西安交通大學

毕业设计（论文）

题　目 基于变分自编码器模型的
轨迹用户链接问题研究

电信 学院 计算机科学与技术 专业 计算机 65 班

学生姓名 王浩辰

学　　号 2161700108

指导教师 孙鹤立

设计所在单位

2020 年 6 月

摘　要

城市计算是近年来新颖且充满应用价值的研究领域，其中，基于地理信息的服务是该领域的一个重要课题，课题中许多重要的下游任务，例如用户的下一个兴趣点推荐、个性化行程推荐等，都需要进行用户的移动模式分析。因此轨迹用户链接是城市计算中的关键环节，也在基于地理信息应用中是一项非常重要的任务。现有对移动模式挖掘的技术主要可以概括为两种，其一是隐马尔科夫模型，另一种是使用循环神经网络对轨迹数据建模。以上两种建模方式，前者基于非相邻位置之间具有独立性的假设，后者则仅研究一种较为浅层的生成过程，然而都忽视了人类轨迹数据通常本身具有稀疏的、高维度的特性，以及可能包含嵌入式层次结构的事实。本文采用基于变分自编码器模型的半监督学习框架解决轨迹用户链接问题，通过使用循环神经网络的隐藏单元结构，随机生成隐式变量来学习人类移动特征。利用变分自编码器模型结合半监督学习方法，借助大规模无标签的数据缓解了数据稀疏性的问题，同时利用高维度的嵌入向量来表示轨迹数据的层次和结构语义。实验验证了在真实的数据上，本文采取的变分自编码器模型同比现有方法，在解决轨迹用户链接问题时提升了性能效率，并且提高了用户轨迹链接的准确度。

关 键 词：深度学习；变分自编码器；地理轨迹信息；城市计算

ABSTRACT

In recent years, urban computing is a new research field, which has been implemented in a number of applications. Among them, geographic information-based services are an important topic in this field. There are many important further tasks in this topic, such as the user's next point of interest (POI) recommendation and personalized itinerary recommendation, and so on. All need to analyze the user's mobile pattern. Therefore, the trajectory user linking (TUL) is a key part of urban computing, and a very important task in applications based on geographic information. Existing techniques for mining mobile patterns can be summarized into two types, one is the hidden Markov model, and the other is to model trajectory data using recurrent neural networks. The above two modeling methods, the former is based on the assumption that non-adjacent positions are independent, while the latter only studies a shallow generation process. However, both of them ignore that human trajectory data is usually featured with sparse distribution and high dimension. Besides, the fact that trajectory data may contain embedded hierarchies is neglected. In this paper, a semi-supervised learning framework based on the Variational AutoEncoder (VAE) model is used to solve the trajectory user linking problem. By using the hidden unit structure of the recurrent neural networks, implicit variables are randomly generated to learn features of human movement. Using the Variational AutoEncoder model combined with semi-supervised learning methods, the problem of data sparsity is alleviated with the help of large-scale unlabeled data. High-level embedding vectors are used to represent the trajectory data hierarchy and structural semantics. Experiments have verified that on real data, the Variational AutoEncoder model adopted in this paper is comparable to the existing method, which improves the performance efficiency when solving the problem of trajectory user linking, and improves the accuracy of user trajectory linking.

KEY WORDS: Deep learning; Variational AutoEncoder; Trajectory; Urban computing

目　录

1 绪论.....5
2 轨迹用户链接问题.....7
2.1 问题定义.....7
2.2 解决方案研究现状.....7
2.2.1 基于隐马尔科夫模型的方法.....7
2.2.2 基于循环神经网络的方法.....8
3 变分自编码器模型.....10
3.1 概述.....10
3.2 数学基础.....10
3.2.1 变分自编码器定义.....10
3.2.2 两个核心问题.....11
3.3 模型结构.....11
3.3.1 构造目标函数.....11
3.3.2 优化目标函数.....12
3.3.3 变分自编码器模型.....13
4 基于变分自编码器模型对轨迹用户链接.....15
4.1 概览.....15
4.1.1 轨迹用户链接整体思路结构.....15
4.1.2 基于变分自编码器轨迹用户链接模型结构.....16
4.2 数据预处理.....17
4.2.1 数据清洗.....17
4.2.2 轨迹数据分段.....18
4.3 轨迹数据编码.....19
4.3.1 轨迹数据嵌入（Embedding）.....19
4.3.2 轨迹数据编码.....19
4.4 使用变分自编码器链接轨迹用户.....23
5 实验结果与分析.....27
5.1 数据集说明.....27
5.2 模型训练.....27
5.2.1 模型设置与参数.....27
5.2.2 训练技术参数.....28
5.3 测试评估与性能比较.....28
6 总结与展望.....33
7 致　谢.....34
8 参考文献.....35

1 绪论

随着移动智能终端的普及与使用场景的扩展，越来越多的便携的、可获取设备定位信息的、可记录人类运动轨迹的智能设备，例如：智能手机、智能手表、运动手环、行车记录仪等，出现在人们的生活中。也就是说，人们生活中产生的轨迹信息被很容易的也很广泛的被记录下来。与此同时，城市计算在大数据技术的发展和推动下开始发挥着强大能力，可以为一座城市的管理和相关企业的策略提供了有效且可靠的参考建议。城市计算也可以看作是一种基于地理信息的服务，承担着许多重要任务，其中将轨迹数据和产生它的用户链接起来是关键环节。

然而，出于对用户隐私的保护的考虑，轨迹与用户的链接关系通常是被隐藏或者被清洗掉的，获得的轨迹数据往往是没有用户标签的位置序列。因此，解决轨迹用户链接问题是基于地理信息来提供服务的应用的重要任务之一。与用户链接后的轨迹数据可以产生更好、更个性化、更准确的兴趣点（POI，Point of Interest）推荐，也可以帮助公安部门从过往旅客轨迹的稀疏时空数据中识别恐怖分子或犯罪嫌疑人。

解决轨迹用户链接问题现已有的方法主要是依靠隐马尔科夫模型或者是循环神经网络（RNN, Recurrent Neural Network）基于用户的历史签到信息对人类运动模式建模。基于隐马尔科夫模型的方法[1]，例如：基于排序的隐马尔科夫模型和基于排序的马尔科夫链的转换，假设在非相邻的位置间存在强独立性，失去了轨迹信息（POI 序列）对时间的依赖性，忽视了人类运动轨迹具有的时序性和符合生活实际的语义，因此限制了捕获长期定位轨迹信息的性能。

另一方面，循环神经网络是一种能够解决变长输入和输出序列的特殊神经网络。基于循环神经网络的用户轨迹链接解决方案，是根据先前的输出，通过对兴趣点序列进行联合概率分布建模，它可以预测下一个输出序列。这被成功地运用在许多时空位置预测的应用中，且近来被用作于轨迹用户链接中识别人类运动模式的建模中[2]。遗憾的是，标准循环神经网络的监督学习轨迹模型缺乏对人类运动的层次语义的理解，并且无法利用未标记的数据进行深度学习，浪费了未标记数据中丰富且独特的个体运动模式特征。

再者，诸如生成对抗网络（GAN，Generative Adversarial Network），自回归模型和变分自编码器（VAE，Variational Autoencoder）之类的深层生成模型，已经在计算机视觉领域的图像识别、图像生成和自然语言处理领域的语义语句建模、机器翻译等方面取得了成功，也受到行业内的认可，并已运用在实际应用中。近期，半监督学习的生成模型受到越来越多的关注，它利用已标签的数据和未标签的数据对复杂的高维度潜在变量进行建模，同时可进行分类和生成任务。变分自编码器在自然语言处理领域中文本分类和语言生成应用上展现出令人鼓舞的性

能，但是，VAE 在人类运动建模中的应用尚未得到充分的研究。并且，受限于数据稀疏性和轨迹序列数据中潜在的语义结构，以前基于变分自编码器的模型无法直接应用于解决轨迹用户链接问题。

在本文中，采用新颖的方法——基于变分自编码器的轨迹用户连接方法（TULVAE，Trajectory User Linking via Variational AutoEncoder），对人类轨迹数据建模，并且将轨迹数据与产生其的用户链接起来，以解决上述存在并提出的问题，以满足目前基于地理信息的应用中的轨迹数据与其用户的链接需求，进而更好地使城市计算进一步地为公众和社会提供高质量、高效率、高可靠性的服务。

本文的主要工作包括如下的四点:

（1） 提出一种基于变分自编码器的优化目标函数建模方法，并利用该模型推断出人类运动模式中的潜在变量，据调查研究所知，本文提出的方法是目前为数不多的变化轨迹数据推理模型，以及为带有时序的轨迹数据挖掘提供了新视角。

（2） 探索并尝试解决基于地理信息应用中轨迹数据稀疏性问题，特别是使用半监督学习将未标签轨迹数据链接到产生其的用户，以提高轨迹用户链接的性能，并使数据得到尽可能完全地利用，充分挖掘有效数据，最大化获取到数据的价值。

（3） 通过轨迹数据中人类活动轨迹的分段，对分段的轨迹内和轨迹间所包含的语义建模，弥补了以往研究中忽略轨迹数据层次结构的不足，解决了轨迹间过于复杂了相互依赖性问题。

（4） 在实验评估中，通过使用三个公开的基于地理信息的社交媒体数据集测试了模型的性能，并将本文中的模型与几种现有模型进行比较，实验结果表明，本文提出的基于变分自编码器的用户轨迹链接模型的方案对于问题的解决所带来的性能和效率的提升。

在本文的后续部分，将在第 2 部分论述轨迹用户连接问题的数学定义，并详细说明现有研究针对该问题给出的解决方案以及问题解决的现状。在第 3 部分讨论本文运用的核心结构——变分自编码器模型的数学原理和模型细节。在第 4 部分介绍并分析基于变分自编码器的轨迹用户链接模型的技术细节，然后再第 5 部分中给出数据集、实验环境说明，可视化的实验结果展示，并将本文使用模型的问题解决效果和其他的轨迹用户链接问题的解决方案效果进行横向比较与评估。第 6 和 7 部分给出本文的结论和对未来研究的展望。

2 轨迹用户链接问题

2.1 问题定义

设 $t_{u_i} = \{c_{i1}, c_{i2}, \dots, c_{in}\}$ 表示用户 u_i 在一段时间内产生并被记录的长度为 n的移动轨迹数据，其中 $c_{ij}(j \in [1,n])$ 是 t_j 时刻用户 u_i 所处的位置（该位置包含很多信息例如：用经度与纬度表示的地理信息，经度与纬度，形如(x_{ij}, y_{ij})）。在本文中，我们将 c_{ij} 称为一个签到点。一条轨迹数据 $\overline{t_i} = \{c_1, c_2, \dots, c_m\}$，若我们不知道是由哪个特定的用户产生，就被称为一条未链接的轨迹数据。于是，轨迹用户链接问题被定义为：假设有许多未链接的轨迹数据 $\mathcal{T} = \{\overline{t_1}, \dots, \overline{t_m}\}$，这些轨迹数据均由一组确定的用户 $\mathcal{U} = \{u_1, \dots, u_n\}$ $(m \gg n)$ 产生。轨迹用户链接的目标是学习一种将未链接的轨迹数据与最可能是产生该轨迹的用户进行链接的映射函数，即： $\mathcal{T} \mapsto \mathcal{U}$。

2.2 解决方案研究现状

收集人类的行为数据，使用计算机技术从数据中挖掘人类的行为模式是人工智能、地理信息系统和推荐系统中共同的一个研究主题，并且轨迹分类，也就是将大量轨迹分类为不同的运动模式，是理解人类运动模式的中心任务之一。轨迹用户链接是比轨迹分类更要复杂的一项问题，因为需要分类的类别数量更多，需要将一条轨迹具体分配到某一个用户。现有的，对人类轨迹数据建模的方法主要使用的模型包括隐马尔科夫模型或者是循环神经网络模型，模型通过学习大量的用户历史签到信息，学习得到用户的移动行为模式。

2.2.1 基于隐马尔科夫模型的方法

隐马尔科夫模型（HMM，Hidden Markov Model）是一种可用于分析带有时间标签的序列数据的技术，是一种随机状态机。HMM 在研究领域和应用领域已经非常成熟。自 2005 年至今，有许多研究文献表明，引入隐马尔科夫模型的识别、分类算法，可以以较高的识别正确率（84%~96%）解决轨迹分类问题[3]，但是对轨迹用户链接任务并没有很好的效果。

根据实际场景中运动轨迹的复杂程度，针对轨迹的类别或产生轨迹的用户建立相应的隐马尔科夫模型，利用数据集训练上述模型来获得可靠的参数。计算并且统计测试样本中的轨迹对于每个 HMM 模型的最大似然概率，比较选择概率值最大的分类（用户）作为轨迹用户链接的结果。

根据轨迹数据的复杂性，隐马尔科夫模型的状态数选取通常为 4，5 或者 6 不等，模型的宏观结构为从左至右且中间没有状态的跳转，初始化开始状态均为初始，终止于末状态。多选用连续混合高斯概率密度函数作为模型的输出似然概

率函数，对每个混合状态单元的概率加权计算，可以起到平滑模型参数的作用，以获得较高的识别率。为了应对轨迹数据稀疏和训练数据不足的问题，可利用多观测序列算法来训练各个隐马尔可夫模型。

对于各个状态的混合高斯概率密度函数中均值矢量、混合权系数和协方差矩阵的初始化都使用 K-means 算法[3]。首先，依据先前选定的状态数，对该数据做平均分段，即将训练集中的轨迹看做预处理后的点集，使用 K-means 算法聚类，其中聚类簇的个数（K 值）等于隐马尔科夫模型的状态数量。然后，将观测序列中属于同一个状态的数据合并组成一个矩阵，再根据混合单元的数量，利用 K-means 算法得到各个状态中的均值矢量、混合权系数和协方差矩阵。最后，可以得到一个具有确定数量状态的连续密度隐马尔科夫模型的参数初始值。

对于轨迹数据的隐马尔科夫模型训练，给定用户的训练轨迹数据样本集，包含若干观测值序列，开始迭代计算：前向变量和后向变量，利用多观测值序列重估公式重新估算模型参数，最后使用条件概率和最大似然函数评价模型是否收敛，若收敛则结束对此轨迹样本的训练，得到该用户的隐马尔科夫模型，否则进行下一次迭代。

基于 HMM 的轨迹用户链接，其实质上是在上述已训练好的隐马尔科夫模型中选择一个可以最佳地描述观测轨迹序列的模型，即：计算测试轨迹数据中相对于各个训练完成 HMM 的条件概率，在得到的条件概率中选取概率值最大的用户类作为该测试轨迹序列的用户链接结果[3]。

然而，HMM 假设在非相邻的位置间存在强独立性，失去了兴趣点对时间的依赖性，对于轨迹数据生硬地平均分段，忽视了人类活动本身具有的时序性和符合生活实际的语义，因此基于隐马尔科夫模型的方法限制了捕获长期定位轨迹信息的性能和效果。

2.2.2 基于循环神经网络的方法

循环神经网络是可以处理任意长度序列的神经网络。对于任意时间步长 t，它都会馈入输入 x_t 并从先前的隐藏状态 h_{t-1} 生成隐藏状态 $h_t = \varphi(x_t, h_{t-1})$，其中 φ 是非线性函数。通过递归展开 h_t，将得到：

$$h_t = \varphi\left(x_t, \varphi\left(x_{t-1}, \varphi\left(x_{t-2}, \varphi(\dots)\right)\right)\right) = f(x_{1:t}) \qquad (2-1)$$

表明循环神经网络的隐藏状态是所有过去输入 $x_{1:t}$ 的函数。

循环神经网络作为深层模型，已有文献[4]表明，它具有对变长序列分布建模的强大优势，例如长短期记忆模型（LSTM，Long-Short Term Memory）可以很好地捕获一段时间内兴趣点序列的独立性特征。LSTM 和门控循环单元（GRU，Gated Recurrent Unit），通过引入循环神经网络的门控机制，来解决梯度消失和梯

度爆炸问题，它比隐式马尔科夫模型等浅层序列模型更强大，因此循环神经网络在对序列的建模中很受欢迎。

使用 LSTM 对轨迹数据建模[5]，通过类似上文提到的生成状态函数，可把记忆单元分别表示终止状态、当前状态、候选嵌入状态，用 Sigmoid 函数作为激活函数，对应地定义输入门、遗忘门及输出门。类似地，使用 GRU 对轨迹数据建模，不单独引入记忆单元，可简化 LSTM 模型。形式上，我们通过最后一个状态 h_{t-1} 与候选嵌入状态 $\widetilde{h_t}$ 之间的线性插值来更新 h_t 的状态，候选嵌入状态 $\widetilde{h_t}$ 的计算与传统循环神经网络单元相似。为了最终将轨迹链接到产生他们的用户，由 LSTM 或者 GRU 生成的轨迹序列表示被传入 Softmax 层，进行多类别的分类，最终解决轨迹用户链接问题。

从近年来的文献来看，有研究团队[6]将循环神经网络引入来解决轨迹用户链接问题，通过使用基于循环神经网络的模型来学习人类移动特征，并按照用户对轨迹进行分类，从而将未链接的轨迹与产生其的用户相关联，这一思路是可行的。但令人失望的是，基于标准循环神经网络的监督性学习轨迹模型缺乏对人类活动分层语义的理解，且无法利用未标记的数据进行深度学习。轨迹恢复问题也以类似的方式进行了研究，使用轨迹统计推断个人的身份，其本质上是等效的轨迹用户链接问题。

更进一步的，直接应用循环神经网络对从基于地理信息的应用所获得的签到序列建模至少存在以下三种缺陷：

（1） 轨迹数据严重的稀疏性：例如从 Foursquare 和 Yelp 获取签到数据的密度通常大约是 0.1%，而 Gowalla 数据的签到密度约为 0.04%[7]。并且，数据集中的签到数据可能同时包括一些无用数据（噪音），会直接影响轨迹用户链接的准确率；

（2） 轨迹数据中结构间的依赖：由于时间步长选取的不同，在签到数据中存在强而复杂的依赖[8]；

（3） 简单循环神经网络的浅层生成：只有顺利通过输出门的生成序列信息（例如：生成兴趣点序列）被采样，模型的可变性或随机性才会出现，否则不具有一般性[9]。

因此，基于简单循环神经网络的方法完成轨迹用户链接任务的思路虽然可行，并且已有相关文献给出论述证明和实验表征，但是在实际数据集中若没有特别有效的数据预处理和巧妙地轨迹分段时，该方法的鲁棒性不足。并且，这种方法所链接的用户数有局限，当用户数量，也就是分类类别数量过多时，这种方法的性能和准确率会大大下降。

3 变分自编码器模型

3.1 概述

近年来，变分自编码器作为一种无监督学习复杂分布的生成式模型广泛应用于数据挖掘和人工智能等领域，变分自编码器基本原理基于贝叶斯推理，并且使用了神经网络结构，可以通过大量的训练数据并且采用随机梯度下降等参数优化的方法，进行训练而受到人们广泛的关注。变分自编码器是一种生成模型，已经在许多复杂数据的生成任务中，包括住宅编码、物理模型场景、分割以及预测静态图像上，显现出优势[10]。

生成式模型是机器学习中的一个重要领域，是在一些潜在的高维空间 χ 中对数据点 X 定义分布 $P(X)$的模型。与其他传统生成模型所基于的强独立性假设（例如蒙特卡洛方法和前文提到的隐马尔科夫模型）不同的是，同样属于生成模型的变分自编码器则进行弱假设，模型的训练通过反向传播以较高的效率快速进行。变分自编码器确实做了近似，但是由这种近似引入的误差对于高容量模型来说是渺小的，这些特点促成了它的迅速普及。

3.2 数学基础

3.2.1 变分自编码器定义

直观地说，变分自编码器模型首先决定在生成一个兴趣点序列之前，应该生成哪个兴趣点 ID，这种决策方法被称为受限于隐式变量的决策。也就是说，在本模型生成任何兴趣点信息之前，先随机地从全体用户样本中采样一个用户标识 z，确保模型生成的所有轨迹信息与该用户标记相匹配，因此z就是模型生成兴趣点序列的决定性因子，被称为隐式变量。将z命名为隐式变量，其原因在于模型直接得到的是生成的结果，而并不需要知道是哪个z生成了这个结果，只需要使用可提取轨迹数据特征的技术，例如循环神经网络，来推断它即可，而没有必要知道如何具体设置隐式变量参数[10]。

从形式上，在高维空间中 Z 有一个隐式变量 z 向量，可以根据定义在 Z 上的概率密度函数 $P(z)$ 对隐式变量进行采样。此时，在向量空间 Θ 中有一个确定函数 $f(z;\theta)$，其中 $f:Z\times\Theta\to\chi$，χ是生成空间。如果 z 随机并且 θ 固定，那么函数 $f(z;\theta)$ 就是 χ 空间中的随机变量。变分自编码器的目标就是：在生成过程中最大化 $P(X)$，即：

$$\max P(X)=\max\int P(X\mid z;\theta)P(z)dz \qquad (3-1)$$

公式中，$f(z;\theta)$ 被一个分布 $P(X\mid z;\theta)$ 所代替，可以通过“全概率公式”推导出 X 对 z 显式的依赖，称为 Z 对 X 的“最大似然”，在变分自编码器中，

通常选择高斯分布作为输出分布，即：$P(X \mid z;\theta) = N(X \mid f(z;\theta), \sigma^2 * I)$，其中 I 是与 X 维度相同的单位矩阵，标量 σ 是超参数。也就是说，在标准变分自编码器中，如果输出是实数向量，通常输出的分布是均值为 $f(z;\theta)$，方差为 σI 的高斯分布。通过使 $f(z;\theta)$ 接近 z 的方法来最大化 $P(X)$，即在训练的过程中，生成结果尽可能接近样本数据。

3.2.2 两个核心问题

1）隐式变量 z 向量的表示

在理想的情形下，隐式变量是由模型自动生成的，从而避免手动设计与选择隐式变量空间中每一个维度所编码的信息，在必要的时候能够让隐式变量的编码带有特定的语义，同时也应该避免隐式变量不同维度之间的依赖关系[10]。变分自编码器通过引入函数映射关系处理这些问题：假设隐式变量空间中的每一个维度是彼此独立的，同时隐式变量满足先验分布$N(0, I)$，其中 I 是单位矩阵。因为只要有一个 n 维标准正态分布的 z，再找到一个足够复杂的函数 $g(x)$，那么 $g(z)$ 就可以表示任意一个 n 维分布。对公式（3-1）而言，$f(z;\theta)$ 就是一个多层神经网络组成的函数逼近器，可以将隐式变量映射到最后的输出 X。

2）$P(X)$积分的计算

采用随机的方式对 z 采样来计算（3-1）式时会发现，对于绝大多数 z 来说，$P(X|z)$ 都接近于 0（因为 z 是对标准正态分布的采样），这些采样对 $P(X)$ 的估计毫无用处，使得这种方式非常低效。变分自编码器解决这个问题的核心思想是：通过构造另一个分布 $Q(z|X)$，使得尝试从该分布中采样出有更大可能产生 X 的 z[10]。因此，在变分自编码器的采样环节中，以给定 X 的信息作为先验，从最有可能生成一个待估计样本 X 的隐式变量分布 $Q(z|X)$ 中对 z 进行采样，并用这些采样来高效估计 $P(X)$。

3.3 模型结构

为了解决两个核心问题，变分自编码器中引入编码函数$g(X)$来对分布$P(z|X)$进行估计[10]，我们记$g(X)$的估计分布为$Q(z|X)$。则$g(X)$输出最有可能生成X的隐式变量z的分布参数，并用这些参数来构建$Q(z|X)$。此时在$Q(z|X)$下进行采样的效率远远高于在$P(z)$下的采样效率，这让我们对$E_{z\sim Q}P(X|z)$的计算更加容易。

3.3.1 构造目标函数

首先利用 KL 散度（Kullback-Leibler Divergence）研究 P(z|X)与 Q(z|X)的关系[10]，KL 散度的相关推述不在本文的研究范围，故不作推导：

$$\mathbb{KL}[Q(z|X) \parallel P(z|X)] = E_{z\sim Q}\big[log\big(Q(z|X)\big) - log\big(P(z|X)\big)\big] \qquad (3-2)$$

由条件概率公式，注意到：

$$P(z|X) = \frac{P(X|z)P(z)}{P(X)} \tag{3-3}$$

将该等式代入（3-2）式，可以得到：

$$\begin{aligned}&\mathbb{KL}[Q(z|X) \parallel P(z|X)] \\ &= E_{z\sim Q}\big[log\big(Q(z|X)\big) - log\big(P(X|z)\big) - log\big(P(z)\big)\big] \\ &+ log\big(P(X)\big)\end{aligned} \tag{3-4}$$

对（3-4）式化简得：

$$\begin{aligned}&log\big(P(X)\big) - \mathbb{KL}[Q(z|X) \parallel P(z|X)] \\ &= E_{z\sim Q}\big[log\big(P(X|z)\big)\big] - \mathbb{KL}[Q(z|X) \parallel P(z)]\end{aligned} \tag{3-5}$$

（3-5）式是变分自编码器的核心公式，下面将对其作详细的阐述。(3-5)式的左边包含了本模型需要优化的两个部分：前一部分$log\big(P(X)\big)$是最大似然概率，后一部分$\mathbb{KL}[Q(z|X) \parallel P(z|X)]$是最小化隐式变量分布$P(z|X)$与预测分布$Q(z|X)$的差异，使得最有可能生成一个待估计样本$X$的分布$Q(z|X)$尽可能的靠近实际分布。(3-5)式的右边是可以用随机梯度下降法来进行优化的目标，Q将X编码为隐式变量z，而P将z解码生成（重构）X，这实际上就是自编码器的工作，在后续会详细分析。

因此，我们确定了采用公式（3-5）作为目标函数时需要最大化$log\big(P(X)\big)$的同时最小化$\mathbb{KL}[Q(z|X) \parallel P(z|X)]$。当隐式变量空间$Z$的条件预测$Q(z|X)$尽量接近条件概率分布$P(z|X)$时，目标函数转换为直接最大化$log\big(P(X)\big)$，此时 3.2.2 中提出的难以估计的分布$P(z|X)$可以用$Q(z|X)$进行简单地计算。

3.3.2 优化目标函数

本节论述目标函数的优化问题，即如何采用随机梯度下降法优化公式(3-5)的右半部分:

$$E_{z\sim Q}\big[log\big(P(X|z)\big)\big] - \mathbb{KL}[Q(z|X) \parallel P(z)] \tag{3-6}$$

计算$\mathbb{KL}[Q(z|X) \parallel P(z)]$，首先对$Q(z|X)$的分布形式进行假设，一般地，选择正态分布$Q(z|X) \sim N\big(z\big|\mu(X;\theta),\sigma(X;\theta)\big)$，其中正态分布参数$\mu$、$\sigma$均为由$\theta$所确定的参数，而参数$\theta$可以从数据中学习到。在实际运用中，通常采用深度神经网络，本文将采用深度循环神经网络来学习参数μ、σ。此时，$Q(z|X)$与$P(z)$都是多元正态分布随机变量，$\mathbb{KL}[Q(z|X) \parallel P(z)]$便可以由正态分布的 KL 散度公式进行计算，化简得到：

$$\mathbb{KL}\left(N\left(\mu(X),\sigma(X)\right)||N(0,I)\right)$$

$$=\frac{1}{2}\left(tr\left(\sigma(X)\right)+\left(\mu(X)\right)^{T}\left(\mu(X)\right)-k-logdet\left(\sigma(X)\right)\right) \qquad (3-7)$$

计算$E_{z\sim Q}\left[log\left(P(X|z)\right)\right]$，需要通过采样的方法估算$E_{z\sim Q}\left[log\left(P(X|z)\right)\right]$，但是要得到一个好的估计结果必须要对隐式变量$z$进行多次采样，并经过多次计算，这样做的计算规模是非常大的。实际工程中，在训练集中选取一个样本X，一般情况下，基于$Q(z|X)$仅对z进行一次采样，然后计算$P(X|z)$，并将其作为期望的估计值[10]。在给定的训练集D中，对于其中的所有样本，最大化目标函数的期望：

$$E_{X\sim D}[E_{z\sim Q}[log(P(X|z))]-\mathbb{KL}[Q(z|X)\parallel P(z)] \qquad (3-8)$$

对（3-8）式进行梯度运算时，得到的梯度也必然是期望值的形式。当给定训练集D中的一个样本X，可以利用$Q(z|X)$对z进行采样，然后计算：

$$log\left(P(X|z)\right)-\mathbb{KL}[Q(z|X)\parallel P(z)] \qquad (3-9)$$

作为(3-8)式的估计，同时也可以计算梯度作为对（3-8）式梯度的期望的估计.因此，考虑选取多组(X,z)进行估算，并求平均，将结果以中心极限定理收敛到(3-8)式的梯度。

3.3.3 变分自编码器模型

至此，变分自编码器的目标函数及优化目标的方法已经完备，但是直接利用(3-7)式中的分布$N\left(\mu(X),\sigma(X)\right)$对隐式变量$z$进行采样，无法完成反向传播算法的计算，因为该采样并不是连续的，故无法计算梯度[10]。随机梯度下降法可以处理随机输入的问题，但是不能处理神经网络中的随机模块。

本文引入“重参数化”来解决上述问题：给定分布$Q(z|X)$的参数$\mu(X),\sigma(X)$，将原采样过程$z=sample\left(N\left(\mu(X),\Sigma(X)\right)\right)$重参数化为：

$$z=\mu(X)+\sigma^{\frac{1}{2}}(X)*\epsilon,\epsilon\sim N(0,I) \qquad (3-10)$$

该式作为隐式变量z的采样值，代入(3-9)，得到：

$$log\left(P\left(X\middle|z=\mu(X)+\sigma^{\frac{1}{2}}(X)*\epsilon\right)\right)-\mathbb{KL}[Q(z|X)\parallel P(z)] \qquad (3-11)$$

此时给定X,ϵ，公式(3-11)是P,Q的确定连续函数，因此可以使用随机梯度下降法对其进行优化。

图 3-1 描述了变分自编码器模型的核心结构模式。

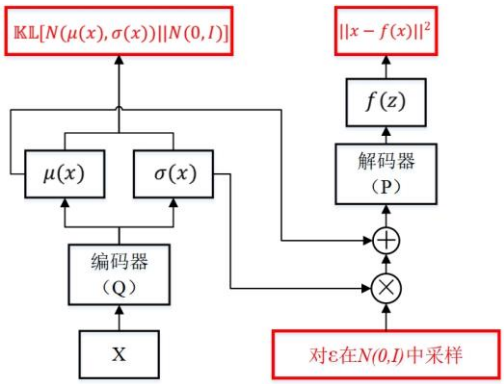


图 3-1　变分自编码器核心结构模式

在本文的轨迹用户链接任务中，人类签到轨迹数据集每一元素由两部分组成 $(t_{u_1},u_1),\ldots,(t_{u_m},u_m)$，其中第 i 个轨迹 $t_{u_i}\in\mathcal{T}$，对应的用户（标签）为$u_i\in\mathcal{U}$。假设观察到的轨迹是由隐式变量 z_i 生成的。根据 Kingma and Welling 于 2014 的研究，只要研究对象是与单个数据点相关的对象，即轨迹，可省略索引 i[11]。链接的轨迹和未链接的轨迹子集上的经验分布分别用 $\widetilde{p_l}(t,u)$ 和 $\widetilde{p_u}(t)$ 表示。目标是在生成模型下最大化训练集中每个轨迹 t 的概率，根据推导完备的公式(3-11)，变分自编码器模型由深度神经网络共同训练[12]，针对轨迹用户链接的变分自编码器，将在 4.4 中详细论述。

4 基于变分自编码器模型对轨迹用户链接

因为直接利用第三部分中论述的变分自编码器模型，而不分析提取人类轨迹数据的语义结构，且不解决数据稀疏性问题，无法应用于解决轨迹用户链接问题。所以，该部分提出基于变分自编码器解决轨迹用户链接问题的方法，并且论述模型实现的细节。

本文将提出基于变分自编码器模型对轨迹用户链接的方法，它同早期对轨迹数据挖掘研究工作的不同之处与亮点在于：

（1） 解决了对人类移动轨迹中隐式变量的推断问题；

（2） 利用循环神经网络学习了人类运动轨迹（签到序列）的层次语义结构；

（3） 以半监督学习的方式利用未标记的数据来识别个体用户的活动模式并解决轨迹用户链接问题。

4.1 概览

4.1.1 轨迹用户链接整体思路结构

解决轨迹用户链接问题主要分三个步骤，另可推广一步做预测，如图 4-1 所示。首先从公开数据集训练样本中获取一群特定用户的轨迹数据，经过数据清洗、预处理后的轨迹数据将被加载到变分自编码器模型中进行训练，通过 LSTM 对序列化数据分层地学习轨迹信息特征，共同训练变分自编码器的编码器和解码器，生成网络模型可以得到更具体和更清晰的轨迹行为特征，与此同时引入半监督学习机制，同时用链接的轨迹数据和未被链接的轨迹数据训练分类器，减轻了数据稀疏性的问题。将待链接的人类轨迹数据通过训练好的变分自编码器模型提取高维信息特征，利用多分类算法链接到具体用户，解决轨迹用户链接问题。本模型可进一步的推广，利用确定已经链接的用户信息，结合变分自编码器的解码器部分，可以预测该用户的下一个兴趣点或者是行程序列，用做未来轨迹预测，该推广属于未来工作，本文不作细致研究。

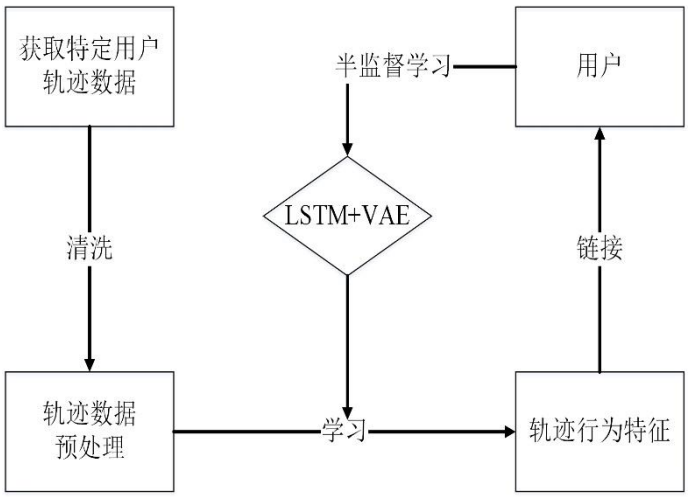


图 4-1 轨迹用户链接的思路结构

4.1.2 基于变分自编码器轨迹用户链接模型结构

前文已经论述过变分自编码器由于无法有效提取人类轨迹数据结构性语义且解码器所生成的向量序列无法直接表示轨迹数据的特征，故不能直接应用与轨迹用户链接任务。现在描述本文提出的基于变分自编码器的轨迹用户链接模型的细节，其中主要包含基于神经网络和深度学习的四个组成部分：循环神经网络编码器、中间层循环神经网络、循环神经网络解码器和多分类分类器，如图 4-2 所示。

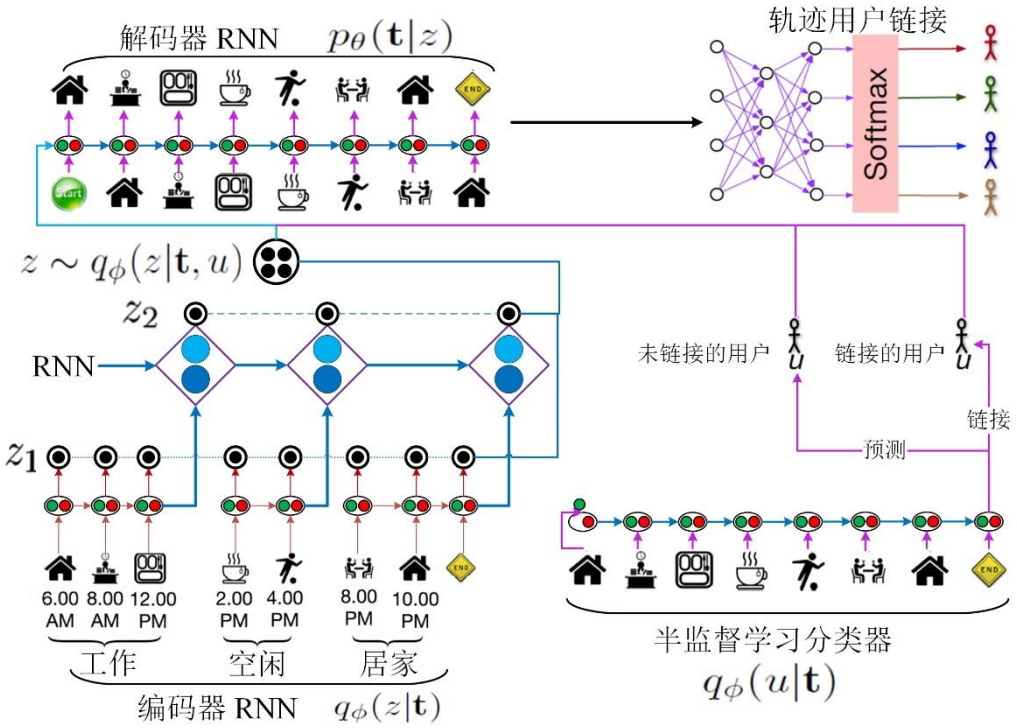


图 4-2 基于变分自编码器轨迹用户链接模型结构

基于变分自编码器的轨迹用户链接模型，首先将原始签到信息数据进行清洗，去除无效与错误的签到数据，并对清洗好的轨迹分段。接下来，使用“词向量”技术将所有轨迹数据表示为一种低维表示的嵌入数据[13]。图 4-2 中，左下部分：部署编码器和中间层循环神经网络学习轨迹数据序列的分层结构，用多层隐式变量串联起来表示隐式空间。左上部分：从先验分布 $q(z|t,u)$ 中采样z，同时与用户标签u被传递到解码器生成网络，通过生成结果来计算概率 $p(t|u,z)$。右下部分：半监督学习模型，未链接的轨迹数据被用来训练分类器 $q(u|t)$，用来表征用户标签分布。右上部分：利用深度神经网络的多分类分类器将给定非链接的轨迹数据链接到产生它的用户。

4.2 数据预处理

本文在数据预处理环节主要对数据集做了数据清洗和轨迹数据分段两方面的工作。

4.2.1 数据清洗

位置采集设备和移动计算技术的进步可以让我们更容易地获取到大量的空间轨迹数据，但是提供位置信息服务的软件和硬件设备可能由于传感器的噪音和其他因素，不可能总是完美地精确记录带有时间和空间的轨迹信息[14]。据观察，在本文采用的数据集中最可能出现的“脏数据”有两种形式：其一，可能会出现数据条目关键信息部分为空（在数据集中表现为确定的用户标签，正常时间戳，空间经纬度信息均为 0，兴趣点标签为“全零”）的无效信息。例如，在 Brightkite 数据集中用户标签为 0 的用户在 2010 年 8 月 6 日 21：22：16 的签到的经纬度坐标信息为(0.0,0.0)，兴趣点标签为“00000000000000000000000000000000”；其二，可能会出现严重违背生活实际的数据条目，在对每个确定用户的签到轨迹条目以时间顺序排列时，发现在较短的时间间隔中出现了极大的位移。例如，在原始 Gowalla 数据集中用户标签为 39602 的用户在约 2 个小时 21 分钟内，经纬坐标由（59.4010641881,16.4434432983）改变到（79.3681452833,12.5328687948），据计算这两个经纬坐标对应的两地相距约 2190.57134 千米，则平均速度约为 940.16 千米每小时，这显然有悖于正常生活实际。

此时需要对待输入模型的轨迹数据进行数据清洗，否则无效信息或者严重偏离生活实际的轨迹数据将会直接影响轨迹用户链接结果的准确率[14]，甚至在前期的实验中，因为在变分自编码器的优化目标和损失函数的计算中包含对数运算，所以原始轨迹条目中无意义的数据对象会导致变分自编码器因崩溃而失效。

本文利用对数据集过滤实现数据清洗。针对包含无效信息的轨迹数据，采用搜索并删除的策略。对所采用的数据集遍历，查找兴趣点标签为空或者为“全零”

的条目，直接删除该条目。针对严重偏离生活实际的轨迹数据，首先引入“探针”，其本质就是临时变量，对每一确定标签的用户中根据时间顺序的相邻两个签到条目，根据时间戳可以得到连续两次签到的时间差，由 Great-Circle 距离，基于球面余弦的公式(4-1)：

$$\widehat{AB} = R \times arccos\left(cos\, A_{lat}\, cos\, B_{lat}\, cos\left(B_{long} - A_{long}\right) + sin\, A_{lat}\, sin\, B_{lat}\right) (4-1)$$

其中R为地球半径，本文中取 6366 千米，A_{lat}、A_{long}分别表示签到点A的纬度和经度，B_{lat}、B_{long}分别表示签到点B的纬度和经度，可以得到该用户每两次签到的直线距离。可以求得该用户在完成两次签到过程中的平均速度，获得“探针”值。根据相关研究文献[14]，本文平均速度阈值设置为 300 千米每小时，当“探针”值不大于 300，则在签到点B处的签到条目是合法的，予以保留；否则“探针”值大于 300，即连续两个签到点间直线移动平均速度超过 300 千米每小时，不符合生活实际，删除该签到条目。该方法默认每一确定标签用户的第一条轨迹签到条目是合法的。

4.2.2 轨迹数据分段

为了降低计算复杂度，同时能够通过后续循环神经网络提取更多的特征信息，我们需要对轨迹数据进行分段，也就是说，对同一用户的轨迹序列划分为若干个子轨迹序列。用户 u_i 的轨迹数据最初以日期分段，即：原始轨迹数据 T_{u_i} 被分段为 k 个连续的子序列$t_{u_i}^1, \ldots, t_{u_i}^k$，其中 k 是用户 u_i 包含签到轨迹的天数。

通过对子轨迹序列的学习，可以挖掘出比整体轨迹序列更多的特征信息。本文对同一用户产生的签到信息，在同一天内考虑两个语义因素，根据时间和空间两个指标进行子轨迹的分段。

首先在时间指标上完成轨迹分段。基于时间间隔的轨迹分段原则，是设置一个时间阈值，在本文中，通过对不同数据集的签到数据的观察和相关研究文献[11]的调查，针对不同数据集设置了不同的时间间隔阈值：对 Brightkite 和 Gowalla 数据集设置时间阈值为 4 小时，即每一个日常轨迹 $t_{u_i}^j (j \in [1,6])$ 基于 4 小时的时间间隔分为六个连续的子序列 $t_{u_i}^{j,1}, \ldots, t_{u_i}^{j,6}$；对 Geolife 数据集设置时间阈值为 6 小时，即每一个日常轨迹 $t_{u_i}^j (j \in [1,4])$ 基于 6 小时的时间间隔分为四个连续的子序列 $t_{u_i}^{j,1}, \ldots, t_{u_i}^{j,4}$。

其次在空间指标上完成轨迹分段。考虑到用户在同一个兴趣点停留时会产生多条轨迹信息，特别是位置记录较为频繁的 Geolife 数据集，通常会在一段时间内采集到该用户许多相近的经纬坐标。根据相关研究文献表明，在 Gowalla 和 Foursquare 数据集中，90%的用户移动距离都小于 $50km$，这意味着用户更趋向于访问与当前邻近的兴趣点[15]。因此，如果连续的签到条目对应的兴趣点相距超

过$50km$，我们将其进一步分割子轨迹 $t_{u_i}^{j,m}(m \in [1,6])$。

4.3 轨迹数据编码

本小节将论述数据预处理完成后的轨迹数据，在传入轨迹用户链接模型之前，按照前文论述及变分自编码器的定义，需对含有语义的轨迹数据做“词嵌入”（Embedding）并利用循环神经网络对嵌入后的向量进行进一步编码。

4.3.1 轨迹数据嵌入（Embedding）

在自然语言处理领域中，类似于文档主题分类和语句级别的对话生成的深入研究和广泛应用，Word2Vec 技术是利用了神经网络将文本数据中的单词通过训练模型转换为最优化的向量，用计算余弦距离的方式，能够很方便容易地将比较两个数据间语义相似程度的过程转换为比较词向量间在某一向量空间中的相似程度的过程，这一转换在文本处理中有利于实现词语词性的提取、词义聚类等应用[16]。

本文受到文本模型层次结构的启发，借鉴 Word2Vec 技术对轨迹数据做嵌入工作，本文借鉴这一技术，将签到轨迹数据的地点经纬度信息、兴趣点序列、时间戳信息通过已经训练好的模型[16]，嵌入到一个特征向量。具体工作为：对分段后的轨迹进行建模，每一用户的完整轨迹由包含语义的子轨迹组成，子轨迹包含了对应用户的时间和空间运动模式，其中一条子轨迹由顺序的兴趣点序列组成：

$$p_\theta(s_1, \dots, s_N) = \prod_{n=1}^{N} \prod_{m=1}^{M_n} p_\theta\left(s_{n,m} \middle| s_{n,1:m-1}, s_{1:n-1}\right) \qquad (4-2)$$

其中，p_θ是由兴趣点组成子轨迹序列的先验概率，s_n 是一个日常签到序列中第 n 个子轨迹，$s_{n,m}$ 是第 n 个子轨迹中第 m 个兴趣点信息，包括时间、经纬度、兴趣点标签，M_n 是第 n 个子轨迹中兴趣点的数量。

4.3.2 轨迹数据编码

根据 3.2 节中对变分自编码器定义的论述，经过处理和分段的轨迹数据再初步嵌入成为特征向量后，会进一步地由编码器编码，本文中使用循环神经网络对轨迹进行分层编码，提取尽可能多的语义特征。为了活动的更细致信息和更丰富的特征，本节将在子轨迹序列的基础上，对兴趣点进行逐个计算。

循环神经网络是一种具有记忆功能的神经网络，适合对序列数据的建模，它的特点为神经元在某一时刻的输出能够作为输入再次输入到神经元。这种循环机制的串联网络结构特别适用于时间序列数据，在训练中可以保持输入序列数据的依赖关系[4]。目前，长短期记忆模型（LSTM）是实际应用中使用十分广泛的循环神经网络，它能够非常有效地解决一般循环神经网络中存在的梯度消失问题，

可以对序列数据中存在的短期和长期的依赖进行建模[17]。因此，本文采用 LSTM 作为编码器的主要模型对轨迹数据进行进一步地的编码，来提取更准确的轨迹层次语义。

在兴趣点级别，用一个兴趣点长短期记忆单元（LSTM cell），或一个门控循环单元（GRU），来编码由兴趣点序列形成子轨迹的隐藏语义主题（例如：图 4-2 中“工作”，“空闲”及“居家”）。兴趣点级编码器的最后隐藏状态被输入到中间层循环神经网络，内部隐藏状态被编码为维度更低的向量，以反映日常出行的结构特征。分层轨迹编码的内部状态中兴趣点 LSTM 和中间层 LSTM 在数学上由(4-3)和(4-4)分别描述为：

$$\begin{cases} h_{n,0}^{POI} = 0, \ h_{n,m}^{POI} = LSTM\left(h_{n,m-1}^{POI}, s_{n,m}\right) & (4-3) \\ h_0^{INT} = 0, \ h_n^{INT} = LSTM\left(h_{n-1}^{INT}, h_{n,M_n}^{INT}\right) & (4-4) \end{cases}$$

其中，对应的 $LSTM(\cdot)$ 是“vanilla” LSTM 函数，其定义如下：

$$i_t = \sigma(W_i v_t + U_i h_{t-1} + b_i) \quad (4-5)$$

$$f_t = \sigma\left(W_f v_t + U_f h_{t-1} + b_f\right) \quad (4-6)$$

$$o_t = \sigma(W_o v_t + U_o h_{t-1} + b_o) \quad (4-7)$$

$$\widetilde{c_t} = \tanh(W_c v_t + U_c h_{t-1} + b_c) \quad (4-8)$$

其中，i_t，f_t，o_t 和 b_* 分别是输入门，遗忘门，输出门和偏移向量；σ 是非线性激活函数 logistic sigmoid 函数；矩阵 W 和 $U(\in \mathbb{R}^{d\times d})$ 是不同的门参数；v_t 是兴趣点 c_t 的嵌入特征向量。

记忆单元 c_t 通过替换现有的记忆单元 $\widetilde{c_t}$来完成一轮计算的更新：

$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c_t} \quad (4-9)$$

$$E_{v_t} = h_t = o_t \odot \tanh(c_t) \quad (4-10)$$

其中 $\tanh(\cdot)$ 函数是双曲正切函数，$\odot$ 是逐个分量乘法。输出编码向量 E_{v_t} 是最终隐藏层特征向量。

本文在变分自编器的编码器部分设置了两层 LSTM 模型用于编码，在 LSTM 层后我们使用批正则化（Batch Normalization）技术，来保证在深度神经网络训练过程中，每一层神经网络的输入都保持相同的分布。因为随着网络深度的加深，在训练过程中非线性激活函数 logistic sigmoid 函数的输入值的分布会发生偏移，导致反向传播时靠前的较低层 LSTM 或者轨迹数据嵌入层的梯度消失。所以需要使用批正则化，将每层神经网络任意神经元的非线性化输入值分布调整到标准正态分布，避免梯度消失，并且大大加快训练的收敛速度。

其次，由于轨迹数据嵌入（Embedding）层中使用了全连接层，而且使用分段后的用户的子轨迹序列，训练数据较为稀疏，极易出现训练模型过拟合的问题。因此，在嵌入层后，本文设置 Dropout 层，以降低过拟合现象发生的可能性。

综合本章节轨迹数据嵌入为特征向量和轨迹数据的编码，结合变分自编码器定义和 3.3 节中论述过的采样准则，现给出基于变分自编码器的轨迹用户链接模型的编码器部分的网络模型和数据结构，如图 4-3 所示。其中，InputLayer 是本文定义的轨迹数据输入，鉴于用户产生签到序列的完整性，输入维度设置为最大序列长度 2101；Embedding 是轨迹数据嵌入层，将嵌入的轨迹数据特征向量定义为 320 维；Flatten 是为了变分自编码器在随机采样时服从一维分布，而将分层提取的轨迹数据编码向量展开；Lambda 是自定义的基于采样均值和标准差的生成层。

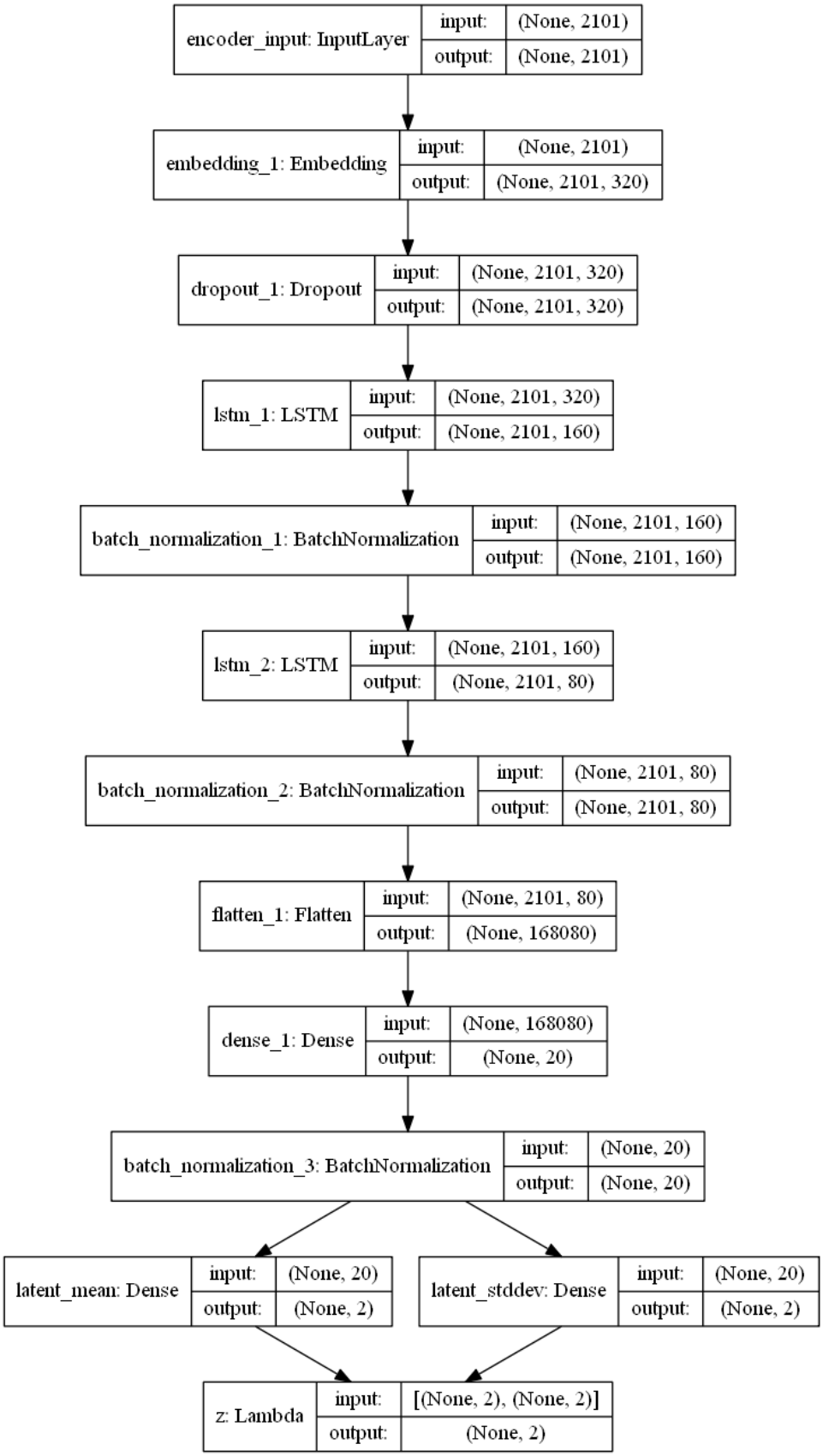


图 4-3 轨迹用户链接模型编码器

4.4 使用变分自编码器链接轨迹用户

本文提出的基于变分自编码器的轨迹链接模型与早期工作的不同之处在于：（1）解决了人类移动轨迹中的对隐式变量的推断问题；（2）可以学习到人类签到序列的层次语义；（3）以半监督学习的方式结合未链接的轨迹数据来识别个体的活动模式并解决轨迹用户链接问题。

为了缓解数据稀疏性问题，可以通过合并未链接的轨迹数据来提高识别链接的性能，采用具有半监督学习性质的变分自编码器能够在一定程度上解决此问题[18]。然而，虽然由于用循环神经网络作为编码器，并从编码向量中采样估计后验概率，这在对隐式变量的层次结构方面建模具有建设性意义，但是我们希望采用半监督学习框架，这可能使编码器模型无法捕获未链接轨迹数据中的抽象特征。同样地，递归堆叠的变分自编码器彼此重叠，受到较低生成层中可能出现的过拟合现象而限制其性能，只有通过学习足够的特征才能够生成重建的轨迹数据。

因此，本文倾向于考虑一种这样的生成序列模式，即：以隐式变量的某些部分为基础，在此之上编码子轨迹的抽象主题，而在其他变量上编码兴趣点，在半监督学习框架下的变分自编码器中学习推理模型。这种对隐式变量分层生成模型的基本思想是受可变阶梯自动编码器（Variational Ladder Autoencoder，VLAE）启发的，其中浅层网络用于表达低级简单特征，深层网络用于表示高级复杂功能，并在网络结构的不同级别和阶段注入高斯噪声[19]。但是，注意到可变阶梯自动编码器是为包含连续隐式变量特征分类学习而设计的，而本文所基于的变分自编码器是针对离散化轨迹数据的半监督学习模型，因此还需要借鉴其思想完善本模型。

更具体地，我们考虑两层隐式变量 z_1，z_2，分别表示兴趣点层（底部）和轨迹数据层（顶部）。先验概率 $p(z) = p(z_1, z_2)$ 是标准高斯分布，并且联合分布 $p(t, z_1, z_2)$ 可以被分解为：

$$p(t, z_1, z_2) = p(t|z_1, z_2)p(z_1|z_2)p(z_2) \qquad (4-11)$$

在生成网络的解码器部分中，由图 4-3 与图 4-4 所示，隐式变量 z 是由并置的两个向量合并而成的:

$$z = [RNN_m(z_2); RNN_n(z_1)] \qquad (4-12)$$

其中 z_2 和 z_1 分别由中间层循环神经网络和兴趣点循环神经网络参数化产生。对于生成网络的推断部分，近似后验概率 $p(z|t)$ 是服从高斯分布 $\mathcal{N}(\mu_i, \sigma_i)(i = 1,2)$ 由三层循环神经网络参数化。

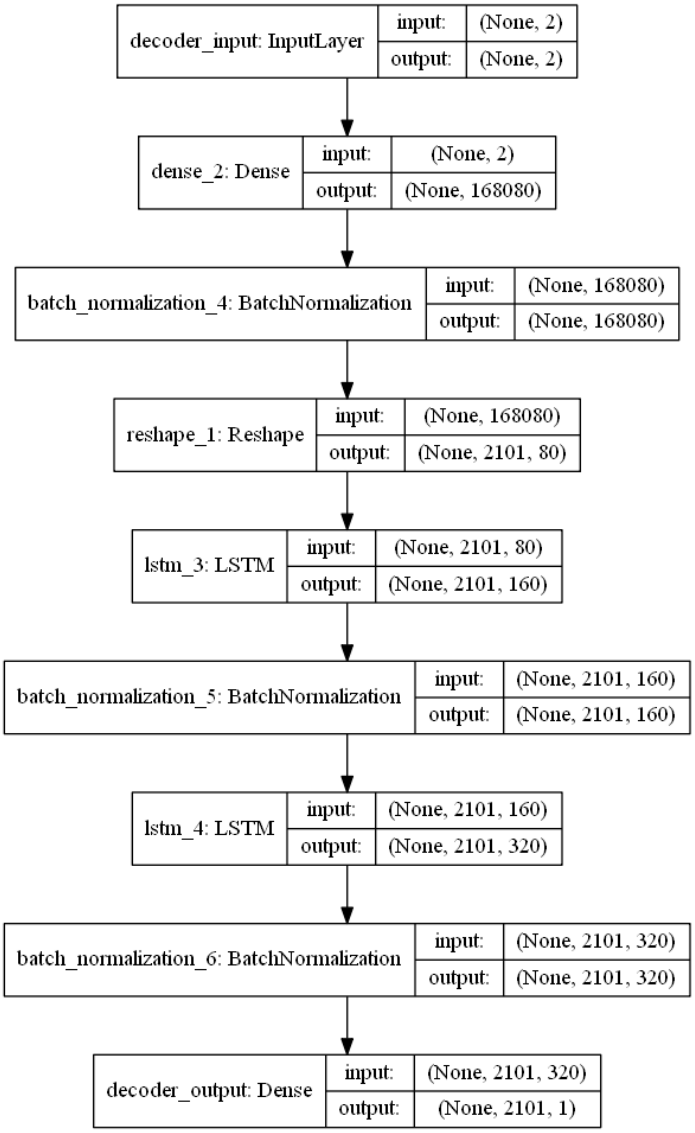


图 4-4 轨迹用户链接模型解码器

现在通过半监督的方式学习轨迹数据，目的是得到给定隐式变量层次结构的潜在表示，并且保证符合深度神经网络模型的生成原则。因此，当观察到与轨迹数据（有标签的数据 D_l）相对应的用户 u，由公式（3-5）得到：

$$\log p(t,u) = \mathbb{E}_{z\sim q}[\log p(t|u,z)] + \log p(u) - \mathbb{KL}[q(z|t,u)||p(z)] + \mathbb{KL}[q(z|t,u)||p(z|t,u)] \quad (4-13)$$

根据第三部分中对变分自编码器模型原理的论述，此时的目标是利用 $q(z|t,u)$ 估算真实先验概率 $p(z|t,u)$，即：最小化$\mathbb{KL}[q(z|t,u)||p(z|t,u)]$。因此，为了估计生成参数，将单个轨迹数据的边际可能性的下限证据（ELBO, evidence lower bound）用作目标$\varepsilon_1^g(t,u)$：

$$\varepsilon_2^g(t,u) = \mathbb{E}_{z\sim q}[\log p(t,u,z)] - \mathbb{KL}[q(z|t,u)||p(z)] \quad (4-14)$$

其中，$\mathbb{KL}[q(z|t,u)||p(z)]$ 是隐式后验概率 $q(z|t,u)$ 和先验概率分布 $p(z)$ 间的 KL 散度，它衡量了对 z 估算先验概率时丢失的信息多少。期望项是重构误差或期望的负对数似然，这可使解码器从隐式分布中重新构建轨迹。

如果是未标签的轨迹数据 D_u，通过使用分类器 $q_\emptyset(u|t)$ 进行后验推断来预测用户身份 u。我们将 u 认为是另一个隐式变量，得到如下的 ELBO $\varepsilon_2^g(t)$：

$$\log p(t) \geq \mathbb{E}_{z\sim q(u,z|t)}[\log p(t,u,z) - \log q(u,z|t)]$$

$$= \sum_u q(u|t)\left(\varepsilon_1^g(t,u)\right) + \mathcal{H}\left(q(u|t)\right) = \varepsilon_2^g(t) \qquad (4-15)$$

其中 $\mathcal{H}\left(q(u|t)\right)$ 是信息熵，分类器 $q_\emptyset(u|t)$ 在训练期间的损失通过预测用户与真实标签之间的 L_2 重构误差来衡量。因此，对整个数据集边界似然的 ELBO ε 为：

$$\varepsilon = -\sum_{(t,u)\sim D_l}\left(\varepsilon_1^g(t,u) + \alpha \log q(u|t)\right) - \sum_{t\sim D_u}\varepsilon_2^g(t) \qquad (4-16)$$

其中左部包含分类器 $q_\emptyset(u|t)$ 当从带标签的数据中学习时的一个附加分类损失，超级参数 α 控制带标签数据学习的权值。

注意到，半监督学习的变分自编码器在数据集中学习到的类别数量呈线性上升，这是通过重新评估训练期间每个类别的生成可能性而提高的。对于文本分类任务，类别数量很少（例如，IMDB 中为 2 个类别，AG News 中为 4 个类别）。然而，轨迹用户链接问题中的类别（用户）的数量非常大，这在计算分类器 $q_\emptyset(u|t)$ 时产生巨大的计算量。为了解决这个问题，我们采用蒙特卡罗方法来估算每个类别的期望评估[20]，其基线准则也可方便地被用作减小基于采样的梯度估计量的方差：

$$\nabla_\emptyset \mathbb{E}_{z\sim q(Z|t,u)}\left[q(u|t)\left(-\varepsilon_1^g(t,u)\right)\right] \qquad (4-17)$$

由于 ELBO $\varepsilon_2^g(t)$ 决定着分类器 $q_\emptyset(u|t)$ 的规模，遵循早期的文本分类工作中的选择，本文在基于变分自编码器的轨迹用户链接模型中使用的准则是 $\varepsilon_2^g(t)$的均值[21]。

综合本章对基于变分自编码器的轨迹用户链接模型的详细论述和推演，与现行常用轨迹挖掘、轨迹用户链接方法相比，其优势表现为：

（1） 本模型通过引入半监督学习的变分自编码器减轻了数据稀疏性问题，利用大量的未标签数据改善了轨迹用户链接任务的性能[18]。

（2） 实例化了一个学习隐式随机变量分布的结构，以对轨迹数据中的观测到的可变性建模。通过将变分推理纳入具有隐式变量的生成模型，本模型揭示了对复杂分布和对兴趣点序列长期依赖关系一种可解释的表

示方法。

（3） 通过利用用户运动轨迹数据展现出高时空规律性这一事实，例如，超过 90%的夜间运动记录是在同一兴趣点生成的[20]，基于变分自编码器的轨迹用户链接模型能够捕捉到表示个体运动独特性的子轨迹数据的语义。

进一步的，用户的移动轨迹数据表现出分层特性，比如说，在子轨迹序列数据中高频出现的兴趣点和一些隐含的模式（例如：目的地的意义），他们会体现在轨迹数据中。我们利用分层次的循环神经网络来提取轨迹语义，编码运动模式特征，改进人类运动识别效果及其效率，为轨迹用户链接任务提供解决办法，同时为轨迹数据挖掘的其它相关工作提供一条可行的思路。

5 实验结果与分析

在这一部分，本文利用三个公开的基于地理社交媒体应用的数据集来测试基于变分自编码器的轨迹用户链接模型，表征本模型的优点，对比和评估效果。

5.1 数据集说明

本文选用三个公开有效的基于地理社交媒体应用数据集进行了实验，分别是 Gowalla，Brightkite 和 Geolife。对于 Geolife，直接选择 Trajectory 数据，而不采用匹配过运动模式的 Label 数据。从数据集中随机选取 $|U|$ 个用户和他们对应的轨迹数据，对每一个数据集进行评估。我们会从中选择两个包含用户数量不同的数据集，这些用户会产生不同的轨迹以进行模型性能的鲁棒性检查。表 5-1 描述了这三个数据集的统计数据，其中$|\mathcal{U}|$表示用户数量；$|T_n|/|T_e|$表示训练和测试的轨迹数量；$|\mathcal{C}|$表示签到数量；$\bar{\mathcal{R}}$表示分段前轨迹的平均长度；$\mathcal{T}_r$表示轨迹长度的范围。

表 5-1 数据集描述

数据集	$\|\mathcal{U}\|$	$\|T_n\|/\|T_e\|$	$\|\mathcal{C}\|$	$\bar{\mathcal{R}}$	$\mathcal{T}_r$
Gowalla	201	9920/10048	10958	219	[1,131]
	112	4928/4992	6683	191	[1,95]
Brightkite	92	9920/9984	2123	471	[1,184]
	34	4928/4992	1359	652	[1,44]
Geolife	50	9378/10366	13652	879	[1,43]
	36	6957/7123	9506	852	[1,31]

5.2 模型训练

5.2.1 模型设置与参数

在基于变分自编码器的轨迹用户链接模型训练的早期阶段，公式（4-13）中的 $\mathbb{KL}[q(z|t,u)||p(z)]$ 项可能会对兴趣点信息编码为隐式变z产生阻碍作用，从而容易导致模型崩溃，我们推测的原因是 Bowman 等人提出自回归模型（如循环神经网络）拥有强大的学习能力和泛化能力。一个有效的解决办法是通过逐渐增加协同效率 β（从 0 到 1）控制 KL 散度项的权值，这也被称作 KL 成本退火

[22]。

在循环神经网络编码器模型中使用的激活函数是 Softplus，即：

$$f(x) = \ln(1 + e^x) \qquad (5-1)$$

，它将非线性化应用到参数向量的每个分量，确保方差有意义。除此之外，因为可变长度数据是基于循环神经网络模型的训练的瓶颈之一，我们使用“桶方法”加速学习过程：按长度对所有轨迹进行排序，然后将长度相似的轨迹放入相同的桶中，在其中将数据填充到相同的长度，并作为批处理传入神经网络中，这可以减少计算时间，提高模型效率。

最后，我们在本模型中使用双向 LSTM，它们可以联合起来，对正向先验概率和反向后验概率建模，同时训练兴趣点序列，从而有可能捕获更丰富的轨迹运动的表示形式和内部语义，并加快模型训练的收敛速度。

在此，给出调试后，基于变分自编码器的轨迹用户链接模型的部分超级参数设置（针对 Brightkite 数据集）：嵌入层维度 enbedding_dim=320，嵌入层 embedding_dropout_rate=0.5，变分自编码器隐式空间维度 latent_dim=2，批处理 batch_size=64，训练轮次 epochs=300，训练验证集划分 validation_split=0.2。

5.2.2 训练技术参数

中央处理器：AMD A8-6410 2.00GHz

内存：16.0GB

操作系统：Windows 8.1

编程语言：Python 3.7、Python 2.7

集成开发环境：JetBrains PyCharm 2019.3

机器学习框架：TensorFlow 1.14.0、Keras 2.3.1

可视化方法：Matplotlib 3.1.1、Graphviz 2.38

5.3 测试评估与性能比较

本文将提出的基于变分自编码器的轨迹用户链接模型与轨迹相似性测量和基于深度学习的分类领域中的几种常用方法进行了比较。为了对比实验，实现了通过嵌入分层 LSTM 和双向 LSTM 提取轨迹数据特征[6]，但是没有变分自编码器的推断。在具体的实现中，所有模型的学习率均以 0.001 初始化，并以 0.9 的速率衰减。KL 成本退火的权值β从 0.5 增加到 1；dropout 率为 0.5。我们将兴趣点统一嵌入到 320 维向量中，所有基于循环神经网络的模型的批处理大小均为 64。

用于对比实验测试的方法可以大致归类两类为：

（1） 传统方法，包括用 SVD 作为矩阵求解器的线性判别分析的 LDA

（Latent Dirichlet Allocation）模型，决策树 DT（Decision Tree），随机森林 RF（Random Forest）和用线性核函数的支撑向量机 SVM（Supported Vector Machine），在相关文献中这几种方法被广泛用于计算流动性模式和对轨迹进行分类[14]。

（2） 基于循环神经网络的轨迹用户连接模型，包括基于 LSTM 的轨迹用户连接模型（TULER-LSTM），基于双向 LSTM 的轨迹用户连接模型（TULER-LSTM-S）以及 Gao 等人提出的双 TULER（Bi-TULER），这几种是用于轨迹用户链接的最新方法[6]。

本文采用的模型评价指标包含：K 折交叉验证 ACC@K，宏查准率 macro-P，宏查全率 macro-R 和宏 F1macro-F1，他们都是信息检索领域中常用的指标。具体地，ACC@K 用于评估轨迹用户链接的准确性：

$$\text{ACC@K} = \frac{\text{正确链接到用户的轨迹条目数量 } @K\text{折}}{\text{轨迹数据条目数量}} \qquad (5-2)$$

以及 macro-F1 是 macro-P 和 macro-R 的调和平均值，对全部类别（轨迹用户链接任务中的用户）计算平均值：

$$macro\text{-}F1 = 2 \times \frac{macro\text{-}P \times macro\text{-}R}{macro\text{-}P - macro\text{-}R} \qquad (5-3)$$

表 5-2、表 5-3 和表 5-4 分别量化地描述了在 Brightkite 数据集、Gowalla 数据集和 Geolife 数据集利用本文基于变分自编码器的轨迹用户连接模型和其他基于循环神经网络的轨迹用户连接模型在选定评价指标上的性能比较，其中最优性能由下划线标出。

表 5-2 基于变分编码器与其他循环神经网络的轨迹用户链接性能比较（Brightkite 数据集）

模型方法	$\lvert\mathcal{U}\rvert = 92$					$\lvert\mathcal{U}\rvert = 34$				
	ACC@1	ACC@5	macro-P	macro-R	macro-F1	ACC@1	ACC@5	macro-P	macro-R	macro-F1
TULER-LSTM	43.01%	59.84%	38.45%	35.81%	37.08%	48.26%	67.39%	49.90%	47.20%	48.51%
TULER-LSTM-S	44.23%	61.00%	38.02%	36.33%	37.16%	47.88%	67.38%	48.81%	47.03%	47.62%
Bi-TULER	43.54%	60.68%	38.20%	36.47%	37.31%	48.13%	68.17%	49.15%	47.06%	48.08%
TULVAE	45.98%	64.84%	43.15%	39.65%	41.32%	49.82%	71.71%	51.26%	46.43%	48.72%

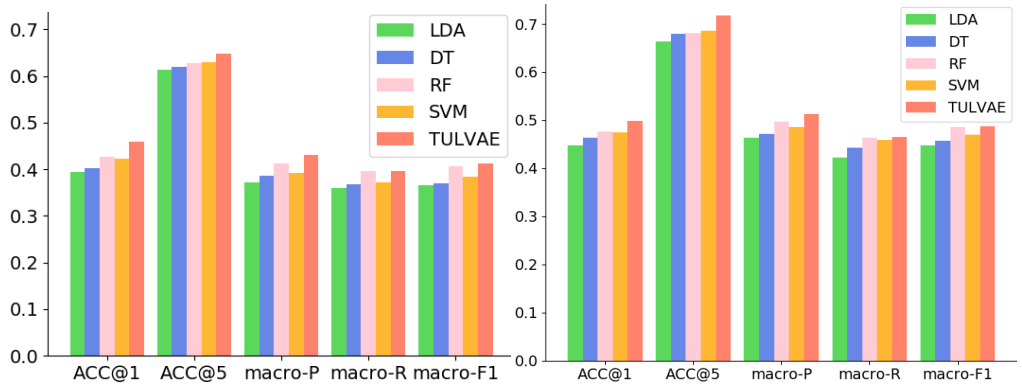
表 5-3 基于变分编码器与其他循环神经网络的轨迹用户链接性能比较（Gowalla 数据集）

模型方法	$\|\mathcal{U}\| = 201$					$\|\mathcal{U}\| = 112$				
	ACC@1	ACC@5	macro-P	macro-R	macro-F1	ACC@1	ACC@5	macro-P	macro-R	macro-F1
TULER-LSTM	41.24%	56.88%	31.70%	28.60%	30.07%	41.79%	57.89%	33.61%	31.33%	32.43%
TULER-LSTM-S	41.22%	57.70%	29.34%	28.68%	29.01%	42.11%	58.01%	33.49%	31.97%	32.71%
Bi-TULER	41.95%	57.58%	32.15%	31.66%	31.90%	42.67%	59.54%	37.55%	33.04%	35.15%
TULVAE	45.40%	62.39%	36.13%	34.71%	35.41%	44.35%	64.46%	40.28%	32.89%	36.21%

表 5-4 基于变分编码器与其他循环神经网络的轨迹用户链接性能比较（Geolife 数据集）

模型方法	$\|\mathcal{U}\| = 50$					$\|\mathcal{U}\| = 36$				
	ACC@1	ACC@5	macro-P	macro-R	macro-F1	ACC@1	ACC@5	macro-P	macro-R	macro-F1
TULER-LSTM	50.69%	62.11%	46.27%	41.84%	43.95%	57.24%	69.27%	49.35%	47.61%	48.46%
TULER-LSTM-S	49.55%	62.65%	43.40%	42.11%	42.75%	57.14%	69.57%	48.48%	47.59%	48.03%
Bi-TULER	52.31%	64.03%	47.15%	44.95%	46.03%	58.31%	71.17%	50.84%	48.88%	49.84%
TULVAE	55.54%	68.27%	51.07%	48.63%	49.83%	59.91%	73.60%	53.59%	50.93%	52.23%

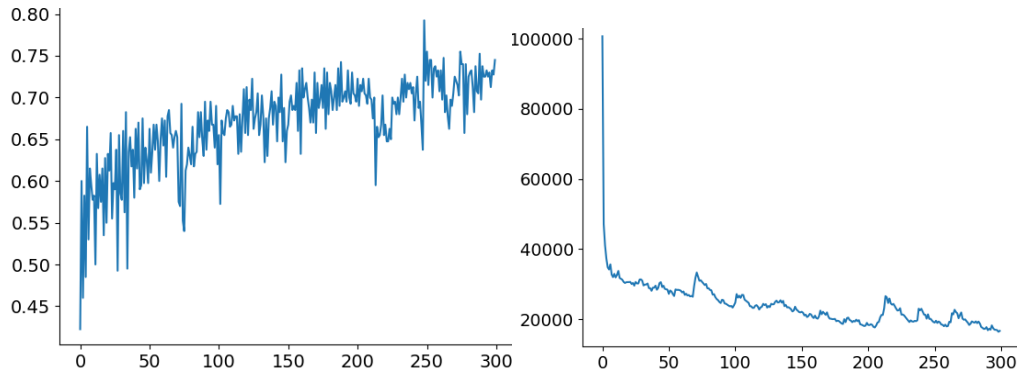
图 5-1 形象直观地描绘了本文模型和传统方法在 Brightkite 数据集上，对选定评价指标上的性能比较。据相关研究文献，在 Gowalla 和 Geolife 数据集上，传统方法已经被证明性能表现不如使用深度学习方法[6]。



（a）$|\mathcal{U}| = 92$　　（b）$|\mathcal{U}| = 34$

图 5-1 基于变分自编码器与传统方法在 Brightkite 数据集上的轨迹用户链接性能比较

我们选取了使用本文提出的基于变分自编码器的轨迹用户连接模型对 Brightkite 数据集中的轨迹数据进行训练时，性能表现最好的一组，做可视化表征，如图 5-2（a）模型准确率最终浮动在 73%左右，图 5-2（b）表现损失函数收敛情况。



（a）准确率随训练轮次的变化　　（b）损失值随训练轮次的变化

图 5-2 基于变分自编码器的轨迹用户链接模型训练可视化表征

从以上表述的这些结果中，可以发现在挖掘轨迹数据、探索人类移动模式方面，对轨迹数据分层次地建模有效提取了潜在语义，通过变分自编码器引入隐式变量，并进行变分推断，在上述评价标准中产生了不错的性能进步，总结如下：

本文基于变分自编码器的轨迹用户链接模型在大多数指标上表现最佳。这一出色的结果是由于它具有学习复杂的轨迹潜在分布并且利用未标记数据的能力。通过在概率生成模型中建模轨迹的分布，本模型能够捕获人类运动模式的基本语义。另外，通过将未标记的数据合并到训练中，本模型中的半监督分类器可以缓解基于地理数据的移动社交媒体应用数据固有的数据稀疏性问题。但是，所学习到的潜在表示通常效果不佳，尤其是在数据量严重稀疏时，例如：在 Brightkite 中 $|\mathcal{U}| = 34$。我们推测这部分是由于编码器产生的纠缠表示（一种具有相互依赖

的复杂化表示），这导致难以表示相对较小且稀疏的数据集的变化。

当更具体地关注本模型和基于一般循环神经网络的模型之间的比较时，我们注意到由于分层轨迹建模而产生了效果的提升。尽管后者使用可变循环神经网络，但是它们在对签到序列建模时，受限于较浅层的特征提取和生成，导致识别和链接效率的下降。

当涉及到各种基于深度学习的轨迹用户链接方法的训练过程时，上述表 5-2、表 5-3、表 5-4 和图 5-2 中的性能表现表明了深度学习在轨迹数据挖掘的性能。这证明了固有生成模型在理解人类运动特性方面的有效性。其他指标和其他数据集也具有类似的结果，但受限于时间，在此省略。

6 总结与展望

本文提出了基于变分自编码器的轨迹用户链接模型，这是一种用于挖掘人类移动模式的生成模型，旨在通过引入半监督学习机制的变分自编码器来学习用户轨迹的隐式层次结构并缓解数据稀疏性问题。本模型在变分自编码器框架中结合循环神经网络的分层轨迹建模来制定轨迹用户链接问题的解决方案。与现有方法相比，本文基于变分自编码器的轨迹用户链接模型大大提高了解决轨迹用户链接问题的性能。

实际上，本文提出的基于变分自编码器的轨迹用户链接模型，还可能有其他几种变形，例如：解码器可以使用可能进一步提高效率的卷积神经网络代替，因为它只需要对离散数据进行一维卷积，例如 Yang 等人使用的空洞卷积神经网络解码器用于文本建模[23]。本文，在轨迹数据链接工作中，着重于推断人类签到数据中的变量分布并提高轨迹用户链接的性能。寻找基于变分自编码器的轨迹用户链接模型的其他高效方案是建立在本文之上的未来的工作，可以通过在潜在空间中合并其他有效信息，例如空间和时间信息，来增强变分自编码器的推断能力，提高轨迹用户链接准确率，我们将其留给未来进一步的研究。

7 致　谢

由衷地感谢指导老师孙鹤立教授在毕业设计中的规划和指导，感谢曹晨学姐对我在模型建立和论文撰写上的耐心指导和中肯帮助，感谢父母在毕业设计期间的照顾和理解。

8 参考文献

[1] Chao Zhang, Keyang Zhang, Quan Yuan et al. Gmove: Group-level mobility modeling using geo-tagged social media[C]. ACM SIGKDD, 2016.

[2] Qiang Liu, Shu Wu, Liang Wang, et al. Predicting the next location: a recurrent model with spatial and temporal contexts[C]. AAAI, 2016.

[3] 潘奇明，程咏梅．基于隐马尔可夫模型的运动目标轨迹识别［J］．计算机应用研究．2008．25（7）：1988-1990．

[4] 杨丽，吴雨茜，王俊丽等．循环神经网络研究综述［J］．计算机应用．2018．38(S2)：1-6．

[5] Wu Hao, Chen Ziyang, Sun Weiwei, et al. Modeling trajectories with recurrent neural networks[C]. IJCAI. 2017.

[6] Qiang Gao, Fan Zhou, Kunpeng Zhang, et al. Identifying Human Mobility via Trajectory Embeddings[C]. IJCAI. 2017.

[7] Huayu Li, Yong Ge, Defu Lian, et al. Learning User's Intrinsic and Extrinsic Interests for Point-of-Interest Recommendation: A Unified Approach[C]. IJCAI. 2017.

[8] Junyoung Chung, Kyle Kastner, Laurent Dinh, et al. A Recurrent Latent Variable Model for Sequential Data[C]. NIPS. 2015.

[9] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, et al. A Hierarchical Latent Variable Encoder-Decoder Model for Generating Dialogues[C]. AAAI. 2017.

[10] Doersch C. Tutorial on variational autoencoders[J]. arXiv. 2016 preprint arXiv:1606.05908.

[11] Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes[C]. ICLR. 2014.

[12] Fan Zhou, Qiang Gao, Goce Trajcevski, et al. Trajectory-User Linking via Variational AutoEncoder[C]. IJCAI. 2018.

[13] Shanshan Feng, Gao Cong, Bo An, et al. POI2Vec: Geographical Latent Representation for Predicting Future Visitors[C]. AAAI. 2017.

[14] Zheng Yu. Trajectory data mining: an overview. ACM Transactions on Intelligent Systems and Technology. 2015 May 12;6(3):1-41.

[15] Fengli Xu, Zhen Tu, Yong Li, et al. Trajectory Recovery From Ash - User Privacy Is NOT Preserved in Aggregated Mobility Data[C]. WWW. 2017.

[16] Qiang Gao, Fan Zhou, Kunpeng Zhang, et al. Identifying Human Mobility via Trajectory Embeddings[C]. IJCAI. 2017.

[17] Chung J, Gulcehre C, Cho K, et al. Empirical evaluation of gated recurrent neural networks

on sequence modeling[J]. arXiv preprint arXiv:1412.3555. 2014 Dec 11.

[18] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, et al. Semisupervised Learning with Deep Generative Models[C]. NIPS. 2014.

[19] Zhao S, Song J, Ermon S. Learning hierarchical features from deep generative models[C]. Proceedings of the 34th International Conference on Machine Learning-Volume 70 2017 Aug 6 (pp. 4091-4099).

[20] Xu W, Sun H, Deng C, et al. Variational autoencoder for semi-supervised text classification[C]. AAAI. 2017.

[21] Zhai S, Zhang ZM. Semisupervised autoencoder for sentiment analysis[C]. AAAI. 2016.

[22] Bowman S R, Vilnis L, Vinyals O, et al. Generating Sentences from a Continuous Space[J]. Computer Science. 2015.

[23] Yang Z, Hu Z, Salakhutdinov R, et al. Improved Variational Autoencoders for Text Modeling using Dilated Convolutions[J]. ICML. 2017.