重庆大学本科学生毕业设计（论文）

基于深度学习的人类出行模式研究



学 生：吴志超

学 号：20144514

指导教师：陈超

助理指导教师：李X X

专 业：计算机科学与技术

重庆大学计算机学院

2018年6月

**Graduation Design(Thesis) of Chongqing University**

# **Study on the model of human travel based on deep learning**



# 

# **Undergraduate: WuZhichao**

# **Supervisor: Prof. ChenChao**

# **Assistant Supervisor : Lecturer Li XX**

# **Major: Computer science and technology**

## **College of Computer**

### Chongqing University

### June 2018

摘 要

在GPS技术发展越来越精确的时代，人类每天会在城市中留下大量的行动轨迹，这些轨迹对研究人类的出行方式有着非常重要的作用。明白城市中人类的每日出行方式对交通堵塞，美丽城市建设都有十分重要的意义。本文主要内容是通过研究GPS轨迹中包含的物理特征，利用深度学习框架，推测出GPS轨迹的原始出行方式。

研究GPS轨迹包含的物理信息，提取出点的速度、段的速度均值、段的速度标准差、点的加速度、段的加速度均值、段的加速度标准差、点的转向、段的转向均值、段的转向标准差、段的最大速度、段的最大加速度、段的最大转向等特征，通过对提取特征的预处理，包括过滤噪声，离散化，归一化等方法，制作TFRECORD数据集，构造三种网络模型：DNN、RNN\_NvN与RNN\_Nv1，利用控制变量法分别挑选出三种模型中最合适的网络与参数，计算三种模型的F1-sorce，比较三种模型的精度。

经过研究发现，DNN模型中精度最高的是不采用dropout与Maxout的双层DNN（12特征），精度可以达到0.88。RNN\_NvN与RNN\_Nv1结构中精度最高的网络为双层双向MaxoutGRU网络，其中RNN\_NvN模式中，9个特征（不包括最大速度、最大加速度、最大转向）输入的双层双向的MaxoutGRU在测试集上精度可达0.939，RNN\_Nv1模式中，9个特征输入的双层双向的MaxoutGRU精度可达0.942。

关键字：GPS轨迹，出行方式推测，深度神经网络，循环神经网络

**ABSTRACT**

空一行

　　英文摘要另起一页，内容应与“中文摘要”对应。使用第三人称，用现在时态编写。

　　（小四号，Times News Roman 字体）

与正文空两行

**Key words：** XXX, XXX, XXX, XXX,

目录

**[1绪论](#_Toc16144_WPSOffice_Level1)** **[1](#_Toc16144_WPSOffice_Level1)**

**[1.1课题背景](#_Toc9381_WPSOffice_Level1)** **[1](#_Toc9381_WPSOffice_Level1)**

**[1.2发展现状](#_Toc15550_WPSOffice_Level1)** **[1](#_Toc15550_WPSOffice_Level1)**

**[1.3本文研究内容](#_Toc24324_WPSOffice_Level1)** **[4](#_Toc24324_WPSOffice_Level1)**

**[1.4本文安排](#_Toc18859_WPSOffice_Level1)** **[5](#_Toc18859_WPSOffice_Level1)**

**[2系统设计](#_Toc8638_WPSOffice_Level1)** **[5](#_Toc8638_WPSOffice_Level1)**

**[2.1系统框架](#_Toc27680_WPSOffice_Level1)** **[5](#_Toc27680_WPSOffice_Level1)**

[2.2.1GEOLife 数据集](#_Toc9381_WPSOffice_Level2) [6](#_Toc9381_WPSOffice_Level2)

[2.2.2数据处理](#_Toc15550_WPSOffice_Level2) [7](#_Toc15550_WPSOffice_Level2)

[2.2.3 深度学习网络构建](#_Toc24324_WPSOffice_Level2) [17](#_Toc24324_WPSOffice_Level2)

**[3实验](#_Toc1304_WPSOffice_Level1)** **[21](#_Toc1304_WPSOffice_Level1)**

**[3.1实验环境](#_Toc16900_WPSOffice_Level1)** **[21](#_Toc16900_WPSOffice_Level1)**

**[3.2训练参数](#_Toc25393_WPSOffice_Level1)** **[22](#_Toc25393_WPSOffice_Level1)**

**[3.3评估方式](#_Toc27017_WPSOffice_Level1)** **[22](#_Toc27017_WPSOffice_Level1)**

**[3.4实验结果与讨论](#_Toc982_WPSOffice_Level1)** **[23](#_Toc982_WPSOffice_Level1)**

[3.4.1 DNN实验](#_Toc8638_WPSOffice_Level2) [23](#_Toc8638_WPSOffice_Level2)

[3.4.2 RNN结构实验](#_Toc27680_WPSOffice_Level2) [27](#_Toc27680_WPSOffice_Level2)

[3.4.3 RNN\_NvN 实验](#_Toc1304_WPSOffice_Level2) [29](#_Toc1304_WPSOffice_Level2)

[3.4.4 RNN\_Nv1实验](#_Toc16900_WPSOffice_Level2) [32](#_Toc16900_WPSOffice_Level2)

[3.4.5 三种模型比较](#_Toc25393_WPSOffice_Level2) [34](#_Toc25393_WPSOffice_Level2)

**[4总结与未来展望](#_Toc11294_WPSOffice_Level1)** **[35](#_Toc11294_WPSOffice_Level1)**

**[参考文献](#_Toc4142_WPSOffice_Level1)** **[1](#_Toc4142_WPSOffice_Level1)**

### 1绪论

## 1.1课题背景

随着经济水平与科技水平的发展，人们的生活水平也在水涨船高越，各种机动车的数量也在急速的增长。2017年，全球的汽车销量首次突破9000万辆，而中国这个市场就销售了1/3，如此巨大的汽车数量的增长为城市的交通带来了非常大的压力。城市的道路规划远远不能满足街头上堵塞的汽车，所以早高峰，晚高峰，成为了北京，上海等大城市居民最为痛恨的城市病。随着共享这个概念的发展，共享单车如雨后春笋般出现在了城市的街头，而共享单车的乱停乱放成为了与建设美丽城市相违背的现象。大量的共享单车挤上街头，占用城市里本来就不赋予的人行道资源，所以在人流密集，人流出入频繁等最需要共享单车的地方设立停放点是必须的。掌握城市居民的出行方式成为了解决以上问题的必须跨过的障碍，掌握城市居民的出行方式的重要途径便是通过城市居民的移动轨迹来推测城市居民的出行方式。

城市居民的移动轨迹在GPS技术未发展成熟的情况下，获取的方式只能是通过便携式穿戴设备，然而这种设备的精度与可靠性都非常的低，获得的数据量小，准确性差，导致出行方式的预测精度低，研究成果停止不前。然而随着GPS技术发展越来越好，以及智能手机广泛普及的情况下，城市居民的出行轨迹的数据获取变得越来越容易，数据精度也越来越高。研究GPS轨迹中时间和空间信息包含的深层次特征为城市居民的出行方式的推测带来了相当大的益处，所以越来越多的学者开始通过GPS轨迹来研究城市居民的出行方式。大量的方式方法被学者们用来研究如何通过GPS轨迹来推测城市居民的出行方式，而其中最为有效的一种方法就是机器学习。机器学习自从被提出以来，就在人工智能方面有着显著的成果，所以使用机器学习来研究本课题也是大势所趋。

## 1.2发展现状

自GPS技术被广泛使用后，国外许多研究院的学者都对GPS轨迹开始进行研究，例如微软研究院早在2007年就采集了许多用户的GPS轨迹来进行研究，并将他们的采集数据开源出来供大家使用，也就是常说的GEOLIFE数据集。

在本课题的研究中，大多都是采用机器学习的方法，而机器学习又分为深度学习与非深度学习。基于这两种方法的研究现状介绍如下：

⑴非深度学习

非深度学习的研究方法主要分为监督学习与非监督学习两大类，其中监督学习的方法有分类与回归、支持向量机、高斯混合模型、朴素贝叶斯网络和决策树等等，而无监督学习中最为有代表性的就是聚类方法。非深度学习是人工智能中非常重要的研究手段，在文本分类，自然语言处理等问题上都有不错的收益，所以许多学者就在这些方法的基础上做了大量的研究，比如：

微软的的ZhengYu等学者就在GEOLIFE的基础上做了大量的研究，定义出了大量的特征，例如，速度，加速度，停止率(SR)[17][18][19][20]等等。并且运用机器学习的方法，包括决策树，支持向量机，贝叶斯网络，条件随机场等来推测出行模式。这些模型中，精度最高的决策树方法达到了0.762的精度。

[M Simoncini](http://xueshu.baidu.com/s?wd=author:(Matteo%20Simoncini)%20Fleetmatics%20Res.,%20Florence,%20Italy&tn=SE_baiduxueshu_c1gjeupa&ie=utf-8&sc_f_para=sc_hilight=person" \t "http://xueshu.baidu.com/_blank) 等学者在他们的通过GPS轨迹来分类车辆的研究中也有提到对GPS轨迹的特征的定义，其中包括段速度，段加速度等。

[TaoFeng](https://www.sciencedirect.com/science/article/pii/S0968090X13002039?_rdoc=1&_fmt=high&_origin=gateway&_docanchor=&md5=b8429449ccfc9c30159a5f9aeaa92ffb" \l "!)等人通过采用GPS轨迹以及加速度计数据，运用贝叶斯置信网，通过将不同的特征分等级来推测8种出行方式，包括[walking,bicycle,running,motorcycle,bus,car,metro,tarm]。

在2010年，由张冶华[26]等学者研究的GPS轨迹的出行信息提取中也早就提到了一些重要特征的定义，以及对GPS原始数据的处理。通过将不同的特征加入到多层感知器网络，贝叶斯网以及决策树中，最终实验得出决策树的精度最高为0.923。

杨飞[28]等学者通过用时间段的方式对GPS轨迹数据进行分段，计算出每段的平均速度、位移、速度最大值，将这些特征运用神经网络识别出分段的交通出行方式，最后根据结果来确定转换点。

沈云[27]等学者通过将GPS轨迹分段，然后将分段数据经过去噪处理和特征（速度，加速度，距离等）提取，然后运用神经网络，决策树，KNN和支持向量机四种模式识别算法进行出行方式的识别。最终四种模型在测试集上分别得到0.803，0.802，0.818，0.887的精度。

⑵深度学习

深度学习作为机器学习的一大门派，在被提出许多年后才被真正的用来做人工智能的研究，而深度学习最重要的概念就是神经网络，自从深度神经网络的梯度消失问题[1][2]被有效的解决以后，深度学习的发展又迎来了新的春天。目前深度学习的框架也有很多，比如Tensorflow,Torch,Theano,Caffe,MXNet等都是发展比较成熟的。深度学习网络主要分为深度神经网络（DNN），循环神经网络（RecurrentNN），卷积神经网络（CNN），递归神经网络（RecursiveNN）。DNN是深度学习中最简单的神经网络结构，也即是常说的全连接神经网络，这种网络对于一些具有离散特征的问题具有很好的学习性能，例如MNIST手写数字的识别，已经具有98%的正确率。（下文中的RNN都指循环神经网络）其中循环神经网络中又包含了很多种变种网络结构，例如LSTM，GRU，SRU。其中LSTM指长短期记忆网络，GRU指门循环单元。LSTM对处理长期的信息有特别独到的效果，但是由于参数比普通RNN多了一倍，在训练时速度较慢，而GRU就是为了解决LSTM参数多，训练速度慢的缺点，在LSTM的结构上做了变形，将LSTM的遗忘门和输入门合成为单一的更新门。循环神经网络最初是为了用来处理文字及语音信息，随着深度学习的发展，人们发现只要能将问题化简为序列的推测问题，用RNN来学习都是是非常有效的，例如MNIST手写数字以及语言的翻译。RNN在推测结构上又可以分为NvN,NvM,Nv1（前面的N代表输入样本个数，后面则代表输出标签个数）。卷积神经网络是在DNN中间加了几层的卷积层与池化层，用来缩减输入向量的大小。因为卷积神经网络一般用来处理图像等的数据，输入向量过大，所以特定的卷积与池化后，可以保留图像的有效信息。目前卷积神经网络已经为公认的比较有效的处理图像的深度学习网络。递归神经网络主要用来处理一些树形结构的问题，例如语法树等问题都可以用递归神经网络来解决。深度学习网络虽然很好，但往往具有严重的过拟合的问题，所以会使用dropout[25],L1或L2正则化[24]等方式来抑制过拟合。

正因为深度学习的学习精度往往好于非深度学习，所以近年来，大量的学者也开始将研究重心转移到深度学习的方法上来，就比如：

YuKi Endo[22]等学者提出了一种新的方式，将轨迹转换成一张灰度图，像素的深度表示停留时间的长短，最后将灰度图转换为一个向量，输入到全连接神经网络中，这种方式的检测精度在GeoLife 数据集上的精度可以达到0.679，在KT 数据集上可以达到0.832的精度。

SinaDabiri等学者用一种新的方式，即CNN来处理GPS轨迹，通过将GPS轨迹映射为CNN的输入向量，通过去除噪点，数据增强，包概念等等，使得模型可以达到0.848的精度。

Toan H. Vu等学者研究了一种称之为SGRNN的神经网络，其中更改了RNN节点的内部结构，改变了每个节点的状态计算公式。这种神经网络在推测五种出行方式（bike,run,still,walk,vehicle)上，在HTC数据集上取得了0.93的精度，在HAPT数据集上取得了0.905的精度。

XiangJiang[10][21]等学者的提出的RNN\_NvN的MaxoutGRU网络也有十分显著的效果，其中的Maxout在GRU网络上的应用极大的增加了GRU网络的学习能力。

## 1.3本文研究内容

本文研究的是通过GPS轨迹推测人类的出行方式，其中主要包括四种出行方式：自行车，汽车，步行，公交车。在具体的过程中，又主要分为系统实现与实验两个部分，其中系统实现的部分包括GPS轨迹的分段，GPS轨迹特征的提取，特征的预处理，以及三种深度学习模型的构建。实验部分主要是对三种深度学习模型的输入特征选取、参数的优化和模型的比较。故本文的主要研究如下所示：

⑴GPS轨迹的分段

本文的GPS轨迹全部来自于微软的GEOLife数据集，这些GPS轨迹包含了几十位志愿者将近一年的出行记录，而这些GPS轨迹都是混合在一起无法直接使用的，所以必须先将所有的GPS轨迹分段。按照一个交通工具从开始记录到最后记录的时间为分段标准，这样所有的GPS轨迹段中的记录点都有相同的标签，而不同的GPS轨迹段中又有数量不等的轨迹点。

⑵特征的提取

GPS轨迹是由一个个独立的记录点组成的，每个记录点中都包含有此点记录的时间与经纬度，正是这一个个的记录点构成了一条条的GPS轨迹，而这些GPS轨迹中又包含了大量的深层次的特征，例如两点可以计算出距离和时间差，通过时间差就可以计算出速度，并将此速度看作是后点的速度，此速度即为深层次的特征，本文通过研究，在GPS轨迹段中提取出了点的速度，段的速度均值，段的速度标准差，点的加速度，段的加速度均值，段的加速度标准差，点的转向，段的转向均值，段的转向标准差，段的速度最大值，段的加速度最大值，段的转向最大值。这些特征将会为以后推测出行方式有着非常大的作用。

⑶特征的预处理

在上一节中寻找到的特征，必须经过一层层的处理之后才能成为深度学习网络的输入，本文中主要先将数据使用hample过滤法进行过滤，去除掉噪音数据，然后再使用min-max归一法将数据压缩到[0,1]之间，最后通过等宽分箱的离散化方法，将特征转换为等长的onehot向量。最终，经过一步步的处理，数据由原始的标量转换为了深度学习网络可以使用的输入向量。

⑷三种深度学习网络模型

由于深度学习在出行方式推测中取得了很好的效果，所以本文采用了深度学习的方法。本文一共构建了三种深度学习的网络模型，其中包括DNN（全连接神经网络）、RNN\_NvN（循环神经网络的many to many模型）、RNN\_Nv1（循环神经网络的many to one 模型）。因为在深度学习中，可以使用多种方法来提升网络的精度，所以本文在DNN中分别实现dropout与maxout的版本，在RNN中又实现了LSTM，GRU，以及MaxoutGRU三种不同的网络结构。

⑸实验

因为本文实现了三种网络结构，所以按照三种网络模型的结构，本文的实验主要分三类：(i)DNN (ii )RNN\_NvN (iii) RNN\_Nv1。在这三种实验中，都主要围绕网络的输入特征选取，隐藏层层数，隐藏层节点个数，不同的网络结构版本比较来进行实验，最后通过实验结果来评估三种模型。

## 1.4本文安排

在第一部分中，主要阐述了本课题的背景，介绍了研究人类出行方式的重要性与迫切性，目出行模式的推测的研究现状，以及本文的主要研究内容和后文的安排。

在第二部分中，主要阐述系统实现的方法，介绍了系统的结构，以及系统的各个部分的主要作用与实现手段。

在第三部分中，主要是通过本文实现的系统所进行的类比实验，评估系统中的不同的网络结构。

在第四部分中，主要总结我们的工作以及对未来工作的思考与展望。

### 2系统设计

# 2.1系统框架

本系统的框架图如2.1所示，GPS轨迹数据首先经过数据分段以后，成为了一条一条具有标签的GPS轨迹段，接着通过对GPS轨迹段的研究，提取出可能对模型学习有用的特征，包括基于点的特征和基于段的特征，然后将提取出的特征进行预处理，包括数据的噪声过滤，归一化以及离散化，得到模型可以接受的onehot向量，因为本系统采用的深度学习框架为TensorFlow，且深度学习的数据量一般都比较大，直接加载入内存会造成系统崩溃，所以又制作了TFRECORD数据集，最后将TFRECORD数据集输入到最终的三种不同的网络结构中，到此为止就是本系统从开始的数据处理到最终的模型训练的全部过程。关于本系统的关键实现会在下文中一一给出。

# 2.2系统实现

GPS轨迹分段

GPS轨迹特征提取

特征预处理

深度学习模型

GPS轨迹数据集

制作TFRECORD数据集

噪声过滤

归一化

离散化

DNN

RNN\_NvN

RNN\_Nv1

最大点速度

····

点速度

图2.1 系统框架图

从系统框架图中可以了解到系统主要分为2大部分:数据处理、深度学习网络构建。其中数据处理主要包括GPS轨迹的分段，GPS轨迹特征的提取，特征的预处理，TFRECORD数据集的制作。在描述系统的主要实现以前，必须清楚的一件事情就是数据的来源与格式。在本系统中，采用的是微软的GEOLife数据集，在此先介绍下GEOLife数据集的组成形式。

## 2.2.1GEOLife 数据集

该GPS轨迹数据集出自微软研究GeoLift项目。从2007年四月到2012年八月收集了182个用户的轨迹数据。这些数据包含了一系列以时间为序的点，每一个点包含经纬度、海拔等信息。包含了17621个轨迹，总距离120多万公里，总时间48000多小时。这些数据不仅仅记录了用户在家和在工作地点的位置轨迹，还记录了大范围的户外活动轨迹，比如购物、旅游、远足、骑自行车。平均每1-5秒或每5-10米产生一个轨迹点。这个数据集可以用来进行用户活动相似度估算，移动模型挖掘，用户活动推荐，基于位置的社交网络，位置隐私，位置推荐。

* 时间：2007年4月～2012年8月
* 数据大小：大概300M.
* 下载地址：[https://www.microsoft.com/en-us/download/details.aspx?id=52367](https://www.microsoft.com/en-us/download/details.aspx?id=52367" \t "_blank)
* 数据格式：

该数据集中一个文件夹存储一个用户的GPS日志，这些日志文件都被转换成了plt格式。为了避免时间区间问题，统一使用了GMT格式的时间表示。其他具体格式如图2.2所示

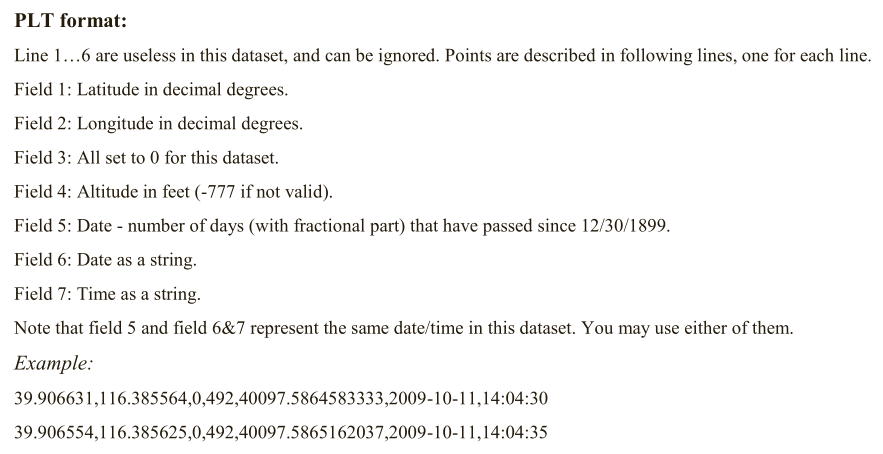
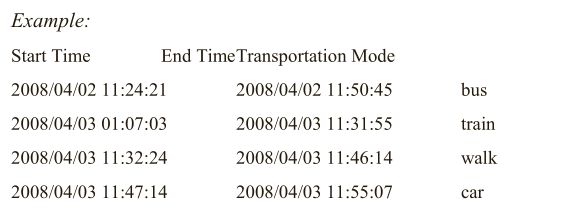


图2.2 轨迹数据PLT文件格式

以及还保存有一个标签文件，主要对应每个时间段的出行方式。如图2.3所示。

图2.3 标签文件格式



## 2.2.2数据处理

明白了数据的来源与格式，接下来重要的步骤就是对GPS轨迹的分段，在源文件中，GPS每一段的轨迹并没有被很好的标记出来，非常影响基于GPS轨迹段的特征的提取，所以进行GPS轨迹分段是至关重要的一步。

▪GPS轨迹的分段

GPS轨迹的分段有很多种，例如1，基于过渡的方法[19]，使用步行划分轨迹。2，基于聚类的方法[15]，测量子轨迹的相似性。3，基于时间的方法[16] 在每个段中使用相等的时间间隔。4，基于距离的方法[11]使用在每个段中行程相等的距离。5，基于轴承的方法[4]，测量轴承方向的变化。6，基于窗口的方法[13]。在本系统中，主要使用的是类似于基于过渡的方法，即在GPS轨迹数据中只要更换了出行方式，即代表一次轨迹的结束。

如图2.4所示，轨迹分段的意思就是为每个GPS轨迹点找到属于自己所属的标签，即出行方式。在本系统中，主要是首先为原始的GEOLife数据集中的PLT格式文件中的每个轨迹点在标签文件中找到对应的标签信息，即如图2.5（a）与2.5（b）所示。在图2.5（b）中找到每个轨迹点的标签信息并一一对应存储下来生成如图2.6所示的csv文件。其中每行代表一个轨迹点的数据，每个属性之间用逗号相隔，包括纬度，经度，占位符，海拔，时间戳，年月日，时分秒，标签，所属段。

为了能够充分的利用GEOLife数据集，对于同一段轨迹可以挑选全部的轨迹点，也可以间隔步长挑选轨迹点，这样就可以形成较小的数据集，但拥有与本段轨迹相似的特征。

图2.4 轨迹分段示意图

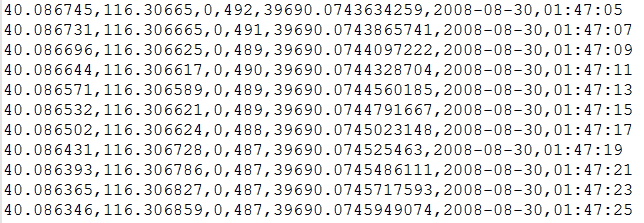
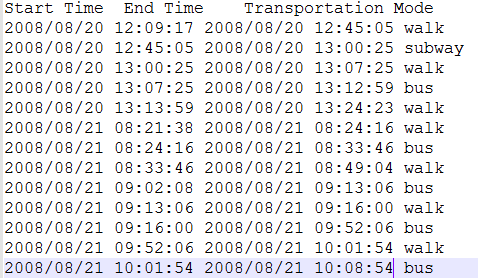
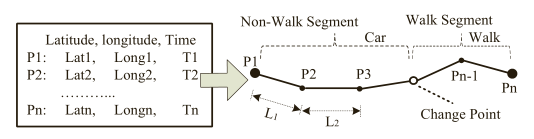


图2.5(b) 标签

图2.5 (a) 轨迹数据格式

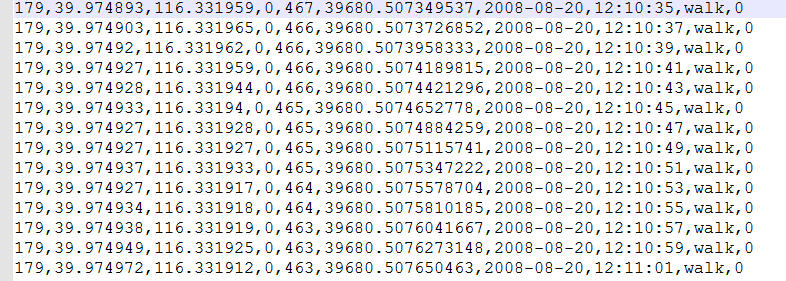


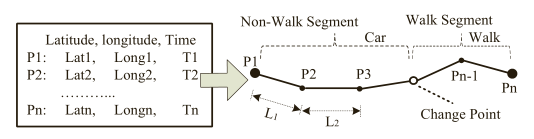
图2.6 GPS轨迹分段文件格式

经过GPS轨迹的分段以后，得到了一段一段的带有标签的GPS轨迹，然而这些轨迹还不能够为我们所用，因为这些轨迹中包含的属性特征还不够全面，所以需要更深层次的提取轨迹段中包含的特征，经过研究，一共提取了12个特征以供模型的输入，包括点的速度，段的速度均值，段的速度标准差，点的加速度，段的加速度均值，段的加速度标准差，点的转向，段的转向均值，段的转向标准差，段的速度最大值，段的加速度最大值，段的转向最大值。为了能够更好的理解这12个特征中所包含的含义，如下给出了这12个特征的定义：

⑴点的速度（v）

在我们的GPS数据中，每段数据中都有若干个点。假设有两个点,,都有经纬度与时间戳，用两个点之间的经纬度算出距离L，T代表两点之间的时间差，则=。如图2.7所示，将GPS数据转换为可以理解的轨迹数据。

图2.7



⑵段的速度均值

在GPS数据中，每一条轨迹中有点集 P = {},经过2.1.1的计算可以得到速度集V={}，即 = 。此处用V的算术平均值而不是用总距离除以时间来得到粗糙的平均速度，这种平均速度对于我们的推测是没有用处的，有时甚至会干扰模型，以至于将汽车的轨迹判断为自行车，甚至步行。所以用所有速度得到的算术平均值可以很好的显现出这条轨迹的运行状态，得到这条轨迹实时的平均速度。

⑶段的速度标准差

在2.1.2中得到了速度集合与速度均值，接下来计算速度的标准差：。速度的标准差是为了来描述这段轨迹速度的离散度，因为在出行方式的不同上，速度的改变是有一定规律的，例如自行车就可以很快的停下，并且可能随时处于加速减速状态，这种状态是汽车不能拥有的，汽车可能不会很快的停下，汽车的速度不会在大多时间呈非线性变化。所以得到速度的标准差对于模型评估是有益处的。

⑷点的加速度

通过2.1.2计算出的速度集合V={}，假定将至之间的时间记为，则加速度。加速度也是一个很重要的属性，因为加速度可以很好的描述出速度的变化情况，在一段轨迹中，速度的变化情况对于推测这条轨迹的分类是很重要的一点，例如，步行的速度变化幅度就不会太大，而公交车因为本身的用途定位，它需要在每一个站台都停车，所以可能在一段轨迹中会出现很多的速度波动，也即加速度的变化，这是，加速度为区分步行或是公交车带来了很重要的依据。

⑸段的加速度

通过2.1.4可以计算得到加速度集合A = {}，则加速度均值：=。在速度均值的时候已经讨论过为什么要计算速度均值，以及速度均值为模型带来的影响。在这里同样求得加速度的均值，加速度的均值也是为了刻画一段轨迹中速度的整体加减速状态。

⑹段的加速度标准差

通过2.1.5计算得到 ：，加速度的标准差是为了得到一段轨迹中加速度的离散情况，也即为了判断这段轨迹中速度的变化是否较大，从而更准确的分析出这段轨迹的分类情况。

⑺转向

将得到的GPS轨迹点通过经纬度映射到平面图中，由于经纬度映射的地理实际距离很大，所以可以直接用经纬度代替平面图上的坐标。假设以段轨迹中存在有三个点,,,设,,由于象限的不同，首先将和都取绝对值，可以计算出，在根据和的正负关系来判断是否要加上象限的角度。即如果>0和>0，则的度数应该加上0;>0和<0，则的度数应该加上90;<0和<0，则的度数应该加上180;<0和>0，则的度数应该加上270。如图2.8(a)所示，用代表的转向角，另，如果<=180,；180<<=360，,前面的负号代表向右转向。

如图2.8(b)转向在步行中是一件很频繁的事情，并且在步行中转向的幅度也会比较大，而在开车的过程中，车是一般不可能大幅度的转向的，一般在直行的过程中，车的转向频率很小，且转向幅度也很小。所以转向这个特征，可以很好的分别转向幅度较大的步行轨迹与行车轨迹。

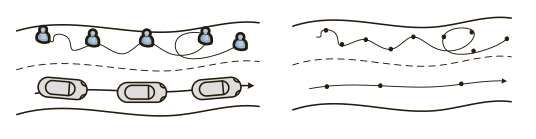


图2.8 Head (b)



图2.8 Head (a)

⑻段的转向均值

令转向均值为2.1.7节算出的集合的平均值，即，用所有转向度数的绝对值来表示每一个点的转向大小，这样求得的转向均值可以表示出这段轨迹的转向程度，所以一般情况下，汽车的转向均值会小，而步行的转向均值则会大于汽车。

⑼段的转向标准差

## 转向标准差即2.1.7中算出的集合的标准差，用来描述一段轨迹转向的离散程度。即，其中。转向的离散程度，可以用来描述一段轨迹的曲折程度，如果离散值小，则并不能说明什么，但若是以段轨迹的标准差很大，则很有可能说明此段轨迹是步行，而不是在开车。

⑽段的最大速度

最大速度即指一段GPS轨迹中的最大速度。即2.1.1中速度集中的最大值。

⑾段的最大加速度

最大加速度即指一段GPS轨迹中的最大加速度，本文中的最大加速度是将所有加速度取绝对值后再取集合中的最大值。

⑿段的最大转向

最大转向是指将2.1.7中转向的集合取绝对值后求出的最大值

明白了每个特征的定义，便可以很轻松的计算出各个GPS轨迹段的特征，为了能够条理清晰，将计算出的特征按用户ID、纬度，精度，点的速度，点的加速度，段的速度标准差，段的速度均值，段的加速度均值，段的加速度标准差，点的转向，段的转向均值，段的转向标准差，段的最大速度，段的最大加速度，段的最大转向，年月日，时分秒，标签，段分类标签存储在csv文件中，每行代表一个GPS轨迹点所拥有的特征。具体格式如图2.9所示

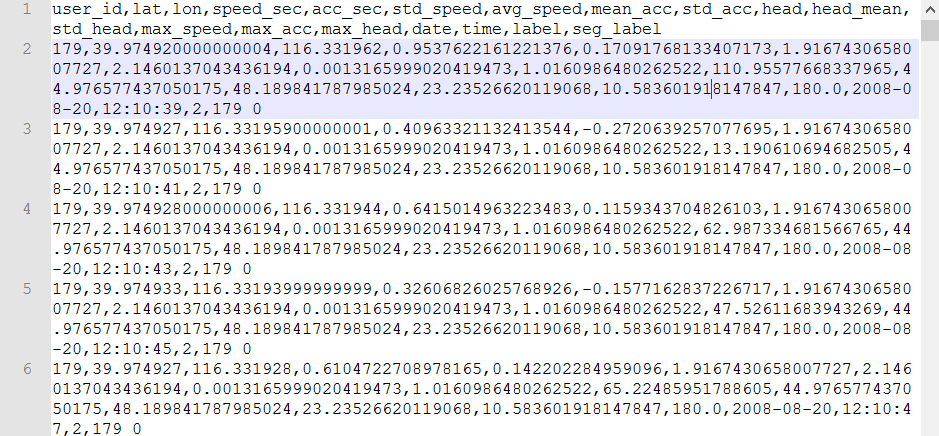


图2.9 特征文件格式图

通过对GPS轨迹特征的提取得到了诸如点的速度等12个特征，然而这12个特征数据存在噪音，单位的不同，大小的不同，以及不能够输入到深度学习网络中的原因，所以必须对特征进行预处理，进行预处理一是为了清洗数据，去除噪音，二是为了去掉量纲，统一大小，三是为了将人们所能理解的特征转换为深度学习网络可以理解的输入，即onehot向量。在本系统中，特征的预处理主要分为以下几个过程：

▪数据过滤

由于GEOLife的数据来源于2010年以前，以及数据在采集过程中的偏差等等，都会造成数据有很大的噪音，就比如在数据的处理过程中，加速度竟然有达到400的，这是很夸张的错误数据要将它去除并且平滑数据分布，所以决定采用Hample[23]过滤器过滤。Hample过滤器在实际的应用中有很好的效果，可以很好的过滤掉噪音，并且平滑数据。Hample滤波也叫箱子滤波，主要是有一个宽度为K的容器，以及指在K/2的数据迭代器，当处于K/2的数据满足一定的条件后，则将调整整个容器内包含的数据，所以这也是为什么说Hample滤波可以平滑我们的数据。由于有些特征会因为数据的不准确而得到难以想象的值，所以在取过滤的阈值的时候经常会取最大阈值为百分位95上的数值，最小阈值会根据实际情况而定。例如，在速度这一特征中，最小值为零，也会出现不为零得情况，所以选择百分位为0的数值作为最小阈值，而在加速度这一特征中，由于与速度有关系，当速度计算出来很夸张时，会导致加速度的变化度十分巨大，在检查数据的过程中，就发现了有-400这一加速度数据出现，所以将最小阈值设定为百分位为1的数值，这样则更好的符合不同特征的筛选，最终决定将速度，速度均值，速度标准差，加速度均值，加速度标准差，转向，转向均值，转向标准差的最小阈值设为百分位为零的数值，最大阈值设为百分位为95的数值，而加速度的最小阈值为百分位为一的数值，最大阈值设为百分位为95的数值。

如图2.10（a）所示，点的速度最大值为1380，而这种速度在城市中是不可能的，很明显的它是噪音数据，所以将最大阈值设为百分位95后，如图2.10（b）所示，经过处理后，点的速度的最大值为31，这符合四种轨迹的正常最大速度。

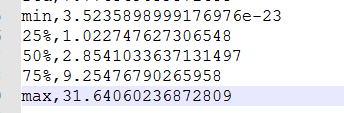


图2.10（a）数据过滤后

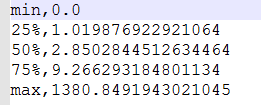


图2.10（a）数据过滤前

▪归一化

经过过滤后的数据已经比较规整，但是不同的特征的单位不同，大小不同，如果将这些量纲不同的数据直接输入到深度学习的网络之中，较大的特征值会对网络产生较大的影响，而较小的特征值对网络产生较小的影响，而这并不符合设计的初衷，因为每一种特征在网络中的权重在初始时应该是相同的，如图2.11所示，归一化对网络训练的影响。所以在深度学习中，归一化是非常重要的过程，归一化是一种简化计算的方式，即将有量纲的表达式，经过变换，化为无量纲的表达式，成为标量。本文将采用线性归一的方法，也即是min-max归一法。例如速度这一特征，首先计算出数据集中经过过滤后的速度的最小值与最大值，然后用计算出每一个数据归一化后的值。

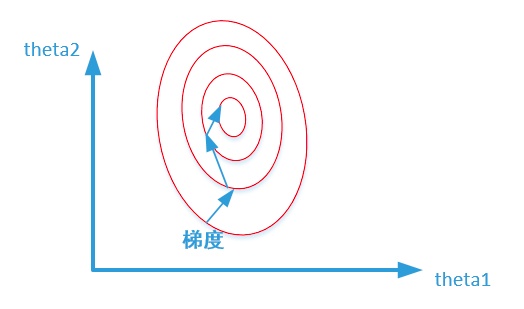


图2.11(a) 未归一化

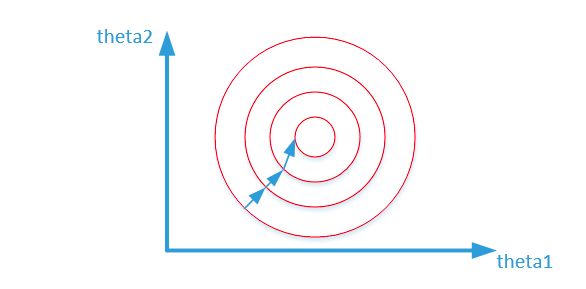


图2.11(b) 归一化后

▪特征离散化

在特征经过归一化后，所有特征都压缩在了[0,1]之间，然而这些特征还是以连续值在表示，在深度学习中，连续值作为网络的输入会导致模型在学习时的震荡幅度变大，而离散化后的数据有以下优势：

1.对异常数据有很强的鲁棒性

比如一个特征是年龄>30是1，否则0。如果特征没有离散化，一个异常数据“年龄300岁”会给模型造成很大的干扰。

2.有利于对非线性关系进行诊断和描述

对连续型数据进行离散处理后，自变量和目标变量之间的关系变得清晰化。如果两者之间是非线性关系，可以重新定义离散后变量每段的取值，如采取0，1的形式，由一个变量派生为多个哑变量，分别确定每段和目标变量间的联系。这样做，虽然减少了模型的自由度，但可以大大提高模型的灵活度。

3.加快运算速度

稀疏向量内积乘法运算速度快，计算结果方便存储，容易扩展。

4.特征交叉

离散化后可以进行特征交叉，由M+N个变量变为M\*N个变量，进一步引入非线性，提升表达能力。

5.模型更加稳定

特征离散化后，模型会更稳定，比如如果对用户年龄离散化，20-30作为一个区间，不会因为一个用户年龄长了一岁就变成一个完全不同的人。当然处于区间相邻处的样本会刚好相反，所以怎么划分区间是门学问。

本文采用等宽分箱的方法将连续特征离散化为onehot向量，如图2.12所示，经过归一化后的特征数据全部落在[0,1]之间，将区间[0,1]划分为10个等长的区间，这样划分的原因一是因为简单快捷，二是因为归一化的方式是线性压缩。当数据(x)落入0.6-0.7中经过离散化可以得到onehot向量[0,0,0,0,0,0,1,0,0,0]。

0

0.1

0.2

0.3

0.4

0.5

0.6

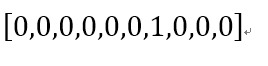
0.7

0.8

0.9

1

x



离散化

图2.12 离散化方式

▪制作TFRECORD数据集

TFRECORD文件是Tensorflow为了解决大数据集时内存不足的问题，提出将所有的数据都分散保存为统一的格式，在制作与读取的时候都通过统一的数据传输协议进行编码与译码的方案。由于是统一的数据格式，所以可以通过异步的方式从各个文件中读取相同格式的数据，由此可以不需要一次性将所有数据都读入内存中，且可以很好的混淆数据集。在制作数据集的时候还有一个非常重要的步骤，就是数据混淆，在深度学习中，数据混淆的好与坏经常会影响网络学习的速度与性能，因为在深度学习中，每次学习都是一批一批的数据，这一批数据如果都是同样的标签，那么网络会偏置的学习到一种情况，而不能充分的学习到别的情况，所以为了能够更好的将数据混淆，在制作TFRECORD文件的时候，存入的轨迹不按照相同的标签存放，这样就可以很好的混淆数据。当数据经过离散化处理以后，由于数据量很大，每一个特征都是一个等宽长度的向量，为了加速网络的训练，必须将网络需要的数据提前制作好。因为网络涉及到RNN，所以我们将长度不等轨迹序列划分为等长的数据序列，然后将切割后的每一条数据与标签存入到TFRECORD文件中，为了能够处理切割后的尾部数据，将每段剩余的数据的尾部填充为零，并且将每个序列的实际长度保存下来，以及为了增加训练的速度，可以提前将数据划分为各个长度的等长度的不同数据集。最终，按照8：1：1的比例划分训练集，验证集与测试集。

## 2.2.3 深度学习网络构建

本文采用深度学习的方法来推测城市居民的四种出行方式，主要构建了三种不同的网络结构，包括DNN，RNN\_NvN，RNN\_Nv1。虽然三种网络有着不同的结构，但是它们也可以简单的，统一的看作都拥有输入层，隐藏层与输出层。在深度学习网络中，为了能够让网络学习到非线性的边界，往往要在每一层隐藏层的输出中加入激励函数，例如sigmod，than，relu，maxout。在近期的深度学习发展中，研究发现，maxout对于凸函数具有良好的拟合能力，所以maxout也在近期被广泛的使用。在深度学习的训练中，经常会发生过拟合的问题，过拟合是每一个深度学习网络都会产生的问题，而为了抑制过拟合也有很多的解决方案，例如：dropout，L1和L2正则化，这些方案被证明都可以有效的抑制深度学习网络在训练集上的过拟合问题。在这三种网络中，DNN与RNN的结构又有所不同，RNN主要用来处理关于序列的问题，所以RNN有很多的变体结构，例如LSTM，GRU以及MaxoutGRU，这些变体结构有了比普通RNN更好的学习能力，能更好的去处理诸如语言处理，语言翻译的问题。所以在以上种种问题上，深度学习是一个庞大的系统，在叙述本文的三种网络结构之前， 必须先了解一下相关的基础知识。

▪深度学习相关知识

⑴Maxout

Maxout激励函数准确来说是一层具有k个单元的网络，这k个单元的结构是相同的，具有相同的输入与输出尺寸。经常在k个单元中学习k个权重不同的感知器，最终，将k个单元的输出比较大小，筛选出最大值做为最后的输出。不同于以往的Sigmod,Relu等激励函数，当使用maxout激励函数之后网络的参数将成k倍的增加，因为Maxout网络有k个单元，可以说，每个单元都是有平常的一个Sigmod层。Maxout虽然有些难以理解，但是Maxout的实现却非常的简单，正如它的神经元计算公式：。由于Maxout对于凸函数具有非常好的拟合性，所以在很多时候Maxout具有很高的学习能力。

⑵Dropout

Dropout[25]简单点说即是给隐藏层的每个节点赋予一个通过概率p,即每个节点都有p的概率将自己的信息输出到下一层，这样就等于是变相的修剪了网络，也等于训练了个网络，其中n代表隐藏层节点的个数，Dropout往往对于深层次的网络有很好的抑制过拟合效果。

⑶L2正则化

为了防止模型过拟合，所以对每个训练的参数都加入了L2正则化系数。L2正则化就是给每一个训练参数都加上了一个正则化参数，这个参数一般都小于1，这样，就可以让训练的参数向较小的方向训练，主要是因为较小的参数的网络抗扰动能力强，最后训练的模型泛化能力会提高。

⑷RNN

RNN是循环神经网络的简称，RNN是因为全连接神经网络无法对语言序列等的问题很好的建模而应用而生的。正如RNN的名字所述一样，RNN中最重要的就是循环，RNN的隐藏层中在不断的循环计算每一个节点的状态，并将节点的状态传给下一个状态，而且每一个节点也会有一个输出，最终可以根据不同的问题采用不同的输出，所以RNN又有四种不同的模型，分为1vN，NvN，Nv1,NvM。因为RNN中节点要计算状态，所以又通过改变状态的计算方法，又变种了LSTM，GRU，SRU等不同的网络结构，所以RNN是变化最多，也应用最宽泛的网络之一。

(4)MaxoutGRU

在上一节中介绍了Maxout网络的基本原理，接下来的介绍的MaxoutGRU[21]即是在普通的GRU网络结构上做了一些改变，其中最主要的改变就是将状态的计算过程中的than激励函数改为了Maxout激励，所示加入了Maxout层，具体公式如等式（1）所示。



等式(1)

介绍了关于深度学习的基本知识，现在可以来阐述系统所使用的三种网络结构了，这三种网络结构是系统的主要组成部分，现在将依次介绍这三种网络结构。

▪DNN

DNN是深度学习中最简单的网络结构，也是人们说的全连接神经网络，全连接即指的是每一层中的节点与上一层的节点都依次相连。一般情况下，全连接神经网络不会有太大的改变，在优化网络的时候，经常会替换不同的激励函数，以及加入Dropout来抑制过拟合，关于DNN的结构示意图如2.13所示

图2.13 DNN







输出层

隐藏层(multi)

输入层





▪RNN\_NvN

RNN\_NvN是RNN中的一种模型，正如NvN所示，这种RNN模型序列中的每一个输入都对应着一个输出，它将每个节点的输出状态传给输入层，故这种模型更加注重对每个节点的判断。







输出层

隐藏层(multi)

输入层







图2.14 RNN\_NvN

▪RNN\_Nv1

RNN\_Nv1型是指在循环神经网络的基础上，将隐藏层最后一层的最后一个一点的状态输出传给输出层，如果网络是双向的结构，则将前后的状态拼接起来传给输出层。这样的网络结构经常用于文本的情感分析，此网络可以将多个输入转化为一个输出，刚好可以满足GPS轨迹的推测方式，即推测一段GPS轨迹的出行方式。

如图2.15为RNN\_Nv1型的网络结构。







输出层

隐藏层(multi)

输入层



图2.15 RNN\_Nv1

### 3实验

# 3.1实验环境

本篇论文的实验环境都在windows8.1+python3.6+tensorflow-gpu1.4+aconda3.6上进行。选用python3.6的原因是因为python语言学习简单，灵活度高，易移植性强，而TensorFlow是谷歌退出的一款开源的深度学习框架，并且谷歌一直在维护它，在可靠性上是毋庸置疑的，以及tensorflow是采用先构建图，然后再运行的机制，所以方便新手学习，虽然目前推出了动态图的模式，但使用率还不是很高，而且静态图的方式运行速度会高一些，所以，在能够用静态图构建的情况下，使用静态图的模式还是会方便一些。aconda是集成了python与许多机器学习需要用的包，因为这些包很多都有乱七八糟的依赖，而aconda直接将所有机器学习要用的库都集成在一起，免去了许多安装库的麻烦，大大减轻了工作量。

# 3.2训练参数

本文一共实现了三种模型，分别是DNN,RNN\_NVN与RNN\_NV1。数据集使用隔5个点选取的轨迹点制作出的数据集。首先给出公共的一些模型参数，测试得到batch size设置为128是比较好的，太小会使模型训练速度过慢，误差收敛震荡，太大会导致占用内存，显存占用过高等问题，所以总体来看batch size设置为128是合理的，学习率设置为0.001，dropout的概率设置为0.5，Maxout的单元数设置为5，L2正则化参数为0.001。以及用minibatch的方式来评估模型，所以每100个batch就评估一次我们的模型。训练集：验证集：测试集 = 8：1：1。对于所有的模型，都用交叉熵作为损失函数，Adam优化器进行梯度下降。本实验主要讨论的是四分类问题，即[bike,car,walk,bus]。

# 3.3评估方式

深度学习模型的评估方式有很多种，例如：错误率与精度，查准率、查全率与F1-score，ROC与AUC，以及代价敏感错误率与代价曲线。其中错误率与精度，查准率、查全率与F1-score是经常使用的模型性能评估的方式，现将这两种评估方式介绍如下：

▪错误率与精度

错误率与精度是分类任务重最常用的两种性能度量，既适用于二分类任务，也适用于多分类任务，错误率是分类错误的样本数栈样本总数的比例，精度则是分类正确的样本数占样本总数的比例。

▪查准率、查全率与F1-score

对于而分类问题，可将样例根据其真实类别与学习器预测类别的组合划分为真正里、假正例、真反例、假反例四种情形，令TP、FP、TN、FN分别表示其对应的样例数，则显然有TP+FP+TN+FN=样例总数。分类结果的混淆矩阵如表4.1所示

|  |  |  |
| --- | --- | --- |
| 真实情况 | 预测结果 | |
| 正例 | 反例 |
| 正例 | TP | FN |
| 反例 | FP | TN |

表4.1 混淆矩阵

查准率P与查全率R分别定义为

查准率和查全率是一对矛盾的度量。一般来说，查准率高时，查全率往往偏低；而查全率高时，查准率往往偏低。所以科学家又使用一种新的度量方式即F1度量：

故本文采用精度与查准率和F1-score得分来作为模型的性能评判标准。

# 3.4实验结果与讨论

本系统的实验主要是按照三种不同的模型而分别进行实验，但总体按照DNN、RNN结构实验，RNN\_NvN实验，RNN\_Nv1实验以及三种模型的对比实验的顺序依次进行实验。

## 3.4.1 DNN实验

▪DNN网络的输入特征个数对比

如图3.1所示，在普通双层DNN模型上进行12个特征的比较测试，将12个特征分为四组数据，分别为3特征（点的速度，段的均值速度，段的速度标准差），6特征（在第一组特征的基础上加上点的加速度，段的加速度均值，段的加速度标准差），9特征（在第二组特征的基础上加上点的转向，点的转向均值，点的转向标准差）和12特征（在第三组特征的基础上加上段的最大速度，段的最大加速度，段的最大转向）。从图中可以看到当12个特征全部输入到网络的时候，网络在测

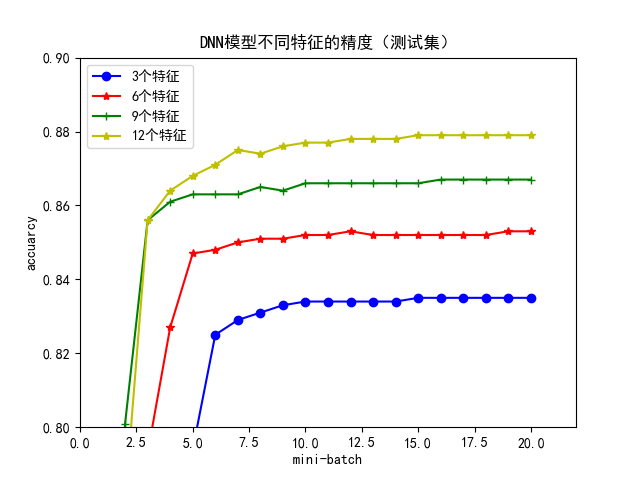


图3.1 DNN模型输入特征个数比较

试集上的精度最高，为0.87。说明这12个特征对DNN网络都有明显的增益作用，我们提取的这12个特征是有用的。

▪DNN网络特征离散化的宽度比较

在上一小节中确定了DNN网络的输入特征的个数以后，还需要对离散化的宽度选取适当的值，因为离散化的宽度是影响网络学习的关键，宽度太小，每个特征的边界会融合在一起，而宽度太大，是对时间与内存的浪费，所以必须选择一个较为适合的离散化宽度。如图3.2所示，网络采取与上一节相同的网络结构，并且根据上一节的实验结果，将网络的特征输入设为12个，当离散宽度为10，20时，网络在测试集的精度没有明显的差别，而当离散宽度为40时，并没有离散宽度选择为30时模型的精度高，所以DNN模型输入特征离散化的宽度设为30是比较合理的，即不会浪费太多的空间，也可以得到更高的精度。

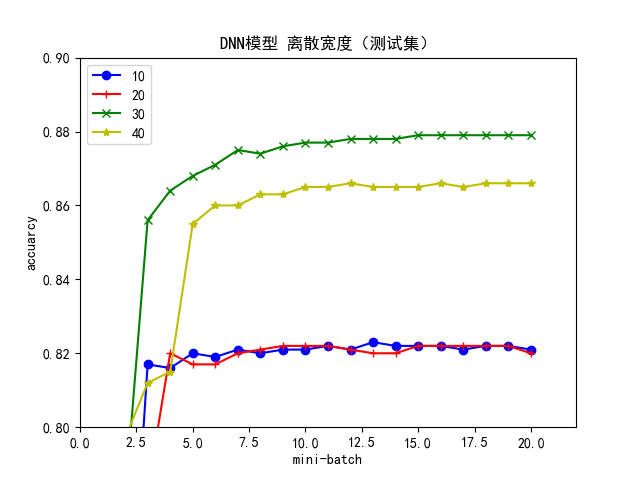


图3.2 DNN模型输入特征离散化宽度比较

▪DNN模型隐藏层层数比较

本文主要实现了普通版的DNN，带dropout的DNN，以及带Maxout的DNN。首先用普通版的DNN来确定隐藏层的数量，因为在深度学习中，隐藏层是至关重要的一层，隐藏层的个数决定了网络的性能，个数太多，会导致学习速率下降，梯度消失等等，个数太少，又造成边界模糊，学习性能欠佳。如图4.3所示，DNN模型的隐藏层个数的测试中，在前10个监测点中，隐藏层个数只有一层时学习速度最快，而当模型学习到后期时，隐藏层个数为两层时精度逐渐提高，并高于隐藏层个数只有一层和三层的时候，所以隐藏层个数为两层时，可以为模型带来最大的学习性能。

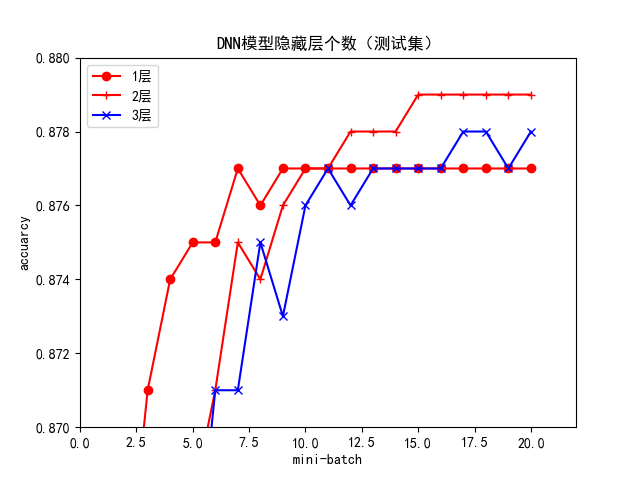


图3.3DNN模型隐藏层个数比较

▪DNN模型隐藏层节点个数比较实验

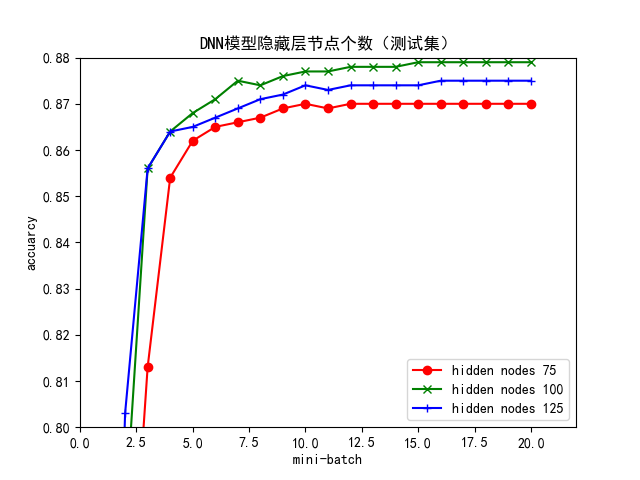


图3.4 DNN模型隐藏层节点个数

确定了隐藏层层数，接下来继续确定隐藏层节点的个数，如果隐藏层层数是宏观调控，那么隐藏层节点个数就是微观调节。而目标只有一个，就是增加模型的精度。如图3.4所示，当隐藏层节点个数为100时，模型的精度略高于隐藏层节点为75和125。所以DNN模型隐藏层节点个数设置为100是合理的。

▪DNN模型不同版本的比较

深度学习中也有很多方法改善模型的精度，例如加入dropout，maxout，以及改变激励函数等等。故本文实现了DNN带dropout与Maxout的版本，将这两种与之前的普通DNN做比较。如图4.6所示，加入dropout与maxout后并没有提升模型的性能，反而模型的精度有所下降。目前只有普通DNN模型得到了最高的精度，因而推测影响模型精度的原因主要有两点，一是特征不够，二是模型的学习能力有限。

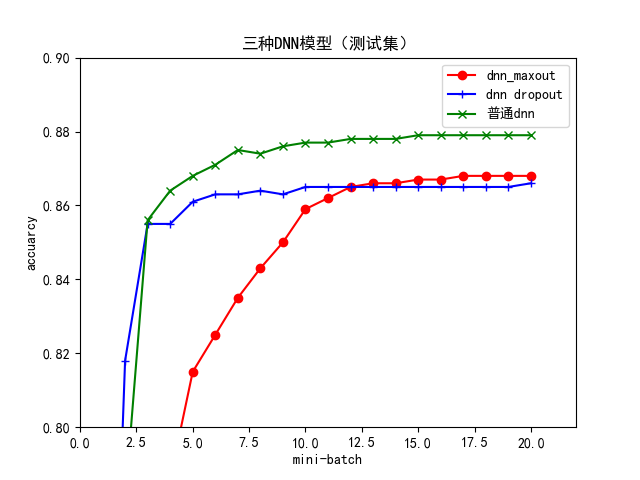


图4.6 三种DNN模型

▪DNN模型的混淆矩阵

经过以上实验，普通的双层DNN模型得到了最高的精度，此模型在运行了20个mini-batch后的混淆矩阵如表3.2所示。从混淆矩阵中可以看到步行的查全率是最高的达到了0.930，而汽车的查全率与查准率都是最低的，导致F1得分很低，只有0.780。在出行方式的推测中，我们比较看重的模型的准确率，所以在准确率方面自行车与步行拥有最高的精度0.913，而因为步行的查全率较高，所以在F1-score上的得分高于自行车，为0.921。可以看出DNN模型的精度还是略低的，主要是对汽车和公交车的分别率太低。初步猜测是因为DNN无法利用一段轨迹中的很多特征，比如速度的起伏等等，网络学习不到这些知识。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 出行方式 | 测试集 | | | | 查全率 |
| Bike | Car | Walk | Bus |
| Bike | 16197 | 0 | 1864 | 1073 | 0.847 |
| Car | 2 | 7378 | 220 | 2089 | 0.761 |
| Walk | 1209 | 50 | 28418 | 892 | 0.930 |
| Bus | 337 | 1788 | 617 | 22140 | 0.890 |
| 查准率 | 0.913 | 0.801 | 0.913 | 0.845 |  |
| F1-score | 0.879 | 0.780 | 0.921 | 0.867 |  |

表3.2 普通DNN混淆矩阵

## 3.4.2 RNN结构实验

▪RNN模型的输入特征选取

在前面DNN的实验中提到了12种特征，为了验证这12种特征是否真正的对RNN模型有增益，所以用RNN\_Nv1结构的双层双向的MaxoutGRU网络来测试这12种特征。特征的划分与DNN实验采取相同的模式，统一将这些网络都运行20个epoch,分别得到四组网络的测试集精度。如图4.7所示。

从图4.7中可以清楚的看到用不同的特征时对网络的精度影响是比较大的，在特征只有点的速度，段的速度均值，段的速度标准差时，网络在测试集上的精度只能达到接近0.85的精度；而将特征数扩大到6个时，网络的精度接近了0.9；9个特征时，网络的精度上升到了0.93；12个特征时，网络的精度还是上升到0.93就停止了。所以将所有的特征都加入到网络并没有带来最大的收益，最后的段的最大速度，段的最大加速度，段的最大转向对RNN结构的网络训练并没有很明显的帮助，反而会造成输入向量过大，训练速度变慢等种种原因，故本文决定在RNN模型中只采用前9个特征来做为模型的输入，暂时舍弃段的最大速度、段的最大加速度、以及段的最大转向。

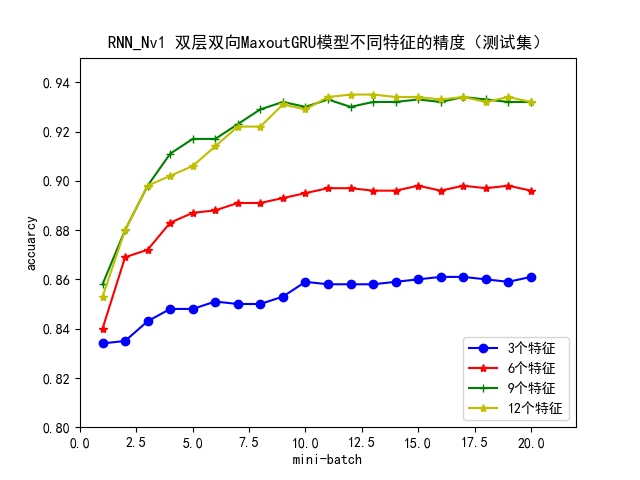


图3.7 RNN网络不同特征个数测试集精度

▪RNN模型输入特征离散化宽度比较

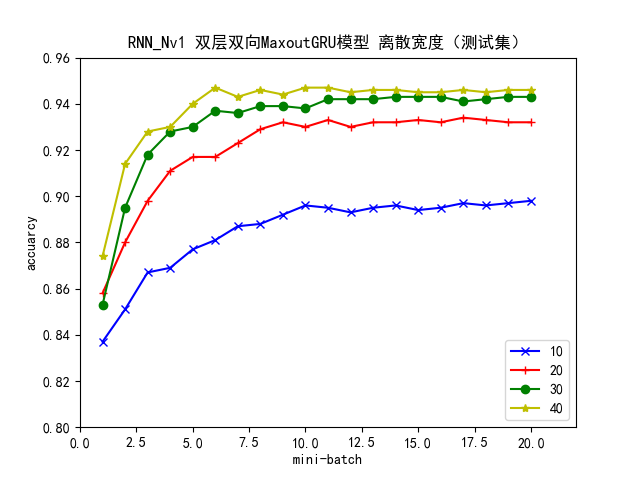


图3.8特征离散化宽度对网络的影响

在上一小节中确定了特征的个数以后，还需要对离散化的宽度选取适当的值，因为离散化的宽度是影响网络学习的关键，宽度太小，每个特征的边界会融合在一起，而宽度太大，是对时间与内存的浪费，所以必须选择一个较为适合的离散化宽度。如图3.7所示，网络采取与上一小节相同的网络结构，并且根据上一小节的实验结果，此网络的特征输入为9个，当离散宽度为10，20，30时，网络在测试集的精度有明显的差别，而当离散宽度为40时，增大离散宽度带来的收益效果并没有显著的提高，所以，离散宽度选择为30是比较合理的，即不会浪费太多的空间，也不会导致模型的精度下降太多。

▪RNN模型网络结构比较

因为在RNN模型中有几种不同的隐藏层结构的变体，所以本文主要实现了三种结构，分别为LSTM，GRU，以及MaxoutGRU。这三种网络都采用双层双向的结构，实验结果如图3.9所示，带有Maxout层的GRU结构具有最高的精度0.94，并且远远领先普通GRU与LSTM，说明Maxout的拟合能力确实很强。故本文的RNN实验采用MaxoutGRU结构。

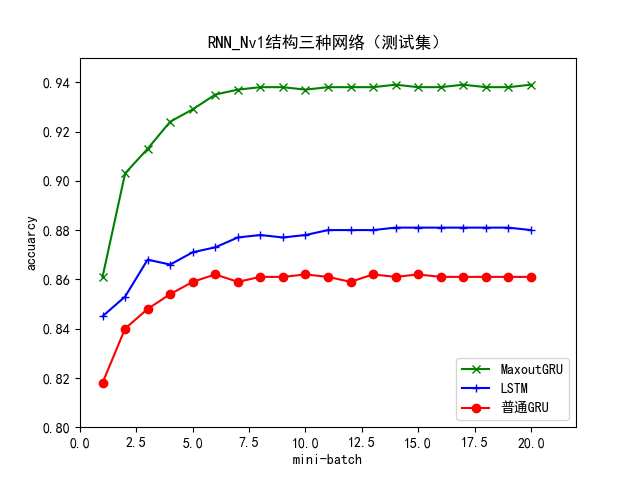


图3.9 RNN\_Nv1三种网络比较

## 3.4.3 RNN\_NvN 实验

▪RNN\_NvN结构单双向实验

在上一节确定了RNN模型所使用的网络结构以后，再进行不同模型的细节比较实验，所以在RNN\_NvN结构中先进行单双向比较的实验。如图3.10所示，双向MaxoutGRU模型的推理能力比单向的要高出一点，所以双向网络对于轨迹的推测还是有一定的帮助的。

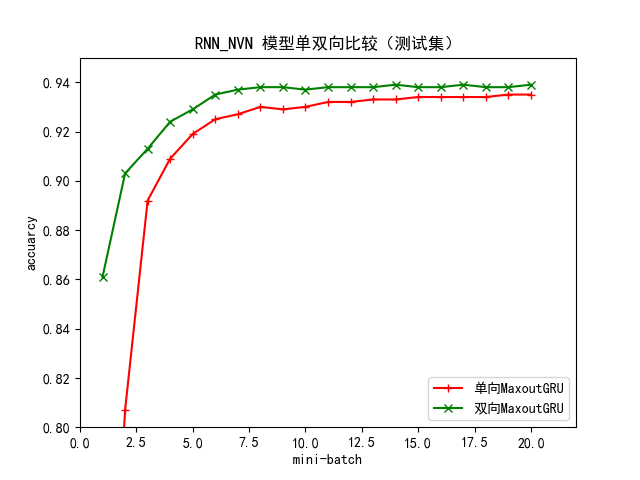


图3.10 RNN\_NvN MaxoutGRU 单双向比较

▪RNN\_NvN隐藏层层数实验

隐藏层层数实验如图3.11所示，双层的MaxoutGRU模型的精度明显高于一层与三层的模型。

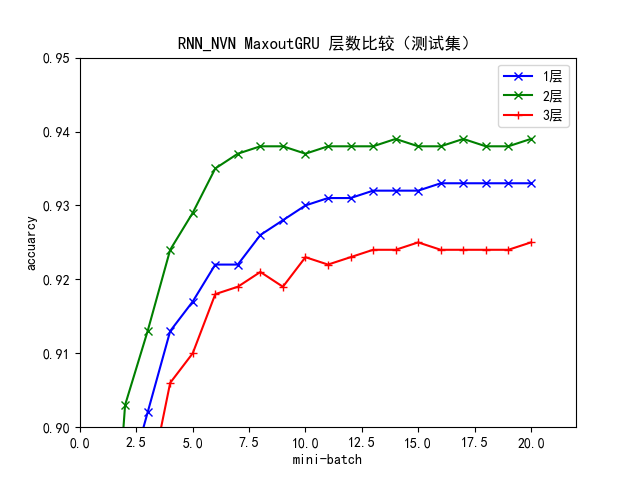
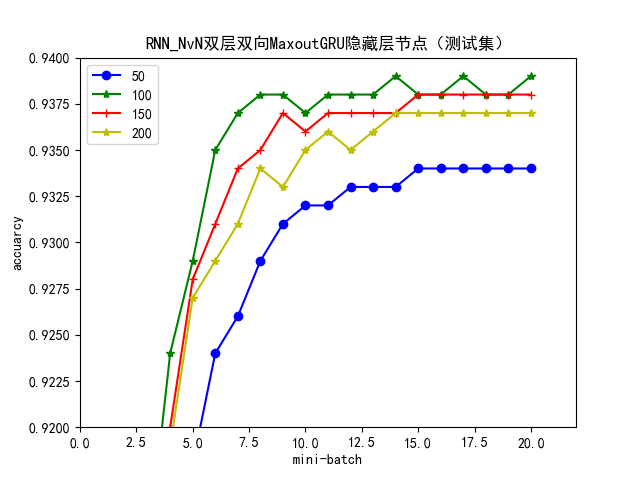


图3.11 RNN\_NvN MaxoutGRU 层数比较

▪RNN\_NvN隐藏层节点个数实验

最后进行隐藏层节点的测试，如图3.12为双层双向MaxoutGRU的隐藏层节点的测试，实验测试了网络隐藏层节点分别为50，100，150，200时的情况。可以看出当隐藏层节点个数为100时，拥有最高的精度，且参数要少于150与200时的模型。所以隐藏层节点个数设置为100。

图3.12 双层双向MaxoutGRU隐藏层节点个数



▪RNN\_NvN实验混淆矩阵

经过以上的实验，在RNN\_NvN中可以确定隐藏层节点个数为100的双层双向MaxoutGRU模型具有最高的精度：0.939。表4.3为此模型的混淆矩阵。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 出行方式 | 混淆矩阵 | | | | 查全率 |
| Bike | Car | Walk | Bus |
| Bike | 18168 | 61 | 429 | 180 | 0.964 |
| Car | 79 | 8771 | 0 | 652 | 0.923 |
| Walk | 651 | 33 | 28537 | 435 | 0.962 |
| Bus | 198 | 377 | 391 | 21735 | 0.957 |
| 查准率 | 0.951 | 0.949 | 0.972 | 0.945 |  |
| F1-sorce | 0.957 | 0.936 | 0.967 | 0.951 |  |

表4.3 RNN\_NvN双层双向MaxoutGRU混淆矩阵

从表4.2中可以看到RNN\_NvN模型的双层双向MaxoutGRU的精度比DNN提升了很大一截，因为从输入输出来讲，DNN与RNN\_NvN都是一个输入点，一个标签相对应，所以证明MaxoutGRU结构还是有一定的帮助的。在混淆矩阵中，步行拥有着最高的准确率0.972，而且拥有着最高的F1-score得分，说明步行的特征比较明显。从表中可以看到汽车的推测准确率有着非常大的提升，达到了0.949，F1-score也达到了0.936，证明MaxoutGRU对于段的特征有一定的学习能力。

## 3.4.4 RNN\_Nv1实验

▪RNN\_Nv1结构单双向实验

如图3.13是MaxoutGRU的单双向测试，从图中可以看到在这个模型中，单双向网络的差别已经越来越小，甚至单向网络追赶上了双向网络的学习能力。但在前20个mini-batch 双向网络的学习能力还是强于单向网络，所以此后的实验暂使用双向网络进行试验。

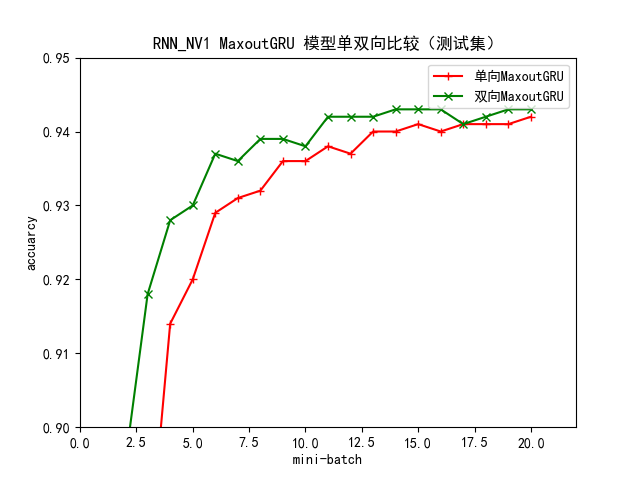


图3.13 RNN\_Nv1 MaxoutGRU 单双向比较

▪RNN\_Nv1隐藏层层数实验

如图3.14所示，MaxoutGRU模型的隐藏层层数的实验中，2层的隐藏层层数的模型的精度微微高于1层和3层的模型，所以决定此网络使用2层的隐藏层。

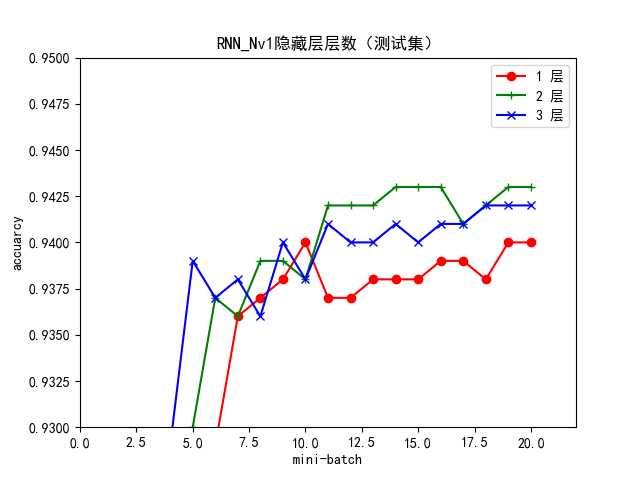


图3.14 RNN\_NvMaxoutGRU隐藏层个数

▪RNN\_Nv1隐藏层节点个数实验

如图3.15所示，隐藏层节点的个数为100或150时，模型的精度较为接近，且个数为100时略精度略高于个数为150时，而个数为50时，精度相差甚远，所以选择隐藏层的节点个数为100。

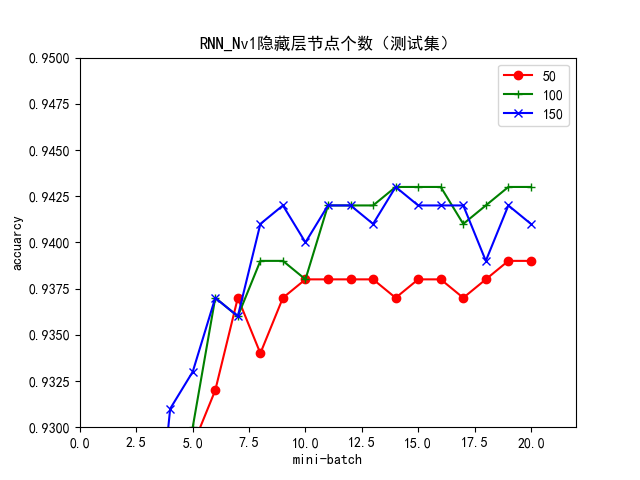


图3.15 RNN\_Nv1隐藏层节点个数比较

▪RNN\_Nv1混淆矩阵

表4.4为RNN\_Nv1结构双层双向MaxoutGRU模型的前20个minibatch的混淆矩阵。共有2048条轨迹，其中自行车462条，汽车200条，步行801条，公交车546条。可以看到在RNN\_Nv1的结构中步行依旧有着最高的准确率，说明步行的轨迹段特征清晰，而汽车的准确率依旧是最低，汽车依旧与公交车有着很高的推测错误率，追究原因可能是因为汽车与公交车的相似度过高，以及轨迹分段的不合理，因为不好的轨迹分段会导致将轨迹中拥有重要特征的段被分离出去。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 出行方式 | 混淆矩阵 | | | | 查全率 |
| Bike | Car | Walk | Bus |
| Bike | 432 | 3 | 16 | 11 | 0.935 |
| Car | 1 | 187 | 1 | 11 | 0.935 |
| Walk | 23 | 1 | 767 | 10 | 0.957 |
| Bus | 9 | 14 | 16 | 546 | 0.933 |
| 查准率 | 0.929 | 0.912 | 0.959 | 0.945 | 0.943 |
| F1-score | 0.932 | 0.923 | 0.958 | 0.939 |  |

表4.4 RNN\_Nv1 MaxoutGRU混淆矩阵

## 3.4.5 三种模型比较

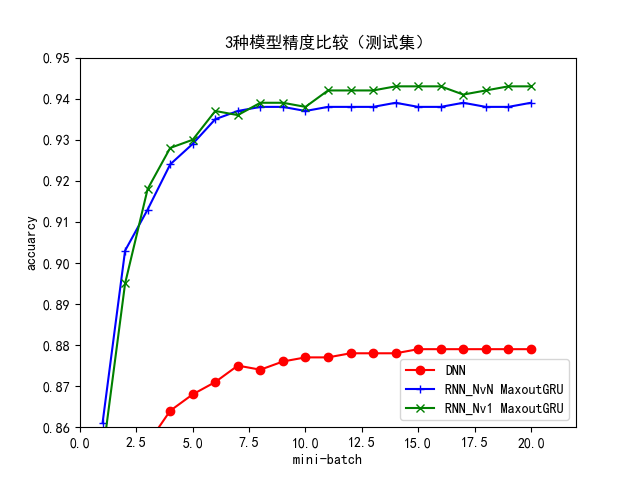


图3.16 三种模型比较

如图3.16所示，在小数据集的测试集上，RNN\_NvN模型与RNN\_Nv1模型拥有着差不多的精度，所以在不同的推测情况下可以选取不同的模型。而DNN模型的精度只有不到0.88，但值得一提的是，DNN模型的训练速度比RNN模型要快很多倍，所以如果能够为DNN模型找到更准确的特征，那么DNN模型将会是我们的首选模型。

### 4总结与未来展望

在本篇文章中主要研究通过GPS轨迹来推测人类的四种出行方式，我们通过对GPS轨迹的分段，GPS轨迹的12种特征的提取，对12种特征进行噪声过滤，归一化，离散化之后转换成深度学习模型的输入。构建了三种不同的网络结构：DNN，RNN\_NvN和RNN\_Nv1，通过大量的实验，我们发现普通版的双层DNN拥有最高的准确率：0.88，以及在RNN中，双层双向的MaxoutGRU也有着最高的准确率，0.945 ，并且在这三种模型中，DNN与RNN\_NvN模型都是推测一个一个的GPS轨迹点的出行方式，而RNN\_Nv1模型是直接推测一段GPS轨迹的出行方式，所以在推测GPS轨迹点的出行方式的时候可以选择RNN\_NvN模型，而在推测GPS轨迹段的出行方式的时候可以选择RNN\_Nv1模型，因为两种模型有着非常接近的精度。我们大费周章的将原始的GPS轨迹一步一步的抽象，剥离，筛选，最终制作成为我们深度学习模型的输入，为我们的模型寻找到了最合适的特征，最合适的参数，只为了能够将模型的精度提高一点点，但我们相信这一切都是值得的。

通过GPS轨迹推测人类的出行方式的研究还并没有结束，GPS轨迹中还存在着大量的未被挖掘出来的特征，通过在我们的模型试验的过程中，我们发现调整参数不如发现一个好的特征，只有寻找到一个好的特征才能够事半功倍。

### 参考文献

1. Yoshua Bengio. 2012. Practical recommendations for gradient-based training of deep architectures. In Neural Networks: Tricks of the Trade. Spri nger,437--478.
2. Y. Bengio , P. Simard , P. Frasconi, Learning long-term dependencies wit-h gradient descent is difficult, IEEE Transactions on Neural Networks,v.5 n.2, p.157-166, March 1994  [doi>10.1109/72.279181] 5
3. Young-Ji Byon, Baher Abdulhai, and AmerShalaby. 2009. Real-time transportation mode detection via tracking global positioning system mobile devices. Journal of Intelligent Transportation Systems 13, 4 (2009), 161--170. 6
4. Erico N de Souza, Johna Latouf, Jiang Xiang, and Stan Matwin. 2017. An Efficient Algorithm to Predict Fishing Activity. To Appear (2017). 10
5. Somayeh Dodge, Robert Weibel, and Ehsan Forootan. 2009. Revealing the physics of movement: Comparing the similarity of movement characteristics of different types of moving objects. Computers, Environment and Urban Systems 33, 6 (2009), 419--434. 12
6. Fosca Giannotti , Mirco Nanni , Fabio Pinelli , Dino Pedreschi, Trajectory pattern mining, Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, August 12-15, 2007, San Jose, California, USA  [doi>10.1145/1281192.1281230] 15
7. Paola A Gonzalez, Jeremy S Weinstein, Sean J Barbeau, Miguel A Labrador, Philip L Winters, Nevine L Georggi, and R Pérez. 2010. Automatin g mode detection for travel behaviour analysis by using global positioning systems-enabled mobile phones and neural networks. IET Intelligent Transport Systems 4, 1 (2010), 37--49. 16
8. Ian J. Goodfellow , David Warde-Farley , Mehdi Mirza , Aaron Courville , Yoshua Bengio, Maxout networks, Proceedings of the 30th International Conference on International Conference on Machine Learning, June 16-21, 2013, Atlanta, GA, USA 17
9. Sepp Hochreiter, The vanishing gradient problem during learning recurrent neural nets and problem solutions, International Journal of Uncertainty, Fuzziness and Knowledge-BasedSystems,v.6n.2,p.107-116,April 1998  [doi>10.1142/S0218488598000094] 22
10. Xiang Jiang, Xuan Liu, Erico N de Souza, Baifan Hu, Daniel L. Silver, and Stan Matwin. 2017. Improving Point-based AIS Trajectory Classification with Partition-wise Gated Recurrent Units. In Neural Networks (IJCNN), 2017 International Joint Conference on. IEEE. 25
11. Lin Liao, Dieter Fox, and Henry Kautz. 2006. Location-based activity recognition.Advances in Neural Information Processing Systems 18 (2006), 787. 30
12. Corrado Loglisci, Donato Malerba, and Apostolos N Papadopoulos. 2014. Mining Trajectory Data for Discovering Communities of Moving Objects. In EDBT/ICDT Workshops. 301--308. 31
13. Emiliano Miluzzo , Nicholas D. Lane , Kristóf Fodor , Ronald Peterson , Hong Lu , Mirco Musolesi , Shane B. Eisenman , Xiao Zheng , Andrew T.Campbell, Sensing meets mobile social networks: the design, implementation and evaluation of the CenceMe application, Proceedings of the 6th ACM conference on Embedded network sensor systems, November 05-07, 2008, Raleigh, NC, USA  [doi>10.1145/1460412.1460445] 32
14. Mirco Nanni , Dino Pedreschi, Time-focused clustering of trajectories of movingobjects, Journal of Intelligent Information Systems, v.27 n.3, p.267-289, November 2006  [doi>10.1007/s10844-006-9953-7] 33
15. Amílcar Soares Júnior , Bruno Neiva Moreno , Valéria Cesário Times , Sta n Matwin , Lucídio dos Anjos Formiga Cabral, GRASP-UTS: an algorithm for unsupervised trajectory segmentation, International Journal of Geographic al Information Science, v.29 n.1, p.46-68, January 2015  [doi>10.1080/13658 816.2014.938078] 41
16. Leon Stenneth , Ouri Wolfson , Philip S. Yu , Bo Xu, Transportation mode detection using mobile phones and GIS information, Proceedings of the 1 9th ACM SIGSPATIAL International Conference on Advances in Geogr ph ic Information Systems, November 01-04, 2011, Chicago, Illinois   [doi> 10.1145/2093973.2093982] 42
17. Yu Zheng, Trajectory Data Mining: An Overview, ACM Transactions on Intelligent Systems and Technology (TIST), v.6 n.3, May 2015  [doi>10.1 145/2743025]
18. Yu Zheng , Licia Capra , Ouri Wolfson , Hai Yang, Urban Computing: Concepts, Methodologies, and Applications, ACM Transactions on Intelligent Systems and Technology (TIST), v.5 n.3, p.1-55, September 2014  [doi>10.1145/2629592]
19. Yu Zheng , Quannan Li , Yukun Chen , Xing Xie , Wei-Ying Ma, Understanding mobility based on GPS data, Proceedings of the 10th international conference on Ubiquitous computing, September 21-24, 2008, Seoul, Korea  [doi>10.1145/1409635.1409677]
20. Yu Zheng , Like Liu , Longhao Wang , Xing Xie, Learning transportation mode from raw gps data for geographic applications on the web, Proceedings of the 17th international conference on World Wide Web, April 21-25, 2008, Beijing, China  [doi>10.1145/1367497.1367532]
21. Jiang X, De Souza E N, Pesaranghader A, et al. TrajectoryNet: An Embedded GPS Trajectory Representation for Point-based Classification Using Recurrent Neural Networks[J]. 2017.
22. Endo Y, Toda H, Nishida K, et al. Deep Feature Extraction from Trajectories for Transportation Mode Estimation[C]// Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Cham, 2016:54-66.
23. Frank R Hampel. 1974. The influence curve and its role in robust estimation.*J. Amer. Statist. Assoc.* 69, 346 (1974), 383--393.
24. Dakun,Yang[a],Yan等.L1/2 regularization learning for smoothing interval neural networks: Algorithms and convergence analysis[J].Neurocomputing,2018,272:122-129.
25. Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: A simple way to prevent neural networks from overfitting[J]. The Journal of Machine Learning Research, 2014, 15(1): 1929-1958.
26. 张治华. 基于GPS轨迹的出行信息提取研究[D].华东师范大学,2010.
27. 沈云. 基于GPS轨迹数据的交通出行方式识别研究[D].北京交通大学,2017.
28. 杨飞, 姚振兴. 基于GPS定位的交通方式组合出行的计算机识别方法:, CN102799897A[P]. 2012.