**基于长短期记忆网络的行车安全综合评价模型系统**

摘 要：随着物联网技术的快速发展和应用拓展，车联网也逐渐步入城市交通网络。车联网获取车辆的属性信息，实时监控驾驶员的行为信息以及车辆的运行状态，是防范交通事故发生的有效途径之一。首先，基于K-means算法进行频道时段聚类分析，构建字段比对库，利用PMUC关键字匹配算法构造出了多时段的频道三级标签和节目三级目录；其次，通过分析用户行为内部关联，划分用户层次并进行用户分组，进而基于协同过滤算法实现打包推荐；最后，通过实例分析，刻画用户画像，挖掘用户深层信息，做出用户行为预测与潜在价值分析。

关键词：精准推荐，用户画像，聚类分析，关联分析，关键字匹配

**Marketing recommendations for TV programs**

Abstract:

Key words: Accurate recommendation， User portrait， Clustering analysis， Correlation analysis， Keyword matching

# 引言

广播电视运营的根本动力来源于对利益的追求，通过各种营销手段吸引用户实现从产品到利润这一过程。然而随着市场经济的不断发展，传统的卖方市场早已经过渡到买方市场，而且随着用户对于产品及服务的个性化需求越来越普遍，整个市场营销环境早已经发生了巨大的变化，从而导致了运营商无法再利用传统的营销方法和手段去满足消费者的个性化需求，进而获得经济效益[1]。如何选择营销策略提高效率、提升效益已经成为了解决的瓶颈问题。

数据在市场营销中的应用早已有之，而在大数据时代，数据在市场营销中的重要性更加突出[2]。运营商可以通过对用户数据的收集和整合，进一步挖掘和筛选具有商用价值的信息，利用这些信息进行市场细分和精准的市场定位，然后面向目标用户采取针对性的营销策略，为其提供差异化的产品和服务，最终获得更多的忠诚用户。

本文一方面从研究的对象及目的出发，通过阅读与大数据挖掘、精准营销等相关的国内外文献，对其进行梳理与总结，掌握本研究领域的一些基础知识，为此次论文写作打下了一定的理论基础。另一方面，通过利用大数据挖掘与精准营销之间的切入点，结合大数据挖掘方法，得出精准营销策略，最后引用一个具体的实例来验证对于基于大数据挖掘的精准营销策略的具体应用[3]。

# 基本思路

问题一的分析

题目要求根据附件1数据，分析用户收视偏好，给出附件2中产品的营销方案。结合数据分析，分三个步骤解决该问题。

（1）将数据分为频道类和节目类数据，以时间、内容纬度描述频道特性，发掘节目特点，通过用户收看频道和节目的记录判断用户收视偏好。拟采用基于K-Means算法的时间聚类划分频道时段，利用网络爬虫获取央广网频道播放内容的参照库，以“频道-时段-内容类别”的形式描述频道特征；通过网络爬虫获取豆瓣电影信息库，构建三级目录体系，利用基于PMUC多模匹配算法确定节目特征。

（2）设置偏好程度评分函数，对用户收视偏好程度进行量化评分。

（3）根据用户收视偏好，结合节目内容和三级目录，利用多模匹配算法从附件2中选取合适的节目并推荐给对应用户。

问题二的分析

题目要求根据附件1~附件3的数据，对相似偏好的用户进行分类，对产品进行分类打包，给出营销推荐方案。参考某电视公司打包定价策略[4]，分三个步骤解决该问题。

（1）根据用户收视行为，结合附件3用户套餐情况，综合收视偏好、收视特点和消费能力三个标准，使用协同过滤算法，对用户进行分组，并设置用户标签。

（2）根据问题一发掘的节目特点和用户分组情况，对节目进行组合打包，并设置节目包标签。

（3）通过用户标签和节目包标签的匹配，将打包节目准确地推荐给用户分组本文解决问题的具体思路，如图2-1所示。



图2-1

# 3.数据预处理

## 数据理解

* + 1. 行车轨迹数据

|  |  |  |
| --- | --- | --- |
| 数据字段 | 指标说明 | 指标解释及统计结果 |
| direction\_angle | 方位角 | 行车方向角指从定位点的正北方向 起，以顺时针方向至行驶方向间的水平夹角。范围从0-360。 |
| Lng、Lat | 经纬度坐标 | 经纬度基本分布在东经110-120左右，存在缺失数据，即轨迹中存在缺失点，需要进行补点操作。 |
| location\_time | 时间戳类型 | 以s为单位，理论上每秒统计一次结果，实际上存在漏打点的情况。 |
| gps\_speed | 数值型 | 速度全程基本呈现出正态分布，部分车辆的速度的分布如图x所示。 |
| mileage | 数值型 | GPS里程统计，存在一定的误差。 |

|  |  |
| --- | --- |
|  |  |

1. 图x 速度全程的统计分布状态

行车轨迹数据包括。。。字段。

* + 1. 自然气象数据

|  |  |  |
| --- | --- | --- |
| 数据字段 | 指标说明 | 指标解释及统计结果 |
| province、prefecture\_city、county | 省/市/县 | 这3个字段进行整合，形成地址合并。确定该地区的经纬度范围，与行车轨迹数据有效结合。 |
| condition | 天气状况 | 原数据给的是几类天气状况，为了后面更好的对数据进行计算，将其转换为数值型数据。对于多云（1）转雷阵雨（2）这种取两者均值（1.5）。 |
| precipitation | 降水量 | 降水量的前提是这一天天气状况是下雨，然后再进行判断天气的恶劣情况。 |
| record\_data | 时间戳类型 | 以天为单位， |

## 数据清洗

硬件采集的也大多是带UTC时间的时间序列，而且采样间隔并不一定均匀，往往还有噪声，对于这样的时间序列，我们第一要做的是清洗。数据清洗(Data cleaning)主要是指过滤该驾驶行为数据的缺失数据,平滑噪声数据,确定或去掉孤立点,解决不一致问题。本文中的驾驶行为数据清洗包括去除孤立点、填补缺失数据两个主要方面。数据清洗的流程如下。

* + 1. 去除漂移点

去除漂移点。由于GPS系统在采集数据时，信号接收可能会受到强弱电磁干预，接受效果不好；GPS天线安装位置不合理或受遮挡；GPS模块工作时数据处理有误差，以上3个因素都会引起数据点漂移现象，引起GPS漂移情况发生将导致车辆定位的位置不准确，造成较大的偏差。

针对这样的情况，考虑到漂移点通常与上一时刻的点的距离一般都是超出正常范围的，在统计了10万条行驶数据后，挖掘出运输车辆正常行驶状态下1s中行驶距离不会超过25m，除此之后，考虑到漂移点的出现可能是GPS信号不好等原因，在这个时间段可能也会存在缺失项的情况，因此将该正常距离范围的阈值设为l=50m。从而筛选出漂移点。

代码及理解

|  |
| --- |
| def remove\_drift\_point(filepath,point\_distance):  test\_vehicle = pd.read\_csv(filepath)  lng\_lat = test\_vehicle[['lng','lat']].values  error\_index = []  for name,index,distance in point\_distance:  if distance > 50:  lng\_lat[index] = np.nan  error\_index.append(index)  test\_vehicle[['lng','lat']] = lng\_lat  return test\_vehicle |

该函数中传入的参数为point\_distance，该参数为三维的数组，包含了点与点之间的距离。对于点距离大于距离阈值l的话，就将其抛弃，设为空值，这些都是在dataframe的基础上进行操作的。

* + 1. 填补缺失值

驾驶行为数据原始数据量较大,不同驾驶员在不同时间点发生的上传时间、车速、预警信息不同,即使同一驾驶员在不同时间内发生的驾驶车速、预警等信息也不同,因此空缺字段较多。为了使得里程计算的更多准确，需要对缺失的数据项进行填充。

1. 统计有多少缺失值（代码，统计450辆车50辆，缺失项结果2行10列）

空缺值根据提供的数据，进行统计分析可得要提交上去的10辆车的缺失项数量如下图所示。因此为了避免在计算行车里程中出现较大的误差，现对于对于空缺字段的处理主要分为三种情况，一种的经纬度坐标，一种是数值型(除了经纬度坐标字段)和布尔型的字段类型。（这个等到所有车辆kalman之后再进行统计）

1. 补时间项（代码，代码理解）

|  |
| --- |
| def kalman\_em\_smoothing(lng\_lat,masked\_index):  lng\_lat\_missing = np.ma.array(lng\_lat, mask=np.zeros(lng\_lat.shape))  for i in masked\_index:  lng\_lat\_missing[i] = np.ma.masked  observation\_matrix = np.array([lng\_lat[0],lng\_lat[-1]])  initial\_state\_mean = lng\_lat[0]  n\_timesteps = 50  kf = KalmanFilter(initial\_state\_mean=initial\_state\_mean,n\_dim\_obs=2)  correct\_point = kf.em(lng\_lat\_missing).smooth(lng\_lat\_missing)[0]  return correct\_point |

3. 补数值型、布尔型代码及理解

对于数值型的字段，利用该空缺项前后时刻的状态的均值进行填充；对于布尔型的字段，利用前一时刻该字段的状态值进行填充。例如、对于驾驶员的gps\_speed字段的空缺值使用其上一时刻的状态与下一时刻的速度均值进行填充，对于acc\_state等布尔型字段均采用上一时刻的数据值进行填充该空缺项数据。

对于经纬度坐标，利用改进型的卡尔曼算法进行填充和光滑数据。

* + 1. 平滑轨迹数据

1．利用期望最大算法(EM Algorithm)来改进卡尔曼滤波算法

针对连续性时间序列的经纬度坐标数据，使用卡尔曼滤波器进行平滑。卡尔曼滤波（Kalman filter）是一种高效率的[递归滤波器](https://zh.wikipedia.org/wiki/%E9%80%92%E5%BD%92%E6%BB%A4%E6%B3%A2%E5%99%A8" \o "递归滤波器)（[自回归](https://zh.wikipedia.org/wiki/%E8%87%AA%E8%BF%B4%E6%AD%B8%E6%A8%A1%E5%9E%8B)滤波器），它能够从一系列的不完全及包含[噪声](https://zh.wikipedia.org/wiki/%E9%9B%9C%E8%A8%8A_(%E9%80%9A%E8%A8%8A%E5%AD%B8)" \o "噪声 (通讯学))的[测量](https://zh.wikipedia.org/wiki/%E6%B5%8B%E9%87%8F" \o "测量)中，估计[动态系统](https://zh.wikipedia.org/wiki/%E5%8A%A8%E6%80%81%E7%B3%BB%E7%BB%9F" \o "动态系统)的状态。它常用于跟踪连续状态空间中的单个对象的无监督算法。 给定一系列噪声测量，卡尔曼滤波器能够恢复被跟踪的下属对象的“真实状态”。卡尔曼滤波会根据各测量量在不同时间下的值，考虑各时间下的[联合分布](https://zh.wikipedia.org/wiki/%E8%81%94%E5%90%88%E5%88%86%E5%B8%83" \o "联合分布)，再产生对未知变数的估计，因此会比只以单一测量量为基础的估计方式要准。卡尔曼滤波器的常见用途包括雷达和声纳跟踪以及机器人中的状态估计。因此，对于轨迹路线这样的一维时间序列，可以利用卡尔曼滤波器对空缺的数据项经纬度坐标进行预测，从而获得精确完整的行车轨迹坐标数据。

但是传统上卡尔曼滤波器需要手动指定的模型参数，这就给卡尔曼滤波的实际应用带来很大的挑战，但是在本实验中，无需任何标记的训练数据，也可以通过将EM算法应用于Kalman滤波中，实现对无标签的数据进行学习训练后获得较为准确的模型参数值。从而对缺失的轨迹经纬度数据进行填充，并对填充完成后的轨迹数据进行平滑。

2．代码实现及理解

|  |
| --- |
| def kalman\_em\_smoothing(lng\_lat,masked\_index):  lng\_lat\_missing = np.ma.array(lng\_lat, mask=np.zeros(lng\_lat.shape))  for i in masked\_index:  lng\_lat\_missing[i] = np.ma.masked  observation\_matrix = np.array([lng\_lat[0],lng\_lat[-1]])  initial\_state\_mean = lng\_lat[0]  n\_timesteps = 50  kf = KalmanFilter(initial\_state\_mean=initial\_state\_mean,n\_dim\_obs=2)  correct\_point = kf.em(lng\_lat\_missing).smooth(lng\_lat\_missing)[0]  return correct\_point |

该函数即为改进型卡尔曼滤波的实现。其中传入的参数lng\_lat为经纬度坐标组成的二维矩阵，masked\_index为缺失项的索引值，应用KalmanFilter Em算法，可以无需手动指定初始参数，Em算法可以通过多次迭代(这里设置的迭代次数为10次)寻求参数的最优解。函数返回经过平滑后的经纬度坐标点correct\_point(二维矩阵)。

1. 显示平滑轨迹图，缺失项补齐的前后对比图

|  |  |
| --- | --- |
|  |  |

图 清洗修正前的轨迹路线 图 清洗修正后的轨迹路线

## 3.3 数据压缩

在对数据清洗结束之后，还需要对数据进行压缩。一辆运输车辆每次运输大约能产生10万左右的GPS数据点，这样的一辆车的数据计算量也可以接受，但是对于几百辆这样的车辆而言，数据规模将达到千万级别。这个时候我们就需要考虑将轨迹数据进行压缩，从而减少时间和空间的计算复杂度，本文数据压缩基于道格拉斯-普客(Douglas-Peucker)抽稀算法。该算法的思想是设置一个距离阈值，当点到线的距离小于该阈值，那么点视为可以删除的点。最后剩下的点即为最终结果。

道格拉斯-普客(Douglas-Peucker)算法过程如下:

1、连接曲线首尾两点A、B；

2、依次计算曲线上所有点到A、B两点所在曲线的距离；

3、计算最大距离D，如果D小于阈值threshold,则去掉曲线上出A、B外的所有点；如果D大于阈值threshold,则把曲线以最大距离分割成两段；

4、对所有曲线分段重复1-3步骤，知道所有D均小于阈值。即完成抽稀。

这种算法的抽稀精度与阈值有很大关系，阈值越大，简化程度越大，点减少的越多；反之简化程度越低，点保留的越多，形状也越趋于原曲线。为了保证求出来的行车里程最为精确，在综合考虑计算复杂度与计算精度两方面的要求后，将阈值设为1m，这样就可以很好的将速度为0、经纬度坐标基本上没有变化的数据项进行了去除。

数据压缩的好处速度，以一个具体的车为例，前后对比时间，和内存大小。

# 不良驾驶行为挖掘

4.1轨迹特征提取

4.1.1里程

地球是一个近乎标准的椭球体，它的赤道半径为6378.140千米，极半径为 6356.755千米，平均半径6371.004千米。如果我们假设地球是一个完美的球体，那么它的半径就是地球的平均半径，记为R。如果以0度经线为基准，那么根据地球表面任意两点的经纬度就可以计算出这两点间的地表距离（这里忽略地球表面地形对计算带来的误差，仅仅是理论上的估算值）。设第一点A的经纬度为(LonA， LatA)，第二点B的经纬度为(LonB， LatB)，按照0度经线的基准，东经取经度的正值(Longitude)，西经取经度负值(-Longitude)，北纬取90-纬度值(90- Latitude)，南纬取90+纬度值(90+Latitude)，则经过上述处理过后的两点被计为(MLonA， MLatA)和(MLonB， MLatB)。那么根据三角推导，可以得到计算两点距离的如下公式：





其中，R和S的单位为km（采用6371.004 km作为半径），如果要使用其他单位，比如英里，需要做单位换算，1 km=0.621371192 mile。

如果仅对经度作正负的处理，而不对纬度作90-Latitude（假设都是北半球，南半球只有澳洲具有应用意义）的处理，公式为：





然后，将每秒两坐标点的距离求和即求出行车里程，公式如下：



，。

4.1.2平均速度

轨迹的平均速度为轨迹包含每段距离的长度总和与起始时间和结束时间之差的比值。公式为：



4.1.3加速度

轨迹平均加速度为当前点速度与之前点速度的差的大小求和与起始时间和结束时间之差取平均。公式为:





4.1.4平均方向变化率

以正北方向为基准，每点的速度方向与正北方向的夹角为轨迹当前坐标当前的方位角，每一点速度方向的变化的平均值即轨迹段平均方向变化率。公式为:



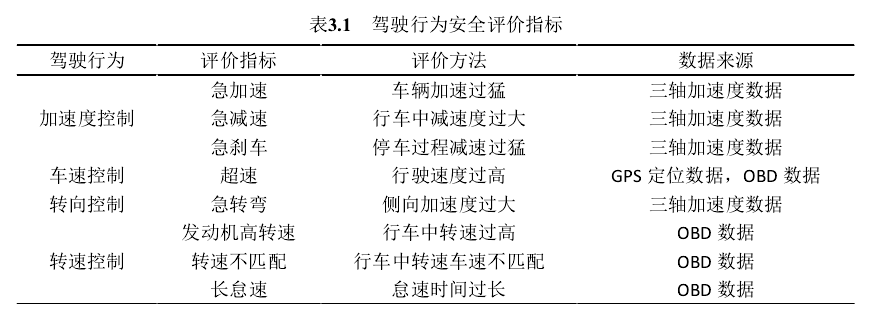
需要图

我们将定义的以上一般属性与速度排列熵属性结合，实验结果显示，排列熵属性在不同路况下能够更加清晰地表达出轨迹的特征，在不同路况下排列熵的属性变化不大，能够作为一个有效的特征值进行使用。排列熵是一种可以反应出一维时间序列复杂度的熵的参数，而实验的轨迹数据类型完全符合一维时间序列的特点。把一个长时间序列划分为多个长度小于当前长度的时间序列，轨迹数据为多个时间的坐标点组成，可以进行小段序列的分解。这些序列中可以有重合部分也可以不重合，然后计算每个子序列的排列熵。如果计算一个准确的时间序列，序列的长度不能够太小，否则不能够完全反应出该序列的特征，长度也不应该过大，否则会增加计算的时间复杂度，降低计算效率。

4.2不良驾驶行为识别

4.2.1 不良驾驶行为评价指标

本文研究的数据来自于某运输企业，因此对驾驶行为评估指标的研究只考虑运输车辆。通过分析，本文提出了影响营运车辆驾驶员异常驾驶行为的超速、急加速、急减速、急刹车、急转弯 5 个驾驶行为安全评价指标和超速、急加速、急减速、急刹车、高转速、转速不匹配，长怠速 7 个驾驶行为节能评价指标，为后面驾驶倾向性识别和驾驶行为评分提供数据来源，每个指标对应不同的评价方法与对应的数据来源见表3.1。



* + 1. 不良驾驶行为识别方法

**（1）疲劳驾驶**

根据《道路交通安全法》，驾驶疲劳是指驾驶人在长时间连续行车后，产生生理机能和心理机能的失调，而在客观上出现驾驶技能下降的现象。驾驶人睡眠质量差或不足，长时间驾驶车辆，容易出现疲劳。驾驶疲劳会影响到驾驶人的注意、感觉、知觉、思维、判断、意志、决定和运动等诸方面。据统计，20%的驾驶事故与疲劳驾驶有关。根据《道路交通安全法实施条例》第六十二条规定，不得连续驾驶机动车超过4小时未停车休息或者停车休息时间少于20分钟。因此，疲劳驾驶主要与累计驾驶时长和休息时长有关。

对于一辆车，从发动到停止为一次驾驶行为，统计出该车驾驶行为总数n（0,1，…），计算每次驾驶行为的累计时长，停车休息时间为，设计算法如下：

1. 观察acc\_state状态判断车辆启动或停止，当acc\_state=1时，i=1，

**（2）急加速**

通过 OBD 设备终端中的三轴加速度计获得的数据，对驾驶过程中经常出现的急加速行为提出一种识别算法，经过采集后由车载终端的控制处理单元将数据传回监控中心进行数据预处理，对倾斜时的重力加速度进行 XY 轴的分量剔除，从而获取车辆准确的加速度值，然后远程监控中心对预处理后的数据进行算法分析，看车辆加速度阈值是否满足急加速判断条件，由于急加速判断只涉及前进方向 Y  轴的输出值，因此对本节只对三轴加速度数据中 ay做处理。

在专利文献[22]中，提出了一种使用加速度传感器远程监控车辆的行驶状态的方法适合用来对文中采集的数据进行处理。通过相关资料与实验数据分析，车辆普通刹车时 Y 轴加速度大小一般维持在 1m/s2到 2m/s2，Y 轴加速度在 2m/s2到 4m/s2之间已经算是快速刹车了，而当刹车加速度超过 4m/s2时，乘客不适感强烈。因此，本文引入急加速行为界定阈值 A，按照行业经验预取

A=3m /s2

为了克服车辆运动中数据噪点，避免由于外界比如道路不平整，减速带等路障而造成错误判断，在急加速行为的判别条件中引入事件持续时间，本文根据数据采样频率取持续时间 TL＞1s，即在算法识别中连续判断两条数据均大于阈值才能识别为急加速，这样做能提高识别准确率，减少错误识别。

当 ay超过阈值 A 为“急加速行为"，为了区分单一急加速急连续急加速行为，中间没有间断的定义为“单一急加速行为”，中间间断时间短，由多个单一急加速行为合并而来的急加速行为定义为“连续急加速行为”。

为判断是否有急加速行为发生，引入急加速行为标记markAcc：



为判断是否发生连续急加速行为，引入连续急加速行为标记 markCon：



引入连续急加速行为合并时间阈值△T 来判断单一急加速是否可合成连续急加速，取0s<=T<=3s

为计算单一急加速的次数，引入参数 n，即在单一急加速中 n 的值为 1。在最后进行急加速次数累计时，连续急加速次数按一次急加速行为计算。

急加速行为判断算法的流程图如图 3.2 所示，算法主流程步骤说明如下：

1)获得采样数据，初始化参数；

2)设 OBD 采集的数据序号为i，初始值为1；

3)判断第 i 与第 i+1 条数据的加速度值 ay是否大于阈值 A。若是，则为单一或者连续急加速行为，记录急加速开始时间 t1，i 自加一；若否，则执行步骤 8；

4)判断 markAcc 是否等于 0，若否，则急加速行为继续，保留开始时间，记录 i+1条数据的采时间为结束时间，并记录持续时间，执行步骤 11；若 markAcc 等于 0，则执行步骤 5；

5)判断是否为该时间段内最后一条数据，并执行步骤 6；

6)判断是否可以查询到数据，若否，则为单一急加速行为的开始，记录第 i 条数据采集时间为开始时间，第 i+1 条数据的采集时间为结束时间，并记录单一急加速事件持续时间，置 markAcc=1，单一急加速行为次数自增 1；若是，则执行步骤 7；；

7)判断急加速行为最后一条数据的结束时间与第 i 条数据定位时间是否小于△T，若是，则为连续急加速行为，记录 i+1 条数据的采集时间为结束时间，并计算持续时间，令markAcc=1，markCon=1，单一急加速行为次数自增 1，并执行步骤 8；若否，返回步骤 6；

8)判断是否为行程最后一条数据，若是则结束；若否，i 自增 1，返回步骤 3 继续判断。

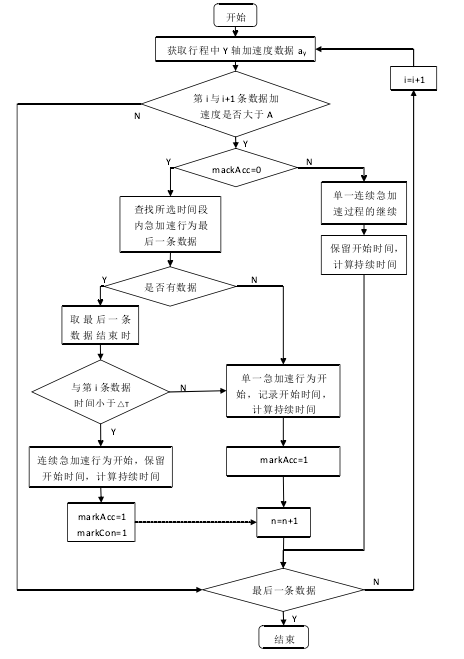


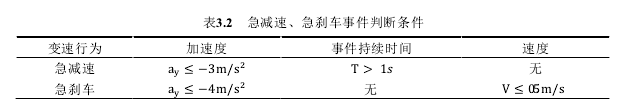
图3.2急加速行为识别算法流程图

**（3）急减速和急刹车**

专利文献[46]提出一种使用加速度传感器，通过测得的加速度大小来从而得到刹车幅度，利用尾灯变化频率来表征不同的刹车幅度。该文献提出，当车辆的加速度值小于-4m/s2时，车辆处于急刹车状态，此时的刹车灯以一定的频率闪烁。

因此，本文设定急减速的加速度阈值为-3m/s2，若减速度值达到设定的刹车减速度阈值时判断是否为急刹车事件，本文急刹车分析中加速度阈值设定为-4m/s2，当且仅当一定时间内车辆停止运动时为急刹车事件，为区分急减速与急刹车行为事件，两者判断条件如表 3.3 所示。

表3.2急减速、急刹车事件判断条件



在进行急减速判断时，同样分为单一急减速行为和连续急减速行为，当两个单一急减速行为间隔小于一定时间阈值时，合并为一个完整的急减速，由于其算法流程和急加速事件判断相同，本文对急减速行为事件算法不做详细说明。

为区分急减速事件与急刹车事件，在急减速行为事件基础上，根据表 3.3 判断条件进行区分，具体流程图如图 3.3 所示，相关说明如下：

1)获取急减速事件数据，对参数进行初始化；

2)判断第 i 条数据的 ay是否小于急刹车判断阈值-4m/s2，若是进入下一个步骤判断，

若否 i 自增 1；

3)判断第 i 条数据的仪表盘时速 v 是否小于 0.5m/s，若是则急刹车事件加一，累计的急减速事件减一；若否则返回步骤 2，i 自增 1；

4)是否为最后一条急减速数据，若是则结束判断；若否则返回步骤 1，判断下个事件。

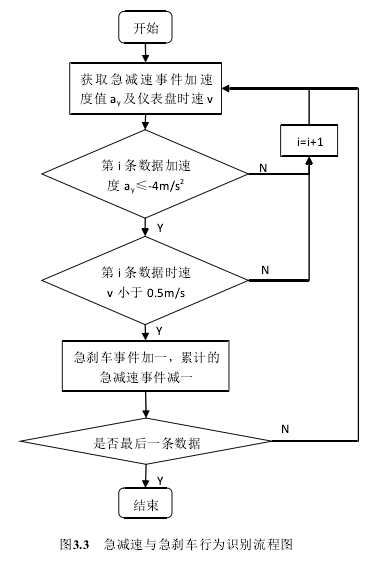


图3.3急减速与急刹车行为识别流程图

**（4）怠速预热**

**（5）超长怠速**

长怠速评价模型主要用来分析车辆的停车怠速时间。《汽车驾驶节能操作规范》中规定停车超过 60s 时发动机应媳火。考虑到城市环境中红绿灯时间很多超过 60s，本文进行长怠速算法识别中，取怠速时间超过 120s 为超长怠速，即长怠速行为的条件为

T>=120s (3.10)

如果怠速时间超过阈值时，则认为出现长怠速情况，应记录怠速累计时间和该事

件行为的次数。长怠速为识别算法流程图如图 3.7 步骤说明如下：

1）获取 OBD 数据中发动机转速和仪表盘速度，初始化参数；

2）判断转速 n 是否不等于零且速度 V 为 0，若是，则记录此条数据采集时间为长怠速发生初始时间。循环判断条件，若是，更新事件持续时间，若否，进入步骤 3；

3）判断事件持续时间是否大于等于 120s，若是，输出事件持续时长，长怠速事件次数加一；

4)判断是否为行程最后一条数据，若否，返回步骤2，若是算法结束。

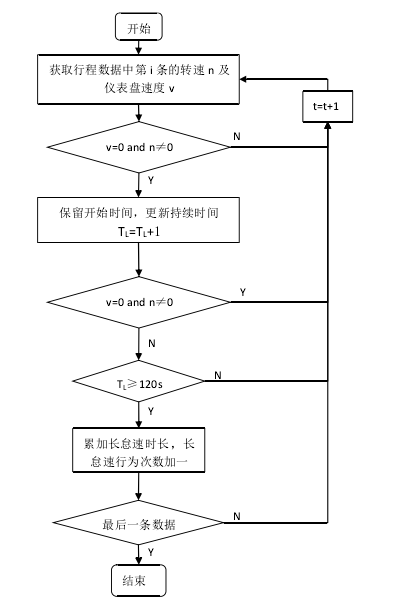


图3.7 长怠速算法识别流程图

**（6）熄火滑行**

acc\_state=0代表熄火，直接对每辆车的行车轨迹数据进行统计，引入变量V，且，如果acc\_state=0时，V=1，则表明为熄火滑行，并记录时间，总熄火滑行时间公式如下：



**（7）超速**

在获取 OBD 数据仪表盘速度的基础上，根据各路段的最高车速限值设定为超速阈值 Va，本文研究的超速车辆基本发生在高速路段，根据客车在一般高速上的速度限制，这里取超速阈值 Va为 100km/h，这里引入超速界定时长 Ta避免个别噪点的影响造成误判，根据 OBD 设备采样频率设定 Ta为 3s。

超速行为识别算法流程图如图 3.1 所示，详细步骤说明如下：

1)获得车辆仪表盘速度数据，初始化参数；

2)判断速度 Vt是否大于速度阈值，若是，则记录此条数据采集时间，循环判断速

度 Vt，若超过阈值，则一直持续超速驶，更新持续时间，若否，超速行驶行为结束，进入步骤 3；

3)判断超速持续时间 TL是否大于等于3s，若是记为一次超速行为，累加超速时长且超速次数加一；

4)判断是否为行程最后一条数据，若否，返回步骤 2，若是输出超速次数和累积超速时长，算法结束。

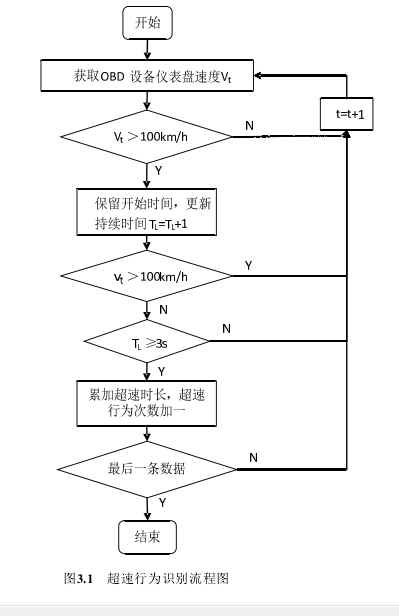


图3.1超速行为识别流程图

**（8）急变道**

当方向角<30度，且时，判断车辆为急变道行为

# 基于长短期记忆网络的行车安全评价模型构建

## 5.1特征提取

4.1节介绍了里程、平均速度、加速度、平均方向变化率等一般属性以及计算公式。文献[1]的实验结果表明，将一般属性和排列熵属性进行结合能够有效地对轨迹进行标识。因此，本章将一般属性和排列熵属性结合，以此作为神经网络的特征输入。

5.1.2 排列熵算法

文献[1]中，Christoph Bandt等人提出了一种衡量一维时间序列复杂度的平均熵参数一排列熵 ( Permutation Entropy，PE)，它在反映一维时间序列复杂度的性能方面与LyaPullov指数相似，并且与LyaPunov数、分形维数等复杂度参数相比，具有计算简单、抗噪声干扰能力强等特点。算法的基本原理如下。设一时间序列，对其进行相空间重构，得到矩阵：

，，。 (1)

式中：、分别为嵌入维数和延迟时间；。矩阵中的每一行可看作一个重构分量，共有K个重构分量。将 X(i) 重构矩阵中的第 j 个重构分量，根据数值大小按照升序重新排列，表示重构分量中各个元素所在列的索引，即

。 (2)

如果重构分量中存在相等的值，即

， (3)

此时就按照、值的大小来排序，即当时，有

。 (4)

因此，对于任意一个时间序列 X( i) 重构所得的矩阵中每一行都可以得到一组符号序列， (5)

式中：，且，维相空间映射不同的符号序列总共有， 符号序列是其中的一种排列。若计算每一种符号序列出现的概率为，则按照Shannon熵的形式，时间序列的种不同符号序列的排列熵可以定义为

， (6)

当时，就达到最大值。为了方便，通常用将进行归一化处理，即

。 (7)

值的大小表示时间序列的随机程度。的值越小，说明时间序列越规则；反之，则时间序列越接近随机。的变化反映并放大了时间序列的微小细节变化。算法总体流程如图1所示。

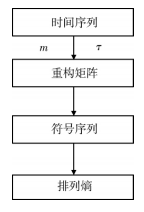


图1排列熵算法流程

5.1.2 基于排列熵的特征提取

**（1）速度排列熵**

对轨迹每点的速度与角度进行排列熵算法的计算，可以得出相应的速度排列熵和角度排列熵的属性，这些属性能够表达出该轨迹的有序程度，能够体现出轨迹内部的细微变化，所以把排列熵作为轨迹的特征值参与轨迹分类。

把轨迹速度排列熵作为轨迹的一种特征值，参与出行模式分类，下面对基于速度排列熵提取的算法进行描述。

(1)获取轨迹轨迹，将轨迹中每个坐标点的速度构成时间序列T=，其中表示第个点的速度;

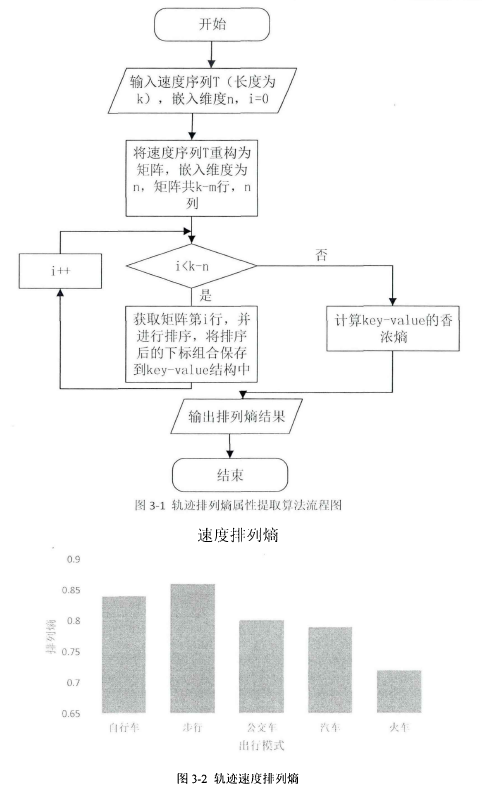
(2)选择合适的维度m，作为求轨迹排列滴的嵌入维度；

(3)将轨迹的速度序列重构为矩阵；

(4)按照升序的顺序对矩阵中的重构分量进行排序；

(5)计算A:个符号向量的香农熵作为Traj的排列熵；

提取轨迹排列熵算法的流程图如图3-1：



**（1）角度排列熵**

……

5.2.3 基于Clara算法的驾驶行为特征聚类

K-means算法的缺点是产生类的大小相差不会很大，不适合于大数据量的计算，因此，应用于GPS行车轨迹数据中效果不佳。于是，在本文中我们采取K-means的改进算法—Clara算法对驾驶行为特征进行聚类。

Clara(Clustering Large Applications，大型应用中的聚类方法)不考虑整个数据集,而是选择数据的一小部分作为样本。Clara算法的优点为在PAM的基础上采用了抽样技术，能够处理大规模数据。它从数据集中抽取多个样本集, 对每个样本集使用PAM,并以最好的聚类作为输出。

CLARA算法的步骤如下：

(1)确定抽样的次数v，重复执行（2）到（4）步骤v次；

(2)随机地从整个数据库中抽取一个N(例如：(40 + 2 k))个对象的样本,调用PAM方法从样本中找出样本的k个最优的中心点；

(3)将这k个中心点应用到整个数据库上，对于每一个非代表对象Oj ,判断它与从样本中选出的哪个代表对象距离最近；

(4) 计算上一步中得到的聚类的总代价。若该值小于当前的最小值,用该值替换当前的最小值,保留在这次选样中得到的k个代表对象作为到目前为止得到的最好的代表对象的集合；

(5) 返回到步骤(1)，开始下一个循环。

算法结束后，输出最好的聚类结果。

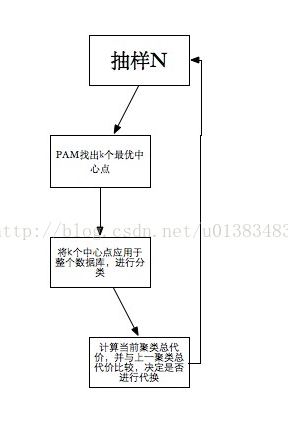


图 Clara的算法流程图

5.2基于LSTM神经网络的行车安全评价模型

5.2.1 驾驶行为特征选取

驾驶倾向涉及的因素众多，本文主要考虑驾驶倾向性中与驾驶行为安全相关的一个部分即驾驶员内在激进性，根据个体特征可分为激进型、普通型和保守型。激进型驾驶员活泼好动，易产生冒险动机，易于高速行车和超车，在行车过程中容易出现急加减速车、急刹车和急转弯情况；温和型驾驶员一般自制力强，驾驶平稳，不违反交通规则；保守型驾驶员比较偏向于开慢车，没有冒险动机，但是遇到紧急情况易惊慌失措[32]。

进行驾驶倾向性识别时需要大量实测行车数据，由于本文采集和经过分析的驾驶倾向性评价指标参数较多，如果考虑将全部用于预测分析驾驶员的倾向性将会大大增加判断时的计算量和复杂性，所以如何选取有效反映驾驶倾向性的特征参数从而减少计算量显得尤为重要。

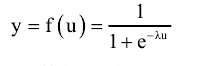
激进型驾驶行为已成为了交通事故的主要原因，包括超速、急变速、急转弯等方面。通过 OBD 设备终端采集得到的车辆数据中包含了很多与驾驶倾向性相关的驾驶行为特征信息。在车速方面，澳大利亚 RTA 相关研究表明，随着车速的提高会导致交通事故风险成倍增长；在加速度方面，它反映了驾驶员对加速踏板和减速踏板的控制行为，而这些驾驶行为不仅会给驾驶员带来不良反应而且会对行车安全带来很大的影响。

因此，通过分析驾驶倾向的外在表现，结合上一章异常驾驶行为分析，选取异常驾驶行为中反应驾驶员操作安全的超速及“四急”（急加速、急减速、急刹车、急转弯）和反应车辆安全状态的平均车速、最高车速作为驾驶倾向性识别特征参数，以 OBD 设备终端采集的实车试验数据分析及前文异常驾驶行为识别结果为基础，建立驾驶倾向特征辨识模型。

5.2.2 RNN神经网络算法及原理

**（1）激活函数**

BP 神经网络隐含层中的激活函数类型共有五种函数和线性函数、Sigmoidal 函数和髙斯画数。本文分类采用的是sigmoidal函数，其函数表达式吧：



式中，其中λ表示为 Sigmoidal 函数的増益。

**（2）BP 神经网络结构**

BP 神经网络采用的是多隐含层结构，在逻辑结构中主要包括输入样本构成的输入层、多个隐含层以及目标输出层，每个结构层又包含多个神经元，各网络层之间的神经元通过相互交叉联系，而每个层内部的神经元是无连接的。如图 4.3 所示为 BP 神经网络结构图。

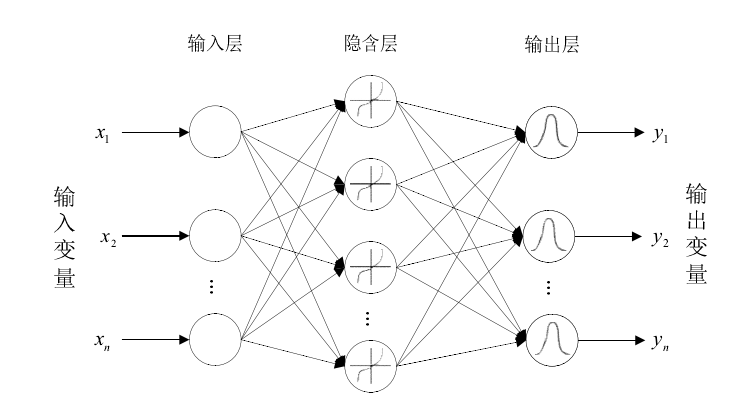
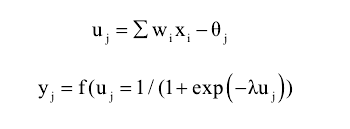


图4.3 BP神经网络结构图

BP神经网络结构的节点计算公式：



1. δ学习规则
2. BP 神经网络权值的更新规则
3. BP 神经网络算法步骤

5.2.3 基于 神经网络的不良驾驶行为评估模型

不良驾驶倾评估模型是采用三层 BP 神经网络，通过提取与驾驶倾向性相关的特征参数数据，利用上一节中……聚类算法得到驾驶行为聚类结果作为 BP 神经网络训练样本，为驾驶倾向性评估的原始样本数据加入类别标签，即以新构造的特征向量作为 BP 神经网络的输入层训练向量，其结构图如图 4.4 所示。

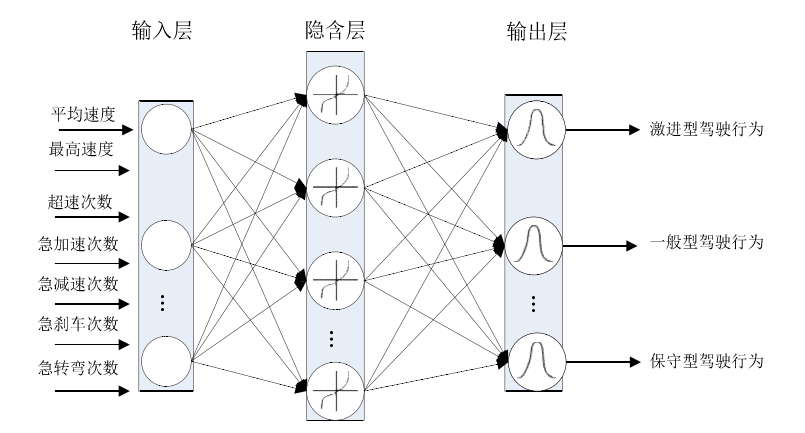


图4.4 BP神经网络结构图

构建BP神经网络驾驶倾向性评估模型，其中隐含层采用的激励函数使用Sigmoidal函数，以 Clara 聚类结果训练 BP 神经网络，选取其中 3000个驾驶数据样本作为测试集，并通过 MATLAB 工具箱对驾驶数据样本进行训练，最后再对 BP 神经网络进行模型准确度验证。评估模型构建后，对于新的行程数据样本，就可以判断该驾驶员是激进型、温和型还是保守型。其流程图如图 4.5 所示。

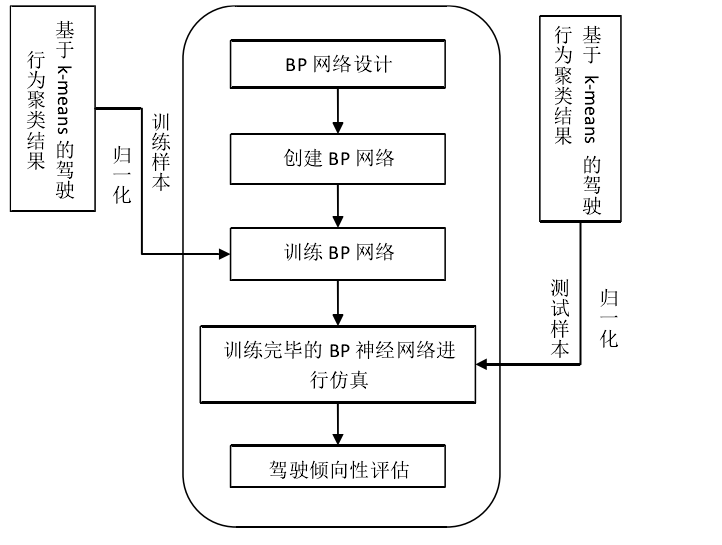


图4.5 驾驶倾向性评估流程图

# 6.基于……的综合评价模型构建

## 6.1综合评价指标体系的建立

## 6.2 基于……综合评价模型

## 6.3

# 7. 问题的求解与验证

7.1 问题的求解

7.1.1问题1的求解

7.2.2问题2的求解

7.2 系统的测评

7.2.1准确率：accuracy

7.2.2 P-R曲线及F1度量（查准率P与查全率R，召回率=查全率）

7.2.3 ROC曲线（TPR,FPR,AUC）

# 8. 结论

# 参考文献

[1]庄薇薇.如何寻找大数据营销的最佳切入点[J].科技资讯，2014，12(30):240.

[2]张楚，冷昕.大数据在电商企业的应用研究——以阿里巴巴为例[J].滁州职业技术学院学报，2017，16(01):60-63.

[3]艾宏.大数据背景下精准营销的探析与思考[J].现代商业，2016(22):34-35.

[4]刘一枞.香港付费电视市场概况及现有付费电视提供商的打包定价策略[J].2012.

[5]龚敏，刘广丹.基于大数据的精准营销应用研究综述[J].市场周刊(理论研究)，2016(07):58-60.