Geomatics Science and Technology 测绘科学技术, 2019, 7(2), 90-100
Published Online April 2019 in Hans. http://www.hanspub.org/journal/gst
https://doi.org/10.12677/gst.2019.72014

Hans汉斯

A Map-Reduce-Based Parallel Approach for Geospatial Data Interlinking in a Semantic Web

Wenyu Yang[1,2]

[1]State Key Laboratory of Information Engineering in Surveying Mapping and Remote Sensing, Wuhan University, Wuhan Hubei
[2]Department of Civil, Geo and Environmental Engineering, Technical University of Munich, Munich Germany
Email: yewenyusheng@gmail.com, wenyu.yang@tum.de

Received: Mar. 20th, 2019; accepted: Apr. 4th, 2019; published: Apr. 11th, 2019

Abstract

The Web of Data represents an intermediate step towards the Semantic Web. Constructing links among different Resource Description Framework (RDF) datasets is a key issue in the Web of Data. An identity link aims to match entities from different datasets and is an important type of RDF link. There are many approaches to constructing identity links between geospatial entities. This paper adopts the Hausdorff distance to compute the location and shape similarity between two entities. Because the computation of the Hausdorff distance is complex and geospatial data intrinsically large, the entire matching process is very time consuming. This paper proposes a Map-Reduce-based framework to parallelize the similarity computation, significantly reducing the runtime. This approach was verified to be effective in an experiment using data from Nomenclature of Territorial Units for Statistics (NUTS) and Database of Global Administrative Areas (GADM). The matching precision was high, and with the utilization of the proposed parallel framework, the runtime was reduced to only approximately 3 h on 8 nodes; in contrast, when run on 1 node, the runtime exceeded one day.

Keywords

Map-Reduce, Data Interlinking, Geospatial Semantic Data, Hausdorff Distance

基于MapReduce的语义网空间数据关联

杨雯雨[1,2]

[1]武汉大学测绘遥感信息工程国家重点实验室，湖北 武汉
[2]慕尼黑工业大学土木地质环境工程系，德国 慕尼黑
Email: yewenyusheng@gmail.com, wenyu.yang@tum.de

文章引用: 杨雯雨. 基于 MapReduce 的语义网空间数据关联[J]. 测绘科学技术, 2019, 7(2): 90-100.
DOI: 10.12677/gst.2019.72014

收稿日期：2019年3月20日；录用日期：2019年4月4日；发布日期：2019年4月11日

摘　要

构建数据网是实现语义网的一种途径，而关联不同的RDF数据集是构建数据网中的重要问题。在RDF关联中，同质关联是一种重要类型，旨在匹配来自不同数据集中的相同实体。构建地理空间实体之间的同质关联有许多方法，本文采用了基于相似性的关联方法，使用Hausdorff距离计算两个实体之间的位置和形状相似度。由于Hausdorff距离的计算十分复杂并且地理空间数据具有大数据的特性，因此整个匹配过程非常耗时。本文提出了一种基于MapReduce框架的并行计算方法，大大减少了运行时间。实验对欧洲领土数据库(NUTS)和全球行政区划数据库(GADM)中的数据进行了同质关联。关联结果精度高，在1个节点上运行时，运行时间超过了一天，而利用拟议的并行框架，在8个节点上运行时间仅3小时左右。

关键词

Map Reduce，数据关联，地理空间数据，Hausdorff距离

1. 引言

语义网旨在构建机器可读的网络，近年来受到越来越多的关注。为了应对具有语义丰富性的各种查询，必须对现实世界的信息和知识进行结构化提取和整合，这仍然是计算机科学中的一个挑战[1]。由结构化 RDF (Resource Description Framework)数据互联而形成的数据网被公认为是一种实现语义网的重要途径[2]。近年来，可用的结构化 RDF 数据增加了几个数量级[3]。根据关联开放数据(LOD)项目显示，截至2018年6月，已有16,136个链接连接着1234个数据集，比如DBpedia，FOAF，MusicBrainz和Geonames，这些数据集被发布和链接，形成了 LOD 云数据网。向数据网添加地理空间数据集会给其他类型的数据集增添更多的有用信息[4]，目前在LOD云中已有多个地理空间RDF数据集，如NationalMap，LinkedGeoData，NUTS 和 GADM，在地理空间数据集和其他类型的数据集之间建立关联可以为其他类型的数据集提供地理空间背景。

为了使人或机器能够浏览数据网，数据集必须要关联起来[5]。RDF 关联将数据网中的数据关联起来，消除信息孤岛，使人或机器能够从一个数据集被导航到另一个数据集。RDF 关联可以分为三种类型：关系关联，词表关联和同质关联。关系关联链接彼此相关的实体，如书籍及其作者或房屋及其所有者。词表关联将描述数据的词汇术语链接起来。同质关联链接标识同一实体的统一资源标识符(URI)。

同质关联目前有两种实现方法。一种是基于关键字的方法，它适用于具有统一命名规则的情况。例如，在出版领域，每本书都有特定的国际标准书号(ISBN)，使书籍之间的关联变得简单。另一种是基于相似性的方法，比较两个实体并量化它们的相似性。如果相似度超过给定阈值，则认为应建立两个实体间的同质关联[6]。

在基于相似度的方法中，必须明确如何度量两个实体之间的相似性。现有的相似性度量方法有两种：一是句法相似度，例如字符串相似性，可以通过 Jaro-Winkler 距离[7]计算相似性，二是语义相似

度[8] [9] [10] [11]，可以由字典或更高层次的本体确定。最简单的匹配方法是使用字符串比较实体的名称[12] [13]，这种方法简单直观，但在匹配地理空间实体时，因为名称重复现象在地理空间领域非常普遍，所以简单的名称匹配效果并不是很好。[14]提出了一种基于 Tversky 对比模型[15]的方法，通过计算地理空间实体的名称，城市和省份属性的相似性的加权和来确定两个地理空间实体之间的相似性。这种方法在一定程度上解决了名称重复的问题。Pschorr 等人[16]通过比较经纬度来构建传感器数据和 GeoNames 数据库之间的关联。Auer 等人[2]通过计算名称和空间相似度来实现 LinkedGeoData 和 DBpedia 之间的匹配，其中，空间相似度由值域在 0 到 1 之间的二次函数确定：如果两个点完全相符，那么函数的值将为 1，如果两点之间的距离达到预定义的最大距离，函数值为 0。但是，每种空间实体的最大距离不是显而易见的，而且要为不同类型的空间实体定义不同的最大距离。Silk [17]是一个关联探测框架，它提供了 Silk-Link 规范语言(LSL)，允许用户自定义应关联哪些数据集，以及应该使用哪些规则来关联它们。Silk 有许多内置的相似度计算方法，如 JaroWinklerSimilarity，numSimilarity 和 taxonomicSimilarity。

因为不同的 RDF 数据发布者对同一个实体可能会有不同的描述，所以数据网允许发布者使用不同的 URI 来描述相同的实体，标识同一实体的不同 URI 称为 URI 别名。因此，数据网中包含许多不同的 URI，却指的是同一个实体。为了消除信息孤岛，构建一个全球的数据网络，这些 URI 别名通过谓语 <http://www.w3.org/2002/07/owl#sameAs> [18]互相关联。这种类型的关联称为同质关联。本文提出了一种基于 MapReduce 的并行计算方法来构建地理空间实体之间的同质关联。

2. 地理空间实体的同质关联

建立地理空间实体的同质关联首先需要一个可区分的特征来计算地理空间实体之间的相似性。在现实世界中，语言是人类沟通的载体，为了准确地交流沟通，所有的实体都被赋予了一个名称，在虚拟世界中，实体的标识符应该更精确、唯一和明确，以实现机器之间的通信。例如，存储在计算机中的文件由整数唯一标识；在出版领域，图书用 ISBN 唯一标识。然而，目前国际上没有公认的唯一标识地理空间实体的标准，名称重复现象在地理空间领域是非常普遍的。据初步统计，美国以 Madison 命名的有 28 个城市，以 Clinton 命名的有 25 个城市，23 个 Washington，16 个 Lincoln，17 个 Jackson。另外，作为一个移民国家，许多市县都是以 Denmark，Sweden，Peru，England，Sydney 等国家或地区名命名。除了名称重复现象之外，一些数据集提供了不同语言的地理空间实体名称。例如，NUTS 是一个为欧盟提供地理信息的数据集，它为地理空间实体提供了不同语言的名称，如德国的名称属性有 Deutschland (德语中的德国)和 Germany (英语中的德国)，意大利的名称属性有 Italia (意大利语中的意大利)和 Italy (英语中的意大利)。如图 1 所示，地理空间数据集 A 和 B 都有四个实体：A 中的 Italia 和 B 中的 Italy 描述同一个地理空间实体，A 中的 Deutschland 和 B 中的 Germany 描述同一个地理空间实体，由于地名重复，B 中，有两个 Madison，如若只匹配地理空间实体的名称，A 中 Madison 将和 B 中两个 Madison 都关联起来，而事实是 B 中只有一个 Madison 与 A 中的 Madison 对应，因此仅仅匹配地理空间实体的名称不是一个完备的策略。在地理空间实体的所有属性中，几何属性(包括位置和形状)对于每个实体来说是唯一的。因此，用这些实体的几何属性来匹配它们是一个较好的策略。本文采用 Hausdorff 距离计算实体之间的几何属性相似度来量化实体之间的相似度。

一些特定类型的地理空间实体，如城市，州和国家，在地理空间数据库中被抽象为多边形，每个多边形由大量的边界点来描述。包含大量边界点的 Hausdorff 距离的计算是复杂且耗时的。MapReduce 框架是一种用于处理大型数据集的并行计算模型，有望加快匹配过程。为了减少匹配过程的运行时间，提高效率，本文提出了一种基于 MapReduce 框架的同质关联方法。

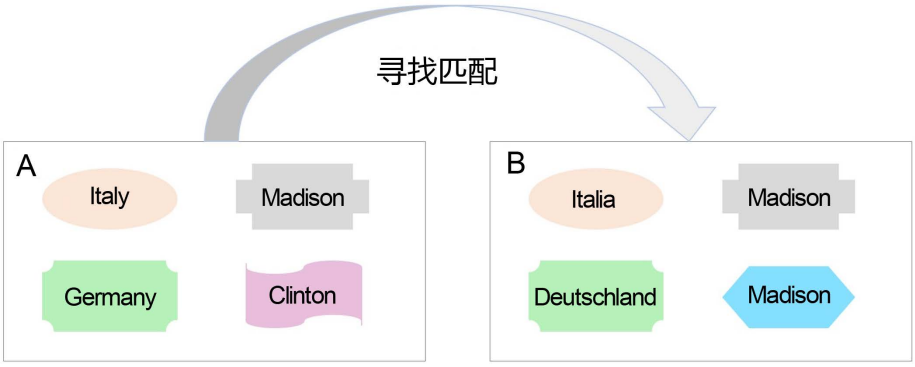


Figure 1. Finding matches between geospatial datasets A and B
图 1. 关联地理空间数据集 A 和 B

3. 方法

如上所述，对于地理空间实体，属性数据如名称不是唯一的，而提供位置信息和形状信息的几何数据是唯一的。因此，本文通过 Hausdorff 距离计算两个地理空间实体之间的空间相似性，通过匹配实体的几何数据构建地理空间实体之间的同质关联，与属性数据相比，几何数据具有大数据的特征，匹配过程十分耗时。因此，为了提高效率，本文提出了基于 MapReduce 的并行框架进行匹配的方法。

3.1. Hausdorff 相似性度量

Hausdorff 距离隐含地计算了实体之间的位置和形状相似度，非常适合用于计算地理空间实体的空间相似度。

集合 A 和集合 B 之间的 Hausdorff 距离定义为 $h(A, B)$和 $h(B, A)$中较大者，如公式(1)所示。

$$H(A,B)=\max\left(h(A,B),h(B,A)\right) \tag{1}$$

其中 $h(A, B)$表示从 A 到 B 的定向 Hausdorff 距离，$h(B, A)$表示从 B 到 A 的定向 Hausdorff 距离，它们的定义如公式(2)和(3)所示。

$$h(A,B)=\max_{a\in A}\min_{b\in B}\|a-b\| \tag{2}$$

$$h(B,A)=\max_{b\in B}\min_{a\in A}\|a-b\| \tag{3}$$

其中 $\|\bullet\|$是距离范数的一种类型。本文采用欧几里得距离。以 $h(A, B)$为例，首先计算集合 A 中的每个元素 a_i 到数据集 B 的最小距离，这个最小距离被定义为 a_i 与 B 中的每个元素 b_i 之间距离的最小值，然后计算这些最小值中的最大值，即为 $h(A, B)$。$h(B, A)$的计算类似。

为了生成阈值从 0 到 1 的相似性度量，本文使用公式(4)将 Hausdorff 距离归一化。

$$sim(A,B)=1-\frac{H(A,B)}{L_D} \tag{4}$$

其中 $H(A, B)$是 A 和 B 之间的 Hausdorff 距离。假设 R_A 是 A 的边界矩形，R_B 是 B 的边界矩形，则 L_D 指能覆盖 R_A 和 R_B 的最小矩形的对角线长度。

3.2. 基于 MapReduce 的并行匹配框架

由于地理空间实体的几何数据具有大数据的特性，所以地理空间实体之间的 Hausdorff 相似性计算非

常耗时。本文提出了一种基于 MapReduce 的并行匹配框架。

MapReduce 是大型数据集并行处理的编程模型[19]。模型的所有输入和输出都是键/值对的形式。该模型有两个主要函数：Map 和 Reduce。Map 接收一个输入键值对，以用户自定义的方式处理输入数据并生成中间键/值对。然后，MapReduce 库将这些中间键/值对分成许多组，每组含有相同的键，并将它们传递给 Reduce。每个 Reduce 都会接收一个键和该键的一组值，以用户自定义的方式处理它们并生成新的键/值对作为输出。

基于 MapReduce 的匹配方法的基本思想如下。给定两个数据集 $A=\{a_1,a_2,\cdots,a_m\}$ 和 $B=\{b_1,b_2,\cdots,b_n\}$，为找到 A 中任一实体 $a_i\ (i=1,\ 2,\ 3,\cdots,\ m)$ 在数据集 B 中的同质实体，首先计算 a_i 和 B 中的每个元素 $b_j\ (j=1,\ 2,\ 3,\cdots,\ n)$ 的 Hausdorff 相似度，然后找到这些相似度值中的最大值，即找到 b_r，使得 Hausdorff 相似度 $sim(a_i,b_r)=\max\left(sim\left(a_i,b_j\right)\right)$，最后，将 $sim(a_i,b_r)$ 与预定义的阈值进行比较，如果超过阈值，则建立 a_i 和 b_r 之间的同质关联。

为了从数据集 B 中找到数据集 A 中任一元素的同质元素，必须计算 A 和 B 的笛卡尔乘积；即必须提取每一个可能的元素对，然后计算每一对的 Hausdorff 相似度。这个过程非常耗时，特别是当数据集很大时。这里提出的基于 Map Reduce 的并行框架可以显著地减少运行时间，提高效率。如图 2 所示，假设数据集 A 有 m 个元素，数据集 B 有 n 个元素，且 n 远大于 m。数据集 B 使用 MapReduce 库分割，数据集 A 被加载到内存中并共享至各个节点。Map 的输入是 B 中的一个元素 b_j 和整个数据集 A，计算 b_j 与 A 中的每个实体之间的 Hausdorff 相似度，并输出 m 个键值对 $\left(a_i,sim\left(a_i,b_j\right)\right)$，其中 i 的取值范围是 1 到 m，j 对于每个 Map 来说是一个常数。然后，Map-Reduce 库将它们分成具有相同键的组，并将它们传递给 Reduce。每个 Reduce 接收 A 中的一个实体 a 和 B 中的所有元素以及 a 与 B 中所有元素的 Hausdorff 相似度，即 n 个键/值对 $\left(a_i,sim\left(a_i,b_j\right)\right)$，其中 j 的范围从 1 到 n，i 对于每个 Reduce 来说是个常数。Reduce 比较接收到的所有 Hausdorff 相似度并找到最大值。如果最大 Hausdorff 相似度超过预定阈值，它将输出键/值对 $\left(a_i,\max\left(sim\left(a_i,b_j\right)\right)\right)$。例如，假设当 j 等于 r 时，$sim\left(a_i,b_j\right)$ 取得最大值，输出键值对 $\left(a_i,sim\left(a_i,b_r\right)\right)$，建立 a_i 和 b_r 之间的同质关联。最后，RDF 三元组 $a_i$owl:sameAsb_r 将被发布。表 1 指明了每个步骤中的键/值对并详细描述了该算法。

Table 1. Hausdorff similarity-matching algorithm
表 1. Hausdorff 相似度匹配算法

算法：Hausdorff 相似性同质关联

```
map(key, value){
/*key: void*/
/*value: identity + coordinates of one entity from GADM*/
for(i=0; i<NUTS. size; i++){
        sim = HausSim(NUTS[i].coordinates, value. coordinates)
emit(NUTS[i].identity, value. identity + sim)
    }
}

reduce(key, value){
/*key: identity of one entity from NUTS */
/*value: identities of entities in GADM + their Hausdorff similarities with the entity in the key */
while(value. hasNext){
            if(value.sim>maxSim){
maxSim=value.sim;
maxIden=value.Iden
            }
      }
      if(maxSim>threshold)
emit(key, maxIden)
}
```

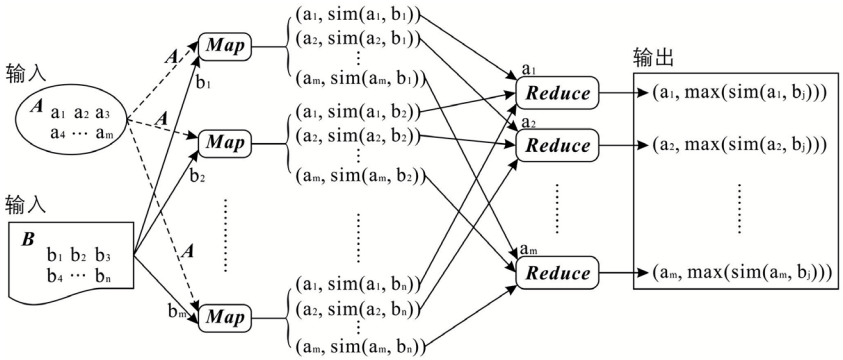


Figure 2. Map-Reduce workflow. Dataset B is split using Map-reduce, whereas dataset A is shared
图 2. Map Reduce 流程图。数据集 B 使用 Map Reduce 库分区，数据集 A 被共享

4. 实验

实验运行在八个计算节点上，每个节点有四个核、8 GB 内存。每个节点上部署 Hadoop，Hadoop 是一个 MapReduce 框架的实现。

实验使用 NUTS 和 GADM 的数据来验证方法的有效性。NUTS 提供欧盟经济领土的地理信息，GADM 提供世界行政区域的地理信息。实验的工作流程如图 3 所示。

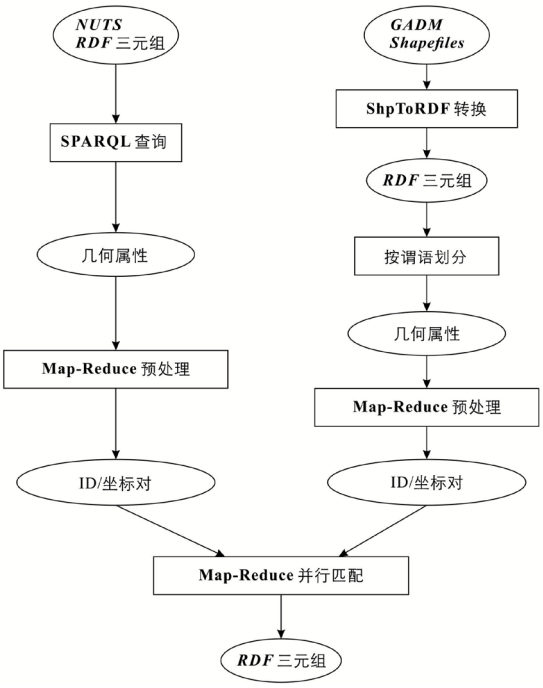


Figure 3. Experiment workflow
图 3. 实验流程图

4.1. 数据预处理

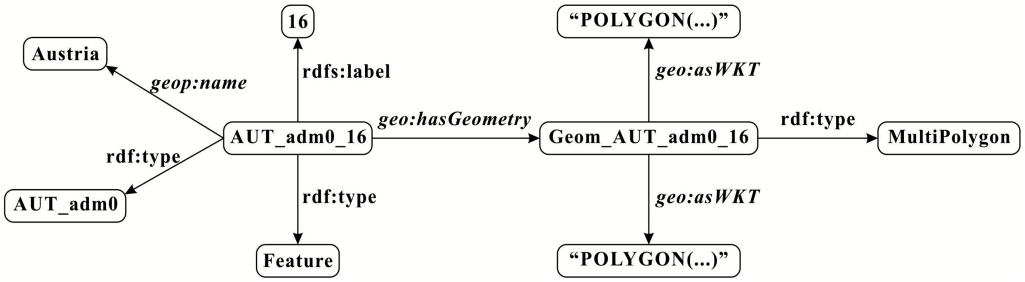
GADM 发布者将其数据转换为 RDF 格式在 GADM-RDF 中发布。然而，从 GADM-RDF 中下载的数据仅包括属性数据，如名称，id 和类型，不包括几何数据。由于 NUTS 数据是欧盟经济领土数据，本文首先从 GADM 上下载了欧洲原始的 Shapefile，然后转换为 RDF。共 98,493 个实体。实验中的奥地利数

据如图 4 所示。该图包含两个核心节点：*AUT_adm0_16* 和 *Geom_AUT_adm0_16*。*AUT_adm0_16* 是一个名为奥地利的实体，标记为 16；它是一种 *geo*: *Feature*，并分类为 *AUT_adm0*。*Geom_AUT_adm0_16* 是表示 *AUT_adm0* 的空间范围的几何体。它是一个 *MultiPolygon*，它的 *WKT* (*Well-known text*)序列化通过谓词 *geo: asWKT* 与其本身关联。

然后，将数据上传到 Hadoop 分布式文件系统(HDFS)，并根据谓词进行拆分。使用相同谓词的三元组被拆分成相同的文件，每个文件都以谓词命名。几何数据被分到名为 geosparql#asWKT 的文件，此文件将用于以后的匹配。

NUTS 为欧盟提供了四个级别的粗粒度数据：NUTS0 提供国家层面的数据，NUTS1 提供主要社会经济区域的数据，NUTS2 提供应用层面的基本区域数据，NUTS4 为特定应用提供小区域数据。NUTS 中奥地利的一部分数据如图 5 所示。由于 GADM 提供世界行政区域的数据，所以使用 RDF 查询语言(SPARQL)来提取 NUTS2 中的实体与之匹配。

接下来，将 NUTS2 中提取的数据和从 GADM 拆分的 geosparql#asWKT 文件处理成键/值对，以供 Map-Reduce 使用，其中键为空间实体的 URI，值是其边界的坐标。



(a)

```
<http://geop.whu.edu.cn/GeoData#AUT_adm0_16>
<http://geop.whu.edu.cn/GeoData#name> "Austria"@en .
<http://geop.whu.edu.cn/GeoData#AUT_adm0_16> <http://www.w3.org/1999/02/22-
rdf-syntax-ns#type>
<http://geop.whu.edu.cn/GeoData#featureWithoutClass> .
<http://geop.whu.edu.cn/GeoData#AUT_adm0_16> <http://www.w3.org/1999/02/22-
rdf-syntax-ns#type>
<http://geop.whu.edu.cn/GeoData#AUT_adm0> .
<http://geop.whu.edu.cn/GeoData#AUT_adm0_16> <http://www.w3.org/1999/02/22-
rdf-syntax-ns#type>
<http://www.opengis.net/ont/geosparql#Feature> .
<http://geop.whu.edu.cn/GeoData#AUT_adm0_16>
<http://www.w3.org/2000/01/rdf-schema#label> "16"@en .
<http://geop.whu.edu.cn/GeoData#AUT_adm0_16>
<http://www.opengis.net/ont/geosparql#hasGeometry>
<http://geop.whu.edu.cn/GeoData#Geom_AUT_adm0_16> .
<http://geop.whu.edu.cn/GeoData#Geom_AUT_adm0_16>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>
 <http://www.opengis.net/ont/sf#Polygon> .
<http://geop.whu.edu.cn/GeoData#Geom_AUT_adm0_16>
<http://www.opengis.net/ont/geosparql#asWKT>
"POLYGON ((10.454459190368652 47.55573654174805, 10.454330444335938
47.55569458007835,...
 10.454459190368652
47.55573654174805))"^^<http://www.opengis.net/ont/geosparql#wktLiteral> .
<http://geop.whu.edu.cn/GeoData#Geom_AUT_adm0_16>
<http://www.opengis.net/ont/geosparql#asWKT>
```

(b)

Figure 4. The data for Austria. (a) Austrian data shown in the RDF graph. (b) Austrian data shown in N-TRIPLES format
图 4. 奥地利数据。(a)奥地利 RDF 图；(b)奥地利 N-TRIPLES 格式数据

```
nuts:AT rdf:type ramon:NUTSRegion .
nuts:AT rdfs:label "AT - ÖSTERREICH" .
nuts:AT ramon:name "ÖSTERREICH" .
nuts:AT ramon:level "0"^^<http://www.w3.org/2001/XMLSchema#integer> .
nuts:AT ramon:code "AT" .
nuts:AT ngeo:geometry nuts:AT_geometry .
nuts:AT_geometry dc:rights "© EuroGeographics for the administrative
boundaries." .
      nuts:AT_geometry rdf:type ngeo:Polygon .
   nuts:AT_geometry ngeo:exterior _:d1e1927 .
   _:d1e1927 rdf:type ngeo:LinearRing .
   _:d1e1927 ngeo:posList (
     [ geo:long "9.620600550000063";

     geo:lat "47.151568250000054" ]
    ...
     [ geo:long "-1.129388949999935";

     geo:lat "46.31027975" ]
) .
```

Figure 5. A portion of the Austrian data in NUTS
图 5. NUTS 中的奥地利部分数据

4.2. 实体匹配

将从 NUTS 和 GADM 提取的实体根据 3 提出的方法进行匹配。从 NUTS 提取的数据相对较小；因此，NUTS 数据作为数据集 A 被加载到内存中并被共享至各个节点，而从 GADM 中提取的数据作为数据集 B 使用 HDFS 分割。相似性阈值设置为 0.8。

4.3. 匹配结果

如表 2 所示，NUTS2 中存在 317 个实体，发现了 169 个匹配项。其中 163 个是正确的。精确度定义为正确匹配数除以发现的匹配总数；召回率定义为正确匹配的数量除以 NUTS2 的总实体数。在表 2 中可以看到精确度高，召回率低，这是希望看到的现象，因为设置不正确的关联比起未找到匹配更为严重。

Table 2. Matching result
表 2. 匹配结果

数据类型	实体数量	匹配数量	正确匹配数量	精确度	召回率
NUTS 2	317	169	163	96.4%	51.4%

最后，这些匹配结果被发表为谓语为 owl:same 的三元组。例如，图 6 显示了意大利在 NUTS 和 GADM 中的数据，意大利最终发布的三元组如图 7 所示。

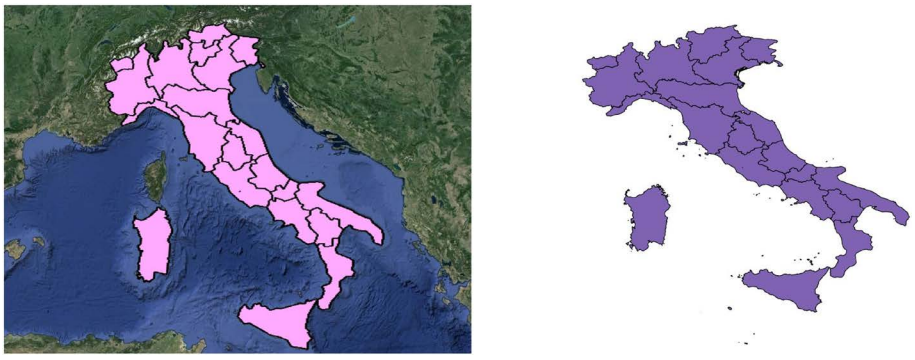


Figure 6. Illustration of the data for Italy: Data from NUTS (left) and data from GADM (right)
图 6. 意大利可视化数据：左图为 NUTS 数据，右图为 GADM 数据

```
<http://nuts.geovocab.org/id/ITC1> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_13>
<http://nuts.geovocab.org/id/ITC2> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm2_103>
<http://nuts.geovocab.org/id/ITC3> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_9>
<http://nuts.geovocab.org/id/ITC4> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_10>
<http://nuts.geovocab.org/id/ITD1> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm2_99>
<http://nuts.geovocab.org/id/ITD2> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm2_100>
<http://nuts.geovocab.org/id/ITD3> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_20>
<http://nuts.geovocab.org/id/ITD4> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_7>
<http://nuts.geovocab.org/id/ITD5> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_6>
<http://nuts.geovocab.org/id/ITE2> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_18>
<http://nuts.geovocab.org/id/ITE3> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_11>
<http://nuts.geovocab.org/id/ITE4> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_8>
<http://nuts.geovocab.org/id/ITF1> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_1>
<http://nuts.geovocab.org/id/ITF2> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_12>
<http://nuts.geovocab.org/id/ITF3> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_5>
<http://nuts.geovocab.org/id/ITF4> <http://www.w3.org/2002/07/owl#sameAs>
<http://geop.whu.edu.cn/GeoData#ITA_adm1_2>
```

Figure 7. Published triples of Italy
图 7. 发布的意大利三元组

4.4. 运行时间和可扩展性

如表 3 所示，随着节点数量的增加，匹配所需的时间显著减少：当只有一个节点需要一天以上的时间进行匹配，两个节点需要 11.35 h，四个节点需要 5.75 h，八个节点仅需要 3.17 小时。图 8 显示了运行时间随节点数量的变化。一开始，运行时间迅速减少，随后降低速度逐渐减慢。使用八个节点可以将运行时间从多于一天减少到仅三个小时左右。

Table 3. Speedup as the number of nodes increases
表 3. 随着节点增加的数据处理增速

节点数	容器	运行时间	增速
1	0	26.34 h	1
2	4	11.35 h	2.32
4	8	5.75 h	4.58
8	16	3.17 h	8.31

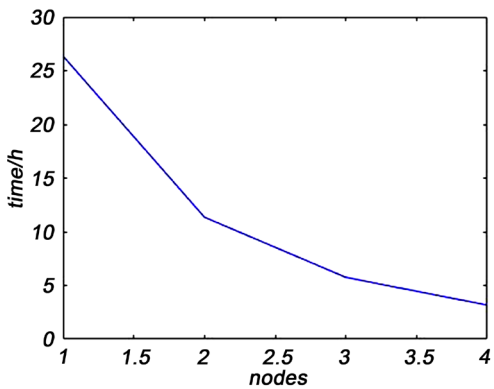


Figure 8. Runtime as a function of the number of nodes
图 8. 运行时间随节点数的变化

本实验使用了 Hadoop-2.5.2。Hadoop 2.x 和 Hadoop 1.x 之间的主要区别是引入了 YARN (Yet Another Resource Negotiator)。YARN 以容器为单位调度和分配集群的资源。容器是资源的封装抽象，为每个任务封装资源，如 CPU，内存和磁盘。YARN 为所有任务动态分配容器，每个任务占用一个容器。基于我们的 Hadoop 配置，一个节点可以同时存在两个容器，因此，八个节点可以同时存在十六个容器。其中一个容器被应用程序主机占用，两个节点可用容器为三个，四个节点可用容器为七个，八个节点可用容器为十五个。实验中增加节点的增速不如理论线性值(理论线性值分别为 3，7 和 15)的主要原因是分配的任务总数为十。当使用八个节点时，十个 Map 任务同时运行。使用四个节点时，首先，七个 Map 任务同时运行，三个任务在队列中等待，有任务完成时，队列中的等待任务再启动。类似地，当使用两个节点时，同时运行三个 Map 任务，剩下七个任务等待，直到有其他任务完成。当所有 Map 任务完成后，Reduce 任务就会启动。另一方面，在节点上启动任务的 Hadoop 开销和节点之间必要的通信也相应减少了增速值。

5. 讨论

本文提出的基于 MapReduce 的并行关联框架适用于一个相对较小的数据集和一个大型数据集之间的匹配，这种情况在现实世界中经常会遇到，例如，在我们的实验中使用的两个数据集 NUTS 和 GADM，NUTS 是一个欧洲小数据集，GADM 是一个全球大数据集。

要执行关联任务，首先要确定某种类型的实体的标识符或描述符。显然，理想情况下，如果标识符或描述符可以唯一地标识实体，关联过程将会很简单，关联结果精度也会很高。比如，ISBN 可以唯一地标识书籍，不同数据库中书籍的关联便简单准确。而对于地理空间实体，不存在与 ISBN 类似的可以唯一地标识实体的标识符。另外，名称重复现象在地理空间领域非常普遍，而且，一些数据集中的实体的名称属性以不同的语言呈现，例如，在 NUTS 中，德国的名称属性有两个，一个是德语中的德国 Deutschland，一个是英语中的德国 Germany。在地理空间实体的所有属性中，理论上，只有传递位置和形状信息的几何属性能唯一地识别地理空间实体。因此，本文采用 Hausdorff 相似性度量，它隐含地计算了两个实体之间的位置相似性和形状相似性。在未来的工作中，还可以计算名称相似度和空间相似度的加权和来建立关联，对于名称属性不局限于一种语言的，可以使用字典将所有名称翻译成对应的英文。

在匹配过程中，必须计算两个数据集中的所有可能实体对的 Hausdorff 距离，即两个数据集的笛卡尔乘积。因此，对于具有 m 个元素的数据集 A 和具有 n 个元素的数据集 B，必须执行 $m \times n$ 个 Hausdorff 相似度计算。Hausdorff 距离的计算是耗时的，当处理大数据集时，需要进行很多次 Hausdorff 相似度计算。在本文的实验数据中，如果只使用一个节点，需要一天多的时间才能完成该任务。本文提出的基于 MapReduce 的并行框架可以显著地减少运行时间，在八个节点上运行时，只需要大约三个小时。为了进一步提高效率，在未来的工作中，可以预定义一个距离阈值，事先筛选一遍可能的匹配，以减少需要 Hausdorff 相似度计算的对数，不过定义这个阈值不是一个容易的任务，不同层次的地理空间实体之间的距离是不同的，比如国家跟国家之间，城市跟城市之间，而且国内省份和省份之间的距离可能和欧洲国家与国家之间的距离大致相同。

此外，本文提出的地理空间匹配方法可以与著名的 SILK 框架集成，这是一个基于属性匹配的框架，基于 MapReduce 的并行匹配框架不仅可以用于构建同质关联，还可以用来构建其他两种类型的关联，词汇关联和关系关联，在构建另外两种关联时，Hausdorff 相似度计算需要由用于其他关联任务算法代替。

6. 结论

本文提出了一种基于 MapReduce 的地理空间实体的并行同质关联方法。由于位置和形状信息，即地理空间实体的几何属性可以唯一地识别一个实体，本文采用 Hausdorff 相似性度量来计算两个实体之间的

空间相似度。另外，由于地理空间数据本身很大，而且 Hausdorff 相似度计算十分耗时，本文提出了一个基于 MapReduce 的并行计算框架。实验通过匹配两个数据集 NUTS 和 GADM 检验了所提出的方法的有效性。当仅使用一个节点时，需要多于一天的时间，但是当使用八个节点并行计算时，只需要大约三个小时。此外，本文提出的基于 MapReduce 的并行框架除了同质关联也可以应用于其它两种类型的关联，此时，Hausdorff 相似度计算需要由用于其他关联任务算法来代替。

参考文献

[1] Auer, S., *et al*. (2007) DBpedia: A Nucleus for a Web of Open Data. *Proceedings of* 6*th International Semantic Web Conference and* 2*nd Asian Semantic WEB Conference*, Busan, 11-15 November 2007, 722-735. https://doi.org/10.1007/978-3-540-76298-0_52

[2] Auer, S., Lehmann, J. and Hellmann, S. (2009) Linked Geo Data: Adding a Spatial Dimension to the Web of Data. *Proceedings of International Semantic Web Conference*, Chantilly, 25-29 October 2009, 731-746.

[3] Mika, P. and Tummarello, G. (2008) Web Semantics in the Clouds. *IEEE Intelligent Systems*, **23**, 82-87. https://doi.org/10.1109/MIS.2008.94

[4] Hoffart, J., *et al*. (2013) YAGO2: A Spatially and Temporally Enhanced Knowledge Base from Wikipedia. *Artificial Intelligence*, **194**, 28-61. https://doi.org/10.1016/j.artint.2012.06.001

[5] Berners-Lee, T. (2006) Linked Data. http://www.w3.org/DesignIssues/LinkedData.html

[6] Heath, T. and Bizer, C. (2011) Linked Data: Evolving the Web into a Global Data Space. Morgan & Claypool, San Rafael.

[7] Winkler, W.E. (1990) String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage. In: *Proceedings of the Section on Survey Research Methods*, American Statistical Association, Alexandria, 354-359.

[8] Rodriguez, M.A. and Egenhofer, M.J. (2003) Determining Semantic Similarity among Entity Classes from Different Ontologies. *IEEE Transactions on Knowledge and Data Engineering*, **15**, 442-456. https://doi.org/10.1109/TKDE.2003.1185844

[9] Varelas, G., *et al*. (2005) Semantic Similarity Methods in WordNet and Their Application to Information Retrieval on the Web. In: *Proceedings of the* 7*th Annual ACM International Workshop on Web Information and Data Management*, ACM, New York, 10-16. https://doi.org/10.1145/1097047.1097051

[10] Nguyen, H.A. and Al-Mubaid, H. (2006) A Combination-Based Semantic Similarity Measure Using Multiple Information Sources. *IEEE International Conference on Information Reuse and Integration*, 16-18 September 2006, 617-621.

[11] Ge, J. and Qiu, Y. (2008) Concept Similarity Matching Based on Semantic Distance. 4*th International Conference on Semantics*, *Knowledge and Grid*, 3-5 December 2008, 380-383. https://doi.org/10.1109/SKG.2008.24

[12] Tejada, S., Knoblock, C.A. and Minton, S. (2001) Learning Object Identification Rules for Information Integration. *Information Systems*, **26**, 607-633. https://doi.org/10.1016/S0306-4379(01)00042-4

[13] Cohen, W.W., Ravikumar, P. and Fienberg, S.E. (2003) A Comparison of String Metrics for Matching Names and Records. *KDD Workshop on DATA Cleaning & Object Consolidation*, Washington, DC, Vol. 3, 73-78.

[14] Zhang, M., *et al*. (2013) An Interlinking Approach for Linked Geospatial Data. *International Archives of the Photogrammetry*, *Remote Sensing and Spatial Information Sciences*, **40**, 283-287. https://doi.org/10.5194/isprsarchives-XL-7-W2-283-2013

[15] Tversky, A. (1977) Features of Similarity. *Psychological Review*, **84**, 327-352. https://doi.org/10.1037/0033-295X.84.4.327

[16] Pschorr, J., *et al*. (2010) Sensor Discovery on Linked Data. *Proceedings of the* 7*th Extended Semantic Web Conference*, Heraklion.

[17] Volz, J., *et al*. (2010) Silk—A Link Discovery Framework for the Web of Data. LDOW, 538.

[18] Bizer, C., Cyganiak, R. and Heath, T. (2007) How to Publish Linked Data on the Web. http://wifo5-03.informatik.uni-mannheim.de/bizer/pub/LinkedDataTutorial/

[19] Dean, J. and Ghemawat, S. (2004) MapReduce: Simplified Data Processing on Large Clusters. *Communications of the ACM*, **51**, 107-113. https://doi.org/10.1145/1327452.1327492

知网检索的两种方式：

1. 打开知网页面 http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD
   下拉列表框选择：[ISSN]，输入期刊 ISSN：2329-549X，即可查询
2. 打开知网首页 http://cnki.net/
   左侧“国际文献总库”进入，输入文章标题，即可查询

投稿请点击：http://www.hanspub.org/Submission.aspx
期刊邮箱：gst@hanspub.org