Managing Congregations of People by Predicting Likelihood of a Person being Infected by a Contagious Disease like the COVID Virus

Pranav Gupta
DPS International, Saket
Delhi, India
pranavgupta2603@gmail.com

Manish Gupta
IBM Global Technology Services Labs
Delhi, India
gmanish@in.ibm.com

***Abstract*—Pandemics such as Covid-19 change the status quo. The things we take for granted become no longer true. When the disease is contagious then in-person communication is fraught with danger for all parties. Despite all these dangers yet people have to meet in person for various reasons that could range from personal to official to government business. Can we have a way to understand the risk of a person being infected as compared to another person so that we can make decisions of segregating the two people or to decline entry to a person?**

In this paper we provide a framework and an approach that caters to estimating a *score* for each person based on GPS locations. A score is computed by an application running in cloud and receiving as input the GPS (Global Positioning System) trajectory information from each person's device (like a mobile or a smart watch). With the scores of any two people in hand one can predict who is more likely to be infected. We complement this approach with extensive simulation results to validate our approach. Our results show that we can achieve a high accuracy (80% to 90%) of predicting which person (of the two being compared) has an infection.

Keywords—Healthcare, COVID, Risk, Mobile, GPS, Application, Smartcities, Virus, Nonpharmaceutical

I. INTRODUCTION

Contagious diseases and Pandemics play havoc with our lives - for e.g. Spanish Virus 1918 [21] and Covid-19 [10]. Clearly people cannot sit at home as that will lead to further deterioration of the standards of living as large parts of businesses and supply-chains that they are involved with will suffer and break. So life and business must move on but move on in a cautious and intelligent manner. Despite the risks many situations demand people to congregate in a room or building in order to conduct their business. When people meet in person can we know which person is more likely to be infected as compared to another person. Knowing this can support decision-making by the event organizers and limit the spread of the infection by taking decisions such as barring some people from attending or making them sit separately.

A recent MIT article [14] talks about new ways of collecting and sharing personal data while preserving individual privacy in order to limit the impact of Covid-19. The importance of mobile phone data and non-pharmaceutical interventions, and the security and privacy issues pertaining to using mobile data are described in [18] [3]. Existing application-based approaches (for e.g., [2][17]) provide an estimate of the proximity risk about individuals who have been infected and also have explicitly declared their health condition or through the analysis of contacts of a person who has been infected. In reality the population at a given time may have a *second* category of persons who may be very recently infected or asymptomatic [22] or have no idea yet that they have contracted the disease, and therefore if we can detect with high accuracy such persons then it will be useful to better planning of in-person congregations.

In this paper, a general framework to detect the second category of persons is provided – that is those who are infected and have not yet declared their health condition or do not yet know about it. Our approach leverages the GPS location history of the people and determines through an analysis, a score, that represents the likelihood of the people being affected because of a contagious disease, and then use that score to devise a strategy for minimizing the likelihood of the spreading the contagion in a congregation. Our approach becomes very important when it is not possible to perform a test of a person for a specific disease on or before the congregation event or even when it is possible to perform the test yet the cost of performing the test may be prohibitive. Our framework becomes valuable when people catch infections from asymptomatically affected people or by touching objects that were already carrying the infection, i.e., situations making it hard to do proximity tracing as in [16]. Thus, our key contributions are:

(a) An approach to determining a relative risk of a person being affected because of coming in contact with asymptomatically affected people or infected places and objects by using only the location data.

(b) Providing a framework wherein we can minimize the likelihood that a maliciously inclined individual can fool the system by giving the device like the mobile or the smartwatch to another person and creating a false trajectory

The rest of the paper is organized as follows. We begin by providing the related work in Section II. Section III provides the formal description of the problem being addressed and the details of the framework for computing the score for each individual. We provide the details of the architecture and the method of computing the score in Section IV. The validation of the approach is provided through extensive simulation in

Section V. We conclude the paper in Section VI wherein we also discuss how our framework can be extended.

II. Related Work

At the time of writing this paper we did not find any research papers or prior work pertaining directly to the approach and framework that we provide in this paper but we have considered all the closest work that can complement or supplement our framework.

The Aarogya Setu [2] app uses a continuous Bluetooth service to establish a close range proximity between two people. When two users come close to each other, the Bluetooth connection between the two smartphones will collect information to only check whether one of the users has been tested Covid positive. If one of them is tested positive, then the other user is alerted. The disadvantage in this case is that this system will only work if one of them has declared itself Covid positive. We will discuss in the conclusions on how our approach can integrate with Aarogya Setu app. But here it suffices to say that we complement each other and our method of computing a score for a person that indicates the person's relative likelihood of being infected is not yet the objective of the Aarogya Setu app.

Fitbit [12] provides a device as part of an ongoing project that purportedly can use the sleep data, heart rate, and other parameters available from the user via the device enabling digital detection of the person having the corona virus. The details are not yet available as to whether they would use a supervised learning approach or any other method. But indeed this clearly would be another extension of the framework that we are providing in this paper. It must however be noted even though our framework allows for getting the health information such as the sleep data or the heart rate but at least in this paper we do not consider it. Only the GPS location is used to calculate the score.

A 'close contact detector' [8] has been launched that tells its users if they have been near a person who has been confirmed or suspected of having the virus. The application is promoted by the Chinese government and there is no method described on how this "close contact detection" is done.

In [4] a biosensor is described that detects the concentration of the covid-19 virus in the environment. At the time of the writing of this paper the sensor is not ready to be used in production. We also believe a lot of work is required where it can be pervasively deployed and despite that there will be a probability that it may result in false positives and negatives.

The articles [15] [11] provide insights into how AI and sensory data can be used to fight the corona virus. The one that is related but complementary to our work is the fever detection by combining computer vision and infrared to detect the forehead temperature. If such devices are available in public areas then our framework can benefit by using them to update the score (that represents the relative likelihood of a person being infected with a virus) of a person to be higher if they show fever, which is expected to improve the prediction accuracy. The article [15] also talks about how to track virus based on NLP (Natural Language Processing) by analysing web sources about the health of humans and animals.

In [9] the approach of "Contact Tracing" is defined wherein the people who come in contact with those who are infected like Ebola or Covid are tracked and who have they been in contact with in the past. Once someone is confirmed as infected with a virus, that person's contacts are identified by asking about the person's activities and the activities and roles of the other people around the person since the onset of the illness. The Contact Tracing information can actually embellish our method by updating the scores of individuals who have been known to be contacts of people who were certified to be infected.

In [20] [6] [16] "Proximity Tracing" is described. It is basically when we track individual entities (including people, animals, vehicles, devices, etc) within a given proximity to other individuals in space and time. The concept of "proximity tracing" is an important component that we leverage in our framework. Earlier work in this area does not talk about how to compute a score that allows for relative comparison of any two entities or human beings in terms of who is more likely to be infected. The "proximity tracing" tool mentioned in [20] analyses point datasets of moving entities and visually shows proximity events that can then be used to apply to contact tracing – i.e., help find potential contact events.

As mentioned earlier, the closest work to the framework we describe in our paper is the Proximity Tracing. Next we described our Theoretical Framework.

III. Theoretical Framework

A. *The Problem Addressed*

What are the chances that a person is affected by a contagious virus? Knowing the chances for every person will allow us to rank them. This ranking will become an important input to creating a plan that allocates a limited set of resources (e.g. rooms, masks, etc) to minimize the risk of the spread. A plan could be as simple as deciding whom to include or exclude in the congregation based on a predefined threshold on the rank. Another plan could be to decide how the different people should be seated, and which kind of gear (masks, etc) to distribute to whom. In another situation the ranking can be used to prioritize testing of individuals – higher rankers are first tested for the disease followed by lower rankers.

Some examples of situations where congregations are required:

- Places of worship
- Stadiums
- Concerts
- Malls & Shopping complexes
- Business parks
- Meeting rooms

We describe our approach by introducing some basic notation.

B. *Notation*

- P_i := The i^{th} person in a region (say a locality, city, etc).
- T_i := Trajectory of P_i. This will be a sequence of nodes, where, a *node* n_{ij} is the GPS location of P_i at

time t_{ij} generated by a client application running on the device (that is with the person). Note $t_{i(j+1)} > t_{ij}$.

- $I_{ij} := t_{i(j+1)} - t_{ij}$. This is the *location emit interval* from a person's device. Clearly, it is dependent on the device and could for example be around 5 mins. Our model allows for different values of I_{ij} for different combinations of indices i and j. The significance of this value is obvious - the larger the interval the more difficult it becomes to impute or interpolate the correct location of a person between any two successive nodes. And the likelihood of a person becoming affected is inversely affected by the magnitude of I_{ij} -- because we do not have information on what the person may have done during the interval $[t_{ij}, t_{i(j+1)}]$.

- C_i := A sequence of confirmations that the person P_i and the device that provides the location are together. C_{ij} is the j^{th} confirmation in the sequence for person i. If a person does not provide this confirmation then clearly our system can be fooled. We discuss in Section IV D automatic ways of confirming that the device is with the person whose likelihood we are computing. Note this is an important aspect that none of the prior work in this space considers.

- A_i := A score for P_i that represents the likelihood that the person is affected by the virus. Higher the score higher the likelihood that the person is affected. Knowing this score for any two people will allow us to compare them in terms of who is more likely to be affected. In Section V our experimental validation will really be validating this hypothesis.

- S := The minimum safe distance between any two people to remain unaffected because of each other.

- E := The error in reporting a location by any device. The error of reporting from a device can adversely affect the likelihood computations. The way our method intends to compute this is by taking multiple measurements from the device while keeping the location fixed. This can be done when the person has actually reached the congregation location and a spot can be reserved to perform this error estimation. It is true that the error may be dependent on the location and time but for simplicity we assume one value for the error.

- K := the maximum duration that an infection once caught is going to be active in a person. This will clearly depend on the contagious diseases for whom we are estimating the likelihood. For simplicity of exposition in this paper we assume that there is only one disease. In the case of Covid-19 this could range from 2 weeks to 6 weeks [19].

- Pub := is the set of all public places which are known to have many people in them. For each place in Pub we know its location as well. Pub_i will be the i^{th} public place and its location will be given by Pub_i.loc. Also an estimate of the footfall in a Pub_i is assumed to be known. For a public location there will be a radius around the Pub_i.loc denoted by Pub_i.radius, and if a person is within that radius then the person qualifies as being in that public location.

C. *Details of the Framework*

Here we discuss the framework and later discuss our architecture and method for computing a score for each person.

- Each person P_i will carry a device (say a smart mobile or a smart watch) that can run an **application client** (based on our method to be described later) which will capture some key information like the GPS location on a continual basis. The collected information can be sent to a secure server/storage in the cloud or kept locally in the device and only sent if required. Note the framework allows capturing health parameters such as sleep time, pulse rate, etc but is out-of-scope of this paper.

- Each person P_i can configure the application client to set the desired **interval** at which the trajectory information is collected.

- If a person P_i who wants to become part of a **congregation** then that person will need to send their T_i for the **last K days** to an **Analysis Server** that performs the computation of the score A_i for the persons attending the congregation. The Analysis Server can run in cloud.

- Once an estimate is available for each person the congregation's organizing body can use these scores in various ways that can lead to minimization of spread of the disease. For instance, the score can be used to decide who to bar from being part of a congregation. If masks are limited in number then based on the ranking it can be decided whom to distribute the masks (or PPE kits) to -- for example, to those who have more likelihood of being affected, i.e., those with relatively lower scores. The ranking can also help decide how to group people together and distance them from each other - for example all the top 10% of the people who have higher scores are seated far away from the remainder.

D. *A Word on Security and Privacy*

Technology (including the one in this paper) is being developed to track and monitor individuals with the aim to contain the covid pandemic however it is raising the concerns of security and privacy [14]. Many regions of the world have already defined how to process and control personal data and actually defined regulations around it (see [3]). For example, GDPR [13] or CCPA [7] are example acronyms of such regulations. These regulations make deployments of a framework like ours very challenging. Collecting fine-grained geolocation information for location tracking with a Global Positioning System (GPS) qualifies as low-risk personal information, but when it is combined with the person's name it is considered as high-risk personal information. As mentioned in [3] we can reduce risk overall by: (a) Ensuring that the Analysis Server complies with the regulations like GDPR, (b) The server should delete the user data after the aggregation event is over, and, (c) while the personal information is in the server it should be encrypted, and any transmission of the information from the client application to the server should also be encrypted. A detailed discussion on security and privacy is out of scope of this paper.

IV. Architecture, Design & Modeling

We first present the architecture a semblance of which is given in the previous section. Thereafter, we discuss the method of estimating the 'score' A_i for a person.

A. *Architecture*

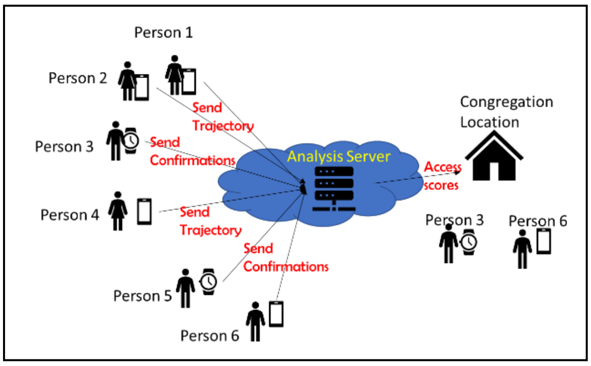

Figure 1: Client Server Architecture

In Figure 1 we show the architecture that brings out three important components – (a) 'Persons' that subscribe to the Analysis Server and have a device that can send trajectory T_i information at regular intervals or send confirmations C_i on demand, (b) The Analysis Server which can run in a cloud and receives all the trajectories from the persons who have subscribed to the server. For example, in Figure 2, the trajectories T_A, T_B, and T_C, respectively, for persons A, B, and C are collected by the devices and can be sent to the Analysis Server when required. It also processes these trajectories to compute a score for each user based on last K days of trajectory history. A component of analysis will revolve around analyzing the trajectories and determining proximity events (as shown in Figure 2). It also may seek information from the subscribed users to understand if the devices are with the *actual* owners (see more on this later in this section), and, (c) The organizer of the congregation event can query the *normalized* scores of the *subset* of persons who are participating in the congregation.

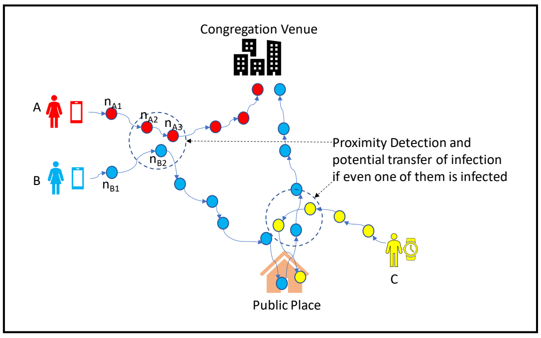
Congregation Venue
A
B
n_{A1}
n_{A2}
n_{A3}
n_{B2}
n_{B1}
Proximity Detection and potential transfer of infection if even one of them is infected
Public Place
C

Figure 2: Trajectories, Proximity Events, and Potential Infection Transfer

B. *Sources of Infected Persons in a Gathering/Event*

For a given virus strain we track the locations of people as also discussed above. Assume that the congregation that will need the scores begins at time X. At time X we assume that the subset of persons participating in a congregation are present at the venue of the conference, and have uploaded their data into the Analysis Server voluntarily. There will be infected and non-infected persons at the venue. The ones who are actually infected (See Figure 3) could comprise of (a) Type A – Asymptomatic, and, (b) Type B – Asymptomatic.

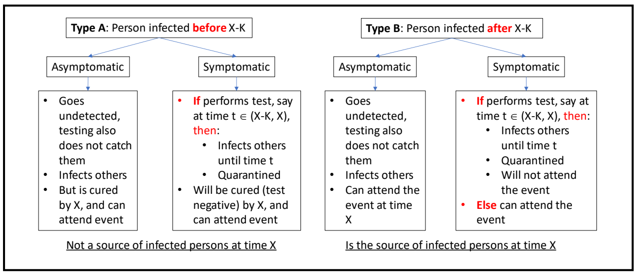
Type A: Person infected before X-K
Asymptomatic
Symptomatic
• Goes undetected, testing also does not catch them
• Infects others
• But is cured by X, and can attend event
• If performs test, say at time t ∈ (X-K, X), then:
• Infects others until time t
• Quarantined
• Will be cured (test negative) by X, and can attend event
Not a source of infected persons at time X
Type B: Person infected after X-K
Asymptomatic
Symptomatic
• Goes undetected, testing also does not catch them
• Infects others
• Can attend the event at time X
• If performs test, say at time t ∈ (X-K, X), then:
• Infects others until time t
• Quarantined
• Will not attend the event
• Else can attend the event
Is the source of infected persons at time X

Figure 3: Sources of Infected Persons

1. **Type A: Person infected <u>before</u> time X-K**: As shown in Figure 3, this is the category of persons who have become infected before time X-K. By the definition of K (see Section III B) the strain of virus or infectious disease for which we are computing the score, will not be active in these infected persons by time X. We further assume that they will not contract the virus again until after the congregation event is over after time X. Type A-Symptomatic persons will be infecting others until they perform a test and at which point in time they will be tested positive and will also be quarantined. Note that in our model (in Figure 3) only the Type A-Symptomatic persons will be tested. The Asymptomatic ones will not be performing the test but will be infecting others until time X. In any case, it must be reiterated that Type A persons will have been cured by time X.
2. **Type B**: **Person infected <u>after</u> time X-K**: As shown in Figure 3, this is the category of persons who have become infected after time X-K. They could have become infected by meeting Type A persons or some previously infected Type B persons. Some of them could have caught the infection by visiting a public place where the probability of catching the infection is higher. While some of them could have been infected because of touching or consuming an object that is carrying the virus. If the person in this category is symptomatic and has a test performed, then this person will be quarantined and will not be able to attend the event at time X. **Clearly the Type B-Asymptomatic or Type B-Symptomatic persons that have not performed a test will be the actually infected individuals at an event at time X**. It must be recognized that asymptomatic cases can infect [22] others, and range between 5% and 80% of the population [5]. Further, there are individuals in every population who may be infected (asymptomatic or not) because of touching of surfaces [22] or objects that are infected.

C. *The Method for Estimating a Person's Normalized Score A_i*

Assumptions:

1. Clock Synchronization: Whenever a GPS location is created by the client application on a person's device,

it also synchronizes with a global clock. Based on whether the clock of the device was *y* time units ahead or behind the global clock, the location's timestamp is, respectively, reduced or increased by *y*.

2. The electronic device which runs the client application and thus provides the trajectory and other information for a person P_i, should remain with the person P_i the whole time. For example, it should not be the case that P_i gives his phone (i.e., the device) to another person and the trajectory submitted therefore is not for person P_i. Later we throw light on ways of increasing the likelihood that the devices is with the rightful owner.

Definitions:

dist(loc1, loc2) := Distance between two GPS locations, say, location loc1 and location loc2.

Note that distance between any two GPS locations can be determined by using a service or a function like [1].

We first present the method and then discuss the method leveraging the notation in Section III B.

Normalized Score Estimation Method:

```
1.  For each person $P_i$:
        1.  interpolate the location of the person $P_i$ at every $t_{rh}$ where $t_{ij} < t_{rh} < t_{i(j+1)}$, and $t_{rh}$ is the time at which another person $P_r$, for $r \neq i$ reported a location $n_{rh}$. Refer to each interpolated location as $n_{ih}$. Note, these interpolated locations (by definition) are not reported from the person's device but our score-estimating method needs to create them for proximity detection.
2.  $T_o$ := sort the list of timestamps $t_{ij}$ for all i and j in the increasing order of $t_{ij}$. In this we also include the timestamps of interpolated locations as described in Step 1
3.  Initialize $A_i := 0$
4.  For each timestamp x in $T_o$:
5.      Pairs := {($n_{lx}$, $n_{mx}$): $n_{lx}$, $n_{mx}$ are locations of persons $P_l$ and $P_m$, $m \neq l$, respectively, at time x}
6.      For ($n_{lx}$, $n_{mx}$) in Pairs:
7.          If $\mathbf{dist}(n_{lx}, n_{mx}) - 2E < S$ then
8.              $A_l := A_m := \mathbf{max}(A_l, A_m) + 1$
9.      PersonLocs(x) := {$n_{px}$: person p has a location, including the interpolated one, at time x}
10.     For z in Pub:
11.         For $n_{px}$ in PersonLocs(x):
12.             If ($\mathbf{dist}(n_{px}, z.loc) - E - z.radius < S$) then $A_p := A_p + 1$
13. Let $A_{max} = \mathbf{max}_i A_i$
14. For every person $P_i$ we find the $I_{imax} := \mathbf{max}_j I_{ij}$. Also let $I_{max} := \mathbf{max}_i I_{imax}$.
15. $A_i := \mathbf{max}(A_i/A_{max}, I_{imax}/I_{max}) + A_i/A_{max} * I_{imax}/I_{max}$
```

Description of the Method:

In Step 1 we find for each person i the interpolated location of person i at timestamps where other persons have reported their locations.

In Step 2 we sort the persons based in the *ascending* order of the timestamps (including the interpolated timestamps generated in Step 1).

In Step 3 the score variables (one for each person) are initialized to 0.

The Step 4 is the beginning of the for-loop where we consider each timestamp in the sorted list T_o.

In Step 5, we consider all the *unique* combinations of persons who have locations at timestamp x in an iteration.

Then in Step 6 we take every pair at timestamp x and find out in Step 7 if the distance between the corresponding two persons l and m, after accounting for error E in each of the reported locations, is smaller than the safe distance S (see Section III B). Note that, for simplicity of exposition, we assume that the interpolated locations too have error not more than E. And if the if-condition is true then it means that the two persons are within the range to catch the infection (i.e., is a Proximity Event like in Figure 2), and thus in Step 8 we set score of each of the persons to the same value which is maximum of the *current* value of two scores but incremented by one. The rationale for Step 8 is that whenever two people come in contact, if one of them is infected the other person is likely to be infected and hence the score of the two people will be considered as identical. The increment of one is to count their coming together in a proximity event. In this exposition we have ignored for simplicity the *duration* that any two people have to be in a proximity event for the infection to be passed on, but the method can be easily enhanced.

The steps 9 through 12 account for persons visiting a public location at timestamp x. Note that even though the previous steps take care of person to person contact even in public places but we still want to separately treat public places as they might have more objects that might be infected because of the footfall, and thus increasing the chances of a person catching the infection.

In Step 9 we identify at timestamp x all the locations each of which corresponds to a person who has a trajectory node at x. Then in Step 10 we begin the for-loop for iterating over all the public locations. In Step 12 we find out if the person p is within the public place and if yes then we increment the score of the person by 1. Note that we could have chosen an increment that is proportional to the footfall in the public place but for simplicity we contend with a constant value of 1.

The steps from 13 through 15 correspond to *normalization* of the scores and also accounting for different intervals at which a person's device sends trajectory locations to the Analysis Server. Refer to the discussion of I_{ij} in the Section III B. In Step 13 we find the maximum of all the scores, i.e., A_{max}. In Step 14 we find for each person's trajectory the biggest interval I_{imax} between any two successive locations in that trajectory. Then we compute the maximum I_{max} across all the values I_{imax}.

Finally, in Step 15, we define the normalized value of the scores following a heuristic. Let's understand the rationale.

For each person i, if A_i/A_{max} is larger than I_{imax}/I_{max} (i.e., based on the previous computation steps) then A_i should be set to that large value. But if I_{imax}/I_{max} is larger i.e., the person has had a trajectory where there are big gaps between sending of the successive location information to the Analysis Server then we will still want to penalize the person proportionally, and set the value of A_i to I_{imax}/I_{max}. Hence we have the term $\max(A_i/A_{max},\ I_{imax}/I_{max})$. Now the second term A_i/A_{max} * I_{imax}/I_{max} addresses taking care of differences between any two persons having similar first term's value but different product value of the two terms, i.e., A_i/A_{max} and I_{imax}/I_{max}. Note that the maximum value of the normalized score cannot exceed 2.

D. How to Ensure that the Device is with the Rightful Owner all the Times?

In Section III B we first introduced the concept of C_i, the confirmations that must be sent in order to ensure that the Analysis Server in Figure 1 has the confidence that the device is with its owner. We now discuss further on this topic. If a person P_i is malicious and in order to fool the system gives the device to another person then the trajectory will not be of P_i but of another person. In another situation a person may just simply forget the device somewhere. These and other similar scenarios will make the computations of the scores worthless. There can be multiple ways in which it can be known if a person is not fooling the system. We present two of such ways below but of course agree that there are potentially more and better approaches possible:

- *Finger-print or Facial Recognition*: If a device has a fingerprint reader or facial recognition then the client application can randomly ask the person to authenticate using the fingerprint. If the person does not do so then that is considered as a violation. The client application on the device retries. Note that when to ask the user may not be entirely random and could depend on factors such as a detection of change in the behaviour of the received locations or lack of any movement of the device, etc. Also note that for this to work it is important that the client application knows which fingerprint or face picture is used for confirmation. On reaching the venue the user will be asked to confirm again using either the fingerprint or the facial recognition. If there is a discrepancy in either of them then it will be known that the person who had authenticated earlier is not the same as the person at the venue.

- *Voice*: Voice samples could be collected automatically by the client. On reaching the congregation the user will have to provide voice samples again for the analysis server to match with those collected earlier. The client application on the device will definitely ask the user to speak randomly a phrase, which will help confirm that the device is with the owner.

In the above, the C_{ij} for P_i can be finger-print or facial recognition validation or voice samples. How should we factor periods of time when it can't be confirmed that the device is with its rightful owner in our calculation of the scores? We address this question along with discussing one more important aspect below.

Sleep Time

An important and related aspect to the above discussion is the question: how to account for when a user goes to sleep? When a user is asleep then obviously cannot respond to authentication requests then how should we factor that in?

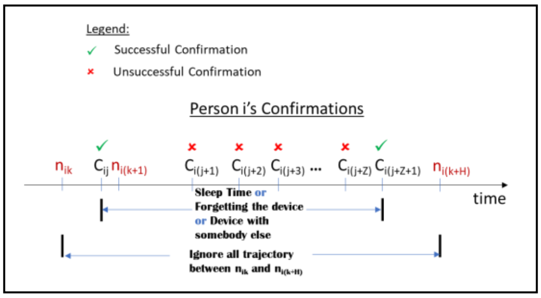
Legend:
✓ Successful Confirmation
× Unsuccessful Confirmation
Person i's Confirmations
n_{ik}
C_{ij}
$n_{i(k+1)}$
$C_{i(j+1)}$
$C_{i(j+2)}$
$C_{i(j+3)}$ ...
$C_{i(j+Z)}$
$C_{i(j+Z+1)}$
$n_{i(k+H)}$
time
Sleep Time or Forgetting the device or Device with somebody else
Ignore all trajectory between n_{ik} and $n_{i(k+H)}$

Figure 4: Factoring Situations such as Sleep Time and Device not with Owner

The period of time when a user cannot or does not respond to authentication requests is determined. For example in Figure 4 we show some Person i's timeline. In this figure you will see that between confirmation epochs C_{ij} and $C_{i(j+Z+1)}$ all the confirmations are unsuccessful. This could be because the person i may be asleep or the device may be with somebody else or the person did not purposefully respond. In our approach we will identify the last node, say n_{ik}, before C_{ij} and the first node information, say $n_{i(k+H)}$, after $C_{i(j+Z+1)}$. We will ignore all the trajectory information between n_{ik} and $n_{i(k+H)}$. The implication is that any updates made by the user's device about the user's location of the device in the trajectory while the user not responding to confirmation requests are not considered. Consequently, the Location Emit Interval (see Section III B) $I_{i(k+H)}$ will now effectively become $t_{i(k+H)}$-t_{ik}. Thus, larger this interval the more penalized the person i will be due to Step 15 of our method. Thus when a person does not respond to successive confirmation requests the person will be automatically penalized by our method.

V. Experimentations & Data Analysis with Discussions

This section creates a simulation model of the system described above, and then runs that simulation model under different scenarios to validate the efficacy of our method.

A. Simulation Model

We model **N** persons, say $P_1, \ldots, P_N$, in the overall system. We create **T** timestamps $(t_1, t_2, \ldots, t_T)$ leading up to the congregation event which is therefore at t_T. Note that t_T-$t_1 \geq$ **K** (see Section III B). At each t_i we generate a random integer J_i ($\leq$ $\mathbf{J_{max}}$, a predefined upperbound) that corresponds to the unique combination or pairs of persons, say $pair_{ij} := (P_a, P_b)$ where $P_a \neq P_b$ and j=1, ..., J_i. Note that $(P_a, P_b) = (P_b, P_a)$. Each pair of persons simulates a proximity event at a given timestamp. A proximity event will result in transmission of the infection from an infected person to the uninfected person (assuming one of them is infected). Out of the N persons we also randomly define **M** persons as being infected, that we call as the "originally infected". These originally infected users have contracted the infection before the timestep t_1 and correspond to the Type A infected persons in Figure 3. This means that by the timestep t_T these originally infected M users have been cured as t_T-t_1 $\geq$ K. But of course these originally infected ones will lead to creation of Type B persons in the interval (t_1, t_T), who in turn will result in more Type B persons, and so on. Once we have generated the N users, the subset of M originally infected users, the T timesteps, and the pairs for each timestep we run our method to compute the scores.

B. Evaluation Methodology

Before we formally define the metric used to evaluate our approach we describe its rationale. A primary objective of having a score A_i for person i is to be able to compare any two persons and predict the person having the higher score as being more likely to be affected. So how accurate are we in this prediction? And, precisely this will be the motivation for the design of the accuracy metric below:

Accuracy: Randomly sample (with replacement) **R** times from the list of N persons a pair of persons (P_i, P_j) where $P_i \neq P_j$, and, at least one of the persons is known to be *actually* infected (i.e., Type B) but not originally infected (i.e., Type A). Recall that the reason for not considering the originally infected is that the originally infected would have been cured by time t_T. For each sample determine if the person with the greater score is actually infected. If the person with the greater score is in fact actually infected then we consider that sample as *positive*. Accuracy is then defined as the **ratio** of samples that are positive to the total number of samples with at least one actually infected. The higher the accuracy, the higher is the confirmation of our approach that any two individuals can be compared using their scores and the higher an individual's score is the higher the likelihood of him/her being infected.

C. Simulation Results

We study how the accuracy varies with number of persons N in the system.

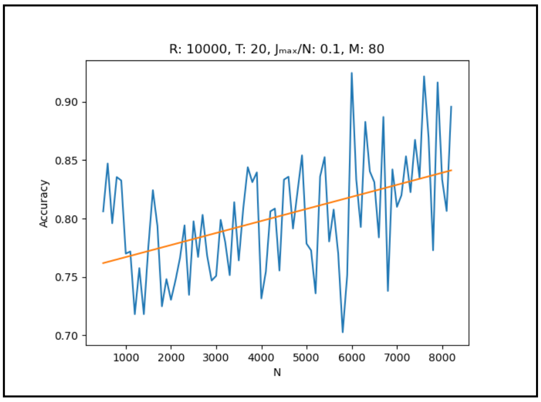

Figure 5: Accuracy vs N

For different values of N ranging from 500 through 8500 in Figure 5, with R = 10000, T=20, J_{max}/N = 0.1, M=80, we see that Accuracy increases. Further, the average Accuracy is greater than 0.80.

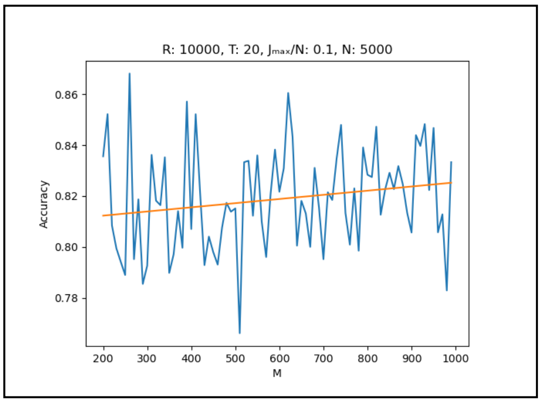

Figure 6: Accuracy vs M

In Figure 6 we explore how accuracy might vary with respect to M, the number of originally infected. For smaller values of M since the chance of picking up a pair with infected is small we see lower values of accuracy. In fact as M increases while N is fixed, we expect that as we sample a pair the chance of picking up an infected increases, leading to an increase in the Accuracy.

In Figure 7, our objective is to understand how Accuracy varies with respect to T, the number of timesteps. As the number of times steps increases the number of people pairing up with infected people increases. This increases the number of people infected at the final timestep. The figure shows that as T increases the Accuracy starts to converge to a value between 0.75 and 0.80.

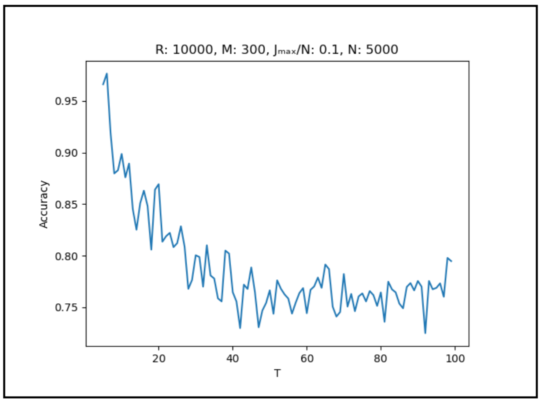

Figure 7: Accuracy vs T

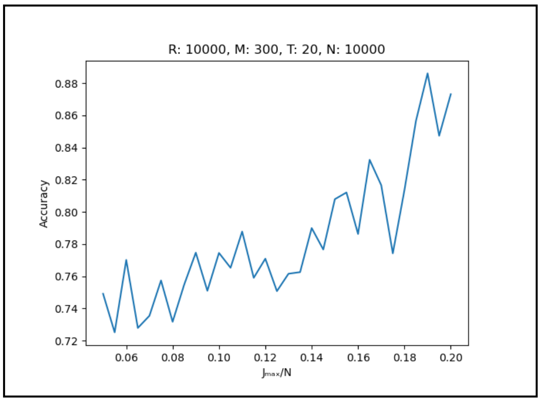

Figure 8: Accuracy vs J_{max}/N

In the Figure 8, we explore the variation of Accuracy with respect to the ration J_{max}/N. Note that J_{max} corresponds to the maximum number of pairs that can be created at each time step. The higher the value of J_{max} the more number of pairs are generated at a given timestamp. This means that more chance of people pairing up with the infected and then getting infected. The net effect of this being that at the final timestamp t_T we have more people who are infected. In the figure we see that with increasing J_{max}/N the Accuracy increases while other factors are constant.

VI. Conclusions & Future Work

Our experimentation (Section V) provides very encouraging results – our accuracy of predicting if a person with the higher score as compared to another person has the infection ranges between 0.75 to 0.90 for different values of the model parameters.

Our framework as presented in the paper only considers GPS locations to compute the scores for each individual but Bluetooth based proximity detection (as in [2], etc) can complement it in two ways:

(a) Whenever two or more individuals' Bluetooth connectivity occurs then they exchange each other's scores. The application running on each individual's device will show the normalized score providing a sense of risk around the individual currently.

(b) A local (i.e., device-level) computing of the score can be enabled. This will ensure that the location information need not be sent to the server end. We have to explore how to use ephemeral ids (see [6]) to compute the scores. This will help preserve the privacy and help better comply with regulations (see [6][16][18]).

It must be noted however that when using only the GPS location as we have done in this paper the advantage is that the GPS locations can be obtained from the mobile network provider (assuming all legal, privacy and security concerns are handled).

Our future work will consider the points made above but also an actual implementation of this client-server application.

VII. References

[1] Calculate distance, bearing and more between Latitude/Longitude points, https://www.movable-type.co.uk/scripts/latlong.html

[2] Aarogya Setu, https://www.mygov.in/aarogya-setu-app/

[3] ArcGIS® Location Tracking Privacy Best Practices, esri, https://coronavirus-resources.esri.com/datasets/7ccaf0d0be7149629c305fbf9d369dad

[4] Biosensor for the Covid-19 Virus, https://www.sciencedaily.com/releases/2020/04/200421112520.htm

[5] Carl Heneghan, Jon Brassey, Tom Jefferson, "COVID-19: What proportion are asymptomatic?," CEBM, April 2020, https://www.cebm.net/covid-19/covid-19-what-proportion-are-asymptomatic/

[6] Carmela Troncoso, et al., "Decentralized Privacy-Preserving Proximity Tracing," https://github.com/DP-3T/documents/blob/master/DP3T%20White%20Paper.pdf

[7] CCPA, "California Consumer Privacy Act," https://www.privacypolicies.com/blog/ccpa-compliance-checklist/

[8] Close Contact Detector, https://www.bbc.com/news/technology-51439401

[9] Contact Tracing, https://www.who.int/news-room/q-a-detail/contact-tracing

[10] Coronavirus, https://www.who.int/health-topics/coronavirus#tab=tab_1

[11] Emily W., "Can Sensors That Detect Coronavirus in the Air Help Economies Reopen Safely?," https://spectrum.ieee.org/the-human-os/sensors/chemical-sensors/devices-monitor-coronavirus-in-the-air

[12] FitBit Partners With King's College To Make App That Can Detect Coronavirus Digitally, https://www.republicworld.com/technology-news/gadgets/fitbit-partners-with-kings-college-to-make-app-that-can-detect-covid.html

[13] GDPR, "General Data Protection Regulation," https://gdpr-info.eu/

[14] Genevieve Bell, "We need mass surveillance to fight covid-19—but it doesn't have to be creepy," MIT Technology Review, April 2020, https://www.technologyreview.com/2020/04/12/999186/covid-19-contact-tracing-surveillance-data-privacy-anonymity/

[15] How people are using AI to detect and fight the coronavirus, https://venturebeat.com/2020/03/03/how-people-are-using-ai-to-detect-and-fight-the-coronavirus/

[16] Marcel Salathe, Viktor von Wyl, Effy Vayena, Edouard Bugnon, "Digital Proximity Tracing,", National COVID-19 Science Task Force (NCS-TF), May 2020, https://ncs-tf.ch/en/policy-briefs/digital-proximity-tracing-15-may-20-en/download

[17] Max S. Kim, "South Korea is watching quarantined citizens with a smartphone app,", MIT Technology Review, March 2020, https://www.technologyreview.com/2020/03/06/905459/coronavirus-south-korea-smartphone-app-quarantine/

[18] Nuria Oliver, et al., "Mobile phone data for informing public healthactions across the COVID-19 pandemic life cycle.Science Advances," 2020, https://advances.sciencemag.org/content/6/23/eabc0764

[19] Recovery Time for Covid-19, https://www.who.int/dg/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---24-february-2020

[20] Sarah Ambrose, Noah Slocum, "Use Proximity Tracing to Identify Possible Contact Events," Health and Human Services, May 2020, https://www.esri.com/arcgis-blog/products/arcgis-pro/health/use-proximity-tracing-to-identify-possible-contact-events/

[21] Spanish Flu, https://en.wikipedia.org/wiki/Spanish_flu

[22] Transmission of SARS-CoV-2: implications for infection prevention precautions, World Health Organization, July 2020, https://www.who.int/news-room/commentaries/detail/transmission-of-sars-cov-2-implications-for-infection-prevention-precautions