and vertical windowed sampling on historical GPS data. Secondly, we use Spark to compute the probability distribution of average speed over each time window. Thirdly, we use Bayesian maximum-a-posteriori estimation to adjust the speed estimate of latest period of time. Experimental results demonstrate that the proposed method can be used for implementing efficient and accurate urban traffic prediction in real time.

Keywords　real time traffic prediction; GPS; Spark; Beidou navigation satellite system

1　引　言

随着城市化发展，城市人口急剧增长，交通拥堵现象愈来愈严重。造成该现象的最大原因是城市交通设施建设与人们日益增长的需求产生了矛盾。对此，智慧城市的概念得以提出，希望能用一些技术手段来分析城市交通路况[1]。目前比较先进的交通数据获取途径如全球定位系统（Global Position System，GPS）、北斗卫星导航系统（BeiDou Navigation Satellite System，BDS）等[2,3]，都是基于工作卫星定位计算的，能实时通过定位装置进行接收和转发到指定服务器，获取更客观全面、实用性很强且成本低，并且在此基础上的研究成果更具有说服力且适用于实际生活场景。目前大多数以出租车和公交车等为代表的典型城市出行交通工具都安装有实时 GPS 定位装置[4]，通过该装置能够实时接收卫星定位，并且将其他的一些附加信息（如车牌号、实时速度、时间等）包装成 GPS 记录，或者时空数据（包括客人上下车地点、时间、收费、速度等信息）发送到指定的服务器存储。随着信息采集设备的改进，浮动车辆的定位信息的获取技术也已比较成熟，数据获取也很容易。但随着车辆的增多，相应地，每一天的 GPS 数据都在急剧增长，导致数据量越来越大、数据来源较多等，再加上地图数据，进一步增加了交通大数据分析以及多源数据融合的难度。而现有的软件技术相对较滞后，同时多数基于磁盘的存储系统在处理大数据时存在大量 I/O 延时，并且实时处理技术不够。其中包含几个难点：首先，地图匹配是一个难点，其匹配的实时性和精确度很大程度上直接影响后续每个步骤的效果[5]。而根据目前的卫星定位信息，计算得到车辆的经纬度其实有一些误差，而地图的划分是准确的，因此地图匹配本身就会存在一些误差，最大限度减少地图匹配误差是一个难点。其次，对于位置信息的计算其实是图数据的计算，对于一块很大区域的数据，数据量大很容易影响计算的效率，因此设计好的计算框架和算法也是难点。

在大数据处理方面，目前国内外还处于不成熟阶段[6]。但大数据时代的到来，出现了专门针对大数据处理的计算框架，其中 Hadoop 平台的 MapReduce 框架，因为拥有较为完善的接口和分布式并行计算能力而非常适合离线批次处理[7, 8]。而目前备受关注的 Spark 分布式内存计算框架，凭借其优秀的分布式内存计算能力受到学术界的青睐，加上 Spark 自身高效便捷的编程语言和兼容性，被许多研究机构作为大数据研究的计算框架[9]。Lin 等[10]提供了一个结合 Hadoop 和 Spark 的云计算平台，用于进行批量日志分析：他们利用 Hadoop 提供分布式文件系统、Spark 进行内存计算、数据仓库等，该平台能提供批量日志分析，以及内存计算等优化性能，但是数据仓库的操作仍然是基于抽取-转换-装载模式的传统数据管理方式；Gu 和 Li[11]通过评估 Spark 和 Hadoop 在做迭代计算时分别在内存和时间上的消耗，比较得出 Hadoop 用于迭代计算时，速度较慢、时间消耗较大，但是内存消耗较小；而 Spark 的运

算速度比 Hadoop 快很多，但是内存消耗很大，如果内存不够大到能载入每次迭代的结果，那么结果将被存储到磁盘上，导致无法体现内存计算的优势。因此，在实际程序设计中，要注意系统内存与中间结果数据集的关系，要尽量让内存能完全存放中间结果，以完美体现 Spark 迭代计算的优势。

在实时交通路况分析方面，学术界获得一些较好的研究成果。例如，Herring[12]于 2010 年利用设备的 GPS 流数据，结合机器学习和交通理论，建立了实时路况分析模型，为其他主要交通道路的规划和建设提供了重要参考；葛晓锋等[13]于 2007 年提出了将 GPS 与地理信息系统(Geographic Information System，GIS)结合起来分析浙江省高速公路的实时路况，他们将 GPS 数据采样后存放在 Oracle 数据库中，最后将路况分析结果以 Web GIS 的形式展示给用户。通过分析不难发现，有的采用静态路况分析的方式，即将传感器设备安装在道路旁、红绿灯处，用于记录车流经过该路段的信息，这种方式因为只能记录到该路段一定范围的路况而存在很大的局限性；有的采用动态路况分析的方式，将设备安装在流动车辆上，利用 GPS 导航系统，实时采集车辆的状态等信息，然后再进一步进行离线分析处理，得到路况信息，计算耗时非常长。

本文针对交通大数据处理技术的不足，契合智慧城市的主题，提出了基于分布式内存计算和浮动出租车 GPS 数据的实时交通路况预测方法。首先，通过分布式并行地对大量历史数据样本进行水平时间窗口和垂直时间窗口切片抽样，每次只需将需要用于计算的数据载入，利用 Spark 分布式内存计算框架进行并行计算，得到历史样本在各个时间段内历史平均速度的概率分布；然后，基于高斯分布采用贝叶斯最大后验估计(Maximum A Posteriori，MAP)对新到的样本进行预测，得到路段在给定时刻拥有的最大概率的平均速度[14,15]，作为路段在该时刻的平均速度的预测值，旨在为城市交通的实时路况进行实时的预测，也为城市交通的改善措施提供理论支持。接下来本文将对所提出的方法进行详细描述，并对实验结果进行分析。

2 基于分布式内存计算的实时路况预测方法

本文提出的实时路况预测方法框架如图 1 所示，涵盖了 GPS 数据源接收、预处理、地图匹配、路况分析模型和实时路况预测等模块。其中实时接收解析模块和地图匹配模块使用的是现有技术，核心模块是多线程并发预处理模块和路况预测模型模块。以下是各个模块的描述。

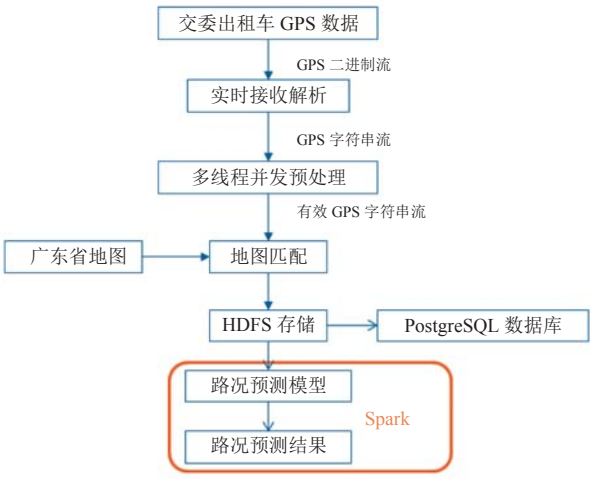

图 1 实时路况预测方法框架

Fig. 1 Framework of real time traffic prediction method

2.1 实时接收解析模块

大体上出租车都安装有 GPS 定位装置，每个 GPS 定位装置定时每隔 30 s 发送一条定位记录，称为 GPS 记录。每一条 GPS 包括车牌号、经度、纬度、汇报时刻、设备号、速度、方向、定位状态、车辆颜色等字段。实时接收解析模块的主要作用是接收深圳市交通运输委员会(以下简称“深圳市交委”)服务器转发的一条一条二

进制形式 GPS 记录并按照转发协议进行解析，对二进制 GPS 流的形式数据进行字段提取和转码，得到一条一条字符串形式 GPS 记录。流入后续的多线程并发预处理模块进行数据清洗。

2.2　多线程并发预处理模块

该模块将实时接收解析模块处理得到的字符串形式的 GPS 流，作为输入，对每一条字符串形式的 GPS 记录进行数据清洗，将字段不全或不合要求的数据做处理，舍弃或者通过简单的统计方法进行补全。本文是通过设置一个多线程缓冲池，利用生产者-消费者模型[16]，开启多个线程作为生产者接收字符串 GPS 流，进行预处理后放入到缓冲池，同时开启多个线程作为消费者从缓冲池中取出 GPS 记录，接入到地图匹配模块。当缓冲池中容量已满时，生产者将等待消费者从中取出数据获取空闲空间；当缓冲池为空时，消费者将等待生产者往缓冲池中放入数据。后续实验证明，当生产者和消费者的线程数量达到一定比例时，能够快速及时地处理接收模块传入的 GPS 记录，不会出现线程等待的情况。

2.3　地图匹配模块

本文用于地图匹配模块的地图是广东省电子地图，其中包含了深圳市 135 138 个路段的信息，包括路段编号、路段名、路段经度、路段纬度、路段宽度、路段长度和路段限速等字段。地图匹配模块的作用是将字符串 GPS 记录中的经纬度与广东省地图中对应路段的经纬度进行匹配，将路段编号以及路段的经纬度字段添加到 GPS 记录中，构成匹配后的 GPS 记录，之后的计算都基于匹配后的 GPS 记录。地图匹配后，GPS 记录将存储于 HDFS 文件中，同时会将用于建模的数据备份到 PostgreSQL 数据库中。

2.4　路况预测模型

该模块是本文的研究重点。本文的交通路况用给定时刻路段上车辆的平均速度来体现，当平均速度小于给定阈值时，视该路段为拥堵。需要明确的是，车辆在每个路段上的行驶速度是随机的，因此客观上不同时间段内的车辆的平均速度分布可以采用高斯分布计算。但是，同一路段在不同时间段内的平均速度会由于不同外界因素而发生各种变化，如下雨、大型活动、车祸等，而且外界因素的影响无时无刻都存在，因此需要通过对历史样本进行抽样来进行概率校验，即使用条件概率来得到最大后验概率。本文采用 MAP[15]来进行实时路况的预测。

2.4.1 最大后验概率估计

最大后验概率估计(MAP)是将估计量的先验分布和条件概率联合起来，选择使得联合概率取最大值的那个估计量作为预测值。那么对于每一个路段 r，根据贝叶斯全概率公式，都有：

$$p(\theta_r, T \mid X)=p(X \mid \theta_r, T)p(\theta_r, T) \tag{1}$$

其中，X 表示路段上车辆的抽样 GPS 记录样本，$X=[X_1, X_2, \cdots, X_n]$，每个样本 X_i 相互独立；θ_r 表示路段 r 的车辆平均速度；T 表示时间段；$p(\theta_r, T \mid X)$ 表示给定样本 X 的情况下，路段 r 在时间段 T 内的平均速度的概率，这是后验概率，是最终实时路况预测的指标，但直接计算需要提前知道待预测时间的样本，这显然计算速度达不到零时间，无法直接计算该指标进行预测；$p(X \mid \theta_r, T)$ 表示给定样本 X 在时间段 T 内路段 r 的速度区间上概率分布，即条件概率；$p(\theta_r, T)$ 表示每个速度区间在每个时间区间的概率分布，由客观统计规律决定，即先验概率分布，符合高斯分布[17]，因此用以下高斯公式进行计算：

$$p(\theta_r, T)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\theta_r)^2}{2\sigma^2}} \tag{2}$$

其中，X 是在时间段 T 内路段 r 上的 GPS 记录样本的速度；θ_r 是路段 r 在时间段 T 内的车辆的平均速度。

首先，通过对公式(1)的分析可知：$p(\theta_r, T \mid X)$ 是要估计的目标值，但是不能便捷地通过抽样统

计得到；$p(X|\theta_r,T)$ 可以通过大量的抽样进行统计很容易计算出来；$p(\theta_r,T)$ 是先验概率分布，用高斯分布能体现出一般性，因此可以直接通过计算得到概率值。然后，本文通过对历史样本按照水平时间窗口和垂直时间窗口进行切片，得到每个路段在时间段 T 内的每个时间窗口的平均速度，进而将多天历史数据的平均速度映射到速度区间，得到这个时间段 T 内的历史平均速度的概率分布，这样就得到了条件概率 $p(X|\theta_r,T)$，然后对上述全概率公式进行变形，得到目标概率为：

$$p(\theta_r,T|X)=\frac{p(X|\theta_r,T)p(\theta_r,T)}{p(X)} \tag{3}$$

将公式(3)中的 X 展开便得到：

$$p(\theta_r,T|X)=\frac{p(X_1|\theta_r,T)p(X_2|\theta_r,T)\cdots p(X_n|\theta_r,T)p(\theta_r,T)}{p(X)} \tag{4}$$

对于给定样本 X，$p(X)$是固定不变的，那么要求的路段 r 在时间段 T 内任意时刻的平均速度预测值根据公式(3)可以得出目标函数是最大后验估计：

$$\theta_{\mathrm{MAP}}=\arg\max_{\theta_r}\left\{p\left(X|\theta_r,T\right)p\left(\theta_r,T\right)\right\} \tag{5}$$

为了方便计算，将公式(5)进行取对数，单调性仍然一致，再将 X 展开式代入公式(5)，就得到：

$$\theta_{\mathrm{MAP}}=\arg\max_{\theta_r}\left\{\sum_{i=1}^{n}\log p\left(X_i|\theta_r,T\right)+\log p\left(\theta_r,T\right)\right\} \tag{6}$$

公式(6)的含义是：将使得等号右边部分取得最大值，即使得后验概率取得最大值的 θ_r 作为路段 r 在指定时间段内的某个时刻的平均速度预测值。

通过上述分析，模型的关键之处是对历史样本进行抽样计算出概率分布 $p(X|\theta_r,T)$，而先验分布采用高斯分布进行概率值计算，然后求解公式(6)，即可得到路段在某个时刻的平均速度预测值。

2.4.2 计算条件概率分布

对于某个时间段 T 内，车辆在路段 r 上的平均速度的概率分布，可以通过对历史样本进行抽样，然后将平均速度映射到速度区间，得到条件概率分布。具体计算过程如下：取出 n 天的 GPS 数据，对于每个路段，将每一天分成多个时间段 T，然后以 *step* 为步长，将每一个 T 划分为多个大小一致的时间窗口 t。接着求出路段 r 在每个时间窗口 t 内的所有车辆的平均速度，记为 θ_{ij}，表示时间段 T 内路段 r 第 i 天的第 j 个时间窗口的平均速度。然后将 n 天相同的时间段内，相同时间窗口的平均速度再求平均值记为 θ_j，表示水平方向的第 j 个时间窗口的历史平均速度。最后将时间段 T 内的多个历史平均速度映射到速度区间，得到路段 r 在时间段 T 内的历史平均速度的概率分布，即条件概率分布。计算过程如图 2 所示。

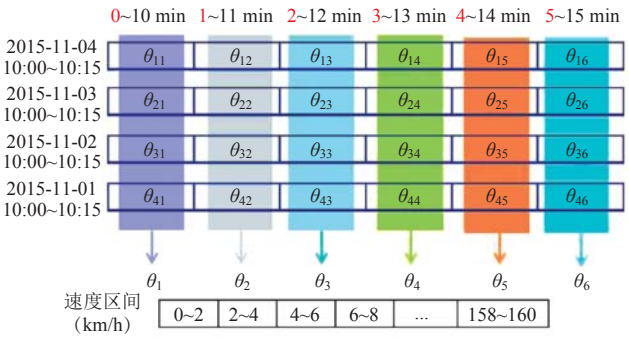

图 2 条件概率计算样例

Fig. 2 An example of Conditional probability calculation

如图 2，样例中选择的参数为：时间段 T =[10:00, 10:15]，时间窗口大小为 t=10 min，步长 *step*=1 min。速度区间大小为 2，从 0 到 160，不重叠。对于每个路段每个时间段都有该计算过程。首先是进行横向时间窗口的划分，得到该路段的所有 GPS 样本，求出速度平均值，然后再纵向求出同一时间窗口的历史平均速度，这样做其实是为了扩大样本容量，降低特殊因素

的影响。最后再将每个历史平均速度映射到速度区间，得到每个历史平均速度在每个速度窗口的概率，即得到了历史平均速度的概率分布，即得到了 $p(X|\theta_r,T)$。

3　实验和分析

3.1　实验数据

首先，选择连续三天时间段为 11 点到 13 点期间的深圳市全部出租车的 GPS 记录，大小为 3.6 G，以 10 分钟为时间窗口大小，1 分钟为步长，构造一共 110×3 个时间窗口。然后，求出每个路段在每个时间窗口的平均速度，将每个路段的历史平均速度映射到长度大小为 2 的速度区间，得到条件概率分布。最后，将得到的条件概率分布放入前文推出的最大后验概率公式(6)进行计算，找到使得后验概率最大的历史平均速度作为路段在目标时刻的平均速度预测值。

3.2　实验结果及分析

本文提出的研究方法中，解析模块和地图匹配模块使用的是已有的研究成果，其中所用的地图匹配算法是 Chawathe[18]提出的基于几何权重的地图匹配算法。本文的重点放在多线程并发预处理模块和路况预测模型模块，因此实验主要对这两个模块进行了效率评估。

3.2.1 多线程并发预处理模块性能评估

实验中，将解析得到的字符串 GPS 流接入该预处理模块，实验中通过设置不同的生产者线程和消费者线程比例进行效率对比，发现设置 10 个生产者线程和 15 个消费者进行数据传输，其吞吐量已能达到 3.462 M/s(54 945 条/秒)。而深圳市交委接收数据的最大速率不超过 1 M/s(3 000 条/秒)。由此可以看出，本文采用的多线程缓冲池已足够满足要求。

3.2.2 计算条件概率效率

整个计算过程都在 Spark 集群进行计算，为了证明 Spark 的高效性，本文增加了 PostgreSQL 数据库和 Spark 集群计算条件概率分布的效率对比试验，输入数据大小为 3.6 G，两者的计算效率对比如图 3 和图 4 所示。

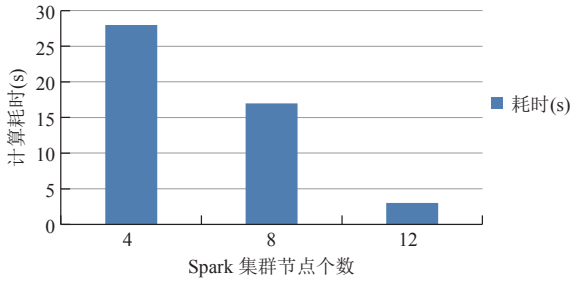

图 3　不同规模的 Spark 集群耗时

Fig. 3　Time consumption by different scale of Spark cluster

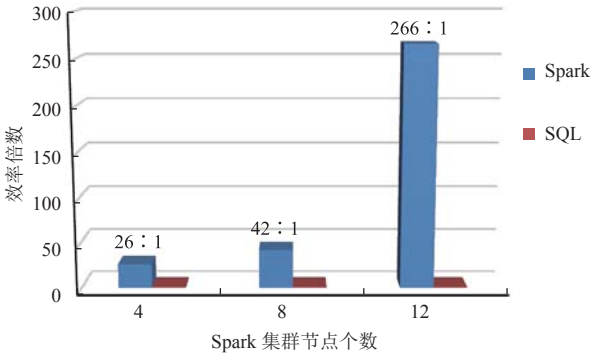

图 4　不同规模的 Spark 集群与 SQL 的效率对比

Fig. 4　Efficiency comparisons between different scale of Spark and SQL

实验过程中发现，编写等效算法的 SQL 查询，由单条 join 语句完成，并创建索引优化查询效果。查询计算条件概率耗时约 720 s，是 4 节点 Spark 集群耗时的 26 倍。因此可以总结得出 Spark 分布式内存计算框架下的计算时间远远比用等效的 SQL 查询语言少，同时该实验证明了选择 Spark 作为本文提出的实时路况预测方法的计算引擎是非常正确的，进一步突显出该方法的高效性。

3.2.3 利用先验概率和条件概率分布进行路况预测的结果及分析

实验中首先选择了 2015 年 10 月底连续三天

的数据进行建模，计算出条件概率分布后，利用公式(6)计算出最大后验概率下平均速度作为预测值。然后对深圳市连续 20 天各路段在时间段 11:00:00～11:05:00 的平均速度进行预测，将其与根据真实的历史数据统计得到的各路段在对应时间段的真实平均速度作比较。由于速度的连续性，此处利用速度阈值将平均速度转换为实时的路况进行量化分析。当路段的平均速度小于给定的速度阈值时，表示该路段处于拥堵状态，反之表示该路段处于通畅状态。根据常识可知，正常情况下，人的步行速度约为 5 km/h，城市出租车的行驶速度约为 30 km/h，因此实现中设置了多个具有代表性的速度阈值来评估路况预测准确率，最后选择了其中三个具有代表性的速度阈值，得到其对应的路况预测准确率曲线如图 5 所示。

图 5 中，横轴表示预测日期，从 11 月 1 日到 11 月 20 日，纵轴表示目标时间段的路况预测准确率。每条曲线使用相同的条件概率分布，速度区间大小为 1 km/h。从图 5 所示的曲线可以看出，不同的速度阈值对应的每天的路况预测准确率不同，但整体的波动趋势一致。相比之下，当速度阈值较小时，如图 5 的 5 km/h(约为人的正常步行速度)，其对应的路况预测准确率在 73% 左右波动，表现最好；而当速度阈值较大时，如图 5 的 12 km/h，其对应的路况预测准确率的整体趋势下降，仅在 63% 左右波动，表现最差。但根据常识，路况预测的速度阈值不会太大，因此本文的实验对比中采用的速度阈值足以证明本文提出的实时路况预测算法的稳定性和高效性。在实验过程中，发现利用 4 节点的 Spark 集群进行实时路况预测的耗时仅为 2 s 左右，该时间对于所要预测的时间段长度可以忽略不计，具有很强的实时性，且最佳的路况预测准确率为 73% 左右。目前最新的相关研究中与本文类似的方案很少，其中算法思路最接近的最新研究是 Pan 等[19]于 2016 年提出的基于真实交通数据并结合历史行为规律的交通预测方法，考虑高峰时段时，路况预测准确率为 67%～78%，但没有体现出很强的实时性。

由于本文实验中所用的地图匹配算法是 Chawathe[18]提出的基于几何权重的地图匹配算法，匹配率在 30% 左右，不算很高，并且本次实验中只使用了 3 天的数据进行建模，导致用于建模的 GPS 数据量较小，从而导致遗漏了有些路段的数据，即条件概率的数据中不包括某些路段，但是该路段却的确出现在要预测路段的集合中，导致该部分路段的预测无效。因此本文提出的实时路况预测方法在保证高实时性的前提下，其路况预测准确率已超出了预期。通过仔细分

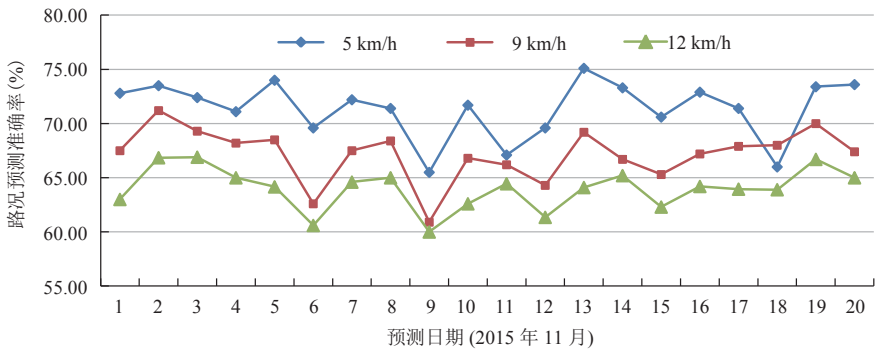

图 5 基于贝叶斯 MAP 的实时路况预测准确率曲线

Fig. 5 Real time traffic prediction accuracy curve based on Bayesian MAP

析，本文提出的实时路况预测模型有以下可改进部分：

(1) 地图匹配：如果能够进一步提升匹配速度和匹配率，那么会得到更多的 GPS 样本，将更加有助于本文所提出研究方案的结果提升。

(2) 使用更多历史样本：如果能用更多天的，如一个月或一个季度，甚至一年的 GPS 数据进行建模计算条件概率分布，那么就不会出现上述路段预测失效的情况，同时必然会使条件概率更具一般统计规律，因此得到的预测准确率肯定会更高。

(3) 优化模型参数：由上述实验结果发现，速度区间大小的设定对路况的预测准确率有很大的影响。实验表明，速度区间越小，则划分越细，那么计算就会越准确。

4　模型讨论

本文提出的实时路况预测方法，能够在有限数据样本的条件下，快速高效地对实时路况进行预测，实时性高，并且最佳的路况预测准确率达到 73% 左右，而本文提出的方法涉及多个参数，主要集中在计算条件概率的过程中。其中，几个重要参数包括时间段 T，时间窗口 t 的大小，步长 $step$，速度区间大小，所用 GPS 历史样本的天数 n 等。本文在相同的参数条件下对不同的速度阈值进行路况预测对比试验，发现当速度阈值为 5 km/h 时，路况预测准确率为 73% 左右，相比于速度阈值为 12 km/h 时的路况预测准确率提升了 10% 左右，由此说明了本文提出的方法的路况预测准确率受速度阈值变化的影响较大。此外，时间段 T 的划分应尽量根据交通的历史规律进行划分，交通高峰时间段的 T 不宜太大，比如早上 7 点至 9 点为上班高峰期，应划分为一个 T，而夜间交通低峰的 T 跨度可以稍微大些，如将 2 点至 5 点划分为一个 T。其他参数不再一一赘述。

5　结论和工作展望

根据实验分析过程可知，在进行路况预测时，需要考虑历史规律和外在客观因素，摒弃传统的通过大量物理传感器的使用，采用 GPS 历史记录来建立模型，将复杂问题简单化。本文提出的基于 Spark 与 GPS 数据的实时路况分析方法，利用贝叶斯概率 MAP，将预测目标值转换为先验概率和条件概率的联合概率，简化了计算难度。在目前有限的样本数量和效果不是最优的地图匹配算法的条件下仍有很好的预测效果，并且有很好的实时性。当时间窗口 t 大小为 10 min、步长 $step$ 为 1 min、速度区间大小为 1、速度阈值为 5 km/h 时，路况预测准确率达到 73% 左右。同时，本文实验所用的条件概率分布的计算在后台进行计算即可，所有涉及计算的模块都在高效的 Spark 分布式内存计算框架中进行，效率非常高。但由于时间等客观因素，本文没有进行大幅度扩展，目前所有的代码运行都在命令行进行，并且只使用了深圳市连续 20 天的历史交通数据。后期将进一步扩展，包括扩大样本量和添加用户操作界面，可视化操作，构建完善的一整套路况预测系统。

参 考 文 献

[1] Su K, Li J, Fu HB. Smart city and the applications [C] // 2011 International Conference on Electronics, Communications and Control (ICECC), 2011: 1028-1031.

[2] 周忠谟, 测绘学, 杰军, 等. GPS 卫星测量原理与应用 [M]. 北京: 测绘出版社, 1992.

[3] 杨元喜. 北斗卫星导航系统的进展、贡献与挑战 [J]. 测绘学报, 2010, 39(1): 1-6.

[4] Binjammaz TA, Al-Bayatti AH, Alhargan A. GPS integrity monitoring for an intelligent transport

system [C] // 2013 10th Workshop on Positioning Navigation and Communication (WPNC), 2013: 1-6.

[5] Li Y, Zhang K, Li T. The research on real-time map-matching algorithm [C] // 2012 International Conference on Industrial Control and Electronics Engineering (ICICEE), 2012: 1973-1976.

[6] Wu XD, Zhu XQ, Wu GQ, et al. Data mining with big data [J]. IEEE Transactions on Knowledge and Data Engineering, 2014, 26(1): 97-107.

[7] Dwivedi K, Dubey SK. Analytical review on Hadoop distributed file system [C] // 2014 5th International Conference-The Next Generation Information Technology Summit (Confluence), 2014: 174-181.

[8] Dean J, Ghemawat S. MapReduce: simplified data processing on large clusters [J]. Communications of the ACM, 2008, 51(1): 107-113.

[9] Zaharia M, Chowdhury M, Franklin MJ, et al. Spark: cluster computing with working sets [C] // Proceedings of the 2nd Usenix Conference on Hot Topics in Cloud Computing, 2010: 1765-1773.

[10] Lin XQ, Wang P, Wu B. Log analysis in cloud computing environment with Hadoop and Spark [C] // 2013 5th IEEE International Conference on Broadband Network & Multimedia Technology (IC-BNMT), 2013: 273-276.

[11] Gu L, Li H. Memory or time: performance evaluation for iterative operation on hadoop and spark [C] // 2013 IEEE 10th International Conference on High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing (HPCC_EUC), 2013: 721-727.

[12] Herring RJ. Real-time traffic modeling and estimation with streaming probe data using machine learning [D]. Berkeley: University of California, 2010.

[13] 葛晓锋, 曹斌. GPS 与 GIS 结合的高速公路实时路况分析系统设计及实现 [J]. 电子技术, 2007, 7(8): 155-157.

[14] Cheeseman P. A method of computing generalized Bayesian probability values for expert systems [C] // Proceedings of the 8th International Joint Conference on Artificial Intelligence, 1983: 198-202.

[15] Gauvain JL, Lee CH. Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains [J]. IEEE Transactions on Speech and Audio Processing, 1994, 2(2): 291-298.

[16] Surhone LM, Timpledon MT, Marseken SF, et al. Producer-Consumer Problem [M]. Betascript Publishing, 2010.

[17] Stein CM. Estimation of the mean of a multivariate normal distribution [J]. The annals of Statistics, 1981, 9(6): 1135-1151.

[18] Chawathe SS. Segment-based map matching [C] // 2007 IEEE Intelligent Vehicles Symposium, 2007: 1190-1197.

[19] Pan B, Demiryurek U, Shahabi C. Traffic prediction using real-world transportation data: US9286793 [P]. 2013-10-22.