

- 这篇文章主要是关于用深度学习的方法来解决蜂窝数据的地图匹配问题
- p1
 - 蜂窝数据：由手机和基站之间产生的包含位置和时间戳的信息
 - 蜂窝数据的地图匹配：将一个基站序列匹配到最合适的在路段上的路径上
 - 目前的地图匹配大部分基于HMM
- p2
 - hmm的问题：1. 人类的运动轨迹并不完全符合马尔可夫性质，下一状态可能依赖于当前和之前好几个状态（看右图）；2. 必须要用一个 $O(n^2)$ 的动态规划算法求解，效率不高
- p3
 - 所以作者就提出使用RNN的方法来做蜂窝数据的地图匹配。这样做有两个好处：一是可以考虑到多个之前的状态，减少信息丢失；二是速度快。
- p4
 - 具体要怎么表示基站序列呢？可以用独热码，但这样的话会导致训练效率不高，而且不能利用已经学习好的数据的空间信息
 - 解决方法：使用一个自编码器，但不是一般的自编码器。基础的自编码器并不能够反映空间邻近信息，因此，作者首先构建一个空间上邻近的基站集，实际上就是设置一个大小为2的窗口，对一个基站，只取前后窗口内的基站和当前基站分别构成两个基站对，然后将这两个基站对放到上面所说的基站集中（基站的序列是按什么规则排的还不是很清楚）。训练时，对于基站对 (A, B) ， A 是输入， B 是expected output，通过不断降低loss，来最大化预测出 B 的概率。那Eq 1到底是什么意思呢？实际上，因为预测出来的基站不一定是窗口内的基站，所以对于基站 x ，在预测完这个点后，结果是他的窗口内的基站的概率总和并不一定是1!!!。所以我们就要最大化预测出的基站在窗口内的概率，所以需要求和!!!而这里的求和，是对 C_x 内的所有基站求和， C_x 就是这个窗口!!!!
- p5
 - 然后是地图匹配模型，这里用到的是一个encoder-decoder的模型，目的是为了变长输入输出的问题。encoder会把一个基站序列转成一系列的隐状态，然后将最后一个隐状态作为decoder的输入。decoder则负责将编码后的基站序列解码成一系列的路段。
 - 基础的地图匹配模型存在的问题是，一个基站序列的所有信息都有最后一个隐状态这一个定长的向量来表征，序列较长的时候就会导致信息丢失，使匹配的准确率下降。解决方法是引入一个校准模型。实际上就是考虑encoder中所有的隐状态，然后再生成一个向量作为decoder的输入。对于decoder的第 i 个隐状态 h'_i ，我们根据以下公式计算该向量：
$$C_i = \sum_{j=0}^{|X|} (a_{ij} h_j)$$

其中， a_{ij} 的计算公式为

$$a_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{|X|} \exp(e_{ik})}$$

e_{ij} 表示的是encoder的第 j 个隐状态和decoder第 $i-1$ 个隐状态之间的匹配情况。
- p6
 - 作者还根据这三点对地图匹配模型做了优化：一是应该选择主干道占比较大的路径；二是要尽可能走一样的方向；三是走比较少转弯的路。
 - 具体来说就是用一个强化学习的模型来将上面这三点应用到地图匹配模型中
- p7
 - 提一下用到了HMM的数据作为ground truth

- p8
 - 提一下SnapNet w/o l是没有使用线性插值的SnapNet
 - 准确率要高于其他三种地图匹配方法，耗时比其他两种减少很多，与没有线性插值的SnapNet差距很小
- p9
 - Location representer（基站表征模型）的作用：可以看到Location representer可以比较明显地提高匹配的准确率
 - RL optimizer（基于强化学习的匹配结果优化模型）的作用：显著提高了匹配准确率
- p10
 - 提出了DMM这一个通过深度学习来做蜂窝数据匹配的框架。
 - 提出用encoder-decoder模型实现地图匹配
 - 提出了一个可以包含空间信息的基站表征模型
 - 提出了一个基于强化学习的匹配结果优化模型