



(12)发明专利申请

(10)申请公布号 CN 106960256 A

(43)申请公布日 2017. 07. 18

(21)申请号 201710165013.8

(22)申请日 2017.03.17

(71)申请人 中山大学

地址 510275 广东省广州市海珠区新港西路135号

申请人 广东恒电信息科技股份有限公司

(72)发明人 王成建 印鉴

(74)专利代理机构 广州粤高专利商标代理有限公司 44102

代理人 林丽明

(51)Int.Cl.

G06Q 10/04(2012.01)

G06N 3/08(2006.01)

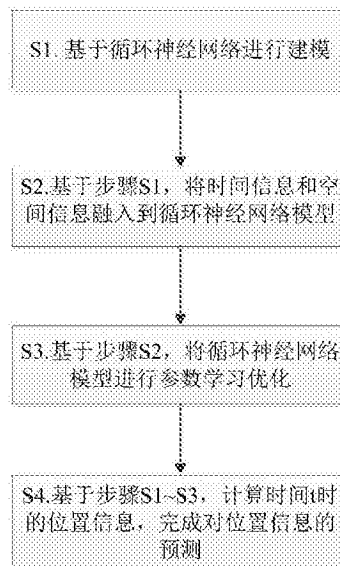
权利要求书3页 说明书7页 附图1页

(54)发明名称

基于时间空间上下文的循环神经网络预测位置的方法

(57)摘要

本发明涉及一种基于时间空间上下文的循环神经网络预测下一个时间的位置的方法;提供的方法通过融入了时间因子和空间因子信息,同时,对循环神经网络进行递归学习保留上一层状态信息,参数自主学习优化;时空循环神经网络能更好地对时空上下文建模而且提供更精准的位置预测。



1. 基于时间空间上下文的循环神经网络预测位置的方法,其特征在于:包括以下步骤:

S1. 基于循环神经网络进行建模;

S2. 基于步骤S1,将时间信息和空间信息融入到循环神经网络模型;

S3. 基于步骤S2,将循环神经网络模型进行参数学习优化;

S4. 基于步骤S1~S3,计算时间t时的位置信息,完成对位置信息的预测。

2. 根据权利要求1所述的基于时间空间上下文的循环神经网络预测位置的方法,其特征在于:所述步骤S1的具体过程如下:

S11. 设循环神经网络模型的输入序列和输出序列分别表示为: $u = (x_1, x_2, \dots, x_T)$ 、 $v = (y_1, y_2, \dots, y_T)$;

S12. 基于步骤S11,在循环神经网络模型中的每个样本都建立一个 (u, v) ;

S13. 根据循环神经网络模型的输入序列和输出序列建立一个历史访问记录库Q,历史访问记录库 $Q = \{q_{t1}, q_{t2}, \dots, q_{ti}, \dots, q_{tn}\}$,其中 q_{ti} 表示用户在时间 t_i 下访问的位置点,每个位置点具有相应的经纬度坐标 (lat, lon) ;

S14. 基于步骤S13,采用矩阵分解方法计算喜好值,设某个时间点,从历史访问记录Q中提取出时间和位置点的信息,将被预测的个体与位置点建立一个二维矩阵,其中,纵向表示被预测的个体集合,横向表示位置点,将二维矩阵分解后相乘,计算被测个体的喜好值;

S15. 基于步骤S11~S14,基于循环神经网络进行建模。

3. 根据权利要求1所述的基于时间空间上下文的循环神经网络预测位置的方法,其特征在于:所述步骤S2的具体过程如下:

S21. 基于循环神经网络基础上加入时间因素数据,引入时间循环神经网络,在二维矩阵的横轴引入时间因素;

S22. 基于步骤S21,用时间转移矩阵 T_1, T_2, \dots, T_n 计算不同时间间隔产生的影响;

S23. 基于步骤S21~S22,基于时间循环神经网络基础上加入空间距离因素数据;引入时间空间循环神经网络,在二维矩阵的横轴引入空间距离;

S24. 基于步骤S23,用空间转移矩阵 S_1, S_2, \dots, S_m 计算不同空间距离产生的影响;

S25. 基于步骤S21~S24,对循环神经网络不断加入时间因素和空间因素。

4. 根据权利要求1所述基于时间空间上下文的循环神经网络预测位置的方法,其特征在于:所述的循环神经网络还包括隐藏序列 (h_1, h_2, \dots, h_T) ,将循环神经网络中的数据信息看做神经单元,所述步骤S3的具体过程如下:

S31. 利用非线性的Sigmoid函数去表示每一个神经单元,其中Sigmoid函数的公式表示为:

$$S(x) = \frac{1}{1 + e^{-x}};$$

S32. 将所有神经单元联系到一起构成复杂的函数网,以函数的迭代与组合构建出复杂的输出,形成一个神经网络;

S33. 循环神经网络的隐藏序列节点之间增加互联;

S34. 循环神经网络存在4条信息流,信息流1从输入序列到达隐藏序列、信息流2从隐藏序列到达输出序列、信息流3从输出序列返回隐藏序列、信息流4包括隐藏序列内的神经单元之间的自连与互连;

S35. 对整个循环神经网络使用梯度下降法则,通过求导进行优化;对神经网络所产生的导数问题使用链式法则进行处理,从而达到学习优化;

S36. 重复步骤S31~S35,进行循环学习优化。

5. 根据权利要求1所述的基于时间空间上下文的循环神经网络预测位置的方法,其特征在于:所述步骤S4的具体过程如下:

S41. 基于步骤S3,被测本体在时间t时的位置点可以表示为:

$$\mathbf{h}_{t,q_t^u}^u = f \left(\sum_{q_{t_i}^u \in Q^u, t-\omega < t_i < t} \mathbf{S}_{q_{t_i}^u - q_t^u} \mathbf{T}_{t-t_i} \mathbf{q}_{t_i}^u + \mathbf{C} \mathbf{h}_{t-\omega, q_{t-\omega}^u}^u \right);$$

其中,S表示空间上下文矩阵,T表示时间上下文的矩阵,C是共享参数,通过不断迭代和反向传播进行参数学习优化C;上式的地理上的距离可以计算为一个欧几里德距离:

$$q_t^u - q_{t_i}^u := \left\| \mathbf{x}_t^u - \mathbf{x}_{t_i}^u, \mathbf{y}_t^u - \mathbf{y}_{t_i}^u \right\|_2$$

其中u为指定用户,x_t和y_t分别表示时间t用户u的地理位置的经度和纬度坐标;

S42. 基于步骤S41,可预测被测本体在时间t到位置点q可表示为:

$$o_{u,t,v} = (\mathbf{h}_{t,q_v}^u + \mathbf{p}_u)^T \mathbf{q}_v;$$

其中,p_u表示稳定本体的表征,表明该本体的兴趣和活动范围,而h参数捕获的是在基于时空上下文的动态兴趣;

上面的C参数和h参数都可以根据个人喜好随便初始化一个值(在0到1之间),模型会通过数据不断学习优化参数的值;

S43. 本模型加入贝叶斯个性化排序(BPR),BPR假设用户对访问过的位置点的喜好值要大于未访问位置点的喜好值,即:

$$p(u, t, v > v') = g(o_{u,t,v} - o_{u,t,v'});$$

其中,v'表示位置负样本,而g(x)是一个非线性函数且g(x) = 1/(1+e^{-x});

S44. 将负对数似然,解决以下目标函数J等价:

$$J = \sum \ln(1 + e^{-(o_{u,t,v} - o_{u,t,v'})}) + \frac{\lambda}{2} \|\Theta\|^2;$$

其中,Θ = {P,Q,S,T,C}表示所有的参数被估算,λ是一个参数来控制正则化的权重;而相对于参数的值可以计算为J的推导:

$$\frac{\partial J}{\partial \mathbf{p}_u} = \sum \frac{(\mathbf{q}_{v'} - \mathbf{q}_v) e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}} + \lambda \mathbf{p}_u;$$

$$\frac{\partial J}{\partial \mathbf{q}_v} = - \sum \frac{(\mathbf{h}_{t,q_v}^u + \mathbf{p}_u) e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}} + \lambda \mathbf{q}_v;$$

$$\frac{\partial J}{\partial \mathbf{q}_{v'}} = \sum \frac{(\mathbf{h}_{t,q_{v'}}^u + \mathbf{p}_u) e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}} + \lambda \mathbf{q}_{v'};$$

$$\frac{\partial J}{\partial \mathbf{h}_{t,q_v}^u} = - \sum \frac{\mathbf{q}_v e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}};$$

$$\frac{\partial J}{\partial \mathbf{h}_{t,q_{v'}}^u} = \sum \frac{\mathbf{q}_{v'} e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}};$$

利用链式法则,使用后向传播算法(Backpropagation algorithm, BP)求导对各参数进行学习:

$$\begin{aligned}\frac{\partial J}{\partial \mathbf{h}_{t-w, a_{t-w}^u}} &= \mathbf{C}^T \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t, a_t^u}} \right), \\ \frac{\partial J}{\partial \mathbf{C}} &= \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t, a_t^u}} \right) \left(\mathbf{h}_{t-w, a_{t-w}^u} \right)^T, \\ \frac{\partial J}{\partial \mathbf{a}_{t_i}^u} &= \left(\mathbf{S}_{a_t^u - a_{t_i}^u} \mathbf{T}_{t-t_i} \right)^T \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t, a_t^u}} \right), \\ \frac{\partial J}{\partial \mathbf{S}_{a_t^u - a_{t_i}^u}} &= \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t, a_t^u}} \right) \left(\mathbf{T}_{t-t_i} \mathbf{a}_{t_i}^u \right)^T, \\ \frac{\partial J}{\partial \mathbf{T}_{t-t_i}} &= \left(\mathbf{S}_{a_t^u - a_{t_i}^u} \right)^T \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t, a_t^u}} \right) \left(\mathbf{a}_{t_i}^u \right)^T.\end{aligned}$$

当所有的梯度已经计算完毕,最后采用随机梯度下降来估计模型参数;这个过程将会重复迭代,直到实现收敛。

基于时间空间上下文的循环神经网络预测位置的方法

技术领域

[0001] 本发明涉及位置预测的技术领域,更具体地,涉及一种基于时间空间上下文的循环神经网络预测位置的方法。

背景技术

[0002] 空间和时间的上下文信息对于分析用户的行为起着关键的作用,而且有助于预测用户将要到达的下一个地点;随着互联网上有效信息的快速增长和系统的收集信息能力不断增强,越来越多基于时间和空间的上下文信息被收集在系统中,使得位置预测问题变得越来越可行和至关重要。

[0003] 如今,时空预测问题已经得到了广泛的研究,但是他们都有各自的局限性;分解个性化马尔柯夫链(Factorizing Personalized Markov Chain,FPMC)已成为序列预测最流行的方法之一,但它是基于不同的因子具有很强的独立性这个假设构,这限制了它的性能;而另一种非常流行的方法,张量分解(TF)已经被成功应用于对时间感知的推荐算法和对时空上下文信息的建模中;在张量分解算法中,时间块和地理位置均被作为张量分解的额外维度,这将会导致在预测未来行动中面临着冷启动的问题。而循环神经网络(RNN)跟FPMC和张量分解这两种算法相比,表现出非常有前景的性能;但是以上所有的方法在连续时间间隔和地理距离的建模上都存在问题。

发明内容

[0004] 本发明要解决的问题是:提供一种更精确的位置预测的基于时间空间上下文的循环神经网络预测位置的方法。

[0005] 本发明实现上述目的所提出的技术方案如下:

[0006] 基于时间空间上下文的循环神经网络预测位置的方法,包括以下步骤:

[0007] S1.基于循环神经网络进行建模;

[0008] S2.基于步骤S1,将时间信息和空间信息融入到循环神经网络模型;

[0009] S3.基于步骤S2,将循环神经网络模型进行参数学习优化;

[0010] S4.基于步骤S1~S3,计算时间t时的位置信息,完成对位置信息的预测。

[0011] 在上述方案中,基于循环神经网络进行建模,将时间因素数据和空间因素数据融入到循环神经网络模型中;然后,将循环神经网络模型进行参数学习优化,基于循环神经网络模型计算时间t时的位置信息,能更精准地预测位置。

[0012] 优选的,所述步骤S1的具体过程如下:

[0013] S11.设循环神经网络模型的输入序列和输出序列分别表示为: $u = (x_1, x_2, \dots, x_T)$ 、 $v = (y_1, y_2, \dots, y_T)$;

[0014] S12.基于步骤S11,在循环神经网络模型中的每个样本都建立一个 (u, v) ;

[0015] S13.根据循环神经网络模型的输入序列和输出序列建立一个历史访问记录库Q,历史访问记录库 $Q = \{q_{t1}, q_{t2}, \dots, q_{ti}, \dots, q_{tn}\}$,其中 q_{ti} 表示用户在时间 t_i 下访问的位置点,

每个位置点具有相应的经纬度坐标 (lat, lon)；

[0016] S14. 基于步骤S13, 采用矩阵分解方法计算喜好值, 设某个时间点, 从历史访问记录Q中提取出时间和位置点的信息, 将被预测的个体与位置点建立一个二维矩阵, 其中, 纵向表示被预测的个体集合, 横向表示位置点, 将二维矩阵分解后相乘, 计算被测个体的喜好值;

[0017] S15. 基于步骤S11~S14, 基于循环神经网络进行建模。

[0018] 优选的, 所述步骤S2的具体过程如下:

[0019] S21. 基于循环神经网络基础上加入时间因素数据, 引入时间循环神经网络, 在二维矩阵的横轴引入时间因素;

[0020] S22. 基于步骤S21, 用时间转移矩阵 T_1, T_2, \dots, T_n 计算不同时间间隔产生的影响;

[0021] S23. 基于步骤S21~S22, 基于时间循环神经网络基础上加入空间距离因素数据; 引入时间空间循环神经网络, 在二维矩阵的横轴引入空间距离;

[0022] S24. 基于步骤S23, 用空间转移矩阵 S_1, S_2, \dots, S_m 计算不同空间距离产生的影响;

[0023] S25. 基于步骤S21~S24, 对循环神经网络不断加入时间因素和空间因素。

[0024] 优选的, 所述的循环神经网络还包括隐藏序列 (h_1, h_2, \dots, h_T) , 将循环神经网络中的数据信息看做神经单元, 所述步骤S3的具体过程如下:

[0025] S31. 利用非线性的Sigmoid函数去表示每一个神经单元, 其中Sigmoid函数的公式表示为:

$$[0026] \quad S(x) = \frac{1}{1 + e^{-x}};$$

[0027] S32. 将所有的神经单元联系到一起构成复杂的函数网, 以函数的迭代与组合构建出复杂的输出, 形成一个神经网络;

[0028] S33. 循环神经网络的隐藏序列节点之间增加互联;

[0029] S34. 循环神经网络存在4条信息流, 信息流1从输入序列到达隐藏序列、信息流2从隐藏序列到达输出序列、信息流3从输出序列返回隐藏序列、信息流4包括隐藏序列内的神经单元之间的自连与互连;

[0030] S35. 对整个循环神经网络使用梯度下降法则, 通过求导进行优化; 对神经网络所产生的导数问题使用链式法则进行处理, 从而达到学习优化;

[0031] S36. 重复步骤S31~S35, 进行循环学习优化。

[0032] 优选的, 所述步骤S4的具体过程如下:

[0033] S41. 基于步骤S3, 被测个体在时间t时的位置点可以表示为:

$$[0034] \quad h_{t, q_t^u}^u = f \left(\sum_{\substack{q_{t_1}^u \in Q^u, t-w < t_1 < t}} S_{q_{t_1}^u - q_t^u} T_{t-t_1} q_{t_1}^u + C h_{t-w, q_{t-w}^u}^u \right);$$

[0035] 其中, S表示空间上下文矩阵, T表示时间上下文的矩阵, C是共享参数, 通过不断迭代和反向传播进行参数学习优化C; 上式的地理上的距离可以计算为一个欧几里德距离:

$$[0036] \quad \mathbf{q}_t^u - \mathbf{q}_{t_i}^u := \left\| \mathbf{x}_t^u - \mathbf{x}_{t_i}^u, \mathbf{y}_t^u - \mathbf{y}_{t_i}^u \right\|_2$$

[0037] 其中u为指定用户, x_t 和 y_t 分别表示时间t用户u的地理位置的经度和纬度坐标;

[0038] S42. 基于步骤S41, 可预测被测本体在时间t到位置点q可表示为:

$$[0039] \quad o_{u,t,v} = (\mathbf{h}_{t,q_u}^u + \mathbf{p}_u)^T \mathbf{q}_v;$$

[0040] 其中, p_u 表示稳定本体的表征, 表明该本体的兴趣和活动范围, 而h参数捕获的是在基于时空上下文的动态兴趣;

[0041] 上面的C参数和h参数都可以根据个人喜好随便初始化一个值(在0到1之间), 模型会通过数据不断学习优化参数的值;

[0042] S43. 本模型加入贝叶斯个性化排序(BPR), BPR假设用户对访问过的位置点的喜好值要大于未访问位置点的喜好值, 即:

$$[0043] \quad p(u, t, v > v') = g(o_{u,t,v} - o_{u,t,v'});$$

[0044] 其中, v' 表示位置负样本, 而 $g(x)$ 是一个非线性函数且 $g(x) = 1/(1+e^{-x})$;

[0045] S44. 将负对数似然, 解决以下目标函数J等价为:

$$[0046] \quad J = \sum \ln(1 + e^{-(o_{u,t,v} - o_{u,t,v'})}) + \frac{\lambda}{2} \|\Theta\|^2;$$

[0047] 其中, $\Theta = \{P, Q, S, T, C\}$ 表示所有的参数被估算, λ 是一个参数来控制正则化的权重;

[0048] 而相对于参数的值可以计算为J的推导:

$$[0049] \quad \frac{\partial J}{\partial \mathbf{p}_u} = \sum \frac{(\mathbf{q}_{v'} - \mathbf{q}_v) e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}} + \lambda \mathbf{p}_u,$$

$$[0050] \quad \frac{\partial J}{\partial \mathbf{q}_v} = - \sum \frac{(\mathbf{h}_{t,q_u}^u + \mathbf{p}_u) e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}} + \lambda \mathbf{q}_v,$$

$$[0051] \quad \frac{\partial J}{\partial \mathbf{q}_{v'}} = \sum \frac{(\mathbf{h}_{t,q_{v'}}^u + \mathbf{p}_u) e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}} + \lambda \mathbf{q}_{v'},$$

$$[0052] \quad \frac{\partial J}{\partial \mathbf{h}_{t,q_u}^u} = - \sum \frac{\mathbf{q}_v e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}};$$

$$[0053] \quad \frac{\partial J}{\partial \mathbf{h}_{t,q_{v'}}^u} = \sum \frac{\mathbf{q}_{v'} e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}};$$

[0054] 利用链式法则, 使用后向传播算法(Backpropagation algorithm, BP)求导对各参数进行学习:

$$[0055] \quad \frac{\partial J}{\partial \mathbf{h}_{t,q_{v'}^u}^u} = \mathbf{c}^T \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t,q_{v'}^u}^u} \right),$$

$$[0056] \quad \frac{\partial J}{\partial \mathbf{C}} = \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t,q_{v'}^u}^u} \right) (\mathbf{h}_{t,q_{v'}^u}^u)^T;$$

$$[0057] \quad \frac{\partial J}{\partial \mathbf{q}_{t_i}^x} = (\mathbf{s}_{q_{t_i}^x \rightarrow q_{t_i}^y} \mathbf{r}_{t_i})^T \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t_i, q_{t_i}^x}^x} \right),$$

$$[0058] \quad \frac{\partial J}{\partial \mathbf{s}_{q_{t_i}^x \rightarrow q_{t_i}^y}} = \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t_i, q_{t_i}^x}^x} \right) (\mathbf{r}_{t_i} \mathbf{q}_{t_i}^x)^T,$$

$$[0059] \quad \frac{\partial J}{\partial \mathbf{r}_{t_i}} = (\mathbf{s}_{q_{t_i}^x \rightarrow q_{t_i}^y})^T \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t_i, q_{t_i}^x}^x} \right) (\mathbf{q}_{t_i}^x)^T,$$

[0060] 当所有的梯度已经计算完毕,最后采用随机梯度下降来估计模型参数;这个过程将会重复迭代,直到实现收敛。

[0061] 与现有技术相比,本发明的有益效果是:

[0062] 本发明提供的方法通过融入了时间因素和空间因素信息,同时,对循环神经网络进行循环学习优化;时间空间循环神经网络能更好地对时空上下文建模和提供更精准的位置预测。

附图说明

[0063] 图1为本发明提供基于时间空间上下文的循环神经网络预测位置的方法具体步骤流程图。

具体实施方式

[0064] 附图仅用于示例性说明,不能理解为对本专利的限制;为了更好说明本实施例,附图某些部件会有省略、放大或缩小,并不代表实际产品的尺寸;对于本领域技术人员来说,附图中某些公知结构及其说明可能省略是可以理解的;为了更好地理解本发明专利,下面结合附图和实施案例对本发明的技术方案做进一步的说明。

[0065] 下面对本发明的具体实施方案进行描述:

[0066] 基于时间空间上下文的循环神经网络预测位置的方法,其具体步骤如图1所示:

[0067] S1. 基于循环神经网络进行建模;

[0068] S2. 基于步骤S1,将时间信息和空间信息融入到循环神经网络模型;

[0069] S3. 基于步骤S2,将循环神经网络模型进行参数学习优化;

[0070] S4. 基于步骤S1~S3,计算时间t时的位置信息,完成对位置信息的预测。

[0071] 在上述方案中,基于循环神经网络进行建模,将时间因素数据和空间因素数据融入到循环神经网络模型中;然后,将循环神经网络模型进行参数学习优化,基于循环神经网络模型计算时间t时的位置信息,能更精准地预测位置。

[0072] 在本实施例中,所述步骤S1的具体过程如下:

[0073] S11. 设循环神经网络模型的输入序列和输出序列分别表示为: $u = (x_1, x_2, \dots, x_T)$ 、 $v = (y_1, y_2, \dots, y_T)$;

[0074] S12. 基于步骤S11,在循环神经网络模型中的每个样本都建立一个 (u, v) ;

[0075] S13. 根据循环神经网络模型的输入序列和输出序列建立一个历史访问记录库Q,历史访问记录库 $Q = \{q_{t1}, q_{t2}, \dots, q_{ti}, \dots, q_{tn}\}$,其中 q_{ti} 表示用户在时间 t_i 下访问的位置点,每个位置点具有相应的经纬度坐标 (lat, lon) ;

[0076] S14. 基于步骤S13, 采用矩阵分解方法计算喜好值, 设某个时间点, 从历史访问记录Q中提取出时间和位置点的信息, 将被预测的个体与位置点建立一个二维矩阵, 其中, 纵向表示被预测的个体集合, 横向表示位置点, 将二维矩阵分解后相乘, 计算被测个体的喜好值;

[0077] S15. 基于步骤S11~S14, 基于循环神经网络进行建模。

[0078] 在本实施例中, 所述步骤S2的具体过程如下:

[0079] S21. 基于循环神经网络基础上加入时间因素数据, 引入时间循环神经网络, 在二维矩阵的横轴引入时间因素;

[0080] S22. 基于步骤S21, 用时间转移矩阵 T_1, T_2, \dots, T_n 计算不同时间间隔产生的影响;

[0081] S23. 基于步骤S21~S22, 基于时间循环神经网络基础上加入空间距离因素数据; 引入时间空间循环神经网络, 在二维矩阵的横轴引入空间距离;

[0082] S24. 基于步骤S23, 用空间转移矩阵 S_1, S_2, \dots, S_m 计算不同空间距离产生的影响;

[0083] S25. 基于步骤S21~S24, 对循环神经网络不断加入时间因素和空间因素。

[0084] 在本实施例中, 所述的循环神经网络还包括隐藏序列 (h_1, h_2, \dots, h_T) , 将循环神经网络中的数据信息看做神经单元, 所述步骤S3的具体过程如下:

[0085] 所述的循环神经网络还包括隐藏序列 (h_1, h_2, \dots, h_T) , 将循环神经网络中的数据信息看做神经单元, 所述步骤S3的具体过程如下:

[0086] S31. 利用非线性的Sigmoid函数去表示每一个神经单元, 其中Sigmoid函数的公式表示为:

[0087]
$$S(x) = \frac{1}{1 + e^{-x}};$$

[0088] S32. 将所有的神经单元联系到一起构成复杂的函数网, 以函数的迭代与组合构建出复杂的输出, 形成一个神经网络;

[0089] S33. 循环神经网络的隐藏序列节点之间增加互联;

[0090] S34. 循环神经网络存在4条信息流, 信息流1从输入序列到达隐藏序列、信息流2从隐藏序列到达输出序列、信息流3从输出序列返回隐藏序列、信息流4包括隐藏序列内的神经单元之间的自连与互连;

[0091] S35. 对整个循环神经网络使用梯度下降法则, 通过求导进行优化; 对神经网络所产生的导数问题使用链式法则进行处理, 从而达到学习优化;

[0092] S36. 重复步骤S31~S35, 进行循环学习优化。

[0093] 其中, 在本实施例中, 所述步骤S4的具体过程如下:

[0094] S41. 基于步骤S3, 被测个体在时间t时的位置点可以表示为:

[0095]
$$h_{t, q_t^u}^u = f \left(\sum_{\substack{q_{t-1}^u \in Q^u, t-w < t_1 < t}} S_{q_{t-1}^u \rightarrow q_t^u} T_{t-t_1} q_{t_1}^u + C h_{t-w, q_{t-w}^u}^u \right);$$

[0096] 其中, S表示空间上下文矩阵, T表示时间上下文的矩阵, C是共享参数, 通过不断迭代和反向传播进行参数学习优化C; 上式的地理上的距离可以计算为一个欧几里德距离:

$$[0097] \quad \mathbf{q}_t^u - \mathbf{q}_{t_i}^u := \left\| \mathbf{x}_t^u - \mathbf{x}_{t_i}^u, \mathbf{y}_t^u - \mathbf{y}_{t_i}^u \right\|_2$$

[0098] 其中u为指定用户, x_t 和 y_t 分别表示时间t用户u的地理位置的经度和纬度坐标;

[0099] S42. 基于步骤S41, 可预测被测本体在时间t到位置点q可表示为:

$$[0100] \quad o_{u,t,v} = (\mathbf{h}_{t,q_v}^u + \mathbf{p}_u)^T \mathbf{q}_v;$$

[0101] 其中, p_u 表示稳定本体的表征, 表明该本体的兴趣和活动范围, 而h参数捕获的是在基于时空上下文的动态兴趣;

[0102] 上面的C参数和h参数都可以根据个人喜好随便初始化一个值(在0到1之间), 模型会通过数据不断学习优化参数的值;

[0103] S43. 本模型加入贝叶斯个性化排序(BPR), BPR假设用户对访问过的位置点的喜好值要大于未访问位置点的喜好值, 即:

$$[0104] \quad p(u, t, v > v') = g(o_{u,t,v} - o_{u,t,v'});$$

[0105] 其中, v' 表示位置负样本, 而 $g(x)$ 是一个非线性函数且 $g(x) = 1/(1+e^{-x})$;

[0106] S44. 将负对数似然, 解决以下目标函数J等价:

$$[0107] \quad J = \sum \ln(1 + e^{-(o_{u,t,v} - o_{u,t,v'})}) + \frac{\lambda}{2} \|\Theta\|^2;$$

[0108] 其中, $\Theta = \{P, Q, S, T, C\}$ 表示所有的参数被估算, λ 是一个参数来控制正则化的权重;

[0109] 而相对于参数的值可以计算为J的推导:

$$[0110] \quad \frac{\partial J}{\partial \mathbf{p}_u} = \sum \frac{(\mathbf{q}_{v'} - \mathbf{q}_v) e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}} + \lambda \mathbf{p}_u;$$

$$[0111] \quad \frac{\partial J}{\partial \mathbf{q}_v} = - \sum \frac{(\mathbf{h}_{t,q_v}^u + \mathbf{p}_u) e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}} + \lambda \mathbf{q}_v;$$

$$[0112] \quad \frac{\partial J}{\partial \mathbf{q}_{v'}} = \sum \frac{(\mathbf{h}_{t,q_{v'}}^u + \mathbf{p}_u) e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}} + \lambda \mathbf{q}_{v'};$$

$$[0113] \quad \frac{\partial J}{\partial \mathbf{h}_{t,q_v}^u} = - \sum \frac{\mathbf{q}_v e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}};$$

$$[0114] \quad \frac{\partial J}{\partial \mathbf{h}_{t,q_{v'}}^u} = - \sum \frac{\mathbf{q}_{v'} e^{-(o_{u,t,v} - o_{u,t,v'})}}{1 + e^{-(o_{u,t,v} - o_{u,t,v'})}};$$

[0115] 利用链式法则, 使用后向传播算法(Backpropagation algorithm, BP)求导对各参数进行学习:

$$[0116] \quad \frac{\partial J}{\partial \mathbf{h}_{t,q_v}^u} = \mathbf{c}^T \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t,q_v}^u} \right);$$

$$[0117] \quad \frac{\partial J}{\partial \mathbf{C}} = \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t,q_v}^u} \right) (\mathbf{h}_{t-w,v_{t-w}}^u)^T;$$

$$[0118] \quad \frac{\partial J}{\partial \mathbf{q}_{t_i}^g} = \left(\mathbf{s}_{q_{t_i}^g - q_{t_i}^g} \mathbf{T}_{t-t_i} \right)^T \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t_i, q_{t_i}^g}^g} \right),$$

$$[0119] \quad \frac{\partial J}{\partial \mathbf{s}_{q_{t_i}^g - q_{t_i}^g}} = \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t_i, q_{t_i}^g}^g} \right) \left(\mathbf{T}_{t-t_i} \mathbf{q}_{t_i}^g \right)^T,$$

$$[0120] \quad \frac{\partial J}{\partial \mathbf{T}_{t-t_i}} = \left(\mathbf{s}_{q_{t_i}^g - q_{t_i}^g} \right)^T \left(f'(\cdot) \otimes \frac{\partial J}{\partial \mathbf{h}_{t_i, q_{t_i}^g}^g} \right) \left(\mathbf{q}_{t_i}^g \right)^T,$$

[0121] 当所有的梯度已经计算完毕,最后采用随机梯度下降来估计模型参数;这个过程将会重复迭代,直到实现收敛。

[0122] 显然,本发明的上述实施例仅仅是为清楚地说明本发明所作的举例,而并非是对本发明的实施方式的限定。对于所属领域的普通技术人员来说,在上述说明的基础上还可以做出其它不同形式的变化或变动。这里无需也无法对所有的实施方式予以穷举。凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等,均应包含在本发明权利要求的保护范围之内。

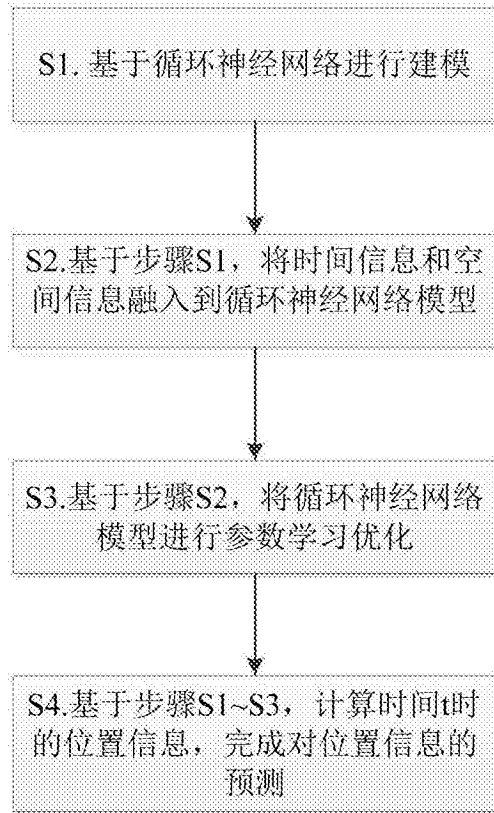


图1