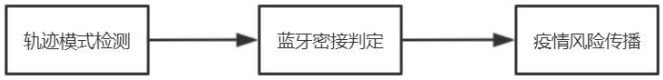
疫情接触风险分析模型设计与实现

小组成员：庄湛 11811721，陈纪元 11811810，江宇辰 11812419　　指导老师：宋轩

1. 系统流程



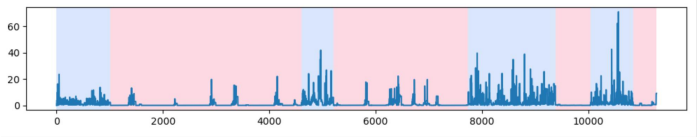
本项目实现了一个基于用户轨迹的疫情风险模型，系统流程如上。该项目风险值计算主要依据用户之间的密切接触（密接）产生的相关数据，通过考虑停留点信息、密接时长、通信强度、交通模式和当前风险值合理性，我们设计了风险值更新函数，并对 27 名活跃用户，共 11583 段轨迹、2688 条密接记录、1 个月内的数据进行了模拟推演计算和动态可视化。其中所用的主要算法都可实现实时计算，可以被利用到移动应用或宏观模拟。

中期答辩前，我们主要分析了交通模式挖掘、停留点分析、传感器数据利用等思路的可行性以及尝试了数据预处理、聚类和标注工具的设计。以下为我们终期答辩的报告内容。

2. 数据预处理

2.1 轨迹分段

首先我们依据用户打开和关闭应用的时间对轨迹数据进行粗分段，但在该分段结果中，每一分段数据仍可能存在多种交通模式的转换。因此，我们利用 ruptures 库中的**基于径向基函数的 Pelt 算法**[2]离线检测状态变更点[1]。该算法是有较低的时间复杂度，可以基于速度的变化对数据进行分段处理，对于较大型数据处理效果较好，但实际应用中我们发现其内存开销较大。如下图，该算法将一段速度的时间序列分为若干段。



图① 轨迹数据分段效果

2.2 轨迹清洗

首先，我们将每一个分段数据(长度不一)写入一个单独的 CSV 文件表，其结构如下表所示：

A	B	C	D	E	F
latitude	longitude	time	lat_change	lng_change	velocity
22.598742	113.998869	2021-01-21 22:00:16	6.479336108	1.645355882	4.6523
22.598741	113.99887	2021-01-21 22:00:17	0.111712692	0.102834742	2.0638
22.598768	113.998857	2021-01-21 22:00:18	3.016242671	1.336851653	2.9324
22.598786	113.998837	2021-01-21 22:00:19	2.010828447	2.056694852	2.9551
22.598802	113.998835	2021-01-21 22:00:20	1.787403064	0.205669485	2.3745
22.598815	113.998822	2021-01-21 22:00:21	1.452264989	1.336851653	2.1438
22.598829	113.998814	2021-01-21 22:00:22	1.563977681	0.822677941	1.9842
22.598723	113.998901	2021-01-21 22:00:23	11.8415453	8.946622605	9.67
22.598729	113.998875	2021-01-21 22:00:24	0.670276149	2.673703308	4.7695
22.598734	113.998863	2021-01-21 22:00:25	0.558563458	1.234016911	2.5359
22.598734	113.998863	2021-01-21 22:00:26	0	0	2.5359
22.598739	113.998861	2021-01-21 22:00:27	0.558563457	0.205669485	0.6984
22.598744	113.99886	2021-01-21 22:00:28	0.558563458	0.102834744	0.5525
22.598726	113.998893	2021-01-21 22:00:29	2.010828447	3.393546505	2.5381
22.598732	113.998898	2021-01-21 22:00:30	0.670276149	0.514173713	1.5526
22.598751	113.998934	2021-01-21 22:00:31	2.122541139	3.702050734	3.4834
22.598754	113.998914	2021-01-21 22:00:32	0.335138074	2.056694852	1.8051
22.598754	113.998914	2021-01-21 22:00:33	0	0	1.8051

表① 分段数据表

然后，我们将所有分段数据做了以下清洗：

1. 删除时间步长大于 2s 的段
2. 删除持续时间较短（小于 5 分钟）的段
3. 删除异常速度（大于 80m/s）的段
4. 修剪段开头和结尾的静止部分

由于我们计划使用由经纬度变化计算而来的**瞬时速度**作为分类的主要依据之一。在实验中我们发现经纬度常常有 GPS 漂移的情况，因此我们对原始的GPS经纬度数据进行了窗口大小为10s的插值运算来减轻 GPS 漂移导致的结果，并以长度为 120s 的滑动窗口在分段数据上进行滑动获取数据来训练后续的深度学习模型。

在特征选取方面，除了计算得到的速度之外，我们还计算了加速度和加加速度，它们统称为**局部变量**。我们同时还对于每一个窗口选择的所有数据进行了平均速度，速度方差，速度最大值，加速度平均值，加速度方差和加速度最大值的计算，以上特征统称为**全局变量**。

此外，根据基于蓝牙检测的密接判定得到的数据，我们整理出每条密接交互数据的信息用于后续模拟计算，其中每条密接记录包括交互时长和平均蓝牙信号强度等，具体见下表。

	user1	user2	time	duration	rssi
1	375	499	2020-12-26 19:26:40	15906	-75.890400
2	535	536	2020-12-27 00:10:53	26693	-48.297600
3	535	537	2020-12-27 13:04:48	72858	-46.205900
4	535	536	2020-12-27 15:07:58	53825	-41.799300
5	375	499	2020-12-27 15:09:46	70986	-69.782200
6	398	499	2020-12-28 11:25:53	5508	-84.558100

表② 密接数据表

2.3 轨迹标注

在对数据分段并清洗后，我们使用 Mobmap 平台[1]的选取功能结合手写 streamlit 预览应用对两千余条轨迹数据进行了交通模式（步行/公交/静止）的手动标注，其中部分标注规则如下：

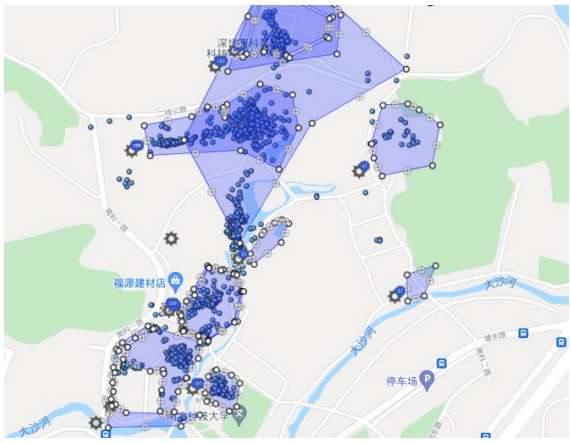
1. 沿轮廓选中经过任一密集停留区域全部轨迹，再除外经过区域外任意区域的轨迹，所剩即为步行模式数据。

2. 选中同时经过任两个较远的校巴停靠站的全部轨迹，再除外两点距离间非校巴路线的区域，所剩即为校巴模式数据[2]。

3. 对于经过部分校巴/汽车抵达不到的区域（步行道、操场等）的轨迹即为步行模式数据。

4. 平均速度大于 10m/s 的轨迹数据即为校巴模式数据，瞬时速度不超过 0.5m/s 者即为静止模式数据。

5. 观察轨迹运动，对于明显校巴轨迹或走动轨迹的数据，我们确定其模式。

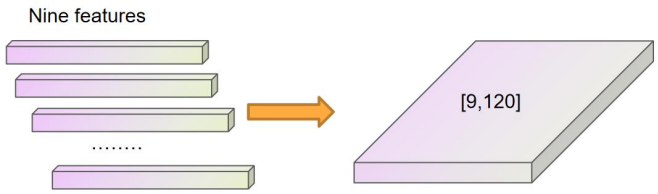


图② 数据标注的具体过程

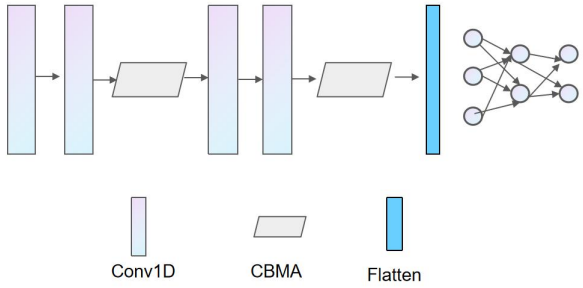
在获得部分带标注的分段数据之后，我们将使用一个深度学习模型进行学习并对剩下的约 80%的数据进行交通模式标注，具体步骤见下小节。

3. 模式分类

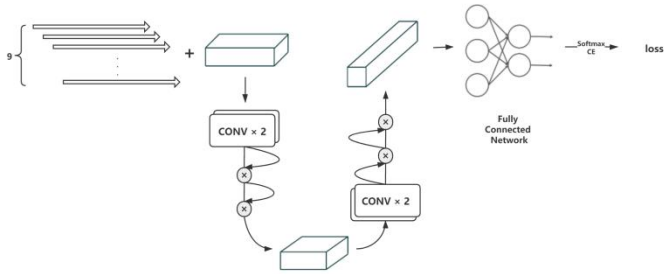
我们设计了一个神经网络模型如图④，输入为九通道（三个局部变量，六个全局变量如图③），网络组成为两个双层卷积神经网络和两层注意力机制(CBMA)以及三层全连接神经网络。



图③ 数据模型



图④ 深度学习模型



图⑤ 总训练流程

最终，训练 6 个 epoch，模型在训练集和验证集的准确率均为 88%左右。利用该神经网络对全部轨迹数据段进行分类，将得到的数据整理得到轨迹模式表格如下：

	id	stime	etime	mode
326	1048	2021-01-15 13:38:43	2021-01-15 13:50:55	1
327	1048	2021-01-15 13:50:57	2021-01-15 14:00:30	0
328	1048	2021-01-15 14:05:51	2021-01-15 18:39:52	0
329	1048	2021-01-15 18:39:54	2021-01-15 18:48:43	1
330	1048	2021-01-15 23:48:41	2021-01-15 23:57:59	0
331	1048	2021-01-16 00:02:50	2021-01-16 00:09:49	0
332	1048	2021-01-16 00:15:05	2021-01-16 00:34:08	0
333	1048	2021-01-16 00:34:10	2021-01-16 00:43:49	0
334	1048	2021-01-16 00:48:38	2021-01-16 01:41:35	0

表③ 轨迹模式数据表

4. 风险计算模型

我们设计了一种基于密接时间、密接信号强度、交通模式和原始风险值的风险值计算模型，并假设这四者对风险值的增益相互独立。则有风险值增加量计算公式为：

$$\Delta r_1 = k \cdot f(r_1, r_2) \cdot g(m_1, m_2) \cdot p(T) \cdot q(D)$$

$$\Delta r_2 = k \cdot f(r_2, r_1) \cdot g(m_1, m_2) \cdot p(T) \cdot q(D)$$

[1] Mobmap Web - 東京大学, http://webapp.mobmap.net/app/

2 注：以此分类方法判断深夜 23：00 至次日 6：00 的轨迹取消标记

其中 r_1, r_2 分别为发生密接的两用户的初始风险值其值域为 [0, 1] ，0 代表核酸检测阴性即未患病、1 代表确诊，m_1, m_2 分别为两用户的交互模式，T 为发生密接交互的时长，D 为密接交互过程中信号强度的平均值，k 为增益系数，默认为 0.1。 f, g, p, q 分别为四种特征对风险值增益影响的四个函数，下面分别介绍其选取：

1. 基于初始风险值

根据分析，保证风险值合理性，f 需满足以下特征:

$$\begin{cases} \dfrac{\partial f}{\partial x} < 0, \ \dfrac{\partial f}{\partial y} > 0 \\ f(x,1) < 1 - x \\ f(x,0) = f(1,y) = 0 \end{cases}$$

最终我们选择了 $f(x,y) = \frac{y(1-x)}{x+y}$

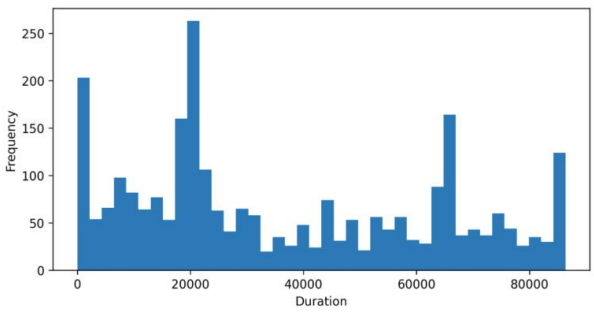
2. 基于用户交通模式

对于静止、步行、公交三种交通工具，我们根据封闭环境下更易于病毒传播的原则和考虑分类容错性，设计了如下评价矩阵 $g(m_1, m_2)$：

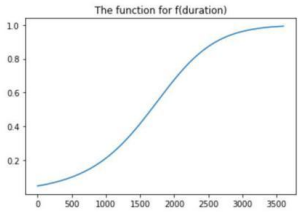
	静止	步行	公交
静止	0.4	0.3	0.6
步行	0.3	0.3	0.2
公交	0.6	0.2	0.8

3. 基于密接交互时长

基于密接交互时长的分布，如图⑥，我们将时长大于 1 小时的数据看作是增益上限，即多于 1 小时按照 1 小时计算。构造了如下函数 $p(T) = \frac{1}{1+e^{-(2-e^{1.5T})*(T-3)}}$,该函数在定义域内的图像如图⑦。



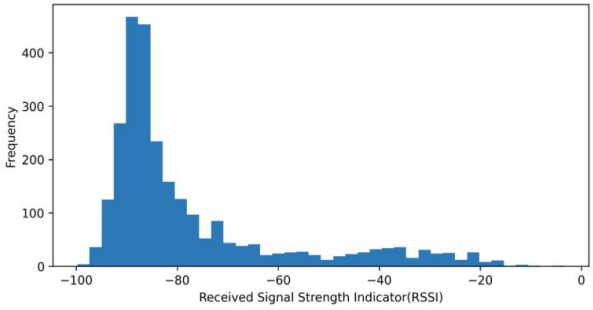
图⑥ 密接交互时长的分布



图⑦ p(T) 函数图像

4. 基于交互信号强度

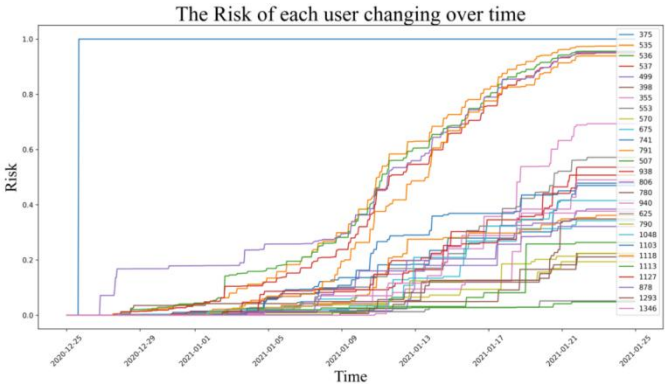
基于交互信号强度的分布，如图，我们将 RSSI 强度除以绝对值最大值后利用 sigmoid 函数向右平移中位数 $D_{median} = -85$ 个长度作为 $q(D) = \frac{1}{1-e^{-(D-D_{median})}}$



图⑧ 蓝牙信号强度的分布

5. 结果与总结

最后我们随机将一位用户（id=375）设为初始患病者，模拟计算了一个月内校园内其他用户的风险值，并记录了每位用户的风险值随时间变化情况如下图：



图⑨ 默认情况下，模拟用户风险值随时间的变化

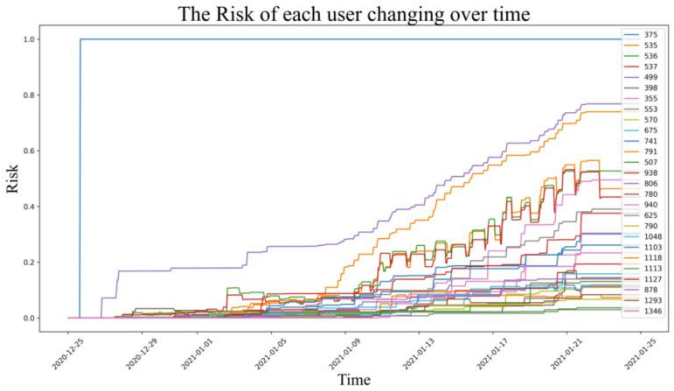
此外，除了交互风险值计算外，我们还确定和提供了两种风险值修改方式：

1. 当用户进行核酸检测或隔离检测后，若患病则风险值设为 1，若不患病则该用户风险值可以降为 0。当用户坚持填写健康申报且没有出现发病症状后，每日其风险值降为昨天的 m 倍，可设 $m = 0.9$。

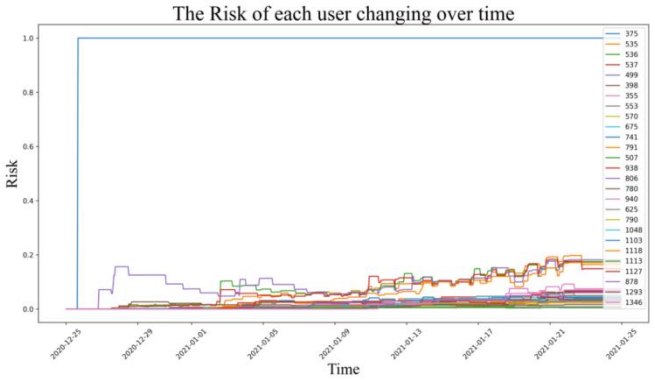
2. 若用户停留在高风险区域，则风险值可随之提高。

据此，我们还设计了以下四种情况，对比用户风险值的变化：

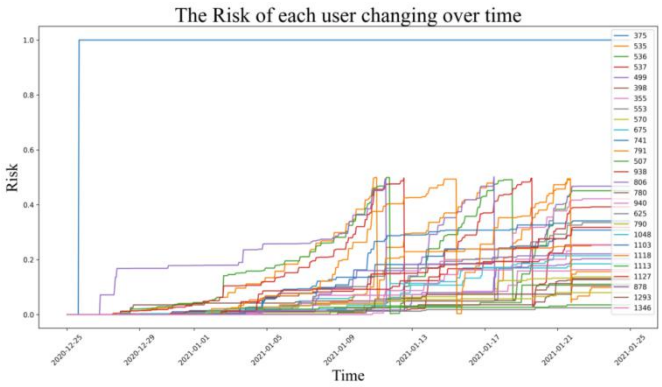
1. 50%用户填写健康申报



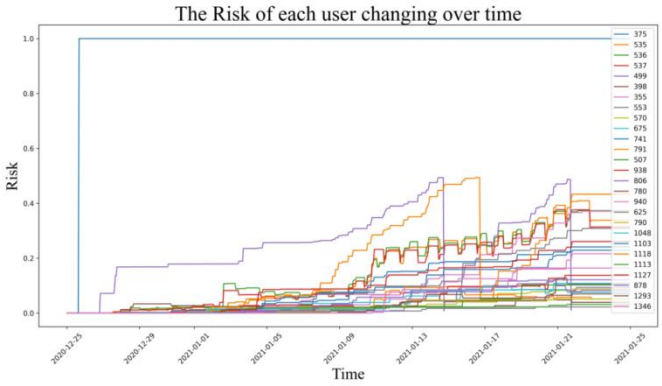
2. 全部同学填写健康申报



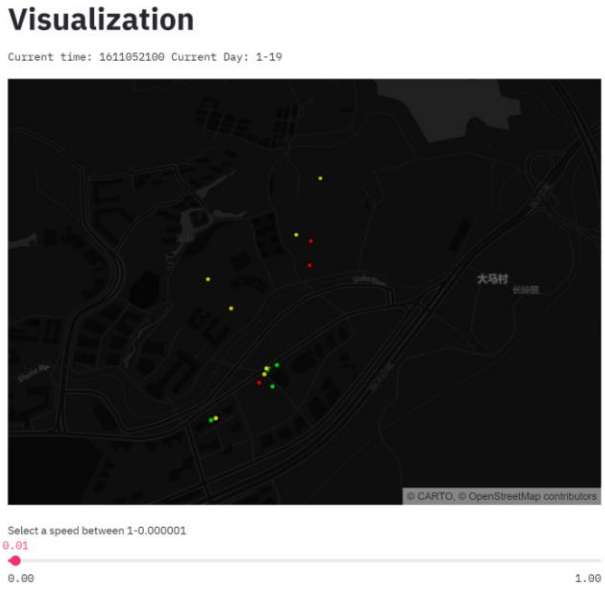
3. 所有人 risk 达到 50%时立刻核酸检测



4. 50%同学填写健康申报，所有人 risk 达到 50%时强制核酸检测



此外，我们利用 streamlit 制作了疫情模拟的动态可视化界面如下图，其中不同颜色代表风险高低。



图⑩ 疫情风险模拟的可视化

6. 贡献与不足

贡献：提供一种基于个人的疫情风险评估模型，相比于传统的地区划分风险值，更具有针对性，一般情况下可以更准确地反应个体风险情况，可以设计相关移动应用，当用户得知自身风险值较高后，建议自行去往医院进行核酸检测。对于政府来说，相比宏观控制，针对性的防疫政策也可以节省财力物力，并及时联系风险较高的民众。

最后，谈一下我们目前的**不足**之处：

1. 在交通模式检测模块中，只利用了轨迹的 GPS 信息，由于前期处理失误，没有利用手机硬件读取的各种加速度。

2. 在风险更新计算的过程中，存在一些系数，还需要专家评判。对于 f, g, p, q 四个函数可以利用泛函、变分等相关知识来选取通过增加约束保证函数的普适性。

3. 考虑构建病毒传播模型，如基于独立级联模型或线性阈值模型结合地点特征等，在病毒传播模型下，测试风险模型的有效性，进而适当调整风险模型参数。

4. 缺乏考虑风险传播的及时性，如在得知新用户确诊时，可以根据历史密接记录或多级密接记录来更新风险值。

参考文献

[1] Truong, C., Oudre, L., & Vayatis, N. (2020). Selective review of offline change point detection methods. Signal Processing, 167, 107299.

[2] Killick, R., Fearnhead, P., & Eckley, I. A. (2012). Optimal detection of changepoints with a linear computational cost. Journal of the American Statistical Association, 107(500), 1590-1598.