系越级	$O(n_b^2)$或 $O(n_n^2)$	$O(n_b)$或 $O(n_n)$
未成对指引	$O(n_n)$或 $O(n_b)$	$O(n_n)$或 $O(n_b)$
非法对称	O(n)	O(n/p)
非法自反	O(n)	O(n/p)

从表中可以看出 Lucene 版的本体推理机的在时间复杂度上基本上都是线性的。由于目前一般的服务器内存空间都足够大，所以借助 Lucene 这个高效的检索引擎，实现 OntoThesaurus 的网络化在线推理完全是可行的。

4 超大型 OntoThesaurus 检索和推理方案的设计与实现

4.1 超大型OntoThesaurus检索方案的设计与实现

OntoThesaurus 的检索实现都是建立在 05CTQ001 项目的成果 OTCSS 基础之上的，功能和界面基本不变（参见参考文献[5]）。我们将超大型本体的检索转化成为对超大型本体的索引文件的检索，对于精确检索、前方一致检索以及任意一致检索，一般通过不同的查询子类(Query)辅以通配符来实现。

下面主要介绍通过主题词（入口词）途径进行任意一致检索的解决思路，其他途径的检索实现可以参照此解决方案类推。使用 Lucene 实现任意一致检索的步骤如下：

Step1: 首先初始化存放 Lucene 索引文件目录的路径；

Step2: 定义指向 Lucene 索引文件目录的搜索器；

Step3: 构建与任意一致检索相关的查询，此时使用通配符搜索(WildcardQuery)。如果为精确搜索则使用 TermQuery，前方一致搜索使用 PrefixQuery；

Step4: 定义存放检索结果的 Hits 类；

Step5: 执行相应的查询；

Step6: 将查询结果进行归并，即调整成若干个完整叙词款目的形式；

Step7: 对叙词款目按拼音（Lucene 默认为按相关度排序）进行排序后输出；

Step8: 计算本次检索所花费的时间。

图 6 为搜索效果展示。另外需要注意的是，从本文构建 Lucene 虚拟文档的方法来看，如果检索条件是英译名、中图法分类号时，则需要通过两次检索才能够搜索出用户所想要的检索结果。

图 6 以“中国”作为关键字的 Lucene 搜索效果展示[16]

4.2 超大型OntoThesaurus的网络术语服务

OntoThesaurus 在语义 Web 界、图书馆界都有着很强的应用背景。本文提供的超大型 OntoThesaurus 的网络术语服务包括供人使用的 OntoThesaurus-TS 和供应用程序使用的 Web Service API（OntoThesaurus-API，目前可提供 18 个服务函数）。功能和界面与原 OTCSS 系统保持一致[5]，但实现方法有所改变，保证了服务的时效性。

网络用户利用 OntoThesaurus-TS[16]的服务，可以检索和获取所需的概念术语及其相关信息（分类号，英译名，同义词，上/下位词或其指定子关系词，相关词或其指定子关系词等），应用于 OPAC 检索、搜索引擎、数据库检索等应用程序，进行扩展查询、分类标引、翻译等工作。在此过程中，还可以随时提交自己对现有词表的修订意见。

OntoThesaurus-API[17]可以帮助应用程序突破机械式字面（关键字）匹配局限于表面形式的缺陷，从词所表达的概念意义层次上来认识和处理用户的检索请求[18]，即实现概念检索（或称智能检索）。例如，我们在深圳大学图书馆的 OPAC 检索系统中，通过调用 OntoThesaurus-API，实现了各种概念检索。以检索“电脑”为例，可以将此入口词转换为正式主题词“电子计算机”实现规范化检索，其效果如图 7 所示。

图 7 以“电脑”作为关键字的智能搜索效果

OntoThesaurus-API 还可以广泛应用于其他应用程序中。例如：在众多支持 Tag 的应用程序中，可以使用 OntoThesaurus-API 为用户输入的标签进行规范化的提示，使用户输入的标签更为规范化；在面向主题词的数据挖掘中，运用 OntoThesaurus-API 可以基于主题词及其款目信息对数据进行挖掘、统计；在机器学习中，通过 OntoThesaurus-API，可以提供同义词辨析机制和主题词分析机制，进行与主题词相关的信息抽取等等。

上述所有的应用都可以通过 Web Service 调用的标准方法，在不同的应用平台上进行实现。[19]

4.3 超大型OntoThesaurus推理方案的设计与实现

超大型 OntoThesaurus 的推理主要运用于一致性检测机制的实现，OTCSS 系统的知识采集、词表管理以及全局检查功能中都用到了一致性检测机制，以保证 OntoThesaurus 在整个生命周期中的正常运行。

文献[6]结合中文叙词表和本体的特点、编制规范和描述逻辑，建立了 OntoThesaurus 的一致性检测机制，即 10 条自定义规则。该文中采用的 Jena/SPARQL 推理方案可以满足一般大型 OntoThesaurus（约 2 万叙词款目以下）的实时推理要求，但无法满足超大型 OntoThesaurus 的推理时效性要求。

本文采用 Lucene 和 Jena 对 OntoThesaurus 进行基于自定义规则的推理，可检测出值域不一致、入口词多次出现、非法自反关系、非法对称关系、未成对指引关系、二元关系冲突和传递关系越级等矛盾问题，以及拼音缺失和叙词未定义等信息缺失问题，并可自动生成族关系。在全局检查阶段，这些问题可以通过网络界面呈现在修订专家面前，根据推理结果的问题信息提示，修订专家可进行相应的处理，达到不断完善的目的。

在知识采集和词表管理的各个阶段，也根据需要不同程度地应用了一致性检测机制。知识采集的过程包括两个部分：知识发送和知识提取，它是 OntoThesaurus 进行充实和完善的有效途径。普通网络用户、领域专家和标引员可以发送修订意见给修订专家参考，其发送信息包括：为原叙词增加入口词（同义词）、新增补叙词（正式主题词/概念）、修改原叙词款目信息、原叙词款目整条删除，以及新增相关关系子关系种类等。修订专家通过提取统计后的修订信息进行处理，决定修订意见的去留。词表管理是供修订专家使用的网络维护界面，对完善 OntoThesaurus 起着非常重要的作用。它通过维护 Lucene 索引来实现，主要包括以下三个部分：新增叙词、修改叙词和删除叙词。

OTCSS 的知识采集和词表管理都是为应对 OntoThesaurus 的动态发展而制定的一套维护机制。知识采集中新增补叙词款目、修改原叙词款目信息和原叙词款目整条删除的处理方式，分别与词表管理的新增叙词、修改叙词和删除叙词一致；为原叙词增加入口词（同义词）需要检验在 OntoThesaurus 中新增的该入口词是否作为概念的实例以及概念的实例的入口词存在，如果合法就只需向索引文件中加入一个三元组虚拟文档即可；新增相关关系的子关系种类则需要通过 Jena 修改 OWL 本体文件的 TBOX 部分，向其中添加该关系种类的基本信息。因此，OTCSS 应具备完善的推理能力，以保证在更新的过程中不至于破坏 OntoThesaurus 的一致性而引起矛盾，其具体的检测效果可以通过访问参考文献[16]进一步了解。

OntoThesaurus 的一致性检测问题的形式化描述请参见参考文献[6]。根据这些形式化描述，我们构造了相应的推理算法来判定这些一致性检测问题。以“传递关系越级”为例，算法如下：

(1)用 Jena API 解析 OntoThesaurus 的 TBOX（OWL 文件），从中搜索出 Narrower、Broader 关系以及它们的扩展子关系集合 R；

(2)定义存储传递关系越级矛盾信息的空集合 ErrorMsg；

(3) 依次取出集合 R 中的任意元素 r，从 OntoThesaurus 中以 r 域搜索出 r 关系中的所有三元组，其中 c_1 和 c_2 分别为三元组的主体和客体，即 $\forall r(r \in R \wedge < c_1, c_2 > \in r \wedge c_1, c_2 \in C)$，若 $\forall r(r \in R \wedge < c_1, c_3 >, < c_3, c_2 > \in r)$ 成立，且 $< c_1, c_3 > \in r$，$< c_3, c_2 > \in r$，无论是直接的还是间接的，将矛盾信息加入集合 ErrorMsg，否则接着检

查该 r 关系中未检查的三元组，直到 r 关系中的所有三元组检查完毕，转下一步；

(4) 集合 R 中的所有元素都处理完毕，则转下一步，否则 goto(3)；

(5)若 ErrorMsg 不为空，输出 OntoThesaurus 中所有传递关系越级的矛盾信息，即 print(ErrorMsg)，否则提示 OntoThesaurus 中不存在传递关系越级的矛盾信息。

最后用 Java 编程实现了这些算法。本方案与原方案均采用 Java 语言进行实现，且没有改变系统的架构，不同之处在于：原方案对于本体模型的读取、检索、推理和写入都采用 Jena 提供的 API 进行实现，而本方案则基于 Lucene 提供的 API 对这些功能进行实现。

5 实验与分析

为了进一步证明本文提出的基于 Lucene 的解决方案对实现超大型中文叙词表本体检索的可行性，我们通过大量实验统计得出叙词数量（本体大小或三元组个数）与检索时间的关系，如表 2 所示。表 2、图 8 和图 9 中的一般检索均以主题词（入口词）作为搜索条件，以“中国”作为搜索关键字，高级检索则以主题词（入口词）“中国”任意一致、中图法分类号 “D” 前方一致作为搜索条件。

表 2 叙词数量与检索时间关系表

检索方式 / 三元组数	前方一致	精确一致	任意一致	高级检索
约 10 万(约 10M)	32ms	47ms	63ms	3572ms
约 20 万(约 20M)	46ms	47ms	78ms	5480ms
约 30 万(约 30M)	47ms	62ms	110ms	6921ms
约 40 万(约 40M)	47ms	63ms	125ms	8853ms
约 60 万(约 60M)	62ms	78ms	203ms	11828ms

表 2 的数据表明，使用 Lucene 实现 60M 以下数据量的检索所用时间基本都在毫秒数量级。本文所采用的实例是基于《中国分类主题词表》一版完整数据建立的 CCT1_OntoThesaurus，叙词数量约 20 万（包括主题词串），三元组约 60 万，本体大小约 60M，使用本文的解决方案完全可行。

在同样的条件下，采用原 Jena/SPARQL 推理方案实现该超大型本体的检索效果很不理想。图 8 为本文 Lucene 方案与原 Jena/SPARQL 方案进行对比的效果图，图 9 为本文 Lucene 方案与改进后的 Jena/SPARQL 方案进行对比的效果图，可以看出，Jena/SPARQL 方案无法满足超大型 OntoThesaurus 的检索需求，Lucene 方案则在时间上具备明显的优势。

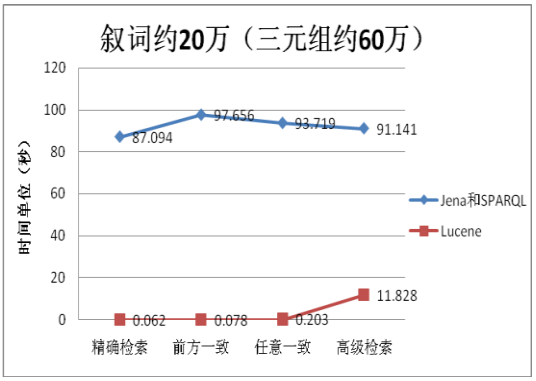

图 8 Lucene 方案和原 Jena/SPARQL 方案的检索效果对比

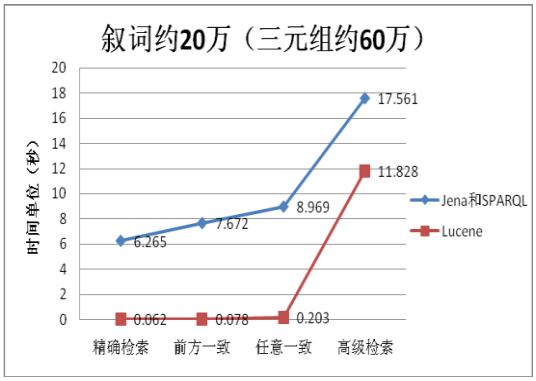

图 9 Lucene 方案和改进后 Jena/SPARQL 方案的检索效果对比

Lucene 强大的检索能力为实现超大型中文叙词表本体的推理奠定了坚实的基础。图 10 是使用 Lucene 方案实现超大型 OntoThesaurus 全局推理的时间效果图，图 11 是全局推理中的“值域不一致”检测运行结果图，而采用原 Jena/SPARQL 方案解决该超大型 OntoThesaurus 的全局推理，其所耗时间在实用中是无法容忍的。

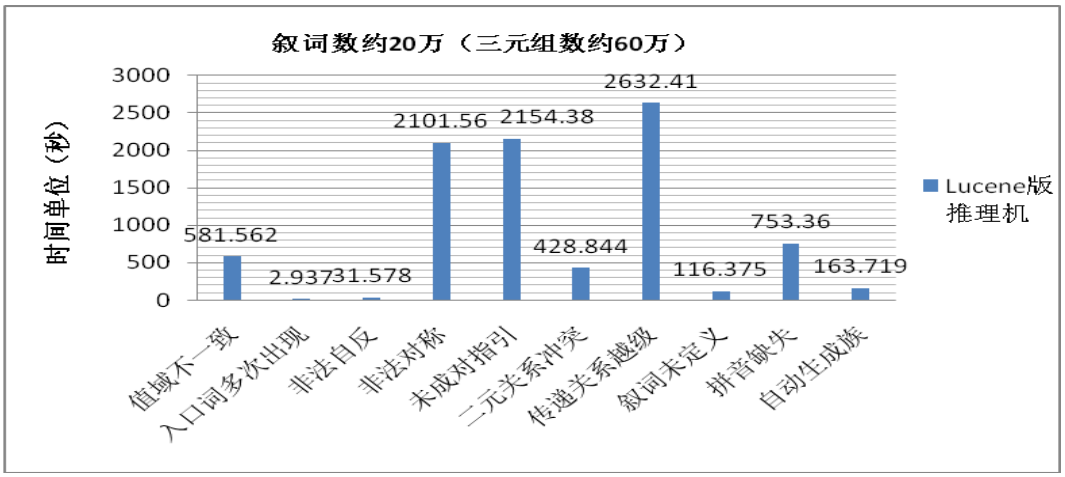

图 10 Lucene 版推理机的效果图

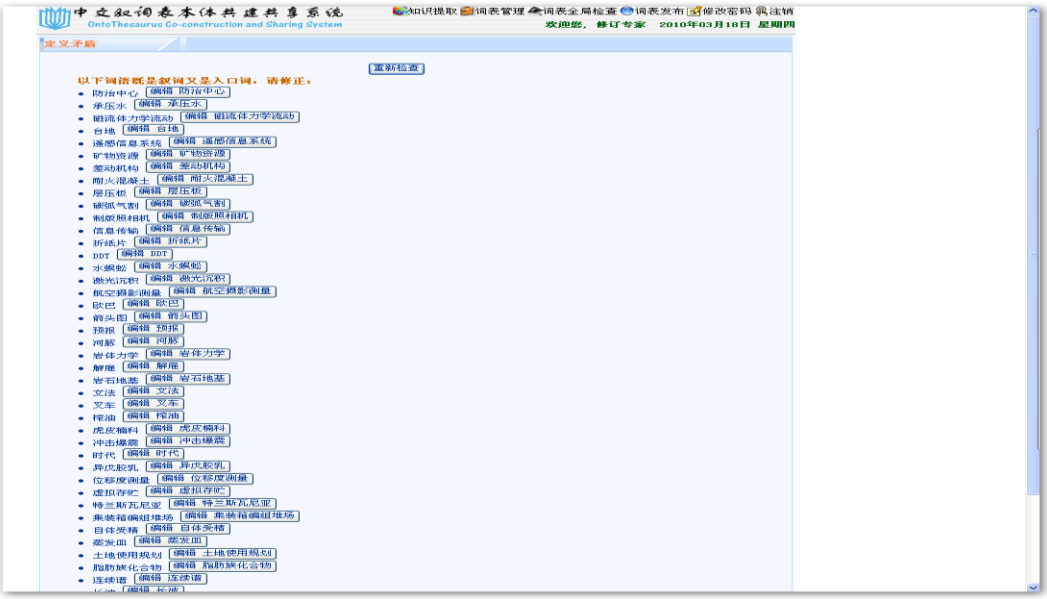

图 11 Lucene 版推理机“值域不一致”检测运行结果

实验结果显示，Lucene 版的推理机对于实现超大型中文叙词表本体的网络化全局推理是基本可行的。对于少数时间复杂度较高的推理，通过网络提交推理结果效果还不是很理想，需要进一步改进，如改为分阶段提交推理结果或非网络方式运行。

6 结语和展望

本文提出了基于 Lucene 的超大型中文叙词表本体检索和推理方案，成功实现了 CCT1_OntoThesaurus 的共建共享系统 CCT1_OTCSS，其各项功能已基本达到了实用要求。该解决方案对于我国现有的超大型中文叙词表快速实现本体化升级、共建共享和动态完善具有良好的应用前景。

基于 RDF 三元组思想设计的 Lucene 索引存储结构，具备扩展性和通用性，对于其他采用 XML/RDF/OWL

表示的大型或超大型知识组织系统（叙词表、本体等），本文的解决方案也具有参考价值。

课题组拟进一步为本解决方案提供支持 SPARQL 本体检索语言的 API，使其具有更广泛的通用性。

参考文献[16]和[17]列出了 CCT1_OTCSS 的登录地址，欢迎使用、测试和批评指正。更多信息和研究进展将在我们的 NKOS 研究网站（http://nkos.lib.szu.edu.cn）上发布，敬请关注。

参考文献

[1] 曾新红等．中文叙词表本体共建共享系统研究[J]．情报学报, 2008,27(3):386-394.

[2] Jena – A Semantic Web Framework for Java[EB/OL]．[2008-5-22]．http://jena.sourceforge.net．

[3] W3C ． SPARQL Query Language for RDF W3C Recommendation 15 January 2008 [EB/OL]. [2008-12-24]. http://www.w3.org/TR/rdf-sparql-query.

[4] 曾新红. 中文叙词表本体——叙词表与本体的融合[J]. 现代图书情报技术， 2009(1):34-43.

[5] 曾新红，林伟明，明仲．中文叙词表本体的检索实现及其术语学服务研究[J]．现代图书情报技术，2008(2):8-13．

[6] 曾新红，林伟明，明仲．中文叙词表本体一致性检测机制研究与实现[J]．现代图书情报技术，2008(5):1-9．

[7] 李新叶，苑津莎．一种快速的 XML 语义检索算法[J]．电子学报，2007，35(11):2220-2225．

[8] 孔令波，唐世渭，杨冬青，王腾蛟，高军．XML 数据的查询技术[J]．软件学报，2007，18(6):1400-1418．

[9] 孔令波，唐世渭，杨冬青，王腾蛟，高军．XML 数据索引技术[J]．软件学报，2005，16(12):2063-2079．

[10] 邵晓宇．基于本体的大型数据资源智能检索研究[D]．合肥：合肥工业大学，2008:10-11．

[11] 汪智勇．本体查询与推理研究及其实现[D]．长沙：中南大学，2007:7-39．

[12] 吴元业．基于信任度的个性化推理机的研究与实现[D]．深圳：深圳大学，2009:8-14．

[13] 推理机 Jess、Racer、Jena 比较[EB/OL]. (2009-1-20). [2009-4-11]. http://blog.csdn.net/hyzhx/archive/2009/01/20/3844741.aspx．

[14] 开放源代码的全文检索引擎 Lucene--介绍、系统结构与源码实现分析[EB/OL]. [2008-10-26]. http://www.lucene.com.cn/about.htm．

[15] 陆建江，张亚非，苗 壮，周 波．语义网原理与技术[M]．北京：科学出版社，2007:136-139．

[16] CCT1_OTCSS 的 OntoThesaurus-TS. http://nkos.lib.szu.edu.cn:8080/ThesaurusProjectForCCTWL/login.jsp.

[17] CCT1_OTCSS 的 OntoThesaurus-API. http:// nkos.lib.szu.edu.cn:8080/ThesaurusProjectForCCTWL/services/ThesaurusService?wsdl.

[18] 宋炜，张铭．语义网简明教程[M]．北京：高等教育出版社，2004:23-139．

[19] 林伟明，曾新红. OntoThesaurus Web Service API 及其应用研究[J]. 图书情报工作，2010,54(2):119-139