目录

[第一章 绪论 2](#_Toc37256484)

[1.1研究背景和意义 2](#_Toc37256485)

[1.2 GIS的概述 2](#_Toc37256486)

[1.3 本文的主要工作及切入点 2](#_Toc37256487)

[1.4本文的组织结构 2](#_Toc37256488)

[第二章 地理位置信息 2](#_Toc37256489)

[2.1地理位置数据简介 2](#_Toc37256490)

[2.2 Openstreemap简介 2](#_Toc37256491)

[2.3数据格式实践 2](#_Toc37256492)

[第三章 最短路径算法 2](#_Toc37256493)

[3.1概念 2](#_Toc37256494)

[3.2常见算法 2](#_Toc37256495)

[3.3 Dijkstra算法 3](#_Toc37256496)

[3.4 A\*算法 3](#_Toc37256497)

[第四章Postgresql数据库技术 3](#_Toc37256498)

[4.1 数据库简介 3](#_Toc37256499)

[4.2数据库特点 3](#_Toc37256500)

[4.3对于gis方面的可扩展性 3](#_Toc37256501)

[第五章 实践 3](#_Toc37256502)

[5.1工具准备 3](#_Toc37256503)

[5.2数据准备 3](#_Toc37256504)

[5.3搭建本地数据库 3](#_Toc37256505)

[5.4设计数据结构导入数据 3](#_Toc37256506)

[5.5呈现在地图上 3](#_Toc37256507)

[第六章 总结 3](#_Toc37256508)

[6.1本文总结 3](#_Toc37256509)

[6.2未来展望 4](#_Toc37256510)

# 第一章 绪论

本章首先介绍了GIS系统的研究背景和意义，其次简单介绍了从算法、数据源、数据结构等各个系统组成部分及其作用，并概述了本文所做的主要工作和贡献以及创新点，在本章的最后介绍了论文的组织结构。

## 1.1研究背景和意义

## 1.2 GIS的概述

地理信息系统（Geographic Information System或 Geo－Information system，GIS）有时又称为“地学信息系统”。它是一种特定的十分重要的空间信息系统。它是在计算机硬、软件系统支持下，对整个或部分地球表层（包括大气层）空间中的有关地理分布数据进行采集、储存、管理、运算、分析、显示和描述的技术系统。

位置与地理信息既是LBS的核心，也是LBS的基础。一个单纯的经纬度坐标只有置于特定的地理信息中，代表为某个地点、标志、方位后，才会被用户认识和理解。用户在通过相关技术获取到位置信息之后，还需要了解所处的地理环境，查询和分析环境信息，从而为用户活动提供信息支持与服务。

## 1.3 本文的主要工作及切入点

随着GIS数据及其各种应用（通常与原始数据集的目的不同）的可用性和可访问性不断扩展，GIS数据集的表征和质量评估变得越来越重要。生成，传输和利用的地理参考数据的数量不断增加，并且在空间数据库中嵌入的不确定性数量已成为具有至关重要的理论重要性和实际考虑因素的主要问题。同时，自愿地理信息（VGI）成为地理信息的重要来源。 OpenStreetMap（OSM）是VGI的最著名来源之一。 OSM的大多数数据是由“非专家”和“业余地理学家”收集的，这引起了专业GIS社区对OSM质量的严重关注。文献中很少出现OSM数据已用于GIS建模，空间分析或空间统计的例子。在等。 [13]等人结合OSM数据和数字地形模型（DTM）数据描述了城市3D模型的开发，但作者评论说，“尽管在德国，OSM街道网络数据库几乎是完整的OSM数据并未在地理信息学”。 Boin和Hunter [2]指出，对于数据使用者来说，使用VGI“他们需要某种程度的数据质量度量，以做出明智的选择，以减少或吸收空间数据中的不确定性”。在本文中，我们研究了OSM质量度量标准的发展：既可用于隔离（不进行地面真实性比较的OSM数据分析），也可用于访问地面真实性数据。实验结果表明OSM数据存在一些严重问题。 OSM成立于2004年，从最初的小规模发展到2009年底的200,000多个贡献者。Haklay [11]显示，到2010年3月，英国OSM在英国的覆盖率已从去年的51.2％增长到69.8％。 。 Zielstra和Zipf [16]评论说，2009年德国的OSM数据量在不到三个月的时间内增长了20％。 Ciepluch等。等人[5]详细介绍了建立OSM数据库和服务器系统所涉及的步骤。收集数据并将其上传到OSM的标准方法是：（1）使用GPS设备收集数据，或（2）从公开的航空影像中跟踪多边形，折线等的轮廓。雅虎！已经同意让OSM将其航拍图像用于OSM追踪。 NASA制作的卫星卫星图像也可以用作OSM的来源。当导入活动扩展到导入政府或制图机构的空间数据时，只要可以在OSM许可证下使用，这两个数据上传到OSM的过程通常会迅速加快。本文的结构如下。在第2节中提供了对空间数据质量的讨论，其中特别着重于VGI域。在第3节中概述了OSM数据的实验分析。本文在第4节结束时，我们在第3节中对结果进行了一些讨论。以及这些结果的可能含义。还概述了正在进行的工作和将来的工作。

## 1.4本文的组织结构

# 第二章 地理位置信息

## 2.1地理位置数据

当移动设备的GPS芯片不能接收到GPS信号时，移动设备就需要与它所连接的手机信号塔通讯和估算它与信号塔之间的距离以不断报告它的地理位置。

美国科技博客下属研究机构BI Intelligence发表了一份报告，分析了地理位置数据是如何被收集的以及地理位置数据将如何改变整个移动行业。

现在，配备GPS功能的智能手机数量已经超过7.7亿部，地理位置数据已经开始渗透到整个移动领域。除了象FourSquare和Shopkick那样的消 费者应用之外，基于地理位置的服务在移动领域还有很多的应用机会。它可以支持广告和许多其他的服务，比如天气、旅游类应用。

报告特别分析了基于地理位置数据的移动广告是如何激发消费者的兴趣，基于地理位置数据的功能是如何提高应用使用率，并且揭开了基础技术和隐私问题的神秘面纱。

纯粹的GPS解决方案以及它所生成的经纬度标签是地理位置数据的公认标准。但是至少还有4种方法可以获得地理位置数据：

1、手机信号塔数据：当移动设备的GPS芯片不能接收到GPS信号时，移动设备就需要与它所连接的手机信号塔通讯和估算它与信号塔之间的距离以不断报告它的地理位置。通过这种方法获得的地理位置数据不如纯粹的GPS数据精确。

2、WiFi连接：这是一种能够精确获得地理位置数据的方法，但是需要用到有效的WiFi热点。WiFi的地址与GPS坐标是一一对应的。它可以准确地标示出用户所处的位置，因此很多零售商才愿意提供免费的公共WiFi服务，那样它们就能够发布店内移动广告了。

3、IP地址：地理位置也可以通过与数据连接有关的IP地址来获得。这种方法的准确性会因运营商而异，可靠性也比上述方法要低一些。

4、用户报告：当用户注册电子邮件或移动应用和服务时，他们通常都会输入自己的地理位置和邮政编码。这些数据可以被转换为GPS坐标，从而建立起用户群的地理位置档案。

## 2.2 Openstreemap简介

osm是一个开放式在线地图平台，Open Street Map（简称 OSM）是一个存储海量XML 数据的数据库（在本文中简称原始 OSM 数据），只要注册账号，

任何人均可以对其后台数据库进行编辑，从而被称为世界的维基地图。尽管众源地图的编辑过程难以监控管理，但通过近年来的完善和修订，OSM的质量

已经得到众多学者的认可。

我们可以在网址（http://download.geofabrik.de/）下载各国的OSM地图，可根据各自的需要下载，下载完成之后将文件改为osm格式。

## 2.3数据格式实践

# 第三章 最短路径算法

## 3.1概念

在图论中，最短路径问题是在图中的两个顶点（或节点）之间找到路径的问题，以使其构成边的权重之和最小。可以将在道路地图上找到两个交叉点之间的最短路径的问题建模为图形中最短路径问题的特例，其中顶点对应于交叉点，边对应于路段，每个边均由路段的长度加权分割。也可以为图定义最短路径问题，无论是无向图，有向图或混合图。在此为无向图定义。对于有向图，路径的定义要求通过适当的有向边连接连续的顶点。其中以Dijkstra算法和A\*算法最为常见。

## 3.2 Dijkstra算法

Dijkstra的算法（或Dijkstra的最短路径优先算法，SPF算法）[2]是一种用于在图形中查找节点之间最短路径的算法，该路径可能代表例如道路网络。它是由计算机科学家Edsger W. Dijkstra于1956年构思的，并在三年后出版。

该算法存在多种变体。 Dijkstra的原始算法找到了两个给定节点之间的最短路径，[5]但更常见的变体将单个节点固定为“源”节点，并找到从源到图中所有其他节点的最短路径，从而产生最短路径树。

对于图中的给定源节点，该算法查找该节点与其他节点之间的最短路径。[6]：196-206它也可以用于通过停止从单个节点到单个目标节点的最短路径。一旦确定了到达目标节点的最短路径，就可以使用该算法。例如，如果图形的节点表示城市，并且边路径成本表示通过直路连接的成对城市之间的行驶距离（为简单起见，请忽略红灯，停车标志，收费公路和其他障碍物），则可以使用Dijkstra算法寻找一个城市和所有其他城市之间的最短路线。最短路径算法的一种广泛应用是网络路由协议，最著名的是IS-IS（中间系统到中间系统）和开放式最短路径优先（OSPF）。它也被用作Johnson算法等其他算法的子例程。

Dijkstra算法使用的标签是正整数或实数，它们是完全有序的。如果后续标签（遍历边缘时生成后续标签）单调非递减，则可以通用使用部分排序的标签。这种概括称为通用Dijkstra最短路径算法。[7]

Dijkstra的算法使用数据结构来存储和查询从一开始就按距离排序的部分解决方案。尽管原始算法使用了最小优先级队列并按时间运行{\ displaystyle O（| V | + | E | \ log | V |）} {\ displaystyle O（| V | + | E | \ log | V | ）}（其中{\ displaystyle | V |} | V |是节点数，{\ displaystyle | E |} | E |是边数），也可以在{\ displaystyle O（V ^ {2}）} {\ displaystyle O（V ^ {2}）}使用数组。 Leyzorek等人也给出了该算法的思想。 1957. Fredman＆Tarjan 1984提出使用Fibonacci堆最小优先级队列来优化运行时间复杂度为{\ displaystyle O（| E | + | V | \ log | V |）} O（| E | + | V | \ log | V |）（其中{\ displaystyle | E |} | E |是边数）。对于具有无界非负权重的任意有向图，这是渐近已知的最快最快的单源最短路径算法。但是，确实可以进一步改进特殊情况（例如有界/整数权重，有向无环图等），如“特殊变体”中所述。

在某些领域，尤其是人工智能，Dijkstra的算法或它的一种变体被称为统一成本搜索，并被表述为最佳优先搜索的更普遍概念的实例。[8]

## 3.3 A\*算法

A \*（发音为“ A-star”）是一种图形遍历和路径搜索算法，由于其完整性，最优性和最优效率而经常在计算机科学中使用。[1]一个主要的实际缺陷是{{displaystyle O（b ^ {d}）} O（b ^ d）的空间复杂度，因为它将所有生成的节点存储在内存中。因此，在实际的旅行路线系统中，它通常比可以对图形进行预处理以获得更好的性能的算法[2]以及具有内存限制的方法要好。但是，在许多情况下，A \*仍然是最佳解决方案。[3]

斯坦福研究所（现为SRI International）的Peter Hart，Nils Nilsson和Bertram Raphael于1968年首次发布了该算法。[4]可以将其视为Edsger Dijkstra 1959年算法的扩展。通过使用启发式方法来指导搜索，A \*可获得更好的性能。

作为Shakey项目的一部分，创建了A \*，该项目的目的是构建一个可以计划自己的动作的移动机器人。尼尔斯·尼尔森（Nils Nilsson）最初提出使用图Traverser算法[5]进行Shakey的路径规划。[6] Graph Traverser由启发式函数{\ displaystyle h（n），} {\ displaystyle h（n），}指导，从节点{\ displaystyle n} n到目标节点的估计距离：它完全忽略{\ displaystyle g { n），} {\ displaystyle g（n），}从起始节点到{\ displaystyle n。} n的距离。 Bertram Raphael建议使用{\ displaystyle g（n）+ h（n）} {\ displaystyle g（n）+ h（n）}之和。[6]彼得·哈特（Peter Hart）发明了我们现在称为启发式函数的可容许性和一致性的概念。 A \*最初设计为在路径成本为边缘成本之和时查找成本最低的路径，但事实证明，A \*可用于为满足成本代数条件的任何问题找到最佳路径。[7]

1968年的原始A \*论文[4]包含一个定理，如果启发式函数是一致的并且适当选择了A \*的平局决胜规则，那么没有A \*类算法[8]可以扩展少于A \*的节点。几年后发表了“修正” [9]，声称不需要一致性，但是在Dechter和Pearl对A \*的最优性（现在称为最优效率）的权威研究中，这证明是错误的。的A \*的启发式算法是可允许的，但与可替代的类似A \*的算法相比，不能任意扩展更多的节点。

# 第四章Postgresql数据库技术

## 4.1 数据库简介

## 4.2数据库特点

## 4.3对于gis方面的可扩展性

# 

# 第五章 实践

## 5.1工具介绍与环境准备

工具：

PostgreSQL: 最核心的数据库工具，由于其非常强大的数据库扩展能力故此选择作为本次数据软件的选择对象。

PostGIS: PostGIS是一个开源软件程序，该程序将对地理对象的支持添加到PostgreSQL对象关系数据库中。 PostGIS遵循开放地理空间联盟（OGC）的SQL简单功能规范。

从技术上讲，PostGIS是作为PostgreSQL外部扩展实现的。它的实现基于“轻量级”的几何形状和优化索引以减少磁盘和内存占用的空间。使用轻巧的几何形状有助于服务器增加从物理磁盘存储迁移到RAM的数据量，从而大幅提高查询性能。

Osm2postgresql: 是一个Linux脚本，用于将openstreetmap数据加载到postgresql数据库中，并在需要时设置服务器和数据库（postgis + hstore）。执行高级处理，简化渲染过程。

Pgrouting:是基于开源空间数据库PostGIS用于网络分析的扩展模块，它只是利用Dijkstra算法实现最短路径搜索，之后慢慢添加了其他的路径分析算法，如A算法，双向A算法，Dijkstra算法，双向Dijkstra算法，tsp货郎担算法等，然后被更名为pgRouting。该扩展库依托PostGIS自身的gis索引，丰富的坐标系与图形类型，强大的几何处理能力，如空间查询，空间处理，线性参考等优势，能保障在较大数据级别下的网络分析效果更快更好

环境准备：

本地环境为 macOS Catalina version 10.15.4

其余软件均由mac系统下的第三方安装工具HomeBrew提供

brew install postgresql

版本 12.2

brew install postgis

版本2.5.0

brew install osm2postgresql

版本0.96.0

brew install pgrouting

3.0.0

## 5.2数据准备

1.访问 <https://www.openstreetmap.org/>下载gis数据源。最便捷的方式是通过网站整理好的各主要国家或城市的聚合数据，并以osm格式的文件下载下来。

A screenshot of a cell phone

Description automatically generated

## 

## 5.3搭建本地数据库

## 5.4设计数据结构导入数据

## 5.5呈现在地图上

# 第六章 总结

## 6.1本文总结

## 6.2未来展望