基于社交属性的时空轨迹语义分析

殷浩腾^{1,2}, 刘洋^{1*}

1. 山东大学计算机科学与技术学院, 济南 250101

2. 北京大学前沿交叉学科研究院, 北京 100871

* 通信作者. E-mail: yliu@sdu.edu.cn

收稿日期: 2017-04-07; 接受日期: 2017-05-10; 网络出版日期: 2017-08-16

国家重点基础研究发展计划 (973) (批准号: 2015CB352502)、国家自然科学基金 (批准号: 61272092, 61572289) 和山东省自然科学基金 (批准号: ZR2015FM002, ZR2016FB14) 资助项目

摘要 时空数据具有多维关联特性, 而深度学习恰恰因其能够对复杂高维数据进行高层抽象处理而备受关注. 本文依据轨迹数据特征给出其形式化定义, 并据此构建基于 Word2vec 的时空语义轨迹模型. 通过模型网络训练位置特征向量, 对不同时间粒度下的用户轨迹进行语义探究. 实验中采取 Top-K 近邻预测和聚类分析等手段验证了轨迹模型在无监督式学习下输出的位置向量具备空间语义且定型良好. 其结果也进一步检验了基于词向量的语言模型迁移至轨迹挖掘的研究具备可行性.

关键词 社交网络, 时空轨迹, 语义分析, 特征向量, 深度学习

1 引言

随着基于位置的社交网络 (location-based social networks, LBSNs) 的兴起, 人们越发热衷于以在线签到的形式将个人地理信息 (如 GPS 坐标、景点地标等) 分享在社交平台上. 由此产生的“签到”数据一般由个人标识、空间位置及时间戳等多元组构成, 愈发展现出巨大的发掘潜力. 依据签到数据可以再现用户兴趣点的时空序列轨迹. 大规模的用户移动轨迹具有群体行为特征, 结合用户的社交属性可以深入探索用户活动趋势, 进而能够预测社会群体的行为模式, 发掘其潜在逻辑.

基于用户兴趣点 (point of interest, POI) 签到数据的研究近年来获得极大关注: 在可视化方向, 基于 POI 绘制社会迁徙图谱¹⁾ 揭示了在不同时空背景下的人流宏观移动趋势; 在数据挖掘领域, 通过 POI 数据对用户活动的时空偏好进行建模^[1]、基于时间感知进行目标位置推荐^[2] 等研究层出不穷. 传统的用户轨迹挖掘主要基于 Markov 模型^[3]、频繁统计模式^[4] 及改进的多阶 Markov^[5] 或隐 Markov 模型^[6] 等方法, 但大部分方法局限于表面描述而忽略了时空数据的深层内涵. 利用话题模型

1) 百度地图, 迁徙人气 —— 全国人口的迁徙路书. <http://qianxi.baidu.com/>.

引用格式: 殷浩腾, 刘洋. 基于社交属性的时空轨迹语义分析. 中国科学: 信息科学, 2017, 47: 1051–1065, doi: 10.1360/N112016-00310
Yin H T, Liu Y. Semantic analysis of spatial temporal trajectory in LBSNs (in Chinese). Sci Sin Inform, 2017, 47: 1051–1065, doi: 10.1360/N112016-00310

进行用户移动模式发现的研究也具有相当的代表性, 如采用 LDA (latent Dirichlet allocation) [7,8] 和 ATM (author topic model) 等主题模型直接挖掘轨迹的隐含信息 [9]. 受限于模型先验分布难以获取等因素的影响, 其近似推断效果往往不佳. 近期, 一些研究将机器学习和神经网络的方法应用到轨迹挖掘中. 例如, 张量模型 (tensor model) [10]、生成模型 (generative model) [11] 和嵌入模型 (embedding model) [12~14]. 张量模型通过张量模拟捕获用户潜在层次偏好, 构建细粒度的感知框架; 生成模型则依赖用户习惯的分布假设确定特征空间; 嵌入模型的核心是将轨迹的上下文关联信息嵌入高维空间, 进行多语境层级的轨迹挖掘分析. 上述方法均不同程度地引入了复合元组来表征用户轨迹的高维信息, 但尚未涉及轨迹时空特性的关联探究. 当前轨迹挖掘课题面临不小的挑战: 面对实时产生的大量用户数据, 传统统计概率模型往往只关注时空轨迹数据本身; 而主题模型等复杂方法先天受限, 无法兼顾数据的时空特性; 已应用的神经网络方法较多关注于用户轨迹的推荐预测, 尚未深入发掘轨迹的语义特性及关联时空因素.

本文受到基于词向量的神经网络模型 Word2vec [15] 的启发: 尝试采用深度学习的方法挖掘具备社交属性的用户签到数据, 结合用户标记、位置点及时间序列等多维数据属性探索时空轨迹的语义特征. 所提出的轨迹模型综合考虑了轨迹挖掘中的空间转移和时序因素: 通过不同时间粒度的划分将时间属性内嵌于位置样本中, 再实施多维数据的组合单位向量化操作; 在向量空间模型 (vector space model, VSM) 中实现定量计算分析, 进而开展用户轨迹的挖掘与处理工作; 在位置向量自定义运算的基础上, 采用 Top- K 近邻预测、聚类分析等手段验证模型的合理性, 以达成时空轨迹语义分析的目标. 轨迹数据的语义特征化表达将时空数据中隐含的用户移动模式与宏观趋势直观地展现出来, 通过调整轨迹模型的参数设置即可实现对用户轨迹的定制化深度探索.

本文的主要贡献如下: (1) 基于时空轨迹的多维属性给出其形式化定义及提取方式, 其基本元兼顾时间和空间属性; (2) 基于 Word2Vec 的神经网络模型首次提出时空语义轨迹模型, 通过该模型训练生成的位置向量进行轨迹分析处理; (3) 在不同的签到数据集上进行实验, 验证了模型训练学习的鲁棒性及映射向量的语义特性.

2 时空语义轨迹模型

2.1 用户时空轨迹

2.1.1 语义概念表示

用户运动轨迹与文本句式具有高度的结构相似性, 均为由离散状态的基本元构成的连续序列. 利用自然语言向量化的类似过程, 可将时空轨迹中的签到地点视作“单词”处理从而获取位置向量. 本文采用 Hinton 于 1986 年提出的“Distributed representation”方法 [16], 将其将单词映射为固定维数向量. 该方法生成的词向量一般都蕴涵一定的空间特性: 所有的词向量共同组成一个词向量空间, 而空间中词向量之间的“距离”(可自行定义)可作为词汇间的相似度量. 采用相同方法对位置序列进行向量化训练, 即可获得具有一定“语言意义”的位置向量.

2.1.2 轨迹形式化定义

时空轨迹记录了用户在连续时段内的空间移动过程, 包含用户标识、空间坐标及时间戳等信息.

因此,可将用户轨迹形式化地描述为:给定一条由 n 个位置 L_i 构成的单一用户轨迹 T 为

$$T = [L_{t_1}, L_{t_2}, \dots, L_{t_n}], \quad t_1, t_2, \dots, t_n \in [t_{\text{sta}}, t_{\text{fin}}],$$

其中, t_{sta} 为采样开始时刻, t_{fin} 表示采样结束时刻. $L_{t_1}, L_{t_2}, \dots, L_{t_n}$ 依次表示轨迹中各签到点在时刻 t_i 下所对应的位置 L_i . 各签到点按时序排列, 描述了用户从前一位置 L_i 在时间间隔 $(t_{i+1} - t_i)$ 后转移至下一位置 L_{i+1} 的移动过程. 轨迹 T 将用户在整个采样周期内的所有签到点汇集为一条完整轨迹作为训练样本, 将时间与空间信息结合起来, 最大限度地降低了信息损耗.

类似地, 单位时间内的用户分段轨迹 $T(\Delta t)$ 为

$$T(\Delta t) = [L_{t'_1}, L_{t'_2}, \dots, L_{t'_n}], \quad t'_1, t'_2, \dots, t'_n \in \Delta t.$$

轨迹 $T(\Delta t)$ 将用户在采样周期内轨迹按标准时间间隔 Δt 进行切分, 将单独的分段轨迹作为独立的训练样本.

进一步推广, 将用户轨迹按更细粒度且蕴含特殊含义的时间片 Δt_m (如节假日、早中晚时段等) 进行切分. 则具有语义特征的用户分段轨迹 $T(\Delta t_m)$ 为

$$T(\Delta t_m) = [L_{t''_1}, L_{t''_2}, \dots, L_{t''_n}], \quad t''_1, t''_2, \dots, t''_n \in \Delta t_m.$$

2.2 轨迹模型构建

获取用户轨迹后需要对其建模以完成语义分析, 下面以单一用户轨迹 T 为例构建时空语义轨迹模型.

传统的统计语言模型是表征语言基本单位的概率分布函数. 类似地, 对于用户轨迹的合理刻画即为求取状态转移之间的概率, 即轨迹中位置点共现的可能性. 则可将签到位置的概率分布作为用户轨迹的生成模型, 利用概率公式 $p(T) = \prod_{i=1}^n p(L_{t_i} | \text{Track}_i)$ 来计算用户轨迹状态概率. 为了简化上述模型, 基于 Markov 假设 (Markov assumption) 可使用 N-gram 模型将上述计算近似为 $p(L_{t_i} | \text{Track}_i) = p(L_{t_i} | L_{t_{i-n+1}}, L_{t_{i-n+2}}, \dots, L_{t_{i-1}})$ [17]. 由于 N-gram 模型受制于样本规模和计算复杂度, 且无法表达样本间的结构相似性, 显然不能满足表征轨迹语义性的需要. 为改进传统模型, Bengio 等 [18] 提出了神经网络语言模型 (neural network language model, NNLM). Word2vec 在 NNLM 模型神经网络的基础上去除了非线性及所有词共享的隐层 [19]. 为了计算用户轨迹状态概率 $p(L_{t_i} | \text{Track}_i)$ 可将 Word2vec 模型进一步拓展为基于时空数据、由 3 层神经网络构建的用户语义轨迹表示模型 Loc2Vec, 其网络结构如图 1 所示.

Loc2Vec 模型的基础结构由输入、投影和输出 3 层构成. 下面以轨迹序列中随机选取的位置 L_{t_i} 前后 $2c$ 个位置序列 ($\text{Context}_{i-c}^{i+c} L_{t_i}, L_{t_i}$) 作为训练样本进行说明.

输入层: 包含 $\text{Context}_{i-c}^{i+c} L_{t_i}$ 中 $2c$ 个位置的随机初始化向量序列 $L_{t_{i-c}}, \dots, L_{t_{i-1}}, L_{t_{i+1}}, \dots, L_{t_{i+c}} \in \mathbb{R}^m$ (m 表示向量的维度).

投影层: 将输入层的 $2c$ 个位置向量 L_{t_i} 进行求和操作, 生成由 m 个位点构成的向量 SL, 即 $\text{SL} = \sum_{k=i-c}^{i+c} \text{Context}(L_{t_k}) \in \mathbb{R}^m$.

输出层: 其结构为一颗完全二叉树, 是由轨迹集合中出现过的位置作为叶子节点, 以各位置在训练集合中出现的次数作为权值所构造的 Huffman 树. 树中每条分支均可视作一次二分类, 建树即为多个二分类拟合多分类的过程. 此处将非叶子节点的左子节点划为负例, 编码为 1, 即 $h_i = 1$; 右子节点为正例, 编码为 0, 则 $h_j = 0$. 该树的叶子节点共有 N 个, 即 $N = 2c$, 每个叶子结点即为位置特征向量; 非叶子结点共有 $N - 1$ 个, 同为归一化形式, 但其本身并不具有任何语义含义.

图 1 (网络版彩图) Loc2Vec 模型网络结构图

Figure 1 (Color online) Loc2Vec model network structure diagram

2.3 参数估计与求解优化

利用极大似然法可基于给定的观察数据评估模型参数. 故可在样本独立同分布的前提下, 依据样本集构建目标函数、最大化其对数似然, 似然函数取得最大值时的参数值即为估计值. 基于上述过程, 可假设 Loc2Vec 模型中的一条轨迹 T 由 n 个位置点组成, 则构建以下概率函数:

$$T = \prod_{i=1}^n p(L_{t_i} | \text{Track}_i). \quad (1)$$

假设模型中轨迹集合由 m 条轨迹 T (每条轨迹 T 包含 n 个位置点) 组成, 则构建模型的目标函数为

$$\text{ST} = \prod_{i=1}^m \left(\prod_{j=1}^{n_i} p(L_{t_j} | \text{Track}_{ij}) \right). \quad (2)$$

其对数似然为

$$\text{st} = \log \text{ST} = \sum_{i=1}^m \left(\sum_{j=1}^{n_i} \log p(L_{t_j} | \text{Track}_{ij}) \right). \quad (3)$$

模型求解与优化过程根据凸优化理论由随机梯度法迭代目标函数最优解^[20]. 为方便表述, 下面以输出层 Huffman 树中的某个叶子节点 L_{t_i} 为例给出符号定义对照表 1.

表 1 输出层 Huffman 树符号参数表
Table 1 Parameters of Huffman tree in output layer

Symbol	Definition
p	The path from the root node to the leaf node L_{t_i}
l	The number of nodes contained in the path p
$p_1, p_2, \dots, p_l$	l nodes in the path p , where p_1 is the root node and p_l is the leaf node
$h_2, h_3, \dots, h_l \in \{0, 1\}$	Leaf node L_{t_i} in the path p corresponds to the Huffman Coding, where h_k represents the k th code, p_1 is not encoded
$\theta_1, \theta_2, \dots, \theta_l \in \mathbb{R}^m$	The vectorization of the non-leaf node in the path p , where θ_k represents the mapping vector for the k^{th} non-leaf node

用户轨迹状态概率 $p(L_{t_i}|\text{Track}_i)$ 计算问题的目标函数可通过投影向量 SL 及 Huffman 树的建树过程来确定, 其时间复杂度为 $O(\log_2 N)$. 在 Huffman 树中有且存在唯一的一条从根节点到叶子节点 L_{t_i} 的路径 p , p 中存在 $l-1$ 条分支. 则路径中非叶节点二分类过程的概率计算可用 sigmoid 函数表示为

$$p(h_i) = \begin{cases} \sigma(\text{SL}^T \theta) = \frac{1}{1 + e^{-\text{SL}^T \theta}}, & h_i = 0, \\ 1 - \sigma(\text{SL}^T \theta), & h_i = 1. \end{cases} \quad (4)$$

将每条分支的二分类概率连乘即可得到所求目标函数

$$p(L_{t_i}|\text{Track}_i) = \prod_{k=2}^l p(h_k|\text{SL}, \theta_{k-1}), \quad (5)$$

其中, $p(h_k|\text{SL}, \theta_{k-1}) = [\sigma(\text{SL}^T \theta_{k-1})]^{1-h_k} [1 - \sigma(\text{SL}^T \theta_{k-1})]^{h_k}$. 将式 (5) 代入式 (3) 可得

$$\begin{aligned} \text{st} &= \sum_{i=1}^m \left(\sum_{j=1}^{n_i} \log \prod_{k=2}^l \left\{ [\sigma(\text{SL}^T \theta_{i,j,k-1})]^{1-h_{i,j,k}} [1 - \sigma(\text{SL}^T \theta_{i,j,k-1})]^{h_{i,j,k}} \right\} \right) \\ &= \sum_{i=1}^m \sum_{j=1}^{n_i} \sum_{k=2}^l \left\{ (1 - h_{i,j,k}) \cdot \log [\sigma(\text{SL}^T \theta_{i,j,k-1})] + h_{i,j,k} \cdot \log [1 - \sigma(\text{SL}^T \theta_{i,j,k-1})] \right\}. \end{aligned} \quad (6)$$

为方便表述, 将式 (6) 中花括号内的内容简记为 $\text{st}(L_{t_i}, k)$, 即

$$\text{st}(L_{t_i}, k) = (1 - h_{i,j,k}) \cdot \log [\sigma(\text{SL}^T \theta_{i,j,k-1})] + h_{i,j,k} \cdot \log [1 - \sigma(\text{SL}^T \theta_{i,j,k-1})].$$

至此, 轨迹模型的目标函数构建过程已推导完成.

下面需要通过优化方法求解函数最大值. 此处采用随机梯度上升法来最大化 $\text{st}(L_{t_i}, k)$, 每迭代一次位置样本 $(\text{Context}_{i-c}^{i+c} L_{t_i}, L_{t_i})$, 就对目标函数中的所有相关参数 SL 和 θ_{k-1} 进行更新^[21].

首先考虑对参数 θ_{k-1} 的梯度计算.

已知 sigmoid 函数的导函数具有 $\sigma(x)' = \sigma(x)[1 - \sigma(x)]$ 的形式. 则有

$$\begin{aligned} \frac{\partial \text{st}(L_{t_i}, k)}{\partial \theta_{k-1}} &= \frac{\partial}{\partial \theta_{k-1}} \left\{ (1 - h_k) \cdot \log [\sigma(\text{SL}^T \theta_{k-1})] + h_k \cdot \log [1 - \sigma(\text{SL}^T \theta_{k-1})] \right\} \\ &= (1 - h_k) [1 - \sigma(\text{SL}^T \theta_{k-1})] \cdot \text{SL} - h_k \cdot \sigma(\text{SL}^T \theta_{k-1}) \cdot \text{SL} \end{aligned}$$

$$= [1 - h_k - \sigma(\text{SL}^T \theta_{k-1})] \cdot \text{SL}. \quad (7)$$

因此, θ_{k-1} 的更新公式为

$$\theta_{k-1} = \theta_{k-1} + \alpha [1 - h_k - \sigma(\text{SL}^T \theta_{k-1})] \cdot \text{SL}, \quad (8)$$

其中, α 表示学习速率, 下文相同.

接下来考虑对参数 SL 的梯度计算.

通过观察 $\text{st}(L_{t_i}, k)$ 可发现, 参数 SL 与 θ_{k-1} 具有形式上的对称性, 只需对两者交换位置即可得

$$\frac{\partial \text{st}(L_{t_i}, k)}{\partial \text{SL}} = [1 - h_k - \sigma(\text{SL}^T \theta_{k-1})] \cdot \theta_{k-1}. \quad (9)$$

SL 为轨迹 Track_i 中各位置向量的投影, 故其迭代梯度也应该分摊至每一个累加分量. 因此, Track_i 中各位置向量 L_{t_i} 的更新公式为

$$\widetilde{L}_{t_i} = \widetilde{L}_{t_i} + \alpha \sum_{k=2}^l [1 - h_k - \sigma(\text{SL}^T \theta_{k-1})] \cdot \theta_{k-1}. \quad (10)$$

模型的求解与优化过程均已明确, 下面给出随机梯度上升法求解目标函数及位置向量迭代更新的流程, 如算法 1 所示. 对于输入位置样本 ($\text{Context}_{i-c}^{i+c} L_{t_i}, L_{t_i}$) 和指定向量维度 m , 算法首先计算投影向量 SL, 初始化累积误差 e 和算符 g . 随后通过参数梯度运算更新 e , 算法遍历所有输入样本后将累积误差分摊至轨迹中各位置向量 L_{t_i} 完成迭代训练过程.

算法 1 Loc2Vec 模型训练算法

Input: Position sample ($\text{Context}_{i-c}^{i+c} L_{t_i}, L_{t_i}$), vector dimension m ;

```

1:  $e = 0$ ;
2:  $\text{SL} = \sum_{k=i-c}^{2c} \text{Context}(L_{t_k}) \in \mathbb{R}^m$ ;
3: for  $k = 2, 3, \dots, l$  do
4:    $g = \alpha[1 - h_k - \sigma(\text{SL}^T \theta_{k-1})]$ ;
5:    $e = e + g \cdot \theta_{k-1}$ ;
6:    $\theta_{k-1} = \theta_{k-1} + g \cdot \text{SL}$ ;
7: end for
8: for  $L_{t_i} \in \text{Context}_{i-c}^{i+c} L_{t_i}$  do
9:    $\widetilde{L}_{t_i} = \widetilde{L}_{t_i} + e$ ;
10: end for
```

Output: Position vector L_{t_i}

3 数据处理与算法实现

3.1 数据预处理

Foursquare²⁾ 是一个为用户提供位置发现与签到分享的移动社交服务平台. 实验从公开数据集^[1]中选取 2012 年 4 月 3 日至 2013 年 2 月 16 日期间约 10 个月的纽约地区 227428 条以及东京地区 573703 条用户签到数据作为数据集. 每条签到数据包含用户编码、街道类型名称、位置坐标及时间戳等多种属性, 添加 * 标数据采用 Foursquare 平台的哈希编码与命名规则, 其具体形式如表 2.

2) <https://foursquare.com/>.

表 2 用户签到数据样例
Table 2 Users' check-in data samples

Attribute name	CHS name	Example
User ID (anonymized)	匿名用户代码	32
Venue ID*	签到街道代码	44af9feef964a5202b351fe3
Venue category ID*	街道类型代码	4bf58dd8d48988d1c1941735
Venue category name*	街道类别名称	Mexican Restaurant
Latitude, Longitude	经纬度坐标	40.747738169430534, -73.98519814526952
Time zone offset in minutes	UTC 时差	-240
UTC time	世界标准时间	Tue Apr 03 18:15:33 +0000 2012

表 3 纽约签到地点类型表
Table 3 Check-in location types (New York)

Category	CHS name	Amount	Example
Food	餐饮类	67	Restaurant; Joint
Shop & service	商贸服务类	61	Shop; Store; Service
Outdoors & recreation	户外休闲类	26	Beach; Garden
Travel & transport	旅游交通类	22	Airport; Travel lounge
Public service	公共服务类	21	Bank; Temple
Education	教育类	20	School; College
Arts & entertainment	艺术娱乐类	20	Museum; Venue
Industry	产业制造类	7	Factory; Facility
Athletic & sport	体育运动类	4	Gym; Stadium
Community	社区类	3	Home; Neighborhood

纽约市的签到数据集中, 街道类型, 即 Venue category ID (以下简称 VCID) 共有 400 个, 分属于 251 个类别名称, 即 Venue category name (以下简称 VCN). VCN 按照功能属性可进一步划分至 10 个标准类, 详见表 3. 数据集包含 1083 名匿名用户由 38333 个签到街道代码即 Venue ID (以下简称 VID) 标记的 219037 次有效签到. 东京市的数据集中共有 247 个 VCN 覆盖 385 个 VCID, 囊括 2293 名用户由 61858 个 VID 标记的签到信息.

初步统计数据可以发现: 如图 2 所示, 在所有的签到记录中, 用户一天中两次签到间距不超过 1 km 的占到 59.4%, 约九成的距离间隔处于 [0, 10 km] 的范围. 时间间隔方面, 47.8% 的用户签到时距短于 25 min, 且随着时间的延后用户的签到意愿逐渐降低, 约七成的用户会在 120 min 内连续签到 (见图 3). 综合两个方面来看, 用户时空签到数据具有明显的短时间连续移动特性, 为接下来的轨迹分析提供了良好的样本.

经过深入分析数据结构可知: 每条签到数据的 VID 与经纬度坐标具有潜在的单一映射关系, 即通过 VID 可以唯一确定用户的运动状态. 进一步梳理发现, 数据集中既存在同一经纬度对应不同 VID (3059 例), 也存在同一 VID 对应不同的经纬度 (5332 例) 的异常情形 (样例详见表 4). 前一种情况大多是由于用户设备定位精度限制导致的位数区分不足或者同一地点受空间限制存在多条街道编码, 不会对 VID 的特异性造成实质影响; 而后一种情况, 经过对误差点进行随机抽样, 结合偏移值统计可知: 异常点通常分布在目标点周边, 96% 以上的异常点偏移直线距离小于 500 m (见图 4). 据此可基

图 2 (网络版彩图) 用户签到数据距离间隔频次分布与概率密度图

Figure 2 (Color online) Frequency distribution and probability density diagram of users' check-in data by distance

图 3 (网络版彩图) 用户签到数据时间间隔频次分布与概率密度图

Figure 3 (Color online) Frequency distribution and probability density diagram of users' check-in data by time period

表 4 用户轨迹异常数据样例
Table 4 Users' trajectory abnormal data samples

Type	VID	LAT (Latitude)	LON (Longitude)	Offset (m)
Same GPS	4b992b04f964a520726635e3	40.683120	-73.975979	0
	5089d4bce4b0f6951cdeb4f0			
Same ID	41390580f964a520dc1a1fe3	40.7420160433638	-74.005163366761	5.876
		40.7419894750989	-74.0051174588942

图 4 (网络版彩图) 异常数据误差距离统计图
Figure 4 (Color online) Error distance statistic diagram of abnormal data

本推定该误差是由设备 GPS 定位漂移造成的, 也不会干扰 VID 的标定特性. 综上所述, 本文将采用 VID 取代 GPS 坐标作为用户轨迹的定位标记.

最后需要对原始数据进行清洗, 剔除错误数据, 填补异常缺失项. 将签到数据中的缺失项根据类别属性通过历史记录进行关联补全和合理填补. 例如, 可以通过 GPS 坐标记录反向查找对应的 VID 和

表 5 主要参数样例
Table 5 Main parameters

Parameter	CHS name	Corresponding parameter	Defaults
size	向量维数	m	100
window	训练窗口大小	$2c$	5
sample	高频降采样阈值	—	$1E-3$
negative	负例采样数	—	5
threads	程序线程数	—	12
min-count	低频词截断阈值	—	5
alpha	初始学习速率	α	0.05
iter	迭代次数	—	5

VCN, 反之亦然. 剔除记录中不符合属性特征和正常阈值的异常记录并修正其中的编码命名错误; 对于时间戳等关键缺失项无法填补的记录予以直接删除. 在不影响原有数据集的基本前提下, 根据实验需求及数据特性进行妥善处理, 最大限度还原数据真实形态.

3.2 参数训练

参数调整是机器学习过程中的重要步骤. 在实际应用过程中需要结合模型算法、数据集与结果反馈对训练参数进行优化调整. 表 5 给出算法 1 实现中的主要参数与 Loc2Vec 模型参量之间的对应关系及其默认值. 在 DELL PowerEdge T630 (Intel Xeon CPU E5-2603 v3 @ 1.6 GHz, 16 G) 服务器上所运行的测试中, 算法 1 的实际平均运行速度约为每秒单线程处理 6 万个样本点.

3.3 语义分析框架

本文从两个维度对用户签到轨迹数据潜在的空间性与语义性进行探索, 图 5 为时空数据挖掘处理的流程图. 首先将签到数据进行分类抽取轨迹样本. VID 向量化后进行聚类操作, 探索签到数据的空间性质. 聚类操作将样本通过定制化的“度量”标准划分为若干子集, 依据表 3 对生成簇中 VID 归属地点类别进行细分. 通过观察簇中的地点类型分布可以获取用户的移动模式同时检验轨迹模型输出是否具备语义特性. 其次, 采集不同时间粒度的用户轨迹, 控制轨迹分割的时间精度进行对照实验. 将切分后的不同轨迹数据作为训练样本输入 Loc2Vec 模型进行训练学习. 利用模型输出的位置向量 L_{t_i} 制定代数运算规则, 以达成用户轨迹量化分析的目标.

4 用户轨迹语义分析

4.1 Top- K 近邻预测

VSM 模型可实现向量的量化比较, 则 Loc2Vec 模型生成的位置向量就能通过定量计算查找其 k 近邻作为关联预测, 其返回候选的地点类别及相关性就可以作为模型空间语义性的评判指标.

本文通过计算给定目标与候选地的位置向量间余弦相似度 (cosine similarity) 来维护一个大小为 k 的小根堆, 选取返回值最小的前 k 个候选作为预测结果. 该算法在堆中进行 $O(\log n)$ 级的插入和删除操作, 对于样本大小为 N 的数据集, 其时间复杂度为 $O(N \log k)$.

图 5 时空数据语义分析框架
Figure 5 Semantic analysis framework of spatial temporal trajectory

在东京地区数据集上以 3 种时间粒度进行目标地点 Top- K ($K = 10$) 近邻预测. 实验挑选了 Japanese Restaurant 这类具有地域性且人流量大的典型测试对象. 表 6 选取了其中比较有代表性的预测结果. 表中 Type 项表示轨迹划分类型: Single (单一用户全记录周期轨迹 T)、24 h (24 h 周期分段轨迹 $T(24\text{ h})$) 和 Workdays / Weekends (工作/节假日的分段轨迹 $T(W)$). Distance 项表示向量余弦距离, GEO distance 项则是实际地理距离 (单位为 m). Category 和 GPS 项分别表示预测位置的地点类型和经纬度坐标.

分别考察不同类型的轨迹近邻预测结果发现: 轨迹 T 中与日式餐厅距离最近的是食饮品商店和教学楼, 均位于目标覆盖范围内. 反查地图数据可知, 其本质上反映了目标周边的实际地理拓扑. 轨迹 $T(24\text{ h})$ 与 T 类似, 但通过分析候选的地点类别变化发现其中蕴含了一定意义的人群日常行为习惯. 轨迹 $T(W)$ 预测结果则展现出更明显的语义特征: 工作日 5 天的预测多为居民社区、教学楼等工作生活区, 而周末两天更倾向于休闲娱乐场所. 上述现象既反映了目标位置周边的空间结构, 又发掘了大众的行为模式.

将总体预测结果进行统计分析 (见表 7) 可发现: 轨迹 T 与 $T(24\text{ h})$ 的预测平均距离较小, 轨迹 $T(W)$ 的平均距离较大且方差更大. 总体而言, 轨迹 T 的预测结果集中反映了实际地理分布可视作基准, 而 $T(24\text{ h})$ 则揭示了大众的日常活动习惯, 即不进行非必要的远距离移动, 集中在生活、工作圈内, 故其预测方差比基准略小. 轨迹 $T(W)$ 则发掘出另一种周期行为模式, 刻画了用户在特殊时段下的移动趋势: 在工作日, 人们被束缚在城市生活圈内; 在休息日, 人们更多地倾向于走出家门、办公室进行户外游艺, 这表现在预测目标间距波动方面十分显著.

4.2 向量聚类分析

聚类可将数据进一步抽象以探索样本间的相似属性. 对有标记的签到数据进行聚类操作, 其分类

表 6 目标位置 Top- K ($K=10$) 近邻预测简表 (Japanese Restaurant)
Table 6 Top- K ($K=10$) neighbor prediction of target location (Japanese Restaurant)

Type	Distance	VID	Category	VCN	GPS	GEO distance
Single	0.674	4b546885f964a52031ba27e3	Food	Food & drink shop	35.654 139.544	24.828
Single	0.638	4bea5deb6295c9b6c05b8608	Education	College academic building	35.656 139.544	230.291
Single	0.631	4b6f7d59f964a520a7f22ce3	Education	University	35.657 139.541	400.026
24 h	0.682	4ec09a4fbe7b04923ccd270d	Outdoors & Recreation	Sculpture garden	35.657 139.544	424.540
24 h	0.678	4b698217f964a5202ea52be3	Food	Chinese restaurant	35.653 139.544	63.646
Workdays	0.680	4cca4775177c370483661534	Education	College academic building	35.658 139.544	468.477
Workdays	0.645	4ffee5dde4b04497d91e7c4c	Community	Home (private)	35.669 139.553	1806.987
Weekends	0.685	4b5a88f1f964a52036ca28e3	Shop & Service	Video store	35.652 139.546	143.953
Weekends	0.612	4cb598229c7ba35db53c8b06	Outdoors & Recreation	Bar	35.651 139.543	337.812

表 7 近邻预测样本分析表
Table 7 Neighbor prediction sample analysis

Type	AVG distance (m)	SD
Single	251.205	150.886
24 h	274.516	121.610
Workdays	462.714	489.566
Weekends	698.784	608.596

结果可以用于检验无监督训练的输出向量是否具备语义逻辑性. 亦即 Loc2Vec 模型输出的位置向量是否在向量空间内具有几何上的特征关系, 即以地点类别的语义相似性进行划分. 下面以表征类似事物与概念划分的可衡量距离来刻画空间语义性.

将纽约地区的样本 VID 向量化后以余弦距离作为度量进行 K -means 聚类, 控制生成簇总数进行梯度实验, 距离间隔设置为 5. 图 6(a) 绘制了聚类后各地点类别的绝对数量分布, 由于样本中地点类别分布本身具有不均衡性, 图 6(b) 展示了与原样本的相对比例分布. 当 $K=5$ 时, 由于组数较少引发语义划分边界模糊, 不具代表性; 当 $K=15$ 时, 组间距较大导致类别分布稀疏, 结果差强人意; 经过比较发现, 当 $K=10$ 时效果最优, 簇核相对集中, 绝对数量与相对比例表征语义性较明显, 可视作有效类别划分. 上述结果表明, Loc2Vec 模型输出的位置向量定型良好, 其空间几何特性反映了地理类别的语义性. 通过标记数据验证了模型在无监督式学习过程下的有效性, 表明无标签数据可借助本模型通过聚类操作完成自动标记.

图 7(a) 为位置向量 K -means 聚类分布的投射效果图, 图中各点表示其的实际地理坐标, 色度代表聚类后所属地点类别 Category_NYC. 可见各类的区域分布基本符合数据采样地纽约市的社会空间组织分布, 较好地反映了位置向量的空间结构性. 与此同时, 其语义性也比较明显, 各簇代表的主地点类型都显著对应于城市功能区域的划分. 为了进一步展现其在城市内部空间的集聚效应, 图 7(b) 重采样了 Top-30 的聚类簇. 抽取后的聚类效果辨析性更强, 侧面反映出 Loc2Vec 模型的训练过程较好地

图 6 (网络版彩图) 位置向量 K -means 聚类分布图 ($K=10$)

Figure 6 (Color online) Location vector K -means clustering distribution ($K=10$). (a) Absolute number; (b) relative proportion

保留了签到数据中的隐含逻辑关系, 即向量空间中的每一个位置向量都与其他的位置向量存在映射关系, 进而表现为空间中的趋同或趋异性.

为进一步衡量模型输出的鲁棒性, 将东京地区的样本 VID 向量化后采用离差平方和法 (Ward's method) 进行层次聚类, 其距离计算方式为

$$d(u, v) = \sqrt{\frac{|v| + |s|}{T} d(v, s)^2 + \frac{|v| + |t|}{T} d(v, t)^2 - \frac{|v|}{T} d(s, t)^2},$$

其中, u 是新加入包含 s 和 t 集群的节点; v 是尚未标记使用的集群; $T = |v| + |s| + |t|$; $|\cdot|$ 表示求模运算. 该方法具有无需事先指定类别数目的特点, 是自底向上的“凝聚”过程. 通过比较聚类层次的语义性同签到地点类型的关联度可反向验证位置向量的空间特性. 图 8(a) 为层次聚类的树状图, 展现了样本集群最后 15 次合并的顺序以及合并时集群的距离. 横轴为集群所包含数据点的数量, 纵轴及合并结点标注了集群间的距离. 图中大距离跨度的合并通常表明被合并的子节点分属不同类别, 据此可得出本例中聚类的临界距离 $\max_{\text{distance}} = 48.25$. 在该临界值下, 原样本可划分为 10 个集群. 恰好同

图 7 (网络版彩图) 位置向量 K -means 聚类地理映射图

Figure 7 (Color online) Location vector K -means clustering geographic map. (a) All points; (b) top-30 points

图 8 (网络版彩图) 位置向量层次聚类效果图

Figure 8 (Color online) Location vector hierarchical clustering diagrams. (a) Hierarchical clustering dendrogram; (b) clustering geographic map

人工划分的地点类别数目和所包含项目相匹配, 这表明位置向量较好地反映了其表征的空间语义性. 图 8(b) 为层次聚类分布的地理可视化效果, 依据签到到数据点的访问频次、聚类类别和 GPS 坐标分别对应其绘点大小、色阶和空间地理位置. 图中浅蓝色部分代表旅游交通类签到地点, 直观地反映了东京地区发达密集的轨道交通网及其带来的巨大人流量; 深蓝色区域则是商超便利店等服务机构的聚集地, 也反映了都市圈附近星罗棋布的商业设施分布. 具有人流量相对较小且分散的特征, 同时与交通设施相邻分布互为补充. 上述结果表明位置向量的空间语义性较好地迎合了东京地区的人流量分布和产业类型部署.

5 总结

用户的移动轨迹具有潜在的规律, 单纯依靠传统的统计挖掘方法并不能显式地提取其隐含的语义

性. 例如文献 [2] 提出的 STAP 模型为了降低问题的复杂度将高维用户轨迹数据中的时间、空间因素分开考虑, 无形中就损失了探索两者之间隐含语义特征的可能性. 本文通过构造基于多维时空数据的语义轨迹模型 Loc2Vec, 经过简单神经网络训练不同时间粒度下的用户轨迹以获取位置向量. 将对时序状态转移的概率计算转化为向量代数运算, 简化了计算采样过程, 实现了轨迹的量化分析. 在此基础上, 通过 Top- K 预测与向量聚类等方式验证了轨迹模型输出的位置向量具备空间及语义的双重特性, 检验了将深度学习领域基于词向量的语义模型迁移至时空轨迹语义分析的可行性, 是跨领域交叉研究的一次初步尝试.

参考文献

- 1 Yang D, Zhang D, Zheng V W, et al. Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs. *IEEE Trans Syst Man Cyb Syst*, 2015, 45: 129–142
- 2 Yuan Q, Cong G, Ma Z, et al. Time-aware point-of-interest recommendation. In: *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, 2013. 363–372
- 3 Nguyen N T, Phung D Q, Venkatesh S, et al. Learning and detecting activities from movement trajectories using the hierarchical hidden Markov model. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Los Alamitos, 2005: 955–960
- 4 Monreale A, Pinelli F, Trasarti R, et al. Wherenext: a location predictor on trajectory pattern mining. In: *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 2009. 637–646
- 5 He J, Li X, Liao L, et al. Inferring a personalized next point-of-interest recommendation model with latent behavior patterns. In: *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, Menlo Park, 2016. 137–143
- 6 Song L J, Meng F R, Yuan G. Moving object location prediction algorithm based on Markov model and trajectory similarity. *Comput App*, 2016, 36: 39–43 [宋路杰, 孟凡荣, 袁冠. 基于 Markov 模型与轨迹相似度的移动对象位置预测算法. *计算机应用*, 2016, 36: 39–43]
- 7 Long X, Jin L, Joshi J. Exploring trajectory-driven local geographic topics in foursquare. In: *Proceedings of the ACM Conference on Ubiquitous Computing*, New York, 2012. 927–934
- 8 Cai H N, Chen C, Wen J H, et al. Personalized location recommendation algorithm research based on user check-ins and geographical properties. *Comput Sci*, 2016, 43: 163–167 [蔡海尼, 陈程, 文俊浩, 等. 基于用户签到和地理属性的个性化位置推荐算法研究. *计算机科学*, 2016, 43: 163–167]
- 9 Wang X. Action recognition using topic models. In: *Visual Analysis of Humans*. London: Springer, 2011. 311–332
- 10 Yang D, Zhang D, Yu Z, et al. Fine-grained preference-aware location search leveraging crowdsourced digital footprints from LBSNs. In: *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing*, New York, 2013. 479–488
- 11 Yin P, Ye M, Lee W C, et al. Mining GPS data for trajectory recommendation. In: *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*, London, 2014. 50–61
- 12 Feng S, Li X, Zeng Y, et al. Personalized ranking metric embedding for next new POI recommendation. In: *Proceedings of the International Joint Conference on Artificial Intelligence*, Menlo Park, 2015. 2069–2075
- 13 Zhou N, Zhao W X, Zhang X, et al. A general multi-context embedding model for mining human trajectory data. *IEEE Trans Knowl Data Eng*, 2016, 28: 1945–1958
- 14 Xie M, Yin H, Wang H, et al. Learning graph-based POI embedding for location-based recommendation. In: *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, New York, 2016. 15–24
- 15 Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space. arXiv: 1301.3781, 2013
- 16 Hinton G E. Learning distributed representations of concepts. In: *Proceedings of the 8th Annual Conference of the Cognitive Science Society*, London, 1986. 1–12
- 17 Mikolov T. Statistical language models based on neural networks. Google, 2012. <http://www.fit.vutbr.cz/~imikolov/rnnlm/google.pdf>

- 18 Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model. *J Mach Learn Res*, 2003, 3: 1137–1155
- 19 Le Q V, Mikolov T. Distributed representations of sentences and documents. In: *Proceedings of the 31st International Conference on Machine Learning*, Beijing, 2014. 1188–1196
- 20 Boyd S, Vandenberghe L. *Convex Optimization*. Cambridge: Cambridge University Press, 2004. 466–468
- 21 Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality. *Adv Neural Inf Proc Syst*, 2013, 26: 3111–3119

Semantic analysis of spatial temporal trajectory in LBSNs

Haoteng YIN^{1,2} & Yang LIU^{1*}

1. *School of Computer Science and Technology, Shandong University, Jinan 250101, China;*
 2. *Academy for Advanced Interdisciplinary Studies, Peking University, Beijing 100871, China*
- * Corresponding author. E-mail: yliu@sdu.edu.cn

Abstract Spatial temporal data has associated multidimensional features. Deep learning has attracted much attention due to its ability to perform high-level abstraction of complex data. In this paper, we give the definition of the track-data based on its characteristics, and build a spatial temporal semantic trajectory model using Word2vec as its foundation. We explore the semantics of different user-tracks under varying time periods by training position vectors in the model network. During the experiments, we use Top- K neighbor prediction and cluster analysis to verify that the position vector has both good semantic meaning and structure. The vector is derived from a trajectory model that employs unsupervised learning. The results also test a word-vector-based language model that can be applied to the study of trajectory mining.

Keywords social networks, spatial and temporal trajectory, semantic analysis, feature vector, deep learning

Haoteng YIN was born in 1993. He received a B.S. degree from Shandong University, Jinan, in 2016. Currently, he is a graduate student at the Academy for Advanced Interdisciplinary Studies at Peking University, Beijing. His research interests include artificial intelligence and deep learning.

Yang LIU received a B.S. degree from Harbin Institute of Technology, China, and M.S. and Ph.D. degrees from York University, Canada. She is currently a professor at the School of Computer Science and Technology, Shandong University, China. Her main areas of research are data mining and information retrieval.