



中国海洋大学
OCEAN UNIVERSITY OF CHINA

顺序号(硕): SS021015
姓名: 李瑞姗
学号: '21100211107
学院: 信息科学与工程学院
专业: 计算机软件与理论

硕士学位论文

MASTER DISSERTATION

基于自然语言处理的多源 POI

论文题目: 数据融合的研究

英文题目: Multi-source POI Information Fusion
Based On Natural Language Processing

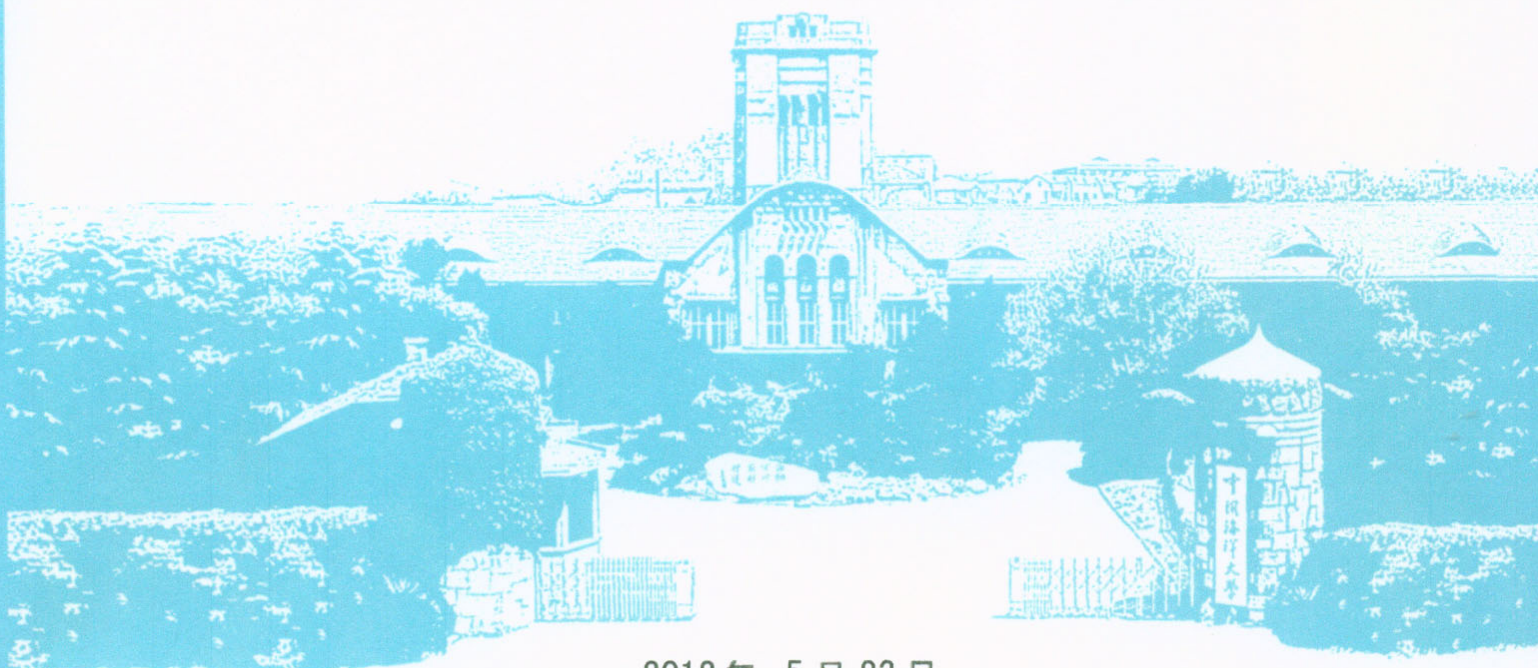
作者: 李瑞姗

指导教师: 张巍

学位类别: 全日制学术学位

专业名称: 计算机软件与理论

研究方向: 并行与分布式理论



2013年 5月 23日

基于自然语言处理的多源 POI 数据融合的研究

学位论文答辩日期: 2013.5.13

指导教师签字: 张磊

答辩委员会成员签字: 赵志刚

唐瑞军
田 爽

王 芳
林文光

知识产权保护协议

依据《中华人民共和国促进科技成果转化法》第二十八条和《中国海洋大学知识产权管理暂行规定（2004.7.20）》的有关规定，研究生李瑞妍（以下简称研究生）与其导师张魏（以下简称导师）就知识产权保护事宜达成如下协议：

1、研究生在校期间从事科研工作所完成的学位论文以及不论是否写入学位论文的其他成果属职务成果。研究生不得对上述职务成果以自己或他人名义擅自向第三方转让或泄漏。

2、研究生离校后三年内，不得擅自将在校期间从事科研工作的相关数据、研究结果和相关技术发表论文，不得擅自向第三方转让或泄漏。

3、研究生离校后三年内，若进行重复及延续在校研究课题的科技项目，必须经导师及中国海洋大学同意并协商知识产权分享事宜后，方可开展工作。

4、若研究生违反上述规定，导师及中国海洋大学有权追究其法律责任，即：要求其停止侵权行为、公开消除影响并予以经济赔偿。

5、本协议双方签字之日起生效，有效期三年。

研究生（签字）：李瑞妍

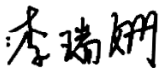
2013年5月10日



2013年6月30日

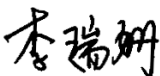
独 创 声 明


本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的
研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其
他人已经发表或撰写过的研究成果，也不包含未获得
(注：如没有其他需要特别声明的，本栏可空)或其他教育机构的学位或证书使
用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明
确的说明并表示谢意。

学位论文作者签名： 签字日期：2013年5月23日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，有权保留并
向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人
授权学校可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用
影印、缩印或扫描等复制手段保存、汇编学位论文。同时授权中国科学技术信息
研究所将本学位论文收录到《中国学位论文全文数据库》，并通过网络向社会公
众提供信息服务。(保密的学位论文在解密后适用本授权书)

学位论文作者签名：

导师签字：

签字日期：2013年5月23日

签字日期：2013年5月23日

谨以此文献给尊敬的张巍副教授以及我亲爱的朋友和同学们！

-----李瑞珊

基于自然语言处理的多源 POI 数据融合的研究¹

摘 要

近年来,由于基于位置的服务快速发展,尤其是对网络电子地图、移动位置服务(LBS)、便携式自动导航(PND)的使用,原有的兴趣点(POI)很难继续支撑这类服务。能否获取高质量的 POI 信息,成为此类服务的命脉所在。随着人们持币消费能力在迅猛增长,在日常消费、出行时,会将更多的注意力放在餐饮、娱乐、旅游等领域。这种不断增长的消费能力催生出了许多面向这一领域的信息提供商,他们所提供的信息内容丰富,并且实时性相对很高。

结合上述背景,如何获取蕴含在 web 中的大量有价值的 POI 信息点成为如今的一个热点问题,对这些已有的 POI 信息进行校正、融合,得到有利用价值的规整数据,这些工作具有重大的理论意义和实际的现实意义。本文在多源 POI 数据融合方面,包括 POI 各特征字段的表示、可融合 POI 的分类、经纬度字段的统一、网络访问受限等方面,进行了深入而系统的研究,具体的研究工作和研究成果如下:

(1) 通过分析 POI 中各特征字段的形式、特点,提出了 POI 特征相似度用以表示待分类 POI 与原有 POI 集的关系,以此进行之后的判断依据。相似度的形式化表示主要由名称、地理信息相似度两部分组成,其中的地理信息包括 POI 中的地址和经纬度。名称部分是通过几种经典字符串匹配方法计算得出的,地址部分根据地址的相似计算得出,经纬度部分利用 POI 之间的距离得出。

(2) 文中用到的 POI 中的经纬度是来源于不同网络电子地图上的坐标,同一实体在不同地图上的坐标不一致,对之后的 POI 融合工作造成了一定的影响。为解决这个经纬度标准不统一的问题,本文提到两种解决方法,即基于纠偏表的方法和基于 API 的方法。

(3) 构建了一个基于规则的分类模型,构建过程中设置 POI 各字段内部系数及阈值,经过回归计算,选取其区分 POI 是否可融合效果最好的一组系数和阈值构建出了判定模型。这个计算过程复杂、耗时,并且不够灵活,不具备自动学

¹Supported by the National Natural Science Foundation of China under Grant No.60602017(国家自然科学基金);Natural Science Foundation of Shandong Province under Grant No.ZR2012FM016(山东省自然科学基金)

习的能力。因此本文又利用机器学习分类器自身主动学习的能力，构造了几种不同的分类模型，比较之后选出了较优分类器，而实现分类性能的有效提升。

论文创新点如下：（1）考虑到因为词语的存在使得不同汉字具有不同的关联性，本文假设中文字符串匹配的最小单位是词，不再延用传统中最小单位是单个汉字的假设。（2）融合了 POI 的非空间信息和空间信息作为判定可融合 POI 的依据，后通过一个基于规则的模型对 POI 进行分类判断。（3）利用机器学习中的分类方法，构建了具备自主学习能力的 POI 可融合分类模型。

实验表明，本文提出的技术方法可以在基本无人工干预下自动、有效地对多源 POI 完成是否可融合判定。

关键词：数据融合；POI；分类；名称；地理信息

Multi-source POI Information Fusion Based On Natural Language Processing²

Abstract

In recent years, due to the rapid development of location-based services, especially on the network map, mobile location services (LBS), automatic portable navigation (PND), it has been difficult that the original point of interest (POI) support such services. Access to the high quality POI information, which was extremely important to location-based services. With the rapid growth in consumption, more attention has been placed on dining, entertainment, tourism and other fields in daily consumption. At the same time, it led to many information providers about this area, and the information they provided was informative and immediate.

In light of above, how to obtain numerous valuable POI information contained in web became a hot issue. Correcting and integrating these existing POI to get the structured valuable data are significant theoretical and practical. This paper systematically has studied multi-source poi information fusion, which includes expressing POI various features, classifying POI-fusion, uniforming coordinate field, solving the problem of limited network access, and so on. Specific research and results are as follows:

(1) Through the analysis of POI various field form and features, paper proposed POI characteristic similarity used to indicate relations between POI-fusion and the original POI collection to complete judgment. POI characteristic similarity mainly comprised of the name similarity, address similarity and coordinate similarity. Name part is calculated from several classic strings matching method, and address part is based on the section similar, and latitude-longitude part is the distance between two POI.

(2) POI's coordinate that appears in this paper comes from different network electronic map, and the coordinates of same entity on different maps are inconsistent that has certain influence to the later POI fusion work. To solve the problem that the

² Supported by the National Natural Science Foundation of China under Grant No.60602017 ;Natural Science Foundation of Shandong Province under Grant No.ZR2012FM016.

coordinate standards are not unified, this paper mentions two solutions. One is based on the correction table, and the other is based on API.

(3) Build a classification model based on rules. In the process, paper sets coefficient and threshold for POI various field, do regression calculations, and select the threshold which distinguish poi-fusion best to build decision model. This calculation process is complex, time-consuming, unflexible, and it do not have auto-learning capability. So paper uses machine learning classifiers which have active learning capacities to structure several different classification models. Then select the better classifier which effectively improve the classification performance.

Paper's innovations are as follows:

(1) Because there exist words and expressions, different Chinese character has the different relatedness. Considering that, this paper supposed the smallest unit that Chinese string match is the word, and no longer extends with the traditional supposition that the smallest unit is a single Chinese character.

(2) Integrated non-spatial and spatial information of POI, and use it as a basis for POI-fusion classification. Then through a model based on rules to categorize POI.

(3) Using the classified method about machine learning, build a POI fusion classification model with self-learning ability.

Experiments show that technique presented in this paper can automatically and effectively classify multi-source POI without human intervention.

Keywords: data;fusion;POI; classification;title;geographic information

目 录

1 引言	1
1.1 研究背景和意义	1
1.2 国内外研究现状	2
1.3 本文主要研究问题和内容	3
1.4 本文的章节安排	4
2 机器学习的分类模型	6
2.1 k-近邻算法	6
2.1.1 KNN 算法	6
2.1.2 距离加权最近邻算法	8
2.1.3 k-近邻的两个实践问题	8
2.2 决策树 C4.5	9
2.2.1 决策树简介	9
2.2.2 ID3 算法	10
2.2.3 C4.5 算法	11
2.3 强分类器 AdaBoost	12
2.3.1 Adaboost 的基本算法	13
2.4 评估方法	14
2.5 Weka 简介	14
2.6 小结	16
3 基于各字段相似度的 POI 匹配	17
3.1 名称相似度	17
3.1.1 莱文史特距离算法	17
3.1.2 Jaccard 相似方法	18
3.1.3 Jaro 距离算法	18
3.1.4 字符串分词处理	18
3.2 地理信息的相似度	20
3.2.1 中文地址的相似度	20
3.2.2 空间地理信息相似度	25

3.3 国内经纬度的统一.....	25
3.3.1 问题描述	25
3.3.2 基于纠偏表的实现	28
3.3.3 基于 API 的实现	29
3.4 各字段匹配结果	31
3.4.1 字符串相似算法之间的比较	31
3.4.2 POI 的三个特征之间的比较	32
3.5 小结	33
4 可融合 POI 的分类	34
4.1 问题描述	34
4.2 基于规则的分类	35
4.2.1 分类模型	35
4.2.2 实验数据介绍	37
4.2.3 分类结果级分析	38
4.3 基于机器学习的分类	40
4.3.1 POI 相似度表示	40
4.3.2 机器学习分类模型	41
4.3.3 机器学习模型训练与分类过程	41
4.3.4 实验结果及分析	42
4.4 小结	43
5 多源 POI 数据融合系统	44
5.1 多源 POI 数据融合系统的简介	44
5.1.1 系统流程	44
5.1.2 网络地图的选择	46
5.1.3 http 代理服务的使用	47
5.2 小结	48
6 总结与展望	49
6.1 总结	49
6.2 展望	49

参考文献	51
致谢.....	54
个人简历	55
学术论文	55
研究项目	55

1 引言

1.1 研究背景和意义

POI(Point Of Interest)即兴趣点^[1],是在地理信息系统(GIS)中表示地理对象的术语,主要是指那些我们日常生活用到的地理实体,如政府部门、景点、学校、医院、银行、商业区、标志性建筑等。POI 表示地理对象时通常将这些实体抽象为一个点。每个 POI 点包含这个实体四个方面的信息:名称、地址、类型、经纬度,同时还可能有电话、评价等信息[2]。最近几年,由于基于位置的服务快速发展,尤其是对网络电子地图^[3]、移动位置服务(LBS)^[4]、便携式自动导航(PND)的使用,使得原有的 POI 显得如此简单,甚至说是简陋,以致其不能继续支撑这类服务。能否获取高质量的 POI 信息,成为此类服务的命脉所在。

目前国内尚处在发展期,政府机关、重要厂矿企业位置信息可能不会变化,但那些小的 POI 点就不能确定了,如沿街的一些小餐馆饭店,这种 POI 可能会频繁的更换名称,更有甚者早以改换门庭成为干洗店或花店。要知道像网络电子地图、LBS、PND 这样的平台不是想更新就能立刻更新,先不说这些服务从采集到制作、加密、审查、出版需要很长的时间,单单是维护这千万级的 POI 点,就需要数千人的队伍保证它的准确性,据了解,目前国内的位置服务企业没有一家具备这个实力。准确度是 POI 质量的一个重要因素^[5],追求每一个 POI 都是真实有效的,就要求 POI 的信息准确,位置准确。

国内经济快速发展的今天,人们持币消费能力也在迅猛增长,在日常消费、出行时,会将更多的注意力放在餐饮、娱乐、旅游等领域,并且他们需要的信息不再单单是店铺的名称、地点、经纬度这些简单信息,还包括电话、营业时间、特色、风格、图片等信息。与此同时,这类不增强的消费能力引导出很多相关领域的信息提供网站,例如美团网、大众点评网等,随着经济的发展其规模也在不断扩大。他们所提供的信息,内容丰富,并且实时性相对较高,利用数据挖掘^{[6][7]}的技术加以提取,规整格式,便可作为位置服务的信息源直接使用,具有较好的应用价值,同时还避免了数据的人工采集过程的低效率、高成本。

结合上述背景,随着 POI 需求的加大和相关 POI 的信息在 web 中的大量涌现,如何获取蕴含在 web 中的大量有价值的 POI 信息点成为如今的一个热点问题,对

这些已有的 POI 信息进行校正、融合^[8]，得到有利用价值的规整数据，这些工作具有重大的理论意义和实际的现实意义。

1.2 国内外研究现状

数据采集工作是 POI 数据生产商的一项重要工作，但是现在绝大多数生产商的采集方式仍旧沿用传统的人海采集方式，过程中雇佣大批量的调查、测绘人员，对一定范围的 POI 进行地毯式调绘作业[9]。其效率低，成本高，并且无法及时更新，因此部分厂家根据自己的经验，创造性地将数据采集工作转移到了室内。比如，文献[9]将 GPS 技术与实景影像相结合，在此基础上开发建立了 POI 快速采集系统，可实现 POI 的快速采集和更新；专业 POI 生产厂商卡贝斯对互联网数据做了实时监测，分类抓取互联网上同 POI 相关的信息[10]。以卡贝斯抓取餐饮企业相关 POI 为例，第一次在互联网上抓取了 22 万个餐饮相关的数据，第二个月继续抓取，拿到了 22.2 万的数据。因为不知道这 2000 个新出现的数据是否真实，所以采集后卡贝斯又通过特有的电话情景脚本进行数据的验真、完善。同样的方法可以完成所有类型的 POI 信息采集。但是对于不同行业的 POI，对应的电话脚本也不一样，且这种电话脚本并非完全自动化，其间还需要人工大量的控制，当需要验正的 POI 数量比较多时，这种采集方式同样不能满足我们对效率的要求。

和卡贝斯一样，大多远程采集机制可以充分把握住新出现的 POI 信息，但忽略了那些原有 POI 信息变化，使得数据的准确度降低。还是以餐饮业为例，餐馆的节假日活动可能会频繁的变化，按照卡贝斯的机制这部分信息就不能在 POI 中被更新，甚至餐馆因为迁址导致地址这一关键字段发生的变化，也不会被更新，造成这个 POI 价值骤减。还有些餐馆因经营不善而关门倒闭，但是他的 POI 信息仍然出现在数据库里，成为无用的“死点”，久而久之便会出现大量的冗余。

正如以上所述，来源于网络的 POI 信息，其准确性和可用性都有待验证，必须通过校正、融合后才能成为具有使用价值的信息。但这些 POI 数量巨大，并且更新频繁，单靠人工进行校正、融合是不可能完成的，针对上述问题，本文采用现有的自然语言处理^[11]的方法，结合机器学习模型^{[12][13]}，模拟人工的校正、融合过程，实现 POI 信息的自动化校正、融合。本文拟定的人工校正、融合 POI 信息过程具体分为以下几个步骤：首先，利用 POI 中的名称字段在多个网络电子地图上进行模糊搜索，将搜索结果中与这个 POI 逐个比较各个字段(名称、类别、地

址、经度、纬度等), 根据经验判断其描述的是否为同一个实体; 然后利用这些描述同一实体的结果经验判断此 POI 是否存在, 是否准确; 其次, 在 web 抽取 POI 时还存在许多与其相关的描述、评论等信息, 查看这些描述、评论的更新时间、数量、相关程度, 进一步判定 POI 的真实性; 最后对那些判断为真实存在的 POI, 利用和它描述同一实体的网络电子地图信息进行融合, 对与影响其准确度的字段进行更正, 对于缺失字段进行补充丰富。

为了提高数据融合结果的质量, 我们要在数据融合前进行一次 POI 是否可进行融合的判断, 对筛选出的可融合 POI 进行融合处理, 验证后进行更新或添加操作; 对于那些不可融合 POI 的相关数据, 进行“死点”判断, 以去除数据库中的部分冗余 POI。本文工作的主要目的就是构建有效的分类模型, 对 POI 数据是否可进行融合进行判断。

1.3 本文主要研究问题和内容

为判断 POI 数据是否能进行融合操作, 本文构建了 POI 可融合分类判定模型, 构建过程中涉及了如下工作:

(1) POI 可融合分类的特征选择

如何更合理、有效的表示出 POI 各特征字段, 决定了之后数据分类结果的好坏。本文通过分析 POI 中各特征字段的形式、特点, 提出了 POI 特征相似度^[14]用以表示待分类 POI 与原有 POI 集的关系, 以些进行之后的判断依据。相似度的形式化表示主要由名称、地理信息相似度两部分组成, 其中的地理信息包括 POI 中的地址和经纬度。

名称部分是指两个不同 POI 名称字段间的相似度, 通过几种经典字符串匹配方法^[15]计算得出, 过程中考虑到因为词语的存在使得不同汉字具有不同的关联性, 本文假设中文字符串匹配的最小单位是词, 打破了传统中最小单位是单个汉字的假设^[16]。

我国尚且没有成熟的地理编码^[17], 既不完整也不精确, 利用经纬度并不能确定两个地址匹配、相似与否。对于地理位置信息的相似程度, 国内主要的根据地址的相似计算得出, 过程中对地址中各特征字段进行匹配^[18], 综合各字段的情况得出地址相似度。本文在考虑地址相似度的同时, 还结合了根据地理空间信息^[19]得出的不同 POI 之间的距离, 弥补了同一 POI 具有多种中文地址描述所导致的

问题。

(2) POI 中经纬度字段的统一

文中用到的 POI 中的经纬度是来源于不同网络电子地图上的坐标, 因为不同电子地图采用的坐标系不同^{[20][21]}, 进行的加密处理也不同, 所以导致同一实体的坐标不一致, 对之后的 POI 融合工作造成影响。为解决这个经纬度标准不统一的问题^[22], 本文提到两种解决方法, 即基于纠偏表的方法和基于 API 的方法。

(3) 分类模型的选择

文中构建了一个基于规则的分类模型, 构建过程中设置 POI 各字段内部系数及阈值, 经过回归计算^{[23][24]}, 选取其区分 POI 是否可融合效果最好的一组系数和阈值构建出了判定模型。这个计算过程复杂、耗时, 并且不够灵活, 不具备自动学习的能力。因此本文又利用机器学习分类器自身主动学习的能力, 构造了几种不同的分类模型, 比较之后选出了较优分类器, 而实现分类性能的有效提升。

1.4 本文的章节安排

本论文的后继章节组织如下:

第二章 机器学习的分类模型

本章介绍了三种机器学习方法(k-近邻、C4.5、AdaBoost)、分类效果评估标准, 以及用于机器学习分类的工具包 WEKA, 为后面可融合 POI 判定技术的实现奠定理论基础。

第三章 基于各字段相似度的 POI 匹配

本章运用自然语言处理的相关知识和技术, 形式化表示出 POI 主要特征字段(名称、地址、经纬度)的相似度, 之后用一定数量的标注了类别的 POI 数据逻辑回归, 找出匹配、分类效果较好的特征及其对应的阈值。最后, 针对不同地图之间的偏差造成 POI 中经纬度标准不统一这个问题, 提出了两种解决方法, 即基于纠偏表的方法和基于 API 的方法。

第四章 可融合 POI 的分类

本章首先对新采集数据真伪性问题进行了介绍, 而后利用基于规则分类方法、基于机器学习的分类方法对新采集数据进行分类, 并用相应的实验对问题进行了仿真, 分析了结果。

第五章 多源 POI 数据融合系统的介绍

本章简单介绍了多源 POI 数据融合系统的主要模块、系统流程，以及分类模型在融合系统中的作用。同时还介绍了系统中网络地图的选择、网络访问限制两个问题。

第六章 总结与展望

首先总结了本文对 POI 可融合分类工作进行研究，然后讨论了存在的不足之处以及未来工作。

2 机器学习的分类模型

机器学习^[25]的核心工作是从特殊的初始训练样例中归纳出一般函数，最终建立能够根据经验自我提高处理性能的计算机程序。学习过程实际是对假设空间进行搜索，使得到的假设最符合已有的训练样例和其他先验的约束或知识。本文的研究重点是将其应用在可融合 POI 分类中，下面简单介绍了三种机器学习方法(k-近邻、C4.5、AdaBoost)及其评估标准，为后面可融合 POI 判定技术的实现奠定理论基础。

2.1 k-近邻算法

k-近邻(k-Nearest Neighbor, KNN)^{[12][26][27]}是一种基于实例的机器学习方法，可用于逼近实值或离散目标函数。它在学习过程只是简单地存储初始的训练数据，一旦遇到新的分类实例，算法就取出与新实例相似的初始训练数据，用此作为参考对新实例进行分类。当目标函数很复杂但却可用不太复杂的局部逼近描述时，使用 KNN 方法进行学习会有显著的优势。KNN 已经被应用到很多任务中，例如，在咨询台上存储和利用过去的经验，根据以前的法律案件进行推理，等等^[28]。KNN 方法存在两个不足，一是分类新实例的开销可能很大，二是从存储器中检索取出相似的训练数据时，会考虑实例的所有属性。

2.1.1 KNN 算法

KNN 算法假定所有的实例对应于 n 维空间 R^n 中的点，并且其中某个实例的最近邻是根据标准欧氏距离定义的。也就是说，把空间中某个实例 x 表示为如下特征向量： $\langle a_1(x), a_2(x), \dots, a_n(x) \rangle$ ，其中， $a_r(x)$ 表示实例 x 的第 r 个属性值。两个实例间的距离就可定义为：

$$d(x_i, x_j) \equiv \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2} \quad (\text{式 2-1})$$

在 KNN 学习中，目标函数值可以用离散数值表示，也可以用实数表示。在本节中我们先考虑离散值目标函数 $f: R^n \rightarrow V$ ，其中 V 是一个有限离散集合 $\{v_1, \dots, v_s\}$ 。表 2-1 是逼近离散值的目标函数的 KNN 算法。其返回值 $\hat{f}(x_q)$ 是对 $f(x_q)$ 的估计，也就是距离：最近的 k 个训练样例中出现次数最多的 f 值。如果

$k=1$ ，“1-近邻算法”就把 $f(x_i)$ 赋给 $\hat{f}(x_q)$ ，其中 x_i 是最靠近 x_q 的训练实例。当 $k>1$ 时，返回前 k 个最接近的训练实例中出现次数最多的值。

表 2-1 逼近离散值函数 $f:R^n \rightarrow V$ 的 k -近邻算法

训练算法：将每个训练样例 $\langle x, f(x) \rangle$ 加入列表 training 中
分类算法：给定一个待分类的实例 x_q
在 training 中选出最靠近 x_q 的前 k 个实例，并用 $x_1 \cdots x_k$ 表示
返回 $\hat{f}(x_q) \leftarrow \arg \max_{v \in V} \sum_{i=1}^k \delta(v, f(x_i))$
其中，当 $a=b$ 时， $\delta(a,b)=1$ ，否则 $\delta(a,b)=0$

图 2-1 展示了一个简单情况下的 KNN，其中的实例是二维空间中的点，目标函数值为布尔型。正反例分别用“+”“-”表示， x_q 为待分类的实例。1-近邻算法把 x_q 分类为正例，而 5-近邻算法将其分类为反例。图 2-1(右)画出了图 2-1(左)中的 1-近邻算法在整个实例空间上导致的决策面，这是围绕每个训练样例的凸多边形合并形成的。对于每个训练样例，其对应的多边形确定了一个查询点集合，其中的点被视为接近这个训练样例，而在这个多边形外的查询点更接近其他的训练样例。

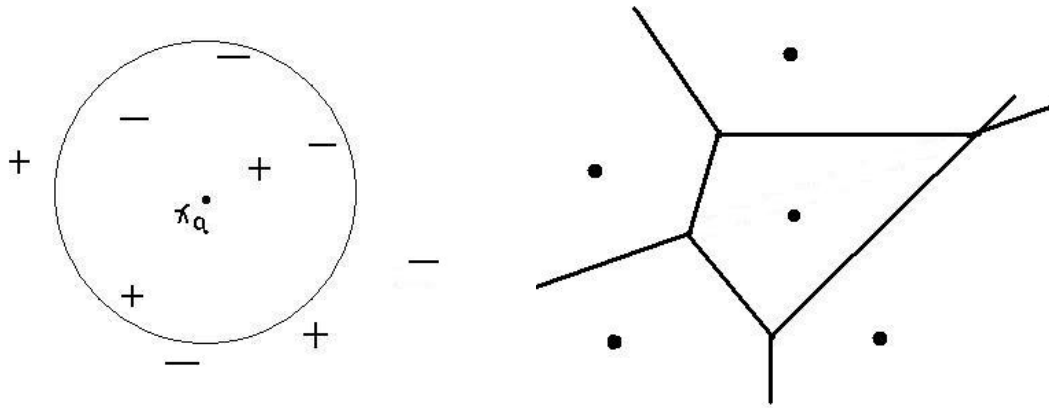


图 2-1 k -近邻算法

对表 2-1 中的算法稍作改动，让算法计算 k 个最接近样例的平均值而不是最普遍的值，也就是将其中的公式替换为：

$$\hat{f}(x_q) \leftarrow \sum_{i=1}^k f(x_i) / k \quad (\text{式 2-2})$$

该算法就可被用于逼近连续值的目标函数 $f:R^n \rightarrow R$ 。

2.1.2 距离加权最近邻算法

对 KNN 算法的一个明显改进是对 k 个最近邻加权值，根据它们相对待分类点 x_q 的距离，将较大的权值赋值给较近的近邻。对于表 2-1 中的逼近离散值函数来说，常用每个近邻与 x_q 的距离平方的倒数作为权值，即把表 2-1 中的公式替换

$$\text{为: } \hat{f}(x_q) \leftarrow \arg \max_{v \in V} \sum_{i=1}^k w_i \delta(v, f(x_i)) \quad (\text{式 2-3})$$

其中

$$w_i \equiv \frac{1}{d(x_q, x_i)^2} \quad (\text{式 2-4})$$

当查询点 x_q 恰好匹配某个训练样例 x_i 时，分母 $d(x_q, x_i)^2 = 0$ ，此时我们令 $\hat{f}(x_q)$ 等于 $f(x_i)$ 。如果有多个这样的样例，则使用它们中占多数的分类。

这里也可以有类似的方法对实值目标函数进行距离加权，把表 2-1 中的公式替换为：

$$\hat{f}(x_q) \leftarrow \frac{\sum_{i=1}^k w_i f(x_i)}{\sum_{i=1}^k w_i} \quad \text{式(2-5)}$$

其中 w_i 与式 2-4 相同。

2.1.1 节中的 KNN 算法只考虑了 k 个近邻用来分类查询点。如果使用距离加权，那么允许所有的训练样例影响 x_q 的分类事实上没有区别，因为较远的实例对 $\hat{f}(x_q)$ 的影响比较小，大幅度减小了噪声对训练数据的影响，具有健壮性，并且在训练数据足够多的时候，该算法也非常有效。考虑所有样例的惟一不足是会导致分类运行得更慢。

2.1.3 k-近邻的两个实践问题

应用 KNN 算法过程中普遍存在着两个问题[12]，其中一个问题是因为实例之间的距离是根据所有属性计算得出的，当紧邻之间距离被大量不相关属性决定时，会导致维度灾难。例如，一组样例中每个实例都有 30 个属性，其中只有两个与实例的分类有关，此时使用 KNN 算法可能会导致两个相关属性值一致的实例相距很远，甚至远远超过了其它不相关实例的距离，最终误导分类。针对这个

问题，一个解决方法是在计算两个实例之间距离过程中对每个属性加权，另一种更强有力的方法是从实例空间中完全消除最不相关的属性。

应用 KNN 算法的第二个实践问题是如何建立高效的索引。因为 KNN 推迟所有的处理，直到接收到一个新的查询，因此处理每个新查询可能需要大量计算。一种索引方法是 kd-tree(Bentley 1975; Freidman et al. 1977)，先把实例存储在树的叶结点内，要求邻近的实例必须存储在同一个或父结点的结点内，之后通过测试新查询选定的属性，树的内部结点就可以把查询结果按顺序存放到相关的叶结点上。目前已经开发了很多类似的索引方法，以便在增加一定存储开销情况下更高效地确定最近邻。

2.2 决策树 C4.5

一些关于归纳推理^[29]的算法已经被广泛应用在现实工作中，决策树^{[12][30]}是其中一个。它是一种有效逼近离散值函数的算法，并且可以很好地处理噪声数据，表现出良好的健壮性，在整个算法学习过程之后会返回一个析取表达式。上节提到的 k-近邻算法，因为延迟了从训练数据中泛化决策的过程，所以被称为是消极的。与之相反，决策树等算法在遇见新的查询实例之前就已经做好了泛化工作，主要是在训练时提交了这些用于定义目标函数逼近的网络结构和加权值，所以决策树等方法是积极的。本节介绍一种使用较为广泛的决策树算法 C4.5^[31]，其中包括了其基本算法、对 ID3^[32]的改进以及应用中的常见的两个问题。

2.2.1 决策树简介

机器学习方法中的 CART 算法、ID3、C4.5、CHAID 等都属于决策树的范围，并且是比较常用的几种算法，它们主要是通过构造决策树进行分类选择的。决策树起源于 1966 年由 Hunt.E.B 等人提出的概念学习系统 CLS (Concept Learning System)。Quinlan.J.R 在 1986 年提出了 ID3(Iterative Dichotomizer 3)算法。因为 ID3 算法在实际应用过程中存在一些问题，1993 年 Quilan 对 ID3 进行部分改进后提出了 C4.5 算法，最后又提出了在商业领域中被广泛使用的 C5.0 算法。但是经实验证明，C4.5 与 C5.0 的决策树产生方法相近，其分类结果存在略微不同，但对其分类性能的影响并不大。

利用决策树分类的过程主要分两骤：（1）根据给定的训练集，构建决策树分

类模型；(2) 用决策树分类模型预测分类数据的类别，也可以用这个模型对数据集进行描述，归纳出分类规则。图 2-2 描述了决策树分类过程，其样例分为训练集和测试集，分别用训练决策树模型、评估这个模型的预测效果。

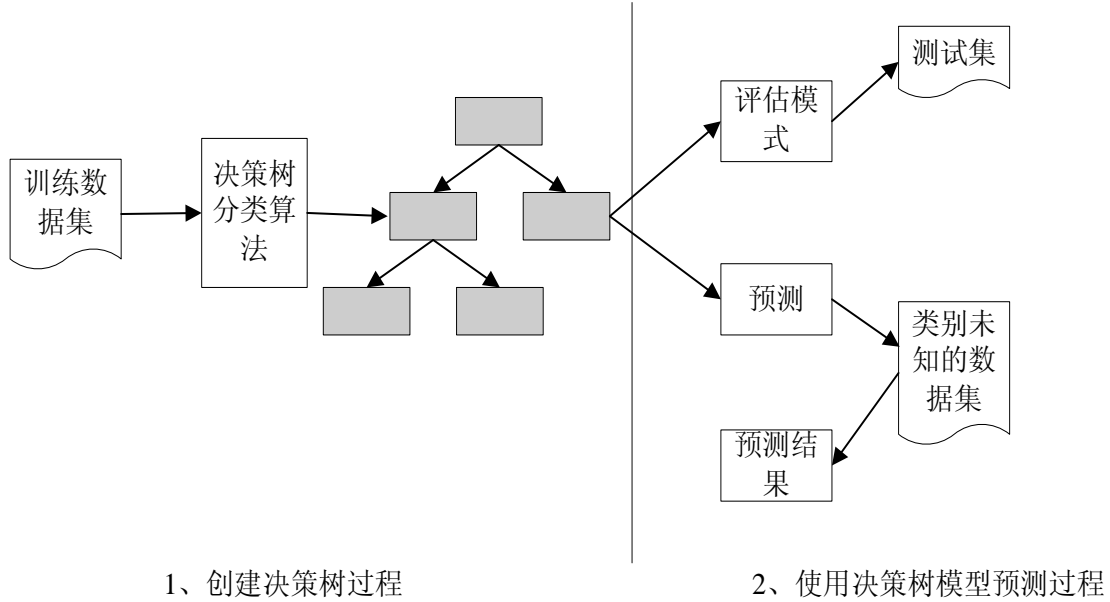


图 2-2 决策树分类模型的工作过程

2.2.2 ID3 算法

ID3 算法通过贪婪搜索的方式生成分类决策树，具体地说，该算法首先选择表现最好的一个属性赋予根节点，用这个属性的各个取值产生对应分支，在每个分支上继续之前的工作，选择出其它表现最好属性作为当前分支节点。ID3 使用信息论领域中信息增益的概念，有效地选择出了待选属性中表现最好的那个。

设 S 是 n 个样例的集合，可被划分为 c 个类别 $c_i (i=1, 2, \dots, c)$ ，每个类别 c_i 含有的样例数目为 n_i ，则 S 划分成 c 个类别所具有的信息熵是：

$$E(S) = \sum_{i=1}^c p_i \log_2(p_i) \quad (\text{式 2-6})$$

其中 p_i 为样例集 S 中的样例属于类别 c_i 的概率，计算如下：

$$p_i = \frac{n_i}{n} \quad (\text{式 2-7})$$

假设将属性 A 的所有取值(不重复)记作集合 X_A ， S 中属性 A 的值为 v 的样例子集记作 S_v ，并将 S_v 作为属性 A 之后分支节点上的样例集，将对样例集 S_v 划分

到 c 个类后所得的信息熵记作 $E(S_v)$ 。选择 A 后, 我们定义期望熵为每个子集 S_v 的信息熵加权后的总和, 基中的加权值等于 S_v 中的样例占原始样例 S 的比例

$|S_v|/|S|$, 期望熵可表示为:

$$E(S, A) = \sum_{v \in X_A} \frac{|S_v|}{|S|} E(S_v) \quad (\text{式 2-8})$$

属性 A 相对样例集 S 的信息增益定义为:

$$Gain(S, A) = E(s) - E(S, A) \quad (\text{式 2-9})$$

$Gain(S, A)$ 的物理意义为: 因为知道属性 A 的值使得熵的期望压缩, $Gain(S, A)$ 越大表示选择属性 A 所提供的信息量越大。ID3 算法正是利用这一点, 在每个节点选择出信息增益 $Gain(S, A)$ 最大的属性, 并将它作为该节点的测试属性。

ID3 算法分类速度快、方法简单、学习容易且理论清晰, 但是它只能用于处理离散值属性。还有一个不足是过度拟合, ID3 使用信息增益选择最好属性, 一般选出的属性是那个取值最多的属性, 但在实际问题中值最多的属性不一定包含最大价值的信息。

2.2.3 C4.5 算法

C4.5 算法既可以处理离散值属性, 也可以处理连续值描述的属性, 同时, 它采用了信息增益比选择最好属性, 是对 ID3 算法不足之处的改进。信息增益比的定义如下:

$$GainRatio(S, A) = \frac{Gain(S, A)}{SplitInfo(S, A)} \quad (\text{式 2-10})$$

其中 $Gain(S, A)$ 与 ID3 中的信息增益概念相同, $SplitInfo(S, A)$ 表示按照属性 A 分裂样例集 S 所表现出的广度及其均匀性, 这个变量可以通过如下公式计算:

$$SplitInfo(S, A) = - \sum_{i=1}^t \frac{|S_i|}{S} \log_2 \frac{|S_i|}{|S|} \quad (\text{式 2-11})$$

其中 S_1 到 S_t 是 t 个不同值的属性 A 分割 S 而形成的 t 个样例子集。

对于某个连续值描述的属性 A_c , 假设某个节点上的数据集样例数量为 $count$ 。将该节点上的样例对连续值属性的具体取值进行升序排序, 得到取值序列 $\{A_{1c}, A_{2c}, \dots, A_{countc}\}$; 分割这个序列可以得到 $count-1$ 个分割点, 并且将第 i 个分

割点值设为 A_{ic} 和 $A_{(i+1)c}$ 的平均数, 可以将该属性对应节点上的数据集划分成了两个子集; 用每一个分割点对数据集进行分割, 并计算出它们各自的信息增益比, 从中可以选取比值最大的分割点来划分数据集。

为了避免过度拟合数据, C4.5 采用了后剪枝方法。该方法通过估计训练样例在剪枝前后的误差来判断是否需要剪枝, 这个误差的计算公式如下所示:

$$P_r\left[\frac{f-q}{\sqrt{q(1-q)/N}} > z\right] = c \quad (\text{式 2-12})$$

其中 N 是样例的数量, $f = E/N$ 是观测后所得到的误差率, E 是 N 个样例分类结果中错误的数目, q 是真实误差率, c 是置信度(默认值取 0.25), z 是 c 对应的标准差。我们可以通过上面公式 2-12 计算得出 q 的置信度上限值, 用这个上限为节点的误差率 e 做出估计, 计算如公式 2-13。根据这个 e 值的大小, 就可以判定是否进行剪枝操作。

$$e = \frac{f + \frac{z^2}{2N} + Z\sqrt{\frac{f}{N} + \frac{f^2}{N} + \frac{z^2}{4N^2}}}{1 + \frac{z^2}{N}} \quad (\text{式 2-13})$$

C4.5 对于 ID3 算法的改进还在于其可以处理缺省值。假如样例集 S 中的一个训练样例 $\langle x, c(x) \rangle$, 其属性 A 的对应值 $A(x)$ 未知, C4.5 就可以为 $A(x)$ 每个可能的取值赋予一个概率。例如给定一个属性 A , 如果样例集中有 8 个 $A=0$ 、2 个 $A=1$ 的实例, 则 $A(x)=0$ 的概率为 0.8, $A(x)=1$ 的概率为 0.2。因此在对实例 x 进行分类时, 它的 80% 将被分到 $A=0$ 的分支上, 其它 20% 将被分到 $A=1$ 的分支上。

2.3 强分类器 AdaBoost

Adaboost(Adaptive Boosting)^[33] 是一种迭代分类算法, 从在线的动态分配模型演化而来, 1995 年由 Yoav Freund 和 Robert E. Schapire 提出。Adaboost 的基本算法思想为: 对多个分类效果一般的弱分类器进行叠加提升, 生成一个分类效果表现较好的强分类器。如果单个弱分类器的分类效果比任意猜测的效果都要好, 那么当弱分类器的个数无穷大时, 经过提升得到的强分类器分类效果将接近完美。基本思想就像中国的一句古话“三个臭皮匠, 顶个诸葛亮”。对 Adaboost 算法的研究及应用大多集中在分类问题上, 如二分类问题、多分类问题、大类单标

签问题，近期也被应用在回归问题上。

2.3.1 Adaboost 的基本算法

Adaboost 算法用全部的训练样例进行迭代学习，给每个训练样例赋予一个权值，用这个权值来表示样例被某个弱分类器选入训练集的概率。分类过程中，如果当前分类器能够正确判断某个样例的类别，则降低这个样例被选入下一个训练集的概率；反之，如果分类错误，则提高这个概率。这样就可以使分类过程重点考虑到那些比较困难的样例，减小过拟程度。

在第一次迭代学习之前，样例被设置了相同的权值，之后进行第 k 次迭代学习时，首先根据样例的权值选择出训练集，利用训练集训练弱分类器 C_k ，再用整个样例集对 C_k 进行测试，测试结果正确的样例降低其相应的权值，错误的则提高相应权值，完成此次迭代学习的同时，也更新了样例集的权值。根据第 k 次迭代学习中更新的权值，可以选出第 $k+1$ 次迭代过程中使用的训练集，继而训练下一个弱分类器 C_{k+1} 。学习过程不断地迭代，直至满足要求为止。

表 2-2 Adaboost 算法伪代码

<p>(1) 初始化 $S = \{(x_1, y_1), \dots, (x_n, y_n)\}$, k_{\max}, $W_i(i) = \frac{1}{n}$, 其中 $i = 1, 2, \dots, n$</p> <p>(2) $k \leftarrow 0$</p> <p>(3) Do $k \leftarrow k + 1$</p> <p>(4) 根据 $W_k(i)$ 在 S 中选出样例子集 S_k, 作为训练集训练弱分类器 C_k</p> <p>(5) 用 S 中所有样例测试 C_k, 计算其训练误差 E_k</p> $E_k = \sum_{i=1}^n W_k(i) \varphi_k(i), \text{ 其中 } \varphi_k(i) = \begin{cases} 1, & \text{分类正确} \\ 0, & \text{分类错误} \end{cases}$ <p>(6) $\alpha_k \leftarrow \frac{1}{2} \ln\left(\frac{1-E_k}{E_k}\right)$</p> <p>(7) $W_{k+1}(i) \leftarrow \frac{W_k(i)}{Z_k} \times \begin{cases} e^{-\alpha_k}, & \text{分类正确} \\ e^{\alpha_k}, & \text{分类错误} \end{cases}$</p> <p>其中 Z_k 是一个归一化系数，使得 $W_k(i)$ 能够成为一个概率分布，α_k 为选取合适阈值，使得误差最小</p> <p>(8) Until $k = k_{\max}$</p> <p>(9) 返回分类器 C_k 和 α_k</p>

设 S 是 n 个样例的集合， x_i 表示 S 中的一个样例， y_i 表示样例的类别标记， $W_k(i)$ 表示第 k 次迭代学习时 S 中所有样例的权值分布， k_{\max} 为整个学习过程迭代

的次数，Adaboost 算法伪代码如表 2-2 所示。

2.4 评估方法

机器学习的核心问题是根据学习模型从训练样例中归纳出一般函数，目的是建立能够根据经验自我提高处理性能的计算机程序。要检测构建的学习模型的性能好不好，即模型分类结果和人工标注结果是否一致，本节介绍了 3 个重要指标，即召回率(Recall，简记为 R)、准确率(Precision，简记为 P)和 F_β ，将在之后模型评测实验中被用到[34]。

表 2-3 为一个二分类列联表：

表 2-3 二分类列联表

实际标注类别 模型分类结果	1	0
1	A	B
0	C	D

则对于类别“1”， R 、 P 和 F_β 的定义分别为：

$$R = \frac{A}{A + C} \quad (\text{式 2-14})$$

$$P = \frac{A}{A + B} \quad (\text{式 2-15})$$

F_β 是将准确率和召回率结合起来，公式如下：

$$F_\beta = \frac{(\beta^2 + 1) \times P \times R}{\beta^2 \times P + R} \quad (\text{式 2-16})$$

其中， β 是一个用来调节准确率和召回率权重的参数。 β 一般取值为 1，则公式 2-16 转化为：

$$F_1 = \frac{2PR}{P + R} \quad (\text{式 2-17})$$

2.5 Weka 简介

Weka^[35]的全名是怀卡托智能分析环境(Waikato Environment for Knowledge Analysis)，由怀卡托大学的 Weka 小组开发，是基于 java 的、用于数据挖掘和知识分析一个平台。Weka 小组曾在第 11 届 ACM SIGKDD 会议上荣获数据挖掘和知

识探索领域的最高服务奖，至此 Weka 平台得到了专业领域内广泛的认可。甚至现在 Weka 仍被列在最完备数据挖掘工具之中。Weka 平台中集合着很多用于数据挖掘工作的经典机器学习算法，如 J48^[35](决策树 C4.5 的 java 版本实现)、IBk(KNN)、AdaBoostM1 算法，用户可以这个平台方便地进行数据的预处理、关联规则、分类、回归、聚类等相关工作。同时，Weka 还封装了一些常用的评估参数，实现了评估算法性能的功能。为方便用户使用，Weka 又提供了可视化的交互式界面。因为 Weka 是开源的，并且具有良好的扩展性和兼容性，用户可以把自有的算法、或者对经典算法改进后的算法集成在 Weka 平台上，进而完成相关数据挖掘工作。

虽然 Weka 也支持 csv、xls、mat 等格式的文件，但是 arff 格式是支持的最好的。因此在使用平台中的算法进行训练模型之前，要将训练数据按照 Weka 的标准写到 arff 文件中。ARFF(Attribute-Relation File Format)的数据文件结构简单，常用于存储测试过程使用的小量数据，文件是由数据定义部分(Header)、描述部分(Data)两部分组成的，图 2-3 是一个简单的例子。数据定义部分中，用@relation 关键词指定该关系的名称，格式为@relation <relation-name>，在 arff 的第一个有效行来定义，如果关系名称字符串中包含空格，它必须加上引号；用@attribute 关键词指定属性名称及数据类型，具体使用格式为@attribute <attribute-name> <datatype>，每行均对应一个属性，最后声明的那个属性将被看作是类别域，会在之后的分类或回归操作中被默认为是目标变量。数据描述部分从“@data”标记开始，每行描述一条样例，样例中的属性使用逗号分割，属性的顺序与数据定义部分定义的属性的出现顺序必须一致，当样例的某属性值缺失时，对应的数据部分应用“?”代替。

```

@relation weather
@attribute outlook {sunny, overcast, rainy}
@attribute temperature real
@attribute humidity real
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}
@data
sunny,85,85,FALSE,no
sunny,80,90,TRUE,no
overcast,83,86,FALSE,yes
rainy,70,96,FALSE,yes
rainy,68,80,FALSE,yes
rainy,65,70,TRUE,no
overcast,64,65,TRUE,yes
    
```

图 2-3 ARFF 格式的数据文件

2.6 小结

本节简单介绍了三种(k-近邻、C4.5、AdaBoost)学习方法、机器学习方法评估标准，以及机器学习领域的一个完备工具 Weka，为后面可融合 POI 判定技术的实现奠定理论基础。

3 基于各字段相似度的 POI 匹配

在对 POI 数据进行维护、更新过程中,用户通常先比较 POI 的各个特征字段,把那些表述相近的 POI 看作是描述同一实体匹配 POI,而那些表述相差甚远的 POI,则看作是描述不同实体、不匹配的 POI。合理地表述出不同 POI 之间的相似关系,利用形式化的、较为直观的参数对 POI 进行相似匹配,对于之后数据维护、更新工作至关重要。

第一章中提到了 POI 的构成及特点,本章将运用自然语言处理的相关知识和技术,形式化表示出 POI 主要特征字段的相似度,而后用一定数量的标注了类别的 POI 数据逻辑回归,找出匹配、分类效果较好的特征及其对应的阈值。同时针对不同地图之间的偏差造成 POI 中经纬度标准不统一这个问题,提出了两种解决方法,即基于纠偏表的方法和基于 API 的方法。

3.1 名称相似度

POI 中名称字段大多比较精短、无明显规则,同时也缺乏语义上的特征,是一类普通的中文字符串。目前,这种中文字符串相似度的计算在中文信息检索^[36]、中文文本校对等领域中已有广泛的应用。衡量两个字符串的相似度,常用的方法有三种,即莱文史特距离算法、Jaccard 相似方法和 Jaro 距离算法[14]。

根据已有资料的分析,现有的这些计算字符串相似度的算法大多基于一个假设:中文字符串匹配的最小单位是单个汉字,这样并没有考虑到汉字中词语对相似度的影响,所以我们将匹配的最小单位假设为词。

3.1.1 莱文史特距离算法

莱文史特距离算法(Levenstein edit distance algorithm)是一种字符串编辑距离算法,指一个字符串通过多少次操作(增、删、改)得到另外一个字符串。例如,字符串 $S1$ 为“aaabc”, $S2$ 为“aabb”, $S1$ 通过‘a’变为‘b’,删除‘c’两步可以得到 $S2$,所以编辑距离等于 2。在这里,我们定义字符串相似度为:

$$edit(S1, S2) = 1 - \frac{distance}{maxLen} \quad (式 3-1)$$

其中 $distance$ 是 $S1$ 、 $S2$ 的编辑距离, $maxLen$ 是 $S1$ 、 $S2$ 字符串长度中较大的那个值。 $edit$ 值越大说明相似度越大,0 表示没有任何相似度,1 则代表完全匹配。

3.1.2 Jaccard 相似方法

这个相似度等于两个字符串中相同词(无重复)的个数与所有词(无重复)个数的比值。也就是说，两个字符串 $S1$ 、 $S2$ ，二者的 Jaccard 相似度可定义为：

$$jacc = \frac{|S1 \cap S2|}{|S1 \cup S2|} \quad (式 3-2)$$

和 *edit* 一样，*jacc* 越大说明相似度越大。

3.1.3 Jaro 距离算法

与上边两种算法相比，Jaro distance 算法的优点在于其考虑到字符不同位置的问题，如“粥全粥到台东三路店”和“粥全粥到三店”，其中的“三”根据位置的不同可判断为不匹配。首先定义匹配窗口：

$$MW = \left(\frac{\max(|S1|, |S2|)}{2} \right) - 1 \quad (式 3-3)$$

其中 $S1$ 、 $S2$ 是待匹配字符串。 $S1$ 、 $S2$ 匹配过程中，若两者中同有字符 x ，并且这两个 x 的距离不大于 MW ，此时可以认为这两个 x 是匹配字符。

Jaro 相似度定义如下：

$$Jaro = \frac{1}{3} \left(\frac{m}{|S1|} + \frac{m}{|S2|} + \frac{m-t}{m} \right) \quad (式 3-4)$$

其中 $S1$ 、 $S2$ 是待匹配的两个字符串， m 是匹配的字符数， t 是换位的数目，其值等于不同顺序的匹配字符数目的一半。

比如：两个字符串“ABCDE”和“EBCDA”做匹配操作，字符串中仅有 B、C、D 三个字符是匹配的，即 $m=3$ 。虽然 A、E 都出现在两个字符串中，但是通过公式得出匹配窗口 MW 为 $(5/2)-1=1.5$ 。而两个字符串中 A、E 字符的距离均大于 1.5，所以不算做匹配。在另一组字符串 $AxByCDz$ 与 $AzBDC$ 。匹配的字符为 A~B~C~I，但在两个字符串中 C~D 两个字符顺序不同，因此 $t=1$ ， $m=4$ 。

3.1.4 字符串分词处理

中文分词技术^{[37][38]}是自然语言处理范畴中的技术，人利用自己日常知识可

以明白、理解句子里哪部分是词语，哪部分不是词语，但如果使计算机也同样能够理解，就需要先根据分词算法对句子进行处理。自然语言处理中分词算法主要分为基于字符串匹配的分词、基于理解的分词以及基于统计的分词三类。因为中文是一种十分复杂的语言，对其分词过程中歧义识别和新词识别这两大难题尚且还未被彻底突破。目前对中文分词进行研究的大多是科研院校，但是进行些项研究的商业公司、机构却几乎没有，除海量科技以外。以下简单介绍几种网络上分词工具：

（1）计算所汉语词法分析系统 ICTCLAS^[39]

ICTCLAS(Institute of Computing Technology, Chinese Lexical Analysis System)主要被用于中文分词、词性标注和未登录词语的识别。它对中文进行分词、标注的速度均可达到 31.5KB/s，分词正确率可达到 97.58%，对未登录词语的识别也可达到 90%以上。该软件是中国科学院计算技术研究所利用多层隐马夫模型实现的，为期一年，并且该软件和中国科学院计算技术研究所的其他 14 项成果均免费对外使用，国内不少免费供用户使用的中文分词工具都多多少少地参考过 ICTCLAS 的功能实现。我们可以在 ICTCLAS 的官方网站(<http://www.ictclas.org>)上下载、使用自己需要的版本。

因为原版的 ICTCLAS 是用 C 语言编写完成的，在现在比较流行的开发工具中使用相当不方便，有些程序员就把 C 语言版的分词器改成了 Java 版本和 C#版本的。Java 版的 ICTCLAS 下载页面为 <http://www.xml.org.cn/printpage.asp?BoardID=2&id=11502>。本文使用了这个分词工具，并使用搜狗的常用词库作为分词词典。

（2）海量智能分词(研究版)

为了让使用中文信息处理技术的研究人员分享自己的成果，共同提高专业水平，海量智能计算技术研究中心发布了海量智能分词器的研究版本。下载页面为 <http://p2s.newhua.com/down/hlssplit.rar>。海量的分词做的不错，不过研究版的速度比之前更慢了，而且只支持 window 开发。

（3）CSW 中文智能分词组件

改组件是用 DLL 形式实现的，可对一段汉语文本进行自动的词语切分，并且实现了指定分隔符号、语义标注、词频标注功能，被广范应用于数据挖掘等相关领域。如果出现如下错误“您当前使用的 CSW 中文分词组件 5.0(标准 C++版)已

超过有效期，请访问我们网站 www.vgoogle.net 获取最新版本或取得使用许可授权！”，把系统时间调到 2008 年 4 月 1 号之前。分词效果还可以，java 下的 20kb/s。

(4) je-anlysis 的分词(java 实现的)

分词效率可达每秒 30 万字，第一次分词需要 1—2 秒加载词典)，在 Lucene 2.0 环境下运行。je-anlysis 可免费安装使用，无限制商业应用，但暂不开源，也不提供任何保证。

3.2 地理信息的相似度

地理信息主要包括两部分，即空间地理信息和非空间地理信息。POI 中的经纬度就是一种典型的空间地理信息，而 POI 中的中文地址则属于地理信息系统中的非空间信息。我国地理信息的相似度主要是根据中文地址的匹配程度得出，但是对于那些具有多种描述情况的地理实体，比如有别名的实体、处于两条路交叉口的实体，这种地址匹配方法就不能得出其真实的相似程度。为解决这个问题，本文借助空间地理信息对这个相似度进行了补充。

3.2.1 中文地址的相似度

地址是各类服务系统中运用自然语言描述空间位置的最常用手段。中文地址是一种具有一定格式的中文字符串，但又不是标准统一格式，对于其相似度的计算，单靠本文提到的中文字符串匹配方法并不能达到很好的效果。目前我国主流的地址匹配方法就是对地址分词，利用各个地址要素进行匹配^[40]。本文基于小词典和特征词对中文地址进行分词，成功分开了中文地址中的各个要素，然后根据设置好的规则，综合所有要素给出其相似程度。

分词过程中用到的小词典是根据行政区划表^[41]构造出来的，主要目的是规范地址中省、地、县、乡级行政区名称，如“崂山区松岭路 238 号”，分词结果为“山东(省)青岛(市)崂山(区)松岭(路)238(号)”，不仅划分出字符串中各个部分，其省略部分也会补充完整。地址字符串中除省、地、县、乡级行政区以外的其它部分，因为信息量太大，严重影响分词速度，况且现在没有合适完整资料来源，所以只对其进行特征字分词。得到最终分词结果格式为“X(省)X(地)X(县)X(乡)X(路)X(号)X(建筑)X(号码)X(其它)”，括号内是其对应地址要素的特征词。实验数据分词结果如图 3-1 所示。

		市南区	燕儿岛路	1号	心海广场		内	(近闽江路)
	青岛市	市北区	延吉路	12号				
		市南区	香港中路	132号				
山东省	青岛市	黄岛区						
山东省	青岛市	城阳区						
			人民路				102-8	
			文化路	387号				
山东省	青岛市	市南区						
			哈尔滨路	55号			丁	
	青岛市		闽江路				183	
山东省	青岛市		香港西路	65号			1602	
			青岛香港中路	18号	福泰广场		a座1306室	
			江西路				106壬	
		市北区	台东八路	25-27号				(近五星电器)
			福清路				1甲	
			桑梓路	1号				
	重庆市	万州区						
			红师大道	10号		附10号	二楼	
	重庆市	铜梁县						
		沙坪坝区					沙坪坝小龙坎正街242号嘉福苑内	
		市南区	南京路	122号	中联广场		G栋106A室	
		市南区	南京路	122号	中联广场		G栋106A室	
	青岛市		伊春路	129号				
山东省	青岛市	市南区						
		近郊城阳	青威路	617号	青特·上豪广场			(城阳家佳源超市, 利群商场北)

图 3-1 基于小词典、关键字的分类结果

对待匹配的两个中文地址，分词处理后对其进行相似度计算，因为分词过程中对乡级及以上行政区字段进行了规范和补充，所以我们认为，该 4 级字段中低级字段若相等，较高级字段也一定匹配。对于其它 5 个字段，先分别计算出相似度，再根据不同权值合算出总的相似度。如果两个中文地址中对应字段不同时存在，就无法进行相似度计算，对于这种情况把相似度计为 -1，表示不考虑该字段。若 $SIM1$ 、 $SIM2$ 、 SIM 分别表示中地址前 4 个字段、后 5 个字段以及整体的相似度， $SIM1$ 、 $SIM2$ 的具体计算过程如图 3-2、图 3-3 所示， SIM 的具体计算流程如下：

Step1 初始化 $SIM1$ 、 $SIM2$ 、 SIM 都为 -1。

Step2 若乡级字段匹配，对 $SIM1$ 赋值为 1，转向 Step3；若不匹配，则匹配县级字段，县级若相等 $SIM1$ 为 0.8，转向 Step3；以同样方法处理地级、省级字段， $SIM1$ 分别为 0.4、0.3；省级字段也不匹配， $SIM1$ 仍为 -1。

Step3 若路级字段对应可比，且两个字段字符串相似度 t 大于 0.8，则将路级、号级字符串的相似度记为 s_1 ， t 小于 0.8 时 s_1 等于 t 的一半；路级字段不可比时， s_1 等于 -1。

Step4 和路级、号级字段一样，计算出建筑级、号码级字段相似度记为 s_2 。

Step5 根据 3.1 中提到的一般字符串相似度算法，计算其它字段的相似度记为 s_3 。

Step6 设置决定 *SIM2* 各字段的权值, s_1 、 s_2 、 s_3 分别对应权值 a_1 、 a_2 、 a_3 ; 若 s_1 、 s_2 、 s_3 中某个变量值为 -1, 表示其对应字段不可比, 我们就使这个字段对应的权值为 0。后 5 个字段的相似度为:

$$SIM2 = \begin{cases} -1 & , a_1 | a_2 | a_3 = 0 \\ \frac{\sum_{i=1}^3 a_i * s_i}{\sum_{i=1}^3 a_i} & , \text{其它} \end{cases} \quad (\text{式 3-5})$$

当 s_1 、 s_2 、 s_3 都等于 -1 时, *SIM2* 为 -1;

Step7 设置 *SIM1*、*SIM2* 字段的权值 b_1 、 b_2 ; 若 *SIM1*、*SIM2* 中有值为 -1, 则使其对应的权值置 0。待匹配两个中文地址的相似度为:

$$SIM = \begin{cases} 0 & , b_1 | b_2 = 0 \\ \frac{b_1 * SIM1 + b_2 * SIM2}{b_1 + b_2} & , \text{其它} \end{cases} \quad (\text{式 3-6})$$

如果 *SIM1*、*SIM2* 值都为 -1, *SIM* 的值定为 0。

我们通过比较不同阈值在上述相似度计算过程中的表现效果, 确定算法中的阈值。因为地址中街道、建筑、其它三个字段对相似程度影响力度差不多, 但是建筑、其它两个字段缺失较为严重, 所以设定 a_1 、 a_2 、 a_3 的值分别为 4、3、3。对于两个地址, 单单知道他们的省、市、区、乡镇字段一致, 并不能说明这两个地址是否相似, 但详细的街道地址等信息相似时, 我们却可以比较确定这两个地址的相似关系, 可见, 上述地址后 5 个字段对相似度的影响明显大于前 4 个字段[42]。所以我们将地址相似度计算过程中的权值 b_1 、 b_2 分别设为 1、3, 以致凸现址后 5 个字段的作

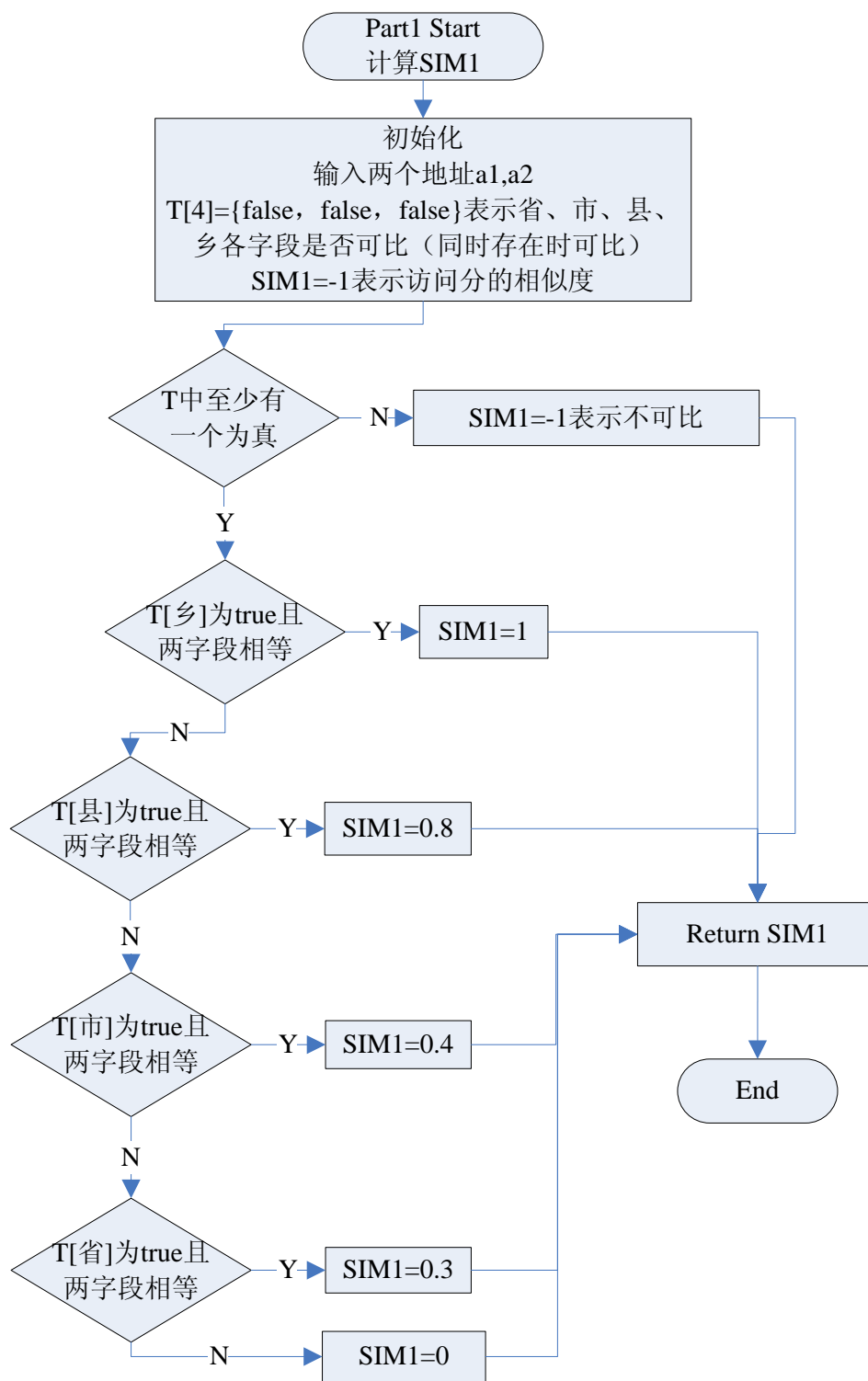


图 3-2 地址中前 4 个字段(省、市、县、乡)相似度 $SIM1$ 的具体计算过程的流程图

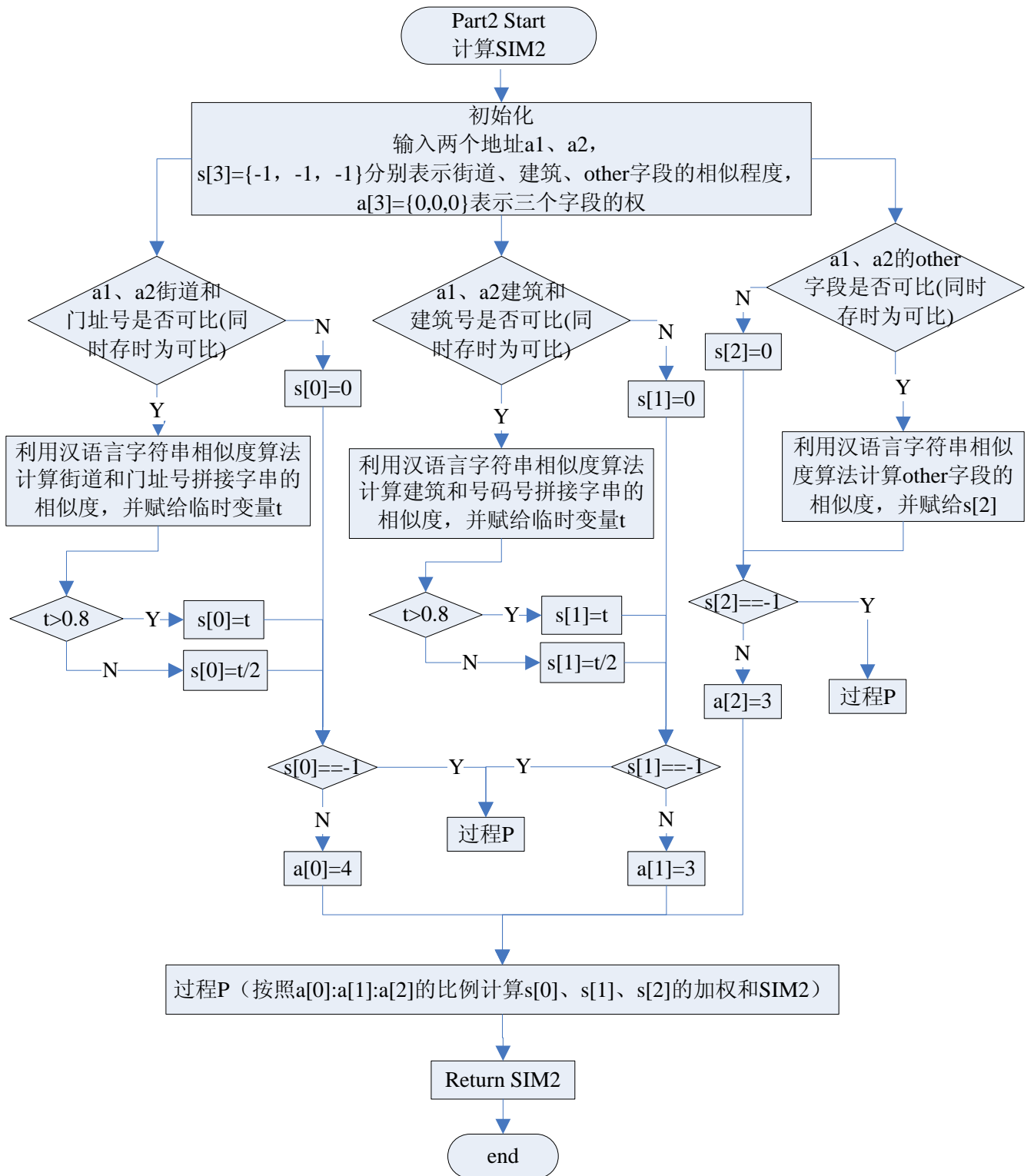


图 3-3 地址中后 5 个字段(街道及号、建筑及号码、other)相似度 $SIM2$ 的具体计算过程的流程图

3.2.2 空间地理信息相似度

经纬度被定义在三度空间的球面上，用来标示地球上的任何一个位置，是一种典型的空间地理信息。POI 中的经纬度作用和地址字段相同，都是用来描述一个位置，只是形式不同。通过经纬度来衡量两个 POI 是否匹配相似，最简单有效的方法就是计算这两点之间的球面距离。该地理坐标相似度定义为：

$$LLsim = \frac{1}{distance(p_1, p_2)} \quad (式 3-7)$$

其中 $distance(p_1, p_2)$ 是匹配的两个 POI 点 p_1 、 p_2 的球面距离。当 $LLsim$ 这个相似度大于阈值时，就认为这两个 POI 相似匹配。为了将相似度控制在 $[0,1]$ 区间内，以及提高计算的精度，本文将上公式修改为：

$$LLsim = 1 + 1/15 \times \ln \frac{1}{distance(p_1, p_2)} \quad (式 3-8)$$

当两点距离为 10 米时， $LLsim$ 的值为 0.85；当距离远至 100,000 米时， $LLsim$ 的值为 0.232。

3.3 国内经纬度的统一

本文用到的 POI 中的经纬度部分来源网络电子地图上的坐标，因为不同电子地图采用的坐标系不同，进行的加密处理也不同，所以导致同一实体的坐标不一致，对之后的 POI 融合工作造成影响。对于坐标不统一的问题，此小节提出两种解决方法。

3.3.1 问题描述

由于采用的椭球基准^[43]不一样，同一实体在不同的地理坐标系上对应的经纬度也不同。但是这些基于不同椭球基准的地球坐标系之间可以自由转换，所有如果知道坐标所处的坐标系，那么因坐标系不同而造成坐标不一致的问题，就可以通过基准转换轻易解决。我国现在常用坐标参照系有北京 54 坐标系、西安 80 坐标系、WGS-84 坐标系[44][45]。例如，目前一般采用布尔莎公式完成 WGS-84 坐标系到北京 54 坐标系的转换，类似地，可以完成三者中任意两个坐标系的转换[20][21]。

要完成坐标之间的统一，首先必须知道这些坐标是在什么坐标系下生成的。

目前,除了知道英文版的谷歌地图采用的是 WGS-84 坐标系和 WEB 墨卡托投影外,其他的网络电子地图都没有公布其采用的坐标系统。

在我国谷歌地图分为两个版本,即中文版谷歌地图(<http://ditu.google.cn/>)与英文版谷歌地图(<http://maps.google.com/>),两者的经纬度定位存在差异。对于北京的人民英雄纪念碑来说,它的地理坐标点是(39.903247,116.391561),在英文版谷歌地图的卫星视图可以标出正确位置,如图 3-4,在地图矢量模式下这一坐标点就没有被标记在正确位置上,出现了 300 米左右的偏差,如图 3-5。正如之前提到的,英文版谷歌地图采用的是 WGS-84 坐标系,两种模式下的偏差并不是由于地理坐标系不同,而是因为两种模式之间存在加偏操作。经调研知,英文版谷歌地图的卫星视图采用的是真实的 GPS 坐标,而地图矢量模式则是采用了经过加偏处理的坐标。这是因为无论是纸质地图还是电子地图,如果需要民用,就必须对其中的一些信息做特殊处理,坐标信息就是其中的一项。目前可知的是,国家测绘部门要求民用公开的电子地图至少使用 GCJ-02 加密算法做一次坐标偏移处理,具体的偏移算法是保密的。这种偏移使得真实坐标与地图上显示出的坐标不一致,且此加偏并非线性加偏,各地的偏移状况都会有所不同。

按照国测局的要求,所有的民用公开电子地图使用国测局制定的 GCJ-02 加密算法进行坐标偏移处理,其中部分电子地图对坐标又实施了二次加偏。例如百度地图,它对真实 GPS 坐标使用 GCJ-02 算法首次加密后,又进行了 BD-09 二次加密措施。各个电子地图的加偏算法不尽相同,对之后的来源于不同电子地图的 POI 融合工作造成影响。为解决不同地图间对应坐标点的偏移问题的问题,本文提到两种解决方法,即基于纠偏表的方法和基于 API 的方法。



图 3-4 英文谷歌地图的卫星视图下坐标点(39.903247,116.391561)在箭头所指处



图 3-5 英文谷歌地图的矢量模式下坐标点(39.903247,116.391561) 在箭头所指处

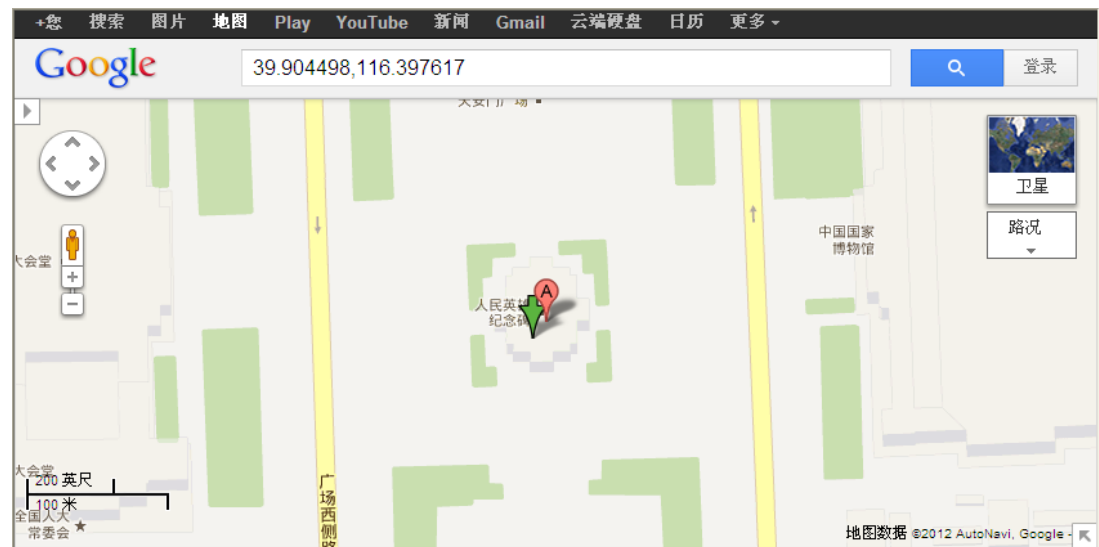


图 3-6 英文谷歌地图的矢量模式下坐标点(39.904498,116.397617) 在箭头所指处

3.3.2 基于纠偏表的实现

电子地图的加密算法整体上看是非线性的，但从局部看，偏差变化缓慢，可以看作是线性变化的。在谷歌地图中指定一个区域，选取合适间隔进行网格搜索，可以得到一批加偏后的经纬度以及其对应的真实 GPS 坐标，两种坐标对应做差得到偏差。将这些真实的 GPS 坐标及其与之对应的偏差数据记录在纠偏表中，之后利用纠偏表可以快速地实现谷歌地图坐标纠偏工作。图 3-7 所示为部分纠偏表。

	A	B	E	F
1	LNG	LAT	OFFSET LNG	OFFSET LAT
73081	116.3	39.6	0.006024	0.001215
73082	116.3	39.7	0.006035	0.001234
73083	116.3	39.8	0.006046	0.001245
73084	116.3	39.9	0.006056	0.001251
73085	116.3	40	0.006073	0.001253
73086	116.3	40.1	0.006083	0.00126
73087	116.3	40.2	0.006099	0.00127
73088	116.3	40.3	0.00611	0.001289
73089	116.3	40.4	0.006121	0.00132
73090	116.3	40.5	0.006137	0.001358
73091	116.3	40.6	0.006148	0.001413
73092	116.3	40.7	0.006158	0.001476

图 3-7 所示为部分纠偏表

纠偏精度取决于纠偏表的精度，即建表时进行网格搜索的间隔。例如 0.1 精度的谷歌地图纠偏，0.1 精度是指偏移数据的间隔，即因为加偏的局部类线性变化，对于那些小数点后一位之前均相同的经纬度，采用同样的纠偏值。尽管这样处理仍会存在偏差，但偏差范围却缩小在 10 米到 20 米之间，与之前百米甚至千米的误差相比，这种精度大可满足民用的需求。精度更高的 0.01 度纠偏表可以使偏差范围缩小到 5 米到 10 米。

对于北京的人民英雄纪念碑，它的地理坐标点是(39.903247,116.391561)，在英文版谷歌地图的卫星视图下显示如图 3-4，在纠偏表中查找坐标点(39.903247,116.391561)对应的偏差为(0.001251,0.006056)，真实坐标加上偏差即可得到英文版谷歌地图的矢量模式下人民英雄纪念碑对应的坐标点(39.904498,116.397617)，在地图上显示如图 3-6，与正确位置仅存在 14 米的偏差。

这种纠偏方法快速、有效，加之地图的偏移算法不会轻易改动，所以偏移表可做到“一次工作，永久有效”。但是同时也存在着不足，电子地图运营商并不会提供真实坐标与偏移后坐标的对应关系，这个对应关系需要花费大量人力查

找、校对，因此得到一个完整有效的纠偏表并不容易。

3.3.3 基于 API 的实现

不同电子地图坐标因采用不同坐标系、不同加密算法，造成同一地理实体在不同地图上的坐标不一致，这个问题使得用户在日常使用这些信息时遇到了麻烦。为了方便用户，大部分电子地图运营商提供了开放的 API(应用程序接口)，部分 API 可以实现不同地图间的坐标转换。经调研，图 3-8 中所示的坐标转换均可实现，具体转换函数或接口，在其地图官方网站 API 中均可找到。例如图 3-8 中的“GPS 坐标 → 百度地图坐标”，它对应的接口为“<http://api.map.baidu.com/ag/coord/convert?from=0&to=4&x=longitude&y=latitude>”，其中 longitude、latitude 参数为 GPS 模式下的经纬度，“from=0&to=4”表示从 GSP 坐标转换到百度坐标，若换成“from=2&to=4”则表示从谷歌坐标转换到百度坐标。同样以北京的人民英雄纪念碑为例，它的 GSP 坐标为 (39.903247,116.391561)，经接口转换结果为 Base64 编码的 {"error":0,"x":"MTE2LjQwNDE4Mzg1NDEy","y":"MzkuOTExMDAwMDA1OTM4"}，解码后的对应的百度坐标(取小数后六位)为：(39.911000,116.404183)，该点在百度地图上显示如图 3-9。

基于 API 的坐标纠偏使用方便，但是它存在不少局限性，具体表现为：

一、不确定、难以控制。因为纠偏工作过渡依赖网络，对数据的处理速度随网络实时变化；

二、可使用 API 较少。有些 API 的坐标转换功能不公开或者有偿提供；

三、对访问次数的限制。有些 API 接口即使免费提供，在访问次数上存在限制，用户不能一次转换大批量的坐标。

本文中提到的经纬度坐标，均按照基于 API 的纠偏方法，转换到了百度地图标准下，以此解决了坐标不统一的问题。

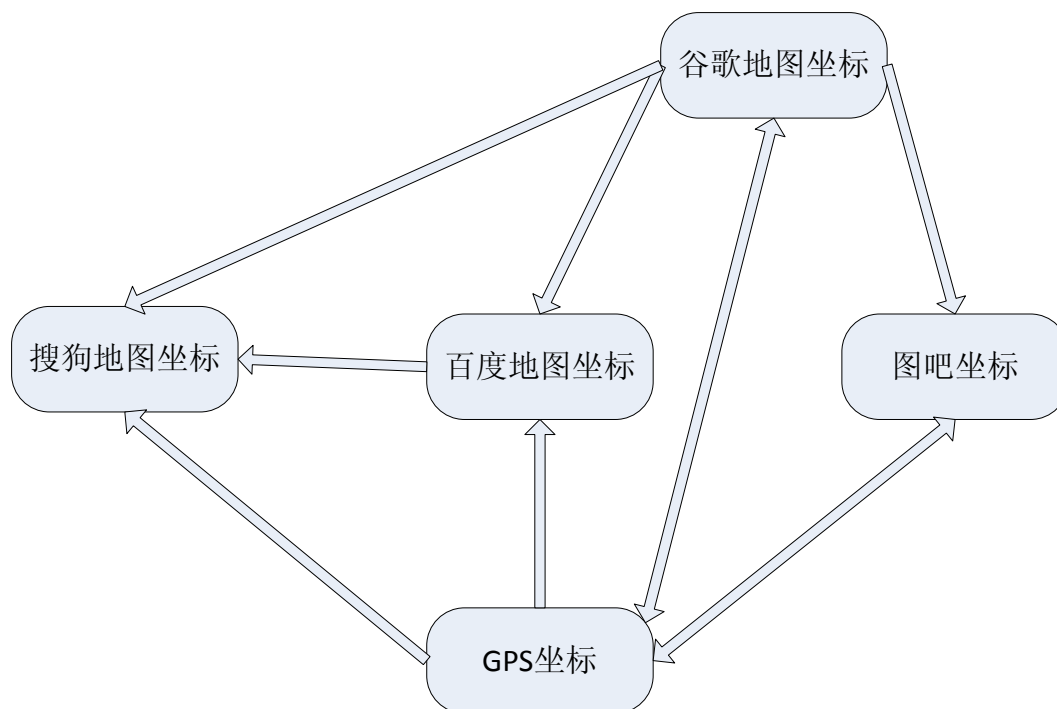


图 3-8 双向箭头表示实现双向转换，单向箭头表示单向转换



图 3-9 GPS 坐标点(39.903247,116.391561)转换为百度坐标
(39.911000,116.404183)

3.4 各字段匹配结果

本文在美团网上抽取了 1095 个完整的 POI，又用这些 POI 的名称字段作为关键字在 google、mapabc、baidu 地图上搜索，共有 8981 条搜索结果。在这些数据中随机选出了 1626 对匹配的 POI 和 2439 对不匹配的 POI 对，作为本小节实验数据。实验过程具体如下。根据之前所提到的相似度定义，计算出这 4065 个 POI 对的相似度。这里提到的相似度共 9 个，包括 edit 相似度、Jaccard 相似度、Jaro 距离相似度、将名称字段分词后的三种相似度(即匹配的最小单位为词的三种相似度)、一般地址相似度、基于小词典的地址相似度、空间地理信息相似度(即经纬度相似度)。根据这些相似度进行了线性回归分析，对不同相似度之间又进行了比较。

3.4.1 字符串相似算法之间的比较

根据 3.1 节中的几种字符串匹配算法计算出 POI 的名称相似度，设置不同的阈值对匹配和不匹配的 POI 进行区分，并计算出其准确率，得到如图 3-10、图 3-11 所示结果。对于 POI 中的名称字符串相似度计算是以字为最小匹配单位，还是以词为最小匹配单位，本文对比这两种假设下的曲线，发现 JW 算法在[0.6,0.8]这部分的结果以词为最小单位的情况好于以字为最小单位，除此之外，其它情况、其它算法都特别相进。

在图 3-10 和图 3-11 中，Levenstein 和 Jaccard 曲线走势相近，在阈值 0.73 处出现最高的准确率 0.66，JW 曲线则在前两条曲线的下方，准确率远不及前两条曲线。0.66 的最高准确度说明，POI 的名称字段的相似程度可以一定程度地分出 POI 是否匹配，但是准确率偏低，仍不能满足本文之后数据融合工作的要求。

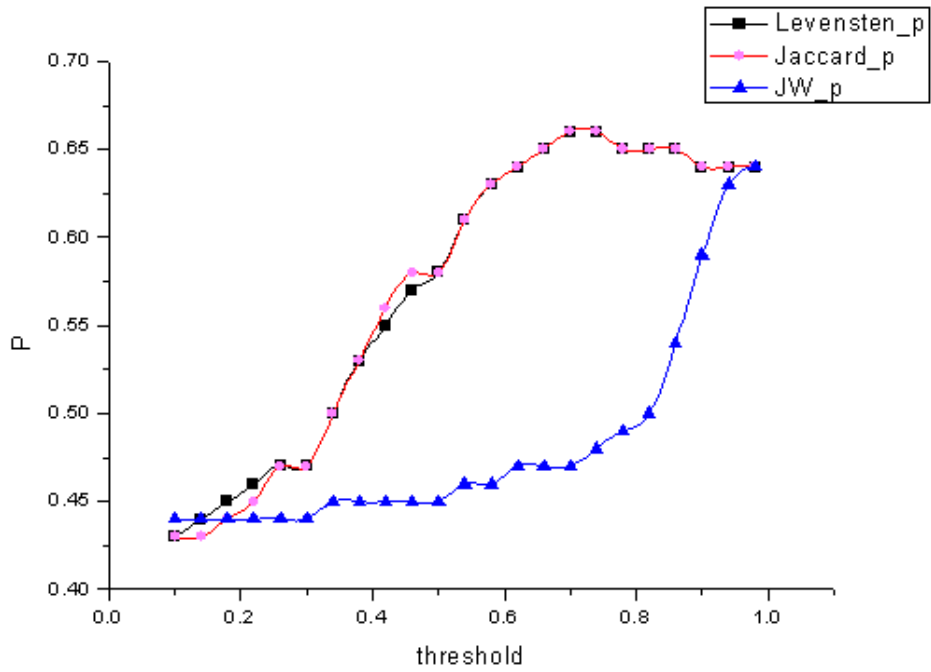


图 3-10 以字为最小匹配单位三种字符串相似算法下的准确率

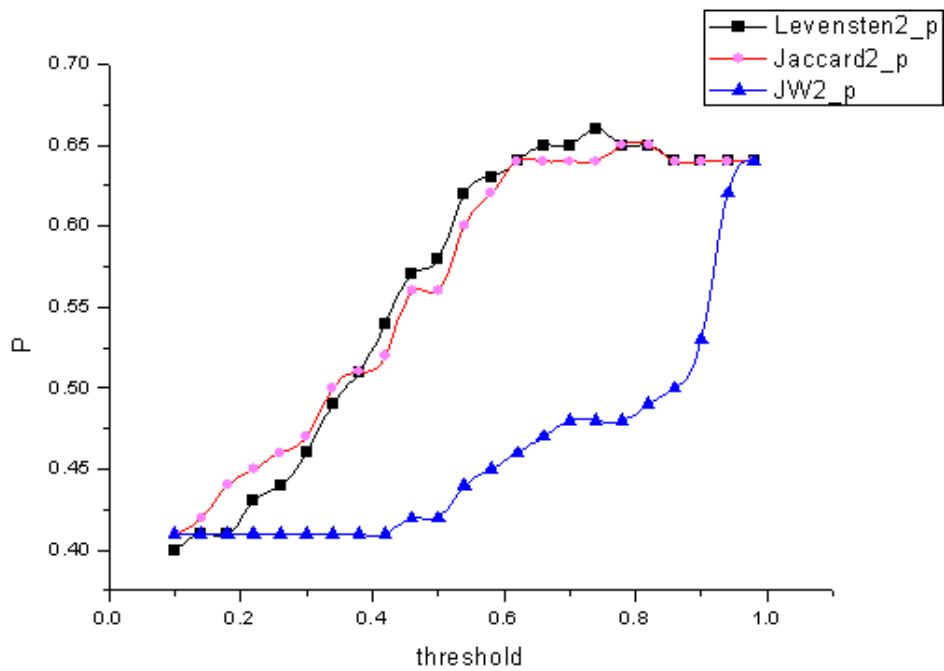


图 3-11 以词为最小匹配单位三种字符串相似算法下的准确率

3.4.2 POI 的三个特征之间的比较

使用 3.2 节中的相似度计算方法，计算出实验数据中 POI 的地址和经纬度相似度，并对其设置了不同的阈值对匹配和不匹配的 POI 进行区分，所得区分结果准确率如图 3-12 所示结果。加上 3.4.1 中 Levenshtein 算法计算得出的名称字段的准确率，可以看出，POI 的地理信息部分可以有效地区分 POI 匹配与否，效果比

名称字段好得多。经纬度字段虽然出现了较高的准确度，但是其总体还是低于地址部分的表现。地址部分的阈值较小时仍能有较高的准确率，这一事实说明现阶段表示不同实体的 POI，其地址相似、或相近的情况不是很多，这也可能与我国的 POI 记录覆盖不全有关。

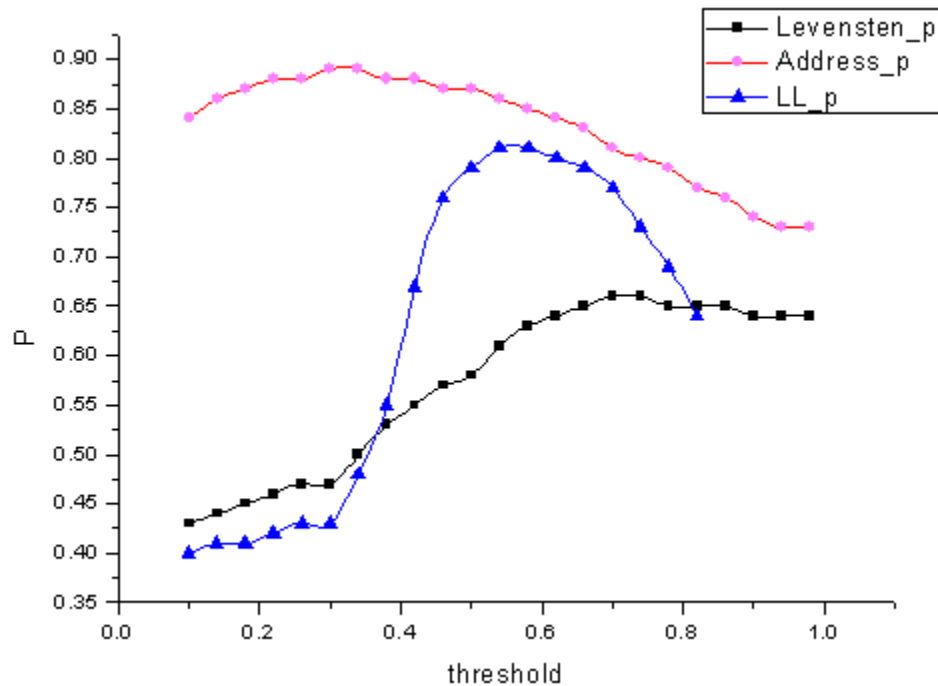


图 3-12 POI 的名称、地址、经纬度相似度在区分、匹配过程中的不同表现

3.5 小结

本章将运用自然语言处理的相关知识和技术，形式化表示出 POI 中的名称、地址、经纬度特征字段的相似度。对于 POI 中经纬度标准不统一这个问题，提出了两种解决方法，即基于纠偏表的方法和基于 API 的方法。最后用一定数量的标注了匹配与否的 POI 数据逻辑回归，找出匹配、分类效果较好的特征及其对应的阈值。结果表明，地址字段的匹配、分类结果较好。

4 可融合 POI 的分类

数据生产商创新性地数据的采集工作转移到室内，实现采集过程的自动化，但是提高了采集速度的同时，也带来了新的问题。新采集的数据是否正确、是否真实存在，是否与之前已有数据相同，这一问决定了这些新数据的实际应用价值。本节使用机器学习领域中分类方法，对新采集的数据是否有实际价值进行区分，将具有实际价值的信息进行融合^{[46][47][48]}操作，得到最终可用的 POI。本节的工作主要是为了提高之后数据融合的准确程度。

本章首先对新采集数据真伪性问题进行了介绍，而后利用基于规则分类方法、基于机器学习的分类方法对新采集数据进行分类，并用相应的实验对问题进行了仿真，分析了结果。

4.1 问题描述

大多数 POI 信息数据生产厂家的数据采集方式主要还是依靠人海战术，雇用大量的调绘、调查人员，对城市进行地毯式作业。这样的作业方式，效率很低，成本很高，并且无法及时更新，因此部分厂家根据自己的经验，创造性地将数据采集工作转移到了室内。像卡贝斯这样的专业 POI 生产厂商，对互联网数据做了实时监测，分类抓取互联网上同 POI 相关的信息。以餐饮企业为例，卡贝斯第一次在互联网上抓取了 22 万个餐饮相关的数据，第二个月继续抓取，拿到了 22.2 万的数据，之后就这 2000 个新出现的数据通过特有的电话情景脚本进行数据的验真、完善。同样的方法可以完成所有类型的 POI 信息采集。和卡贝斯一样，大多远程采集机制可以充分把握住新出现的 POI 信息，但忽略了那些原有 POI 信息变化，使得数据的准确度降低。还是以餐饮业为例，餐馆的节假日活动可能会频繁的变化，按照上边的机制这部分信息就不能在 POI 中被更新，甚至餐馆因为迁址导致地址这一关键字段发生的变化，也不会被更新，造成这个 POI 价值骤减。还有些餐馆因经营不善而关门倒闭，但是他的 POI 信息仍然出现在数据库里，成为无用的“死点”，久而久之便会出现大量的冗余。这些情况下，POI 的准确性就会受到很大的影响。

为解决以上问题，本节使用机器学习领域中分类方法予以初步解决。在互联网上抓取数据之后，筛选出 POI 中相关字段信息，根据这些信息与原有 POI 的关

系对其进行分类处理。主要分为可融合和不可融合两类。可融合是指该 POI 信息已经存在, 只需对这些信息进行融合, 对部分字段进行更新处理; 不可融合则是指该 POI 信息不在现有数据集中, 可能是新出现的 POI 信息, 也可能是错误不真实的 POI。对这些不可融合 POI 信息真伪的验证, 可以像卡贝斯那样通过电话情景脚本的方式实现, 也可以运用自然语言处理相关技术实现。对于验证为正确、真实存在的 POI, 可以进行融合处理, 最后便可作为一条有效信息添加到数据集中。而对于那些被验证为错误、甚至不存在的 POI, 本文不作任何处理, 但本文会对原有数据集中那些与之相似的 POI 进行一次验证, 并去除其中的“死点”。

4.2 基于规则的分类

为了满足 POI 融合阶段对数据精度的要求, 本文对待融合的 POI 进行了可融合与不可融合的分类。本小节根据人工对待融合 POI 分类时一般要经过的几个步骤, 制定了一些分类规则, 并将这些规则模型程序化, 对其进行了仿真实验, 证明该规则分类模型的有效性。

4.2.1 分类模型

通常, 判断一个 POI 是否为可融合 POI, 主要是根据 POI 的名称、地址和经纬度进行判断。根据这三个条件进行判断时, 存在以下三种情况:

- (1) 已有 POI 中存在“名称、地址、经纬度完全相同”的数据, 本文认为是可融合 POI, 对除这三个字段的信息进行融合后, 可更新原有的 POI 信息。
- (2) 已有 POI 中存在“名称相同, 地址、经纬度不同(可能该 POI 已迁址)”或“地址、经纬度相同, 名称不同(可能是该 POI 已经更名或使用了别名)”的数据, 本文则需要对这三个关键字段进行更正、确认, 而后再对其它字段进行融合, 更新原有 POI 信息。
- (3) 已有 POI 与这个待融合 POI “名称、地址、经纬度完全不相同”, 本文认为这是新增数据, 不可融合, 需要对其进行特殊形式的确认。

这三种情况中的“相同”是指 POI 对应字段的相似度大于文中选定的阈值, 具体相似度的计算采用第三章中的方法, 这些阈值的确定放在之后的回归实验中实现。

在数据集中对待融合 POI 进行模糊搜索, 得到的 POI 数据集记作集合 R , R

中 POI 的个数记为 $total$ ，并计算出 R 中每个 POI 与这个待融合 POI 各字段的相似度。假设名称、地址、经纬度三个字段对应的相似度阈值分别是 T_title 、 $T_address$ 、 T_LL 。根据可融合 POI 判断的三种情况，本文制定了一些分类规则，具体则如下：

- (1) 当 R 中存在相似度大于三个阈值中任意一个的 POI 时，判断为可融合 POI。
- (2) 当 R 中的地址相似度、经纬度相似度均小于阈值时，对 R 中 POI 根据经纬度相似度进行降序排序，选取前 N 个，统计其中名称相似度大于 $a * T_title$ 的个数 M ，其中 $0 < a < 1$ 。当 $M > 0$ 时，判断其为可融合 POI。
- (3) 当 R 中的名称相似度均小于阈值，最大的地址相似度 $> b * T_address$ ，或者最大的经纬度相似度 $> c * T_LL$ 时，统计 R 中名称字段的相似度大于 $a * T_title$ 的个数 K ，如果 $K > 0$ ，则判断其为可融合 POI。
- (4) 当 R 中的名称相似度均小于阈值，并且最大的地址相似度 $\leq b * T_address$ ，最大的经纬度相似度 $\leq c * T_LL$ 时，统计 R 中名称字段的相似度大于 $a * T_title$ 的个数 K ，如果 $K > 0.5 * total$ ，则判断其为可融合 POI。
- (5) 其它情况均判断为不可融合的 POI。

对于过程中的相似度阈值 T_title 、 $T_address$ 、 T_LL ，以及变量 a 、 b 、 c 的具体取值，我们根据实验中数据线性回归的具体情况确定。

以上规则对应的流程图如图 4-1 所示。

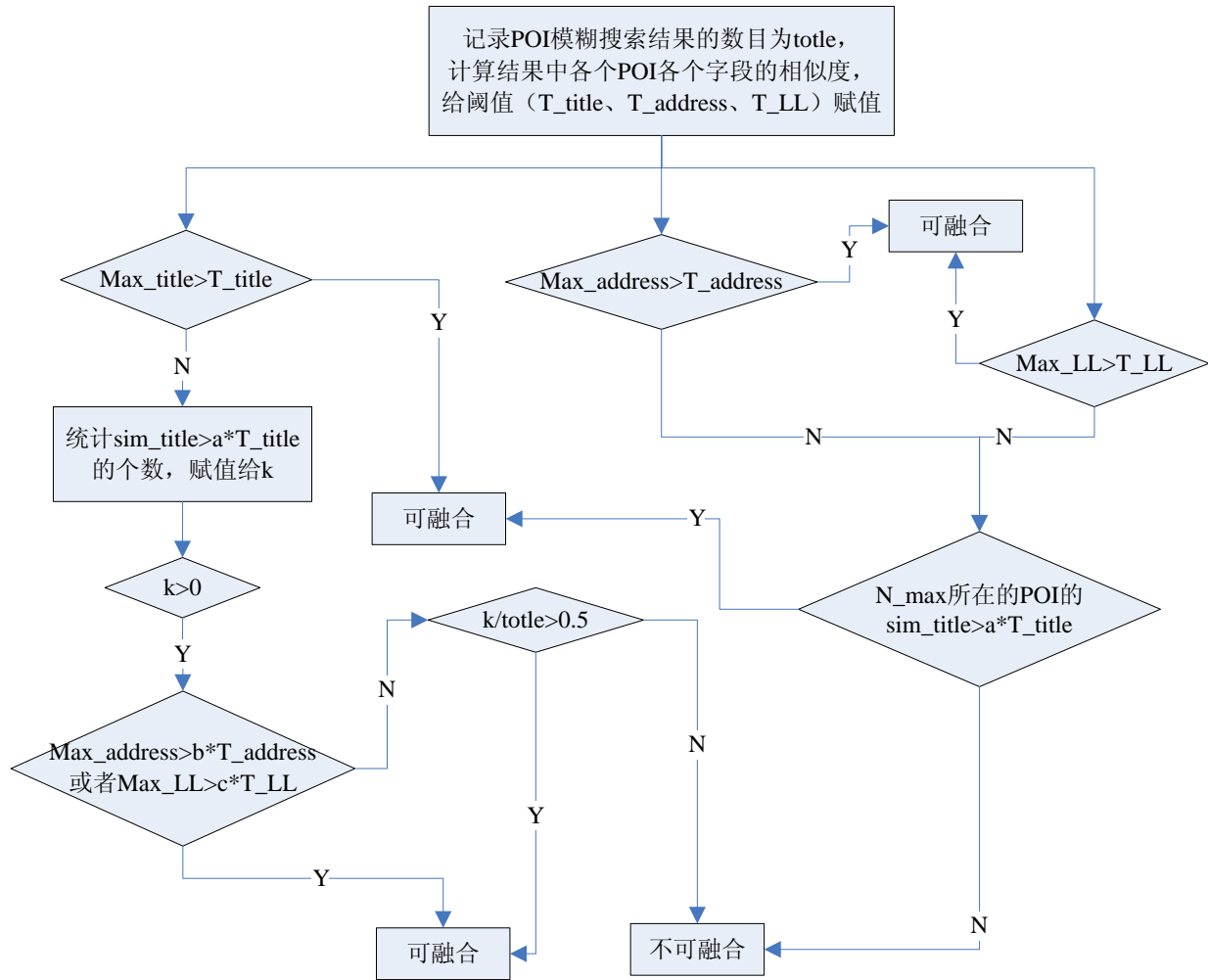


图 4-1 基于规则的可融合 POI 的判断过程

4.2.2 实验数据介绍

目前，人们不断增长的消费能力催生出了许多面向餐饮领域的信息提供商，诸如大众点评网、去哪儿网、团购网等，并且随着时间推移它们的规模也在不断扩大。它们所提供的信息内容丰富，并且实时性相对较高，利用数据挖掘技术加以提取、规整格式、融合各方面信息，便可作为 POI 信息源直接使用，具有较好的应用价值。实验中，本文从美团团购网站上抽取了 1095 个页面，每个页面上有一个 POI 信息，即之前提到的待分类 POI，随后在 google 地图、mapabc、baidu 地图上按照待分类 POI 中的名称字段进行模糊搜索，将搜索结果集作为现有 POI 集合。本文对于这些数据进行了人工标注，根据现有 POI 集合判断其对应的待分类 POI 是否可被融合，标注结果中 744 个 POI 是可融合的，238 个是不可融合，其余 113 个 POI 模糊搜索没有结果，本文不予以考虑。将以上可融合的和不可融

合的(共 982 个)POI 转换成向量集, 作为本节实验的数据集。

4.2.3 分类结果级分析

本文首先分别对 POI 中的各特征字段的相似度进行线性回归, 通过设置不同的阈值进行分类, 得到每个特征相似度单独参与分类的表现(见图 4-2、4-3、4-4):

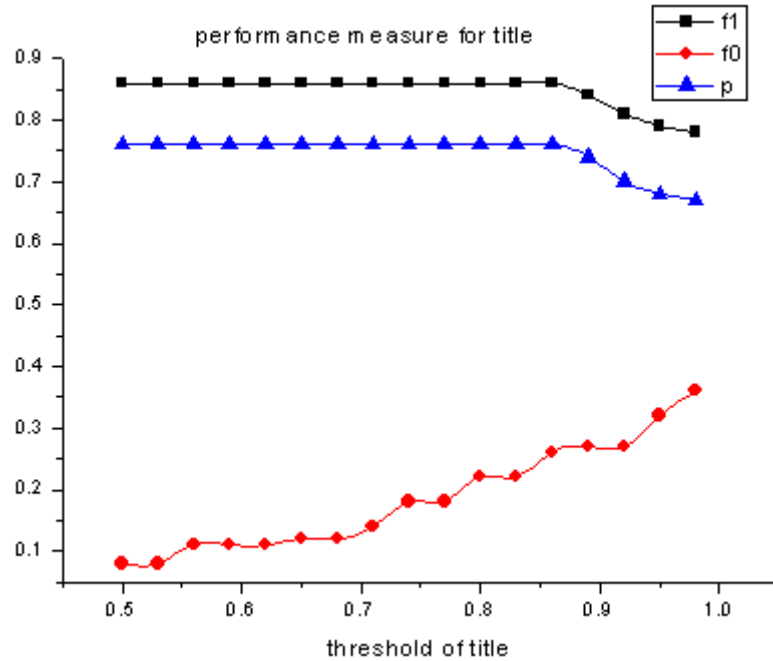


图 4-2 名称字段分类结果

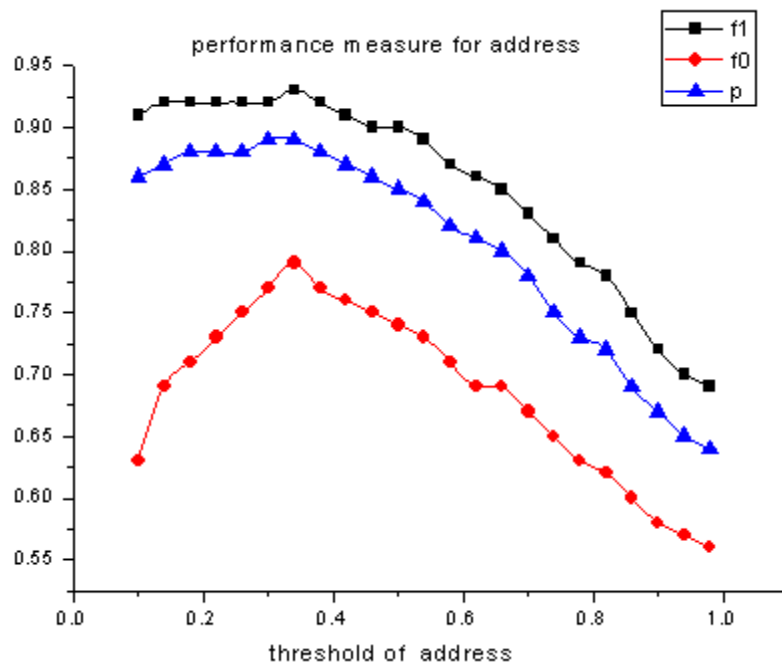


图 4-3 地址字段分类结果

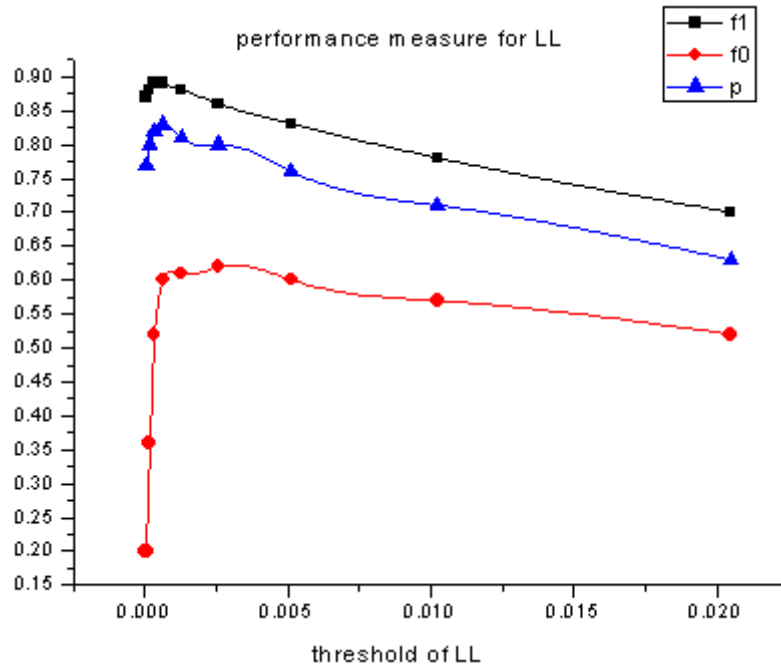


图 4-4 经纬度字段分类结果

图中 $f1$ 是可融合的 F 值, $f0$ 是不可融合的 F 值, p 为整个分类结果的准确率。从三个图中可以看出, 无论是哪个字段, p 和 $f1$ 的变化趋势是一样的, 且 $f1$ 总是处于最上方, $f0$ 总是处于最下方。因为可融合的 POI 占大部分, 所以 $f1$ 会更大程度地影响整体分类结果。图中的峰值并不是说此时的 $p1$ (可融合集的正确率) 或 $r1$ (可融合集的召回率) 是最大值, 而是说 $p1$ 和 $r1$ 处在一个最佳的平衡点, 不至于两个值一个过太一个过小。对于 $p0$ 、 $r0$ 也是一样。在图 4-2 中, p 和 $f1$ 在 $[0.85, 1]$ 区间内逐渐减小, 对应的 $f0$ 不断增大, 但最大值仍旧很小, 此时所有 POI 分类的结果为可融合。在图 4-3 中, 三个曲线同增减, 并在 0.36 处出现峰值。图 4-4 中的三条线变化趋势也相同, 且在 0.001 处出现峰值, 同样是这种情况下的平衡点。具体结果见表 4-1。

从上述结果分析可知, POI 中各字段在区分可融合、不可融合分类过程的表现不同, 其分类效果由弱到强分别是名称、经纬度、地址字段。名称字段之所以比较差, 主要因为现有 POI 集中的 POI 是根据待分类 POI 的名称进行模糊搜索得到的, 它们的名称相似度已经很高, 不足以有效区分 POI。其中对地址和经纬度字段进行了融合, 其结果表现的最佳。

表 4-1 根据不同字段分类的最佳阈值及结果

POI 字段 (阈值)	指标 分类	准确率	召回率	F 值	总 准 确 率
名称 (0.96)	可融合	0.79	0.77	0.78	0.69
	不可融合	0.34	0.37	0.35	
地址 (0.36)	可融合	0.95	0.90	0.92	0.89
	不可融合	0.73	0.85	0.78	
经纬度 (0.001)	可融合	0.87	0.90	0.89	0.82
	不可融合	0.65	0.60	0.62	
地址、经纬度 (0.32、0.08)	可融合	0.93	0.94	0.93	0.90
	不可融合	0.80	0.77	0.79	

4.3 基于机器学习的分类

在之前的可融合分类过程中，设置 POI 各字段内部系数及阈值，经过回归计算，选取其区分 POI 是否可融合效果最好的一组系数和阈值构建模型。这个计算过程复杂、耗时，并且不够灵活，不具备自动学习的能力，因此，本节接下来用一些经典机器学习算法训练模型，实现自动、高效、灵活的判定模型。

4.3.1 POI 相似度表示

在互联网上抓取感兴趣的网页，筛选出其中与 POI 相关的字段信息，之后对其进行分类处理，本文将 POI 分为可融合和不可融合两类。分类的依据则是这个 POI 信息与现有 POI 集的关系。这里的现有 POI 集并不是数据库中所有数据，而是通过在互联网上抽取的 POI 名称字段在电子地图提供的数据库中进行模糊搜索的结果集。

为了方便构建模型，本文将之前提到的待分类 POI 与现有 POI 集的关系转换成为一个向量，该向量中包括这个待分类 POI 和现有 POI 集中的各特征字段相似度的最大值，即

$$\left(\text{edit}(p, p_i), \text{jacc}(p, p_j), \text{jaro}(p, p_k), \text{address}(p, p_l), \text{LLsim}(p, p_m) \right) \quad (\text{式 4-1})$$

其中 p 是待分类 POI， p_x 是现有 POI 集合中的某个 POI， i, j, k, l, m 可以不同，但必须使得其所在的函数值在组内最大； $\text{edit}(p, p_i)$ 、 $\text{jacc}(p, p_j)$ 、 $\text{jaro}(p, p_k)$ 分别表示之前提到的两个 POI 名称字段 Levenstein 相似度、Jaccard 相似度、Jaro 相似度，

$address(p, p_l)$ 是两个 POI 的非空间地理信息相似度, $LLsim(p, p_m)$ 是两个 POI 的空间地理信息相似度。

4.3.2 机器学习分类模型

通过大量数据转换得到的向量集, 将作为训练集, 依据机器学习的方法构建出分类模型, 将 POI 实例分为可融合和不可融合两类。可融合是指该 POI 信息已经存在, 只需对这些信息进行融合, 对部分字段进行更新处理; 不可融合则是指该 POI 信息不在现有数据集中, 可能是新出现的 POI 信息, 也可能是错误不真实的 POI。对这些不可融合信息真伪性的验证, 可以像卡贝斯那样通过电话情景脚本的方式实现, 也可以运用自然语言处理相关技术实现。对于验证为正确、真实存在的 POI, 对其进行融合后便可作为有效信息添加到数据集中。而对那些验证为错误、甚至不存在的 POI 不作任何处理, 是对原有数据集中那些与之相似的 POI 要经过验证, 最终去除其中的“死点”。

在实验中, 我们运用了几个不同的分类器, 其中包括 k-近邻、C4.5 分类器、Adaboost 提升分类器。每个分类器都有各式各样、复杂的标准, 利用这些标准构造不同的模型, 本文的第二章对其进行了详细介绍。比如, C4.5 采用信息增益比作为选择测试属性的标准, 从根节点开始, 赋予最好的属性, 再将该属性的各个取值产生出相应的分支, 这些分支上又生成新的节点, 加之一些剪枝方法构造出决策树, 使其最大程度地拟合训练集。C4.5 分类器产生的分类规则非常便于理解, 准确率也非常高, 但是在构造决策树时需要多次地扫描数据集, 还要进行排序, 因此导致这种算法的效率十分低下。除此之外, C4.5 还要求数据集能够驻留在内存中, 但是当训练数据集非常大, 以至于无法在内存容纳, 这时 C4.5 算法程序就不能正常运行了。

4.3.3 机器学习模型训练与分类过程

对 POI 数据进行是否可融合的人工分类, 标注分类结果, 之后就可以根据实际需要, 训练不同的基于机器学习的分类模型, 并利用训练好的分类模型进行 POI 可融合分类。模型的训练和分类的流程见图 4-5。

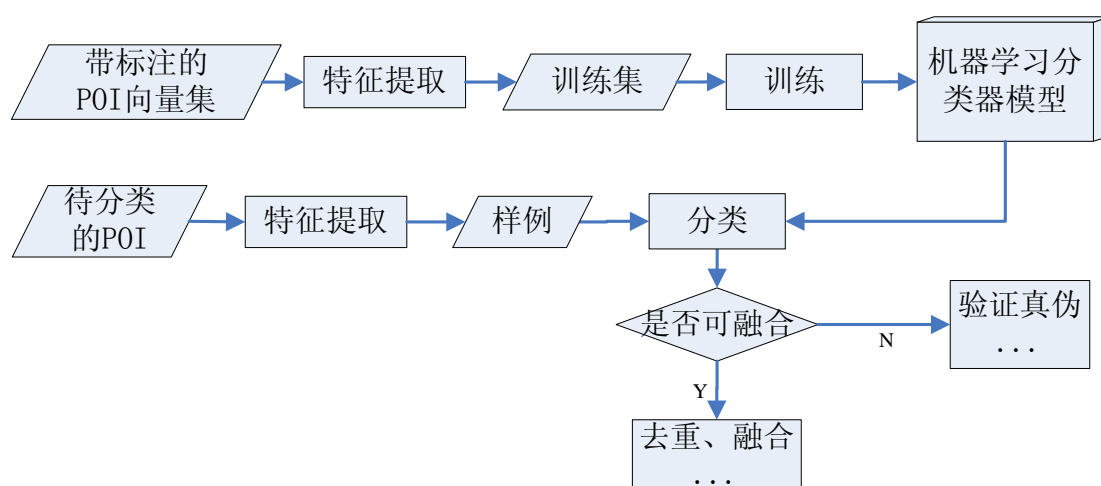


图 4-5 模型训练与分类

训练模型的过程如下：

输入：人工标注的 POI 集合

输出：机器学习的分类模型

将带标注的 POI 集合通过样例集自动生成模块进行特征提取，生成训练集，利用 weka 提供的对象训练分类器。对于 C4.5 分类算法来说，weka 提供了 `weka.classifiers.Classifier.trees.J48` 对象，首先用 `Classifier m_Classifier = new J48();` 创建一个 J48 的对象，然后利用 Classifier 对象的 `buildClassifier` 方法训练决策树模型。对于 K-近邻算法，weka 提供了 `weka.classifiers.Classifier.lazy.IBk` 对象；对于强分类提升算法，它提供了 `weka.classifiers.Classifier.meta.AdaBoostM1` 对象，其具体使用方法和 J48 相同。

分类模型的分类过程如下：

输入：一个待分类的 POI(未标注)

输出：预测的分类，或者是可融合的 POI，或者是不可融合的 POI

用训练好的分类模型创建一个 `weka.classifiers.Classifier` 对象，利用 Classifier 对象的 `classifyInstance` 方法，对样例集自动生成模块生成的一条样例进行预测分类。

4.3.4 实验结果及分析

本节实验中，继续使用 4.2.2 节提到的向量集作为实验数据，运用 k-近邻、C4.5、Adaboost 三种分类器对数据集进行了训练、测试，因为数据有限，所以在

这里采用了十折交叉验证的方法。分类结果(见表 4-2)中可看出, 各分类器效果差不多, 对可融合的 POI 分类较好, 但对不可融合部分各指标还是偏低。总体来说, C4.5 效果较好, 总的准确率达到了 90.3%, 适合应用在这个分类中。

表 4-2 不同分类器的分类结果

分类器	指标 分类	准确率	召回率	F 值	总准确率
IBk	可融合	0.9	0.897	0.898	0.847
	不可融合	0.68	0.689	0.685	
C4.5	可融合	0.916	0.964	0.939	0.903
	不可融合	0.864	0.723	0.787	
AdaBoostM1	可融合	0.937	0.921	0.929	0.895
	不可融合	0.765	0.807	0.785	

4.4 小结

本节根据 POI 各个特征字段的相似度, 构造出 POI 可融合判断模型, 并对网络上抽取的 POI 数据进行有效分类。最后实验结果准确率可达到 90%左右, 验证了根据相似度构建分类模型的正确性和可行性。同时还说明对 POI 各字段进行适当的融合, 对其分类可以起到一定的积极作用。

5 多源 POI 数据融合系统

基于以上研究成果,本文初步完成了多源 POI 数据融合系统的可融合判定部分,这部分工作的目标是对抽取部分得到 POI 的三个关键字段进行特征表示,通过构建的分类模型,完成自动分类。本章简单介绍了多源 POI 数据融合系统的流程,以及分类模型在融合系统中的作用,同时还介绍了系统中网络地图的选择、网络访问限制的问题。

5.1 多源 POI 数据融合系统的简介

整个系统完成对用户站点信息抽取、信息模式化表示、可融合判定,最后对 POI 中各字段融合,得到具有实际使用价值的 POI 数据。整个过程主要分成了三个功能模块:

(1) 根据用户提供的门户网站进行 POI 信息自动抽取^{[49][50]}模块

(2) 在电子地图中对抽取得到的 POI 进行模糊搜索,根据 POI 和搜索结果的关系,进行 POI 可融合判定的模块

(3) 根据不同的判定结果,对 POI 所有字段融合的模块

5.1.1 系统流程

之前章节中已经详细介绍 POI 各特征字段的形式化表示、可融合判定模型的构建以及可融合 POI 的分类过程,所以这一节主要介绍了多源 POI 数据融合系统其它部分。具体过程如图 5-1 所示。图中 POI 的关键字段指的是其名称、地址、经纬度部分,没有考虑 POI 类别字段的主要原因是,整个融合过程中我们只涉及了 POI 信息变换频繁的餐饮、娱乐等领域,类别单一,信息量少。POI 的其它字段主要有店铺的电话、营业时间、特色、风格、评论等信息。

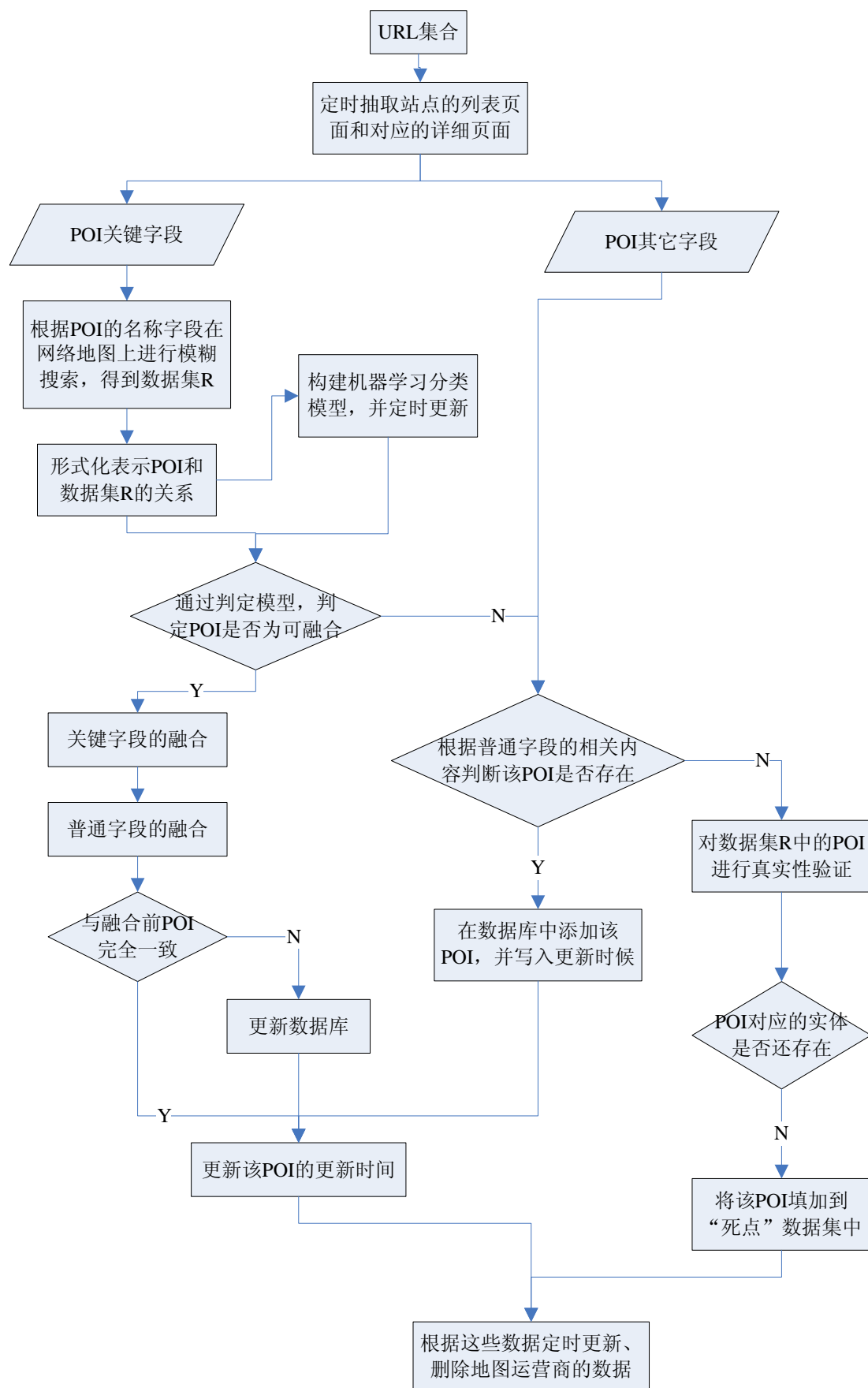


图 5-1 多源 POI 融合系统的流程图

表 5-1 国内常用地图的基本情况

API 提供商 功能	MapABC	百度地图	51 地图	Google 地 图	365 地图	MapBar	备注
地图接口	√	√	√	√	√	√	基本的地图操作, 包含测距功能
搜索功能	√	√	√	√	√	√ (收费)	模糊搜索功能, 含周边查询
GPS 坐标标注	√ (收费)	△	—	△	—	—	通过 GPS 的 ID, 解析坐标并在地图上显示
地址解析和逆地址解析	√	√ (次数限制)	√ (逆地址解析收费)	√ (次数限制)	√ (收费)	√ (收费)	地址转换成经纬度坐标信息和坐标信息转换成城市等地址信息
环境要求	IE 7+、 FireFox 3+、 Flash Player 10+	IE 6.0+、 Firefox 3.0+、 Opera 9.0+、 Safari 3.0+、 Chrome	IE5.5+、 FireFox1.0+、 Opera 8.0+	IE 6.0+、 Firefox 2.0+、Safari 3.1+	IE 6.0+、 Firefox 2.0+	IE 6.0+、 Firefox 2.0+	对于操作系统的要求不再给出, 同时有些其他的要求譬如 Flash Player 的要求可能未给出
接口语言	JS、AS3	JS	Iframe、JS、 Http+Xml、 WebService	JS、.NET、 AS	JS	JS	由于各个提供商的描述标准不同, 可能描述有偏差
更新周期	1-2 次/年	1-2 次/年	△	热点地区更新较快, 其他地区较慢, 1-2 次/年	△	1-2 次/年	地图更新取决于基础地图数据供应商的更新速度
“√”表示提供此服务, “—”表示没有此服务, “△”表示情况不明							
附注:	雅虎的地图目前不支持中国地图开发, 因此不予考虑。						

5.1.2 网络地图的选择

在多源 POI 数据融合系统中, 本文用到了网络地图中的部分信息, 主要通过地图的模糊搜索功能完成。表 5-1 中详细列出了几个目前我国常用地图的信息, 综合考虑, 本文选则了谷歌、百度、MapABC, 原因如下所列:

- (1) 三者都提供了 API 接口, 便于我们根据自己需要进行使用。
- (2) 提供了免费的模糊搜索功能, 可以满足数据融合系统的要求。三者的数据更新周期相对明确, 同时, 谷歌还有自己的数据维护、更新机制。
- (3) 考虑到融合过程中地理信息部分信息的缺失, 即地址、经纬度只有一个的情况, 我们可以利用地图自己所提供的地址解析、逆地址解析功能, 进行信息的完善。
- (4) 三者都可在 IE 浏览器中使用, 并且都可以用 JS 作为接口语言, 这样就

可以方便地在 WebBrowser 控件(在 .NET Framework 2.0 版中新增的控件类)下集成使用, 进而完成数据融合系统中网络地图信息的批量、自动抽取。

5.1.3 http 代理服务的使用

和谷歌提供的地址解析功能一样, 很多数据查询功能提供商对用户的使用进行了限制, 或是次数限制, 或是使用频率限制。这些限制影响了本文多源 POI 数据融合系统中获取网络信息的相关工作的顺利进行。因此, 本文使用广泛存在于网络世界的代理机制, 解决了网络访问限制的问题。

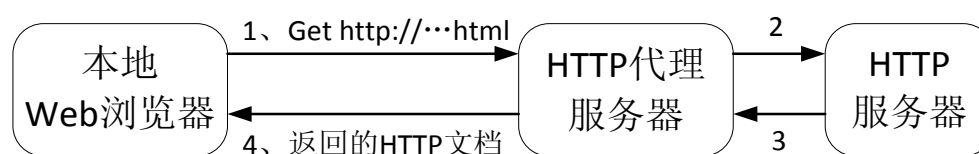


图 5-2 HTTP 代理服务器通信过程

代理服务器^{[51][52]}实际是转发机构, 它位于本地客户端和远程服务器之间, 客户端提出服务请求后, 代理服务器接收这个请求, 紧接着把同样的请求转给服务器, 服务器响应请求并把结果返回到代理服务器处, 最后由代理服务器把结果交至客户端。假如这个代理服务器提供的是 HTTP 代理功能, 那么就称它为 HTTP 代理服务器, 其通信过程如图 5-2 所示, 具体可分为 4 步: (1) 本地 Web 浏览器向 HTTP 代理服务器发送请求, 请求中包含了要访问的 URL 地址; (2) HTTP 代理服务器读取这个 URL 后将之前接收的请求再传送给 HTTP 服务器; (3) 代理服务器接收来自网络上终点计算机的响应; (4) 将 HTTP 响应文档传回本地 Web 浏览器。这样通过代理服务器, 可以对另一个服务器进行访问, 同时隐藏本地客户端的具体信息。

本文涉及的程序在 Java SDK 1.6 版本中调试通过, 其中 HTTP 代理是通过设置 Java 环境变量来实现的, 也就是 JVM 的系统属性。用户也可以参照 HTTP 代理设置方法, 完成 FTP、socket 等其它网络代理的设置。

设置 HTTP 代理需要两个属性, 分别是代理服务器的 IP 地址 proxyHost 和代理服务器的端口地址 proxyPort。有两种方法设置这种系统属性, 一种是命令行下运行 Java 程序的时候, 通过参数设置, 格式如下^[53]:

```
java -Dhttp.proxyHost=187.111.11.6 -Dhttp.proxyPort=8080 MyJavaApp
```

另一种方法, 就是直接在源程序中设置, 如下:

```
import java.util.Properties;
.....

.....

prop.setProperty("http.proxyHost", "187.111.11.6");
// 设置 http 访问要使用的代理服务器的地址
prop.setProperty("http.proxyPort", "8080");
// 设置 http 访问要使用的代理服务器的端口
.....
```

在网络中存在大量的免费代理服务器的 IP、端口，我们可以选取其中一些相对速度快、效率高的代理，定时轮换使用，避免代理服务器被限制，以达到持续使用这些代理的效果。

5.2 小结

本章介绍了多源 POI 数据融合系统主要分为 POI 自动抽取模块、可融合判定的模块、所有字段融合模块三部分，同时还简单介绍了该系统的流程，以及分类模型在融合系统中的作用。

本章通过比较现在国内几个主流网络地图的功能特征，选择了 Google 地图、百度地图、MapABC 地图作为可融合判定模块中网络模糊搜索的平台。最后针对网络访问限制这一问题，本章介绍了代理机制，并使用 http 代理服务器解决了这个问题。

6 总结与展望

6.1 总结

本文在多源 POI 数据融合方面进行了深入而系统的研究，最终完成了可融合 POI 自动、有效的分类，实现了多源 POI 数据融合系统的部分功能。本文的主要研究内容和创新点如下所述：

(1) 本文利用自然语言处理的方法，根据 POI 中各特征字段的形式、特点，定义了 POI 各字段的相似度，并以此来表示待分类 POI 与原有 POI 集的关系。在计算 POI 名称字段相似度的过程中，本文对基于字面相似的方法进行了改进，对字符串进行快速分词后，以词为最小单位进行相似度计算，打破了传统中以字为最小匹配单位的假设，避免了汉字自身的差异对中文字符串相似度的影响。实验结果证明了改进方法的有效性。另有实验表明在 POI 三个关键字段中，地址字段的匹配、分类结果较好。

(2) 针对不同地图之间的偏差造成 POI 中经纬度标准不统一这个问题，提出了两种解决方法，即基于纠偏表的方法和基于 API 的方法，并通过实验验证了该方法的有效性。

(3) 融合了 POI 的非空间信息和空间信息，即融合了名称、地址、经纬度三个字段的相似度，作为判定可融合 POI 的依据。之后通过一个基于规则的模型对 POI 进行分类判断。实验结果证明融合后的信息可有效的对 POI 进行分类，准确率可达到 90%左右。

(4) 利用机器学习中的分类方法，构建了具备自主学习能力的 POI 可融合分类模型。实验表明分类算法的结果相当不错。

(5) 在整个多源 POI 融合系统中涉及了多处大量访问网络的工作，这些工作因网络访问的各种限制而受到影响，本文使用了 http 代理服务器解决了这个问题。

6.2 展望

多源 POI 数据融合系统中的三大模块中的前两个模块，即信息自动抽取模块和 POI 可融合判定的模块，已经初步完成，并通过实验证明了其间各项工作的可行性。完成最后模块后，可以设计并实现一个较为高效的多源 POI 数据融合系统，

系统提供一个简洁友好的用户界面，对用户提供的站点信息进行抽取、校正、融合，返回给用户准确性较高、具有标准结构化的 POI 数据，以及保证数据的查全率和查准率。

本文中各项工作只用到了 POI 的三个关键字段，因为其它字段缺失比较严重所以尚未涉及到 POI 融合判定工作中，但是这些字段同样存在了一些有价值的信息，应该对这些字段内容进一步分析，挖掘更多有用的规则，以达到更好的判定效果。

文中对空间地理信息相似度的计算采用了最简单有效的方法，即计算这两个 POI 点之间的球面距离。但因为这种球面距离计算过程中存在 10 到 20 米的误差，当 POI 点相对密集时，这个误差将严重影响空间地理信息相似度的分类效果。后期工作可以利用地理信息学中知识，寻找出更有效的处理方法。

参考文献

- [1] J.Krosche, S.Boll. The xPOI Concept[C]. Location and Context Awareness, Springer, 2005:113-119.
- [2] 张玲. POI 的分类标准研究[J]. 测绘通报. 2012(1):82-84.
- [3] 易明华, 祝红英, 徐玉玲. 网上电子地图现状浅析[C]. 华东六省一市测绘学会第十一次学术交流会论文集, 2009:394-399.
- [4] 吕志平, 赵冬青. 位置服务系统(LBS)的构建[J]. 测绘科学. 2005(4):92-93.
- [5] 王庆社, 邓南, 刘宁. 兴趣点的检查算法研究与实现[J]. 北京测绘. 2009(4):37-39.
- [6] 万方, 尹为民, 吴迪. 网络数据挖掘及其新技术探讨[J]. 信息技术. 2002, (1)10-11.
- [7] Bing Liu, Robert Grossman, Y Zhai. Mining Data Records in Web Pages[J]. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2003:601-606.
- [8] S.Vivek. Entity Resolution in Geospatial Data Integration[J]. ACM-GIS, 2006, 11: 10-11.
- [9] 王海波. 基于 GPS 与实景影像的 POI 快速采集技术[J]. 中国科技信息, 2007(12):121-122.
- [10] 程桔华. 张政:寻找 POI 行业最有价值的增长点[J]. 中关村. 2007(11):66-68.
- [11] 刘开英, 郭炳炎. 自然语言处理[M]. 北京:科学出版社. 1991.
- [12] Tom M.Mitchell. Machine Learning[M]. 曾华军, 译. 北京:机械工业出版社. 2005:2-10, 38-56, 165-177.
- [13] Ryszard S.Michalshi, Ivan Bratko. Machine Learning and Data Mining:Methods and Applications[M]. 朱明, 译. 北京:电子工业出版社. 2004:67-94, 114-117.
- [14] 戴冬冬. 基于地址匹配方法的 POI 数据更新研究[J]. 电脑知识与技术. 2010(1):1-3.
- [15] 牛永洁, 张成. 多种字符串相似度算法的比较研究[J]. 计算机与数字工程, 2012, 03: 14-17.
- [16] 宋玲, 徐白. 中文检索系统的相似匹配技术研究和实现[J]. 计算机科学. 2010, 37:46-48.
- [17] 江洲, 李琦. 地理编码(Geocoding)的应用研究[J]. 地理与地理信息科学. 2003, 03:24-27.
- [18] 孙亚夫, 陈文斌. 基于分词的地址匹配技术[C]. 中国地理信息系统协会第四次会员代表大会暨第十一届年会论文集. 2007:114-125.
- [19] C.Beerli, Y.Kanza, E.Safra. Object Fusion in Geographic Information System[C]. Proceeding of the 30th VLDB Conference, Toronto, Canada. 2004:816-827.
- [20] 刘恩信. 厦门市二调数据成果 1980 西安坐标转换[J]. 测绘与空间地理信息. 2009, 32(02):198-200.
- [21] 成英燕, 程鹏飞, 秘金钟, 等. 大尺度空间域下 1980 西安体系与 WGS84 坐标系转换方法研究[J]. 测绘通报. 2007(12):8-8.
- [22] 王玉成, 胡伍生. 坐标转换中公共点选取对于转换精度的影响[J]. 现代测绘. 2008, 31(5):13-15.
- [23] Kuzmanovski, I. and Aleksovska, S.. Optimization of artificial neural networks for prediction of the unit cell parameters in orthorhombic perovskites. Comparison with multiple linear regression. Chemometrics and Intelligent Laboratory Systems. v67. 167-174.

- [24] Kashid, D. N.,Kulakarni,S. R.. A more general criterion for subset selection in multiple linear regression[C]. Communications in Statistics: Theory and Methods. 2002,31(5): 795-811.
- [25] Su Jinshu , Zhang Bofeng , Xu Xin. Advances in machine learning based text categorization [J]. Journal of Software . 2006,17(9): 48-59.
- [26] S.B. Kotsiantis. Supervised Machine Learning: A Review of Classification Techniques. Informatica 31(2007). 2007:249-268.
- [27] 曹勇,吴顺祥. KNN 文本分类算法中的特征选取方法研究[J]. 科技信息(科技.教研). 2006(12):26-28.
- [28] 张宁,贾自艳,史忠植. 使用 KNN 算法的文本分类[J]. 计算机工程. 2005,31(8):171-173.
- [29] Quinlan J R. Induction of decision tree[J]. Machine Learning. 1986(1):81-106.
- [30] Quinlan J R. Simplifying Decision Trees[J]. Internet Journal of Man- Machine Studies. 1987,27:221-234.
- [31] Quinlan J R. C4.5: Program for Machine Learning[M]. Morgan Kaufman. 1992.
- [32] Han J, Kamber M. 数据挖掘概念与技术[M].范明,孟小峰,译. 北京:机械工业出版社. 2001.
- [33] Robert E. Schapier, Yoram Singer. Improved Boosting Algorithms Using Confidence-rated Predictions[J]. Machine Learning. 1999, 37(3):297-336.
- [34] 李荣陆. 文本分类及其相关技术研究[博士学位论文]. 上海:复旦大学. 2005.
- [35] Witten I H, Frank E. Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations [M]. Morgan Kaufman. 2003.
- [36] 孙建军等. 信息检索技术[M]. 北京:科学出版社. 2004: 238-240.
- [37] 费洪晓,康松林,朱小娟等. 基于词频统计的中文分词的研究[J]. 计算机工程与应用. 2005 (7) : 67-68.
- [38] 刘群,张华平,俞鸿魁,程学旗. 基于层叠隐马模型的汉语词法分析[J]. 计算机研究与发展. 2004(8).
- [39] http://ictclas.org/hottopic_011.html. ICTCLAS 官方网站.
- [40] 张林曼,吴升. 地理编码系统中地名地址分词算法研究[J]. 测绘科学. 2010,35(2):46-48.
- [41] 中国地名委员会编《中华人民共和国地名录》. 中国社会出版社. 1994.
- [42] Foo S, Li B. Chinese word segmentation accuracy and its effects on information retrieval[J]. Information Processing and Management. 2004,4(1):161-190.
- [43] 蒋景瞳,刘若梅. 地理信息数据质量的概念、评价和表述[J]. 地理信息世界. 2008(2): 5-10.
- [44] 蒋景瞳,何建邦. 地理信息国际标准手册[M]. 北京:中国标准出版社. 2004:53-55.
- [45] International Standard organization. ISO 19101:Geographic Information – Reference Mode[S]. USA, Reston. 2002:5-7.
- [46] V. Sehgal. Entity Resolution in Geospatial Data Integration [J]. ACM-GIS'06. 2006,11,10.
- [47] A. Steinberg, C. Bowman, F. White. Revisions to the JDL data fusion model[C]. Proceedings of the SPIE Sensor Fusion: Architectures, Algorithms, and Applications III, Orlando, FL. 1999(3): 430-441.
- [48] F. White. Data fusion lexicon, Joint Directors of Laboratories, Technical Panel for C3, Data Fusion Subpanel, Naval Ocean Systems Center, San Diego, CA, 1987.
- [49] 董永权. Deep Web 数据集成关键问题研究. 山东大学. 2010.

- [50] D J Buttler, L Liu, C Pu. A Fully Automatic Object Extraction System for the World Wide Web[C]. Proceedings of the 21st International Conference Distributed Computing Systems, Washington DC, USA. 2001:361.
- [51] 曾明,李建军. Internet 访问管理与代理服务器. 北京:人民邮电出版社.1999:3-6.
- [52] Northrup A. NT network plumbing: routers, proxies, and web services[M]. IDG Books Worldwide, Inc. 1998:293-303.
- [53] 刘彦明,李鹏. 实用网络编程技术. 西安:西安电子科技大学出版社.1998:102-114.

致谢

感谢尊敬的导师张巍副教授，感谢他精心的指导和关怀，让我受益匪浅。感谢他对我教导和帮助，为我在学术的道路上指明方向，教会了我进行科研的方法。张巍副教授孜孜以求的治学精神和学术上敏锐的洞察力，对我的科研工作给予了巨大指导和帮助；同时，感谢身边所有的老师和同学们对我生活上的关心，在我困难的时候，总是给我勇气和鼓励；使得我的科研能够顺利完成。

感谢计算机系的全体老师，感谢你们平时的指导和帮助。

感谢同一个科研小组的王秋红同学和高新院同学，感谢他们为该项目所作的工作，以及为解决科研中遇到的问题所付出的努力。

感谢实验室陪伴我师兄师姐周广超、杜冉冉、邓烨以及研二、研一的实验室兄弟姐妹们，对你们平时学习和生活上的帮助表示衷心的感谢。

感谢我身边的同学和朋友，感谢你们对我各方面的帮助。

感谢我的家人，感谢他们对我的支持，感谢他们对我的养育之恩，感谢他们为我付出的一切。

最后，对所有帮助我的老师、同学、朋友和家人表示崇高的敬意和由衷的感谢！

个人简历

1988 年 4 月 7 日出生于山东省聊城市东昌府区。

2006 年 9 月进入青岛大学，2010 年 6 月本科毕业,获得工学学士。

2010 年 9 月进入中国海洋大学信息科学与工程学院计算机软件与理论专业攻读硕士学位至今。

学术论文

- [1] 张巍,李瑞珊,高新院. 基于相似度模型的可融合兴趣点分类研究[J]. 中国海洋大学学报自然科学版,2013,43（12）.(已被中国海洋大学学报.自然科学版录用,核心期刊)
- [2] 张巍,高新院,李瑞珊. 空间位置信息的多源 POI 数据融合[J]. 中国海洋大学学报自然科学版,2013,43（10）.(已被中国海洋大学学报.自然科学版录用,核心期刊)

研究项目

- [1] 国家自然科学基金项目（No.60602017）
- [2] 山东省自然科学基金项目（No.ZR2012FM016）



中國海洋大學
OCEAN UNIVERSITY OF CHINA

硕士学位论文

