DEVELOPMENT OF MACHINE LEARNING-BASED INTELLIGENT COVID-19 CONTACT TRACING TOOLS

by

Wilson Kosasih

A project report submitted in conformity with the requirements
for the degree of Master of Science, Information Technology

Department of Mathematical and Physical Sciences
Faculty of Graduate Studies
Concordia University of Edmonton

DEVELOPMENT OF MACHINE LEARNING-BASED INTELLIGENT COVID-19 CONTACT TRACING TOOLS

Wilson Kosasih

Approved:

Supervisor: Dr. Baidya Nath Saha Date

Committee Member Date

Dean of Graduate Studies: Dr. Alison Yacyshyn Date

DEVELOPMENT OF MACHINE LEARNING-BASED INTELLIGENT COVID-19 CONTACT TRACING TOOLS

Wilson Kosasih
Master of Science, Information Technology
Department of Mathematical and Physical Sciences
Concordia University of Edmonton
2022

Abstract

With the easing of COVID-19 regulations in Alberta, people are gradually back to their normal life. However, we cannot be careless, since there is a possibility for a new variant. The technology could help to prevent the further spread of the pandemic. In addition, in terms of contact tracing, most people's concerns are about privacy and security, where people's data tend to be tracked and used without their consent. The proposed COVID-19 contact tracing method uses the clustering method, one type of machine learning algorithm that is used to segregate the data into different groups based on their characteristics. In this research, different types of clustering algorithms, Density-based Clustering (clusters are formed based on the density of the region) and Hierarchical-based Clustering (clusters are formed using hierarchy structure), and K-Means Clustering (clusters where the preset clusters are set. Besides the algorithms mentioned before, we will also compare Epidemic Modeling, Agent-Based Modeling, and Exposure Notification. They are used and a comparative study has been conducted on these algorithms in terms of efficiency and privacy. This research would not only be useful to trace potential infections of COVID-19, but it would also, build a process to suppress the spread. In the future, we would like to develop a hybrid model to enhance the efficacy of the tools further in terms of both accuracy and security by exploring more sophisticated Machine Learning (ML) techniques, and it could be useful for other infectious diseases as well.

Keywords:Coronavirus, COVID-19, Privacy, Contact Tracing, Clustering, Machine Learning classifiers, Agent-Based, Exposure Notification, Epidemic Modeling

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

The first case of COVID-19 was first found in Wuhan, China in November 2019. As of May 2022, the virus has spread to all of the countries in the world. The United States tops the chart with 81.4 million cases with 993,000 deaths, India comes next with 43.1 million cases and 524,000 deaths, meanwhile, Brazil makes third place with 30.5 million cases and 664,000 deaths. The number of total cases has reached 515,082,443 with total deaths of 6,241,652 [1].

Although the effects of Covid-19 in 2022 are not as bad as 3 years ago, the life and activities of people have taken a drastic change. People have to wear a mask while doing their activities outside, social distancing has to be applied in public, people have lost their jobs which leads to loss of income, etc.

Contact tracing is an essential public health tool for controlling infectious disease outbreaks, such as those caused by the COVID-19 virus. Contact tracing can break the chains of transmission through the rapid identification, isolation and clinical care of cases, and providing supported quarantine of contacts, meaning that virus transmission can be stopped [2]. Contact Tracing occurs in three steps [3]:

- Identifying the Contact: identify people that had contact with people with confirmed positive cases.
- Listing the Contacts: List possible contacts of infected individuals and inform them.
- Contact Follow-Up: Necessary follow-up to patients that had contact with infected individuals and those that are positive.

And based on the survey done by Morning Consult and summarized by Statista, most adults in the United States were concerned about third-party groups having their health data and location. The rest of the concerns are summarized in Figure 1.1:

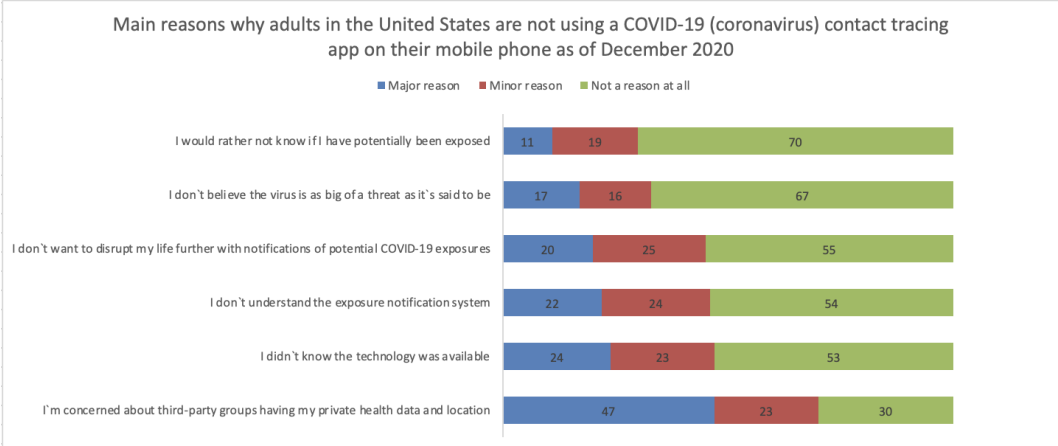


Figure 1.1: Main reasons why adults in the United States are not using a COVID-19 (coronavirus) contact tracing app on their mobile phone as of December 2020 [4]

People's curiosity about Contact Tracing also had a downward trend since the usage of Contact Tracing apps has decreased exponentially as seen in Figure 1.2 where people tend to search about Contact Tracing in December 2021, whereas it started to go down by 2022.

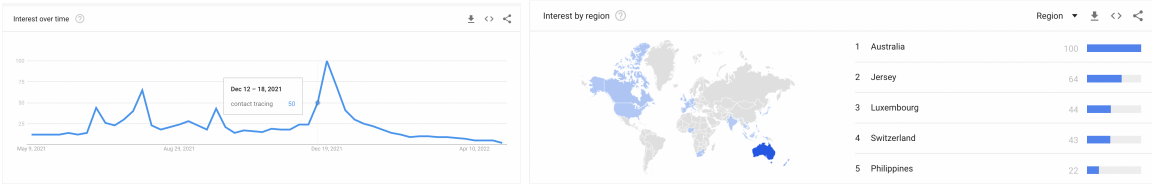


Figure 1.2: Google Trends on COVID-19 and connection to Contact Tracing from May 2021 till April 2022

As of July 2020, 21.6% population of Australia had downloaded government endorsed the COVID-19 Contract Tracing app, meanwhile Thailand, Vietnam, and Philippines rank bottom 3 with a total of 1.3% as shown in Figure 1.3 [5].

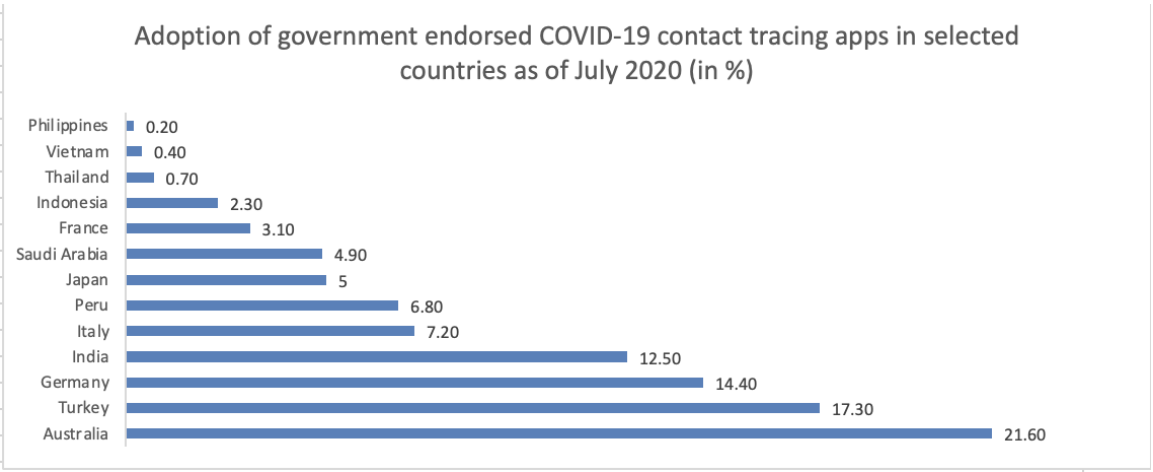


Figure 1.3: Adoption of Contact Tracing Apps in select countries [5]

1.2 Problem Statement

With the use of Contact Tracing apps, there has been an increasing concern about its privacy. From third-party apps having users' private health and location, including the mandatory activation of GPS and Bluetooth, to people not knowing the app even existed. Since the data for the user's GPS isn't provided by the government, dummy data will be used to execute the algorithms.

This research work will compare the effectiveness of multiple machine learning algorithms and which is providing more privacy with its result. These algorithms will be useful to trace infections of Covid-19 and in the future, the same algorithm will be developed to trace other infectious diseases.

1.3 Contribution of the thesis

In this research, we implemented and evaluated the effectiveness of six different contact tracing algorithms (shown in Figure 1.4), Forward and backward Contact Tracing, Clustering, Graph-Based Epidemic Modeling, Agent-Based Modeling, Exposure Notification, and K-core Superspreader. The findings of these algorithms are listed below.

- Forward tracing or conventional contact tracing only traces forwards (downstream) from the first infected case. This approach isn't as effective as the combination of forwarding and backward tracing, where with the combination of the two, new cases could be traced.

- The research work involves the creation of dummy data that includes longitude

and latitude, using Python's library, including NumPy, pygal, and pandas, the creation of a new model, and executing the algorithm. The collected data were processed, by using longitude and latitude on the x-axis and y-axis. A cluster was created based on the data's longitude and latitude. Each person in the data was represented by a color and a cluster was created within a radius of 6 feet. Machine Learning algorithms were performed with the data, including Density-Based Clustering, Hierarchical-Based Clustering, and K-Means Clustering.

- Graph-based Epidemic Modeling provides the term Social Network and Super-Spreaders, where this algorithm traces infected cases from nodes with the visualization of a graph.
- The creation of model and agent's movement in which agents can move from their initial position (home) to the final destination. Agents have their statuses, such as Susceptible, Infected, Recovered, and Death. Agent's status could change while they move.
- Google and Apple's Exposure Notification API preserve users' privacy by not tracking their location, and it gives users the freedom to opt-in or opt-out from the app.
- Using k-core technology which is a network to determine how each node has k connections to other nodes.

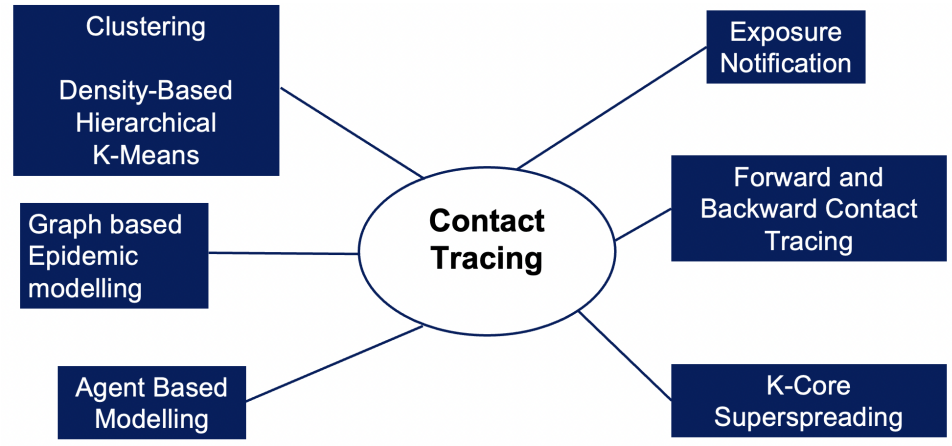


Figure 1.4: Contribution of the Thesis

1.4 Organization of the thesis

For this research work, there are nine (9) chapters. Chapter 1 is the Introduction which provides a detailed background of the research work, explaining the reasons for the research work, and the problem statement. Chapter 2 carries the Literature Review. Chapter 3, explains how Forward and Backward Contact Tracing works and its effectiveness. Chapter 4 talks about Clustering. Chapter 5 explains the implementation of Graph-Based Epidemic Modeling. Chapter 6 discusses Agent-Based Modeling and how it helps Contact Tracing. And Chapter 7 shows the effectiveness of Exposure Notification. Chapter 8 discusses about K-Core and it's connection to Superspreading. Chapter 9 talks about the results and discussions the analysis. Every step of the analysis is broken down here to understand different insights that were extracted from the data. Chapter 10 concludes this research work and also discussed future works for further analysis.

Chapter 2

Literature Review

Contact Tracing is a key strategy to interrupt the infections of Covid-19. Individuals who have been identified with the Covid-19 case are instructed to quarantine. The trigger of a contact tracing is when a case is identified, based on the Figure 2.1 [6].

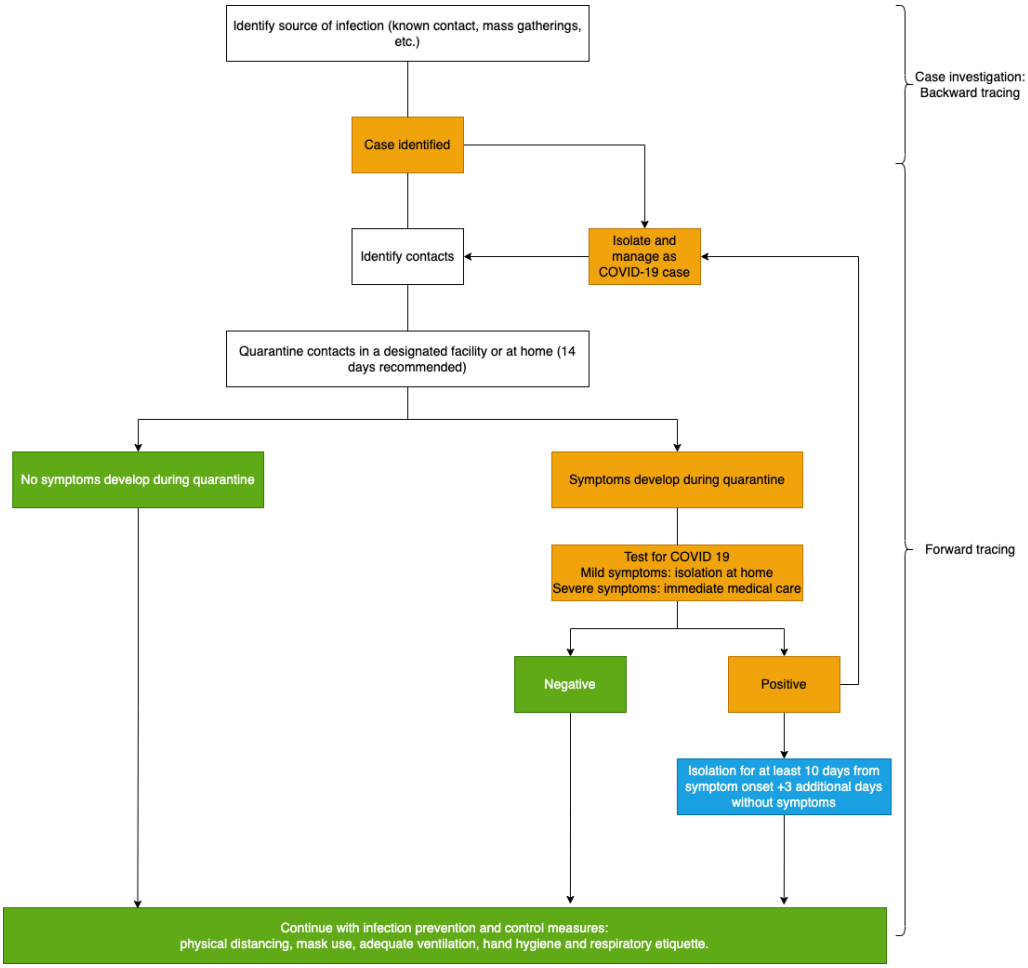


Figure 2.1: Chain of events for tracing, monitoring and caring for contacts of probable and confirmed COVID-19 cases [6]

2.1 Advantages & Disadvantages

People are worried about the privacy issues of Contact Tracing apps. Fortunately, not everything is negative about Contact Tracing. Below are the advantages and disadvantages of what the apps can do [7].

- Advantages
  - Some success in breaking chain of infection

    Countries like South Korea have managed to implement the apps which resulted to be quite effective to contain the virus - albeit at the cost of user privacy. Germany and Ireland have the highest adoption rate of the app, but it's still not enough to be fully effective.
  - Faster means of alerting people

    Exposure notifications are delivered automatically and effectively, rather than calling each person individually, an app may be more effective to alert people close to the proximity of an infected person, but every people have to use the app for it to work. The apps have eased the job of on-the-field contact tracers in South Korea.
- Disadvantages
  - Privacy issues

    South Korea has been the most successful for contact tracing, but at the same time, it was the worst offender of privacy issues. The app managed to expose cheating partners and other details about infected people's lives.
  - Security flaws

    Contract tracing apps are still a work in progress because many governments forced the solutions on app developers. The previous version of UK NHS's app was filled with security issues, in favor of the framework built by Google and Apple.

2.2 Challenges of Contact Tracing Implementation

There are quite a several implementations that focus on contact tracing. One of the implementations was to halt the transmission of the Ebola virus. There were some challenges faced by the researchers, as shown below [8]:

- Contact-person identification

The immediate identification of all potential contact persons is the first step of contact tracing - can be hindered by mistrust of contact tracing personnel and limited understanding of a disease, especially in settings facing the disease for the first time. A community's misperceptions regarding a disease could contribute to a person's hesitancy to reveal the names of contact persons. To increase trust and reduce fears, all countries that attempted the contact tracing have attempted to include local officials, and/or religious officials in its implementation.

- Locating contact-persons

  Once contact persons are identified, logistical issues can obstruct the process of locating persons without addresses, no street name, difficult terrain, telecommunication unavailability, and countries without national identification programs.

- Enrolling contact-persons

  Enrolling contact persons depends on their availability for 21 days. This is one of the reasons that people tend to flee from follow-up, other reasons include the stigma associated with contact-person, from family and friends, as well as financial and social pressures.

- Managing personnel

  Hiring staff that is trained is another huge challenge, especially in areas that are heavily affected, has been limited by education and managerial proficiencies, and provision of job-specific training. Additionally, staff has an increased risk of the disease infection.

- Contact tracing performance

  Ensuring contact tracing performance is another issue faced by the staff. The exhausting schedules of the staff can decrease the motivation to meet the quality of contact tracing results. This can lead to staff falsifying or providing inaccurate results on purpose. To improve the results, all countries should perform a quality assurance check on their data.

2.3 Privacy Issues of Contact Tracing

The government in each country has started to build contact tracing apps since the beginning of the COVID-19 pandemic. In most apps, the users' location history was monitored and tracked without their consent. Besides the location, there are many privacy issues, including data breach, data collection, and obscure data flow [9]:

- Voluntary or Mandatory

  Contact Tracing apps should be voluntary, considering the concern of data privacy, unnecessary data collection, location tracking, and other issues.

- Information Destruction

  Apps should automatically delete users' records, usually between 14-21 days. Otherwise, users have control to delete the data manually.

- Transparency

  The process of data collection, usage, and storing should be transparent to users. While developing an application, it should have data flow, databases, and open-source code for transparency.

- Data Collection

  Many apps collect unnecessary data from users, for example, name, phone number, age, gender, profession, and details of countries visited in the last 30 days. Also, apps usually collect GPS location and Bluetooth.

- Disclosure of Location

  Location illustrates the interaction between individuals by representing the individuals as nodes and the connection between the nodes as endpoints indicating that users may have been in proximity. An adversary can build a social graph by mining the data to infer the user's contact profiles. Disclosure of the location or the social graph is undesirable, however, some countries such as India have done so despite concerns from civil society. Both centralized and decentralized architecture is vulnerable to the disclosure of location details.

Some apps tend to get users' data without consent, primarily the use of the apps' real-time location. Based on the survey done by Sowmiya, 49% of people were concerned with privacy and have yet to use the contact tracing app, 38% are concerned about the mandatory to enable GPS/ Bluetooth, and 40% are concerned about the false sense of Security [9].

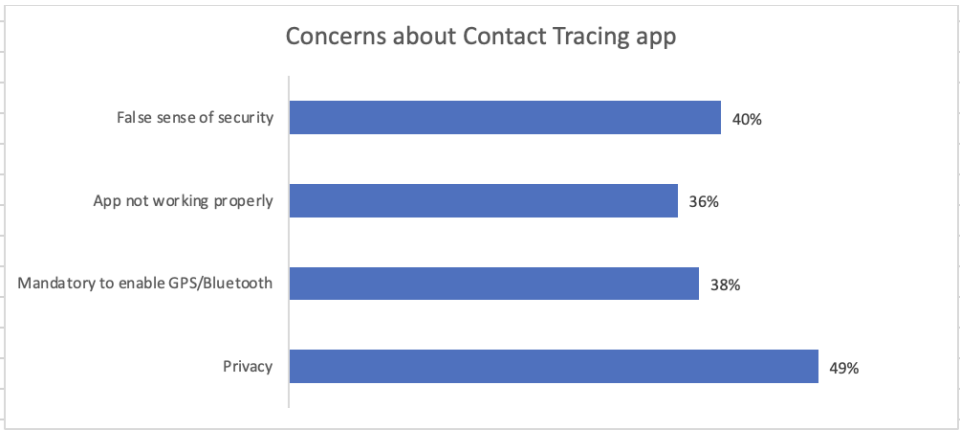


Figure 2.2: Concerns about Contact Tracing Apps [9]

Meanwhile in Canada, almost 65% of respondents didn't want to use the app because of privacy concerns, 44% responded that they didn't want the government to access their location, 24% respondents didn't believe people would install the app, and just 4% of respondents felt that they won't catch the virus [10].

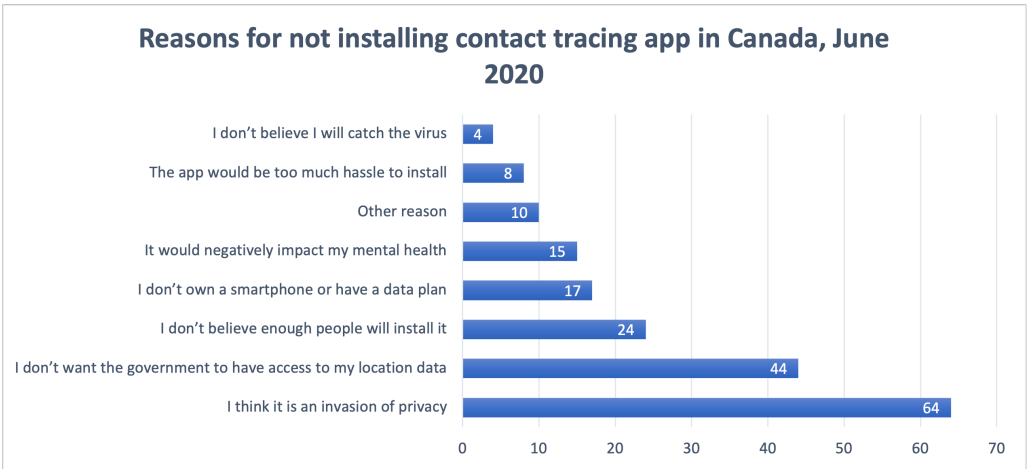


Figure 2.3: Reasons for not installing contact tracing app in Canada [10]

2.4 Ethical Issues of Contact Tracing

Besides privacy, contact tracing apps also face ethical issues. WHO has released ethical guidelines that every contact tracing should apply. The list is shown below [11]:

- Time limitation

  Implementation must be temporary and limited in scope. If the epidemic is proven to be over, measures of contact tracing should be over immediately.

- Testing and evaluation

  Contact tracing technology should be properly tested before it is released to be used by the citizens. The function of the app should be ensured, technically robust, and have no security flaws. Evaluation of the technology should be continuously monitored throughout the pandemic.

- Proportionality

  Collection of personal and health data must be protected by law, this includes, justified by the health system's objectives, suitable to achieve goals, necessary, reasonable and proportionate to the aims pursued.

- Data Minimization

  Data collection and processing should be limited to the minimum necessary use of data. Data collection doesn't require the identity or location of the user or time stamp.

- Use restriction

  Data collection and processing should be limited to the minimum necessary use of data. Data collection doesn't require the identity or location of the user or time stamp.

- Voluntariness

  An individual's decision to download and use the app is completely voluntary. Governments shouldn't force/demand individuals to download the app by providing additional incentives. Individuals also shouldn't be denied services or benefits from the government by refusing to use the app. An individual is free to delete the app at any time, without consequences.

- Transparency and explainability

  Data collection and processing should be transparent, which means individuals should give consent on the purpose of the collection, how the data will be processed and stored, and how the data will be retained.

- Privacy-preserving data storage

  The data storage should be decided whether it's going to be centralized or decentralized. Both approaches may preserve privacy, but issues like the security of the collected data should be considered as well. The decentralized approach enhances security since it provides the user with greater control over what type of data that user may share with health authorities.

- Security

  Every effort should be made to ensure to be encrypted of every data collected and processed on every device, server, network, etc. A third-party audit should be performed to test whether the security is ready.

- Limited retention

  Data retention should be limited to a period of pandemic response, except for pandemic research or subject to regulation. Data used for research should be anonymized where possible.

- Infection Reporting

  The reporting of an individual that has tested positive for COVID-19 can be done through several scenarios. In many scenarios, notification of the app should require the consent of the individual. In other scenarios, the user of the digital proximity app could self-report and should be confirmed by a health professional. Alternatively, a medical professional could notify through the app (with the individual's consent).

  Notification

  Notification of individuals who have been in contact with a person infected with COVID-19 could be delivered directly to the app. The notification should preserve the individual's private information. Notification should be provided with clear information and instruction on how the individuals should do if they receive the notification. It should also be available in different languages and accessible to people with disabilities.

- Tracking of COVID-19 positive cases.

  After an individual has tested positive for COVID-19, the app should not be used to track an individual's movement during infection and recovery.

- Accuracy

  Algorithmic models used to process data must be reliable, verified, and validated. Applications should be tested by third parties to establish parameters for duration and proximity before contact is recorded and should be adjusted and improved over time.

- Accountability

  Individuals must be allowed to know about and challenge any COVID-19-related measured to collect, aggregate, retain and use data.

- Independent oversight

  There should be independent oversight, including ethical and human rights aspects, of both the public agencies and the businesses that develop, and operate digital proximity tracking applications or use information obtained with them.

- Civil society and public engagement

  COVID-19-related responses that include data collection efforts should include free, active, and meaningful participation of relevant stakeholders, such as experts from the public health sector, civil society organizations, and the most marginalized groups.

Based on one of the points above (Privacy-preserving data storage), it is divided into two approaches, which are centralized and decentralized. Key differences between centralized and decentralized contact tracing data storage are shown in Table 2.1:

Table 2.1: Key differences in Centralized and Decentralized[12]

	Centralized	Decentralized
Source of temporary IDS to send out	Receives list from server	Generates its own list
When a temporary ID is deleted from the app's list of sent out IDs	Deletes immediately after it's last sent out	Deletes a few weeks later
Alerting user of positive contacts	Centralized server performs verification	Apps locally perform the verification
Information stored on the server	Can determine the long-term identifier of a user from its temporary ID, and knows the temporary IDs of whom an infected user has come int contact with	Knows the temporary IDs of infected users

Table 2.2 is the overview of both centralized and decentralized approach, and which is more preferred based on some comparisons, as described below [13]:

* (+) approach is preferred, (-) approach is not preferred.

Table 2.2: Preferences of Centralized and Decentralized[13]

	Centralized	Decentralized
Tracking	(-) Put user's data at risk, possibly data theft.	(+) Infected user's data at risk, theft on device data.
Identification (diagnosed people)	(+) Difficult, if not impossible	(-) Identification diagnosed people is impossible.
Pressure to opt-in	(-) Pressured user always results in privacy risk	(+) Users retain to opt-in if health app is designed with an OTK authorization.

2.5 Effectiveness of Contact Tracing

Jonatan Almagor & Stefano Picascia researched the effectiveness of the Contact Tracing App into different methods and categories, as shown below [14]:

- Testing without tracing

  With this method, testing capacity increases from 0% to 3%, and overall infections decrease from 44% to 31%. Further increase in capacity over 3% doesn't affect the decrease of infections further.

- Contact Tracing App that prioritizes symptomatic cases

  With the implementation of the Contact Tracing app, users who have been notified seek testing. When symptomatic cases are prioritized, overall infections decrease. For example, if 80% of the population uses the app, overall infections decrease from 45% to 15%.

- Contact Tracing App with no priority to symptomatic cases

  When testing capacity is restricted and symptomatic cases are not prioritized, it does not always mean that overall infections decreases (testing capacities between 0 and 6%). In the case of 3% testing capacity, 40-80% of app adoption rates result in more infections than the case with no app users. When symptomatic cases are prioritized, the results of positive tests are higher for any testing capacity, even though the prevalence of infection is lower than in no priority scenario. Scenarios with unlimited tests always show a substantial advantage in using the app.

- Contact Tracing App users' compliance with self-isolation

  When users' compliance to self-isolation of app users is high, overall infections are only 3-5% lower when compliance is low. Compliance level with self-isolation influences the higher number of users. With 1% testing, overall infections reach 23% when compliance is high and 23% when compliance is low. With the peak of the epidemic, overall infections are reduced to 80% when compliance is high and 74% when compliance is low.

2.6 Contact Tracing Implementation

Centers for Disease Control and Prevention recommends aggressive contact tracing to control the COVID-19 pandemic that including medical, nursing, and public health students. And the workflow from that research is shown in Figure 2.4.

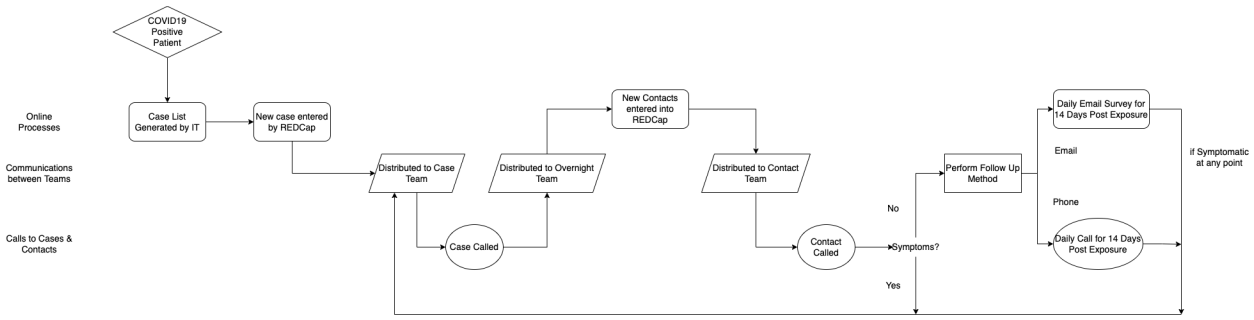


Figure 2.4: Contact Tracing Workflow [15]

By late April 2020, the United States tried to implement Contact Tracing but failed, although some countries successfully managed to implement the app. Below are the reasons why it failed to decrease the COVID-19 spread [16]:

- Lack of Coordination

  From the beginning of the pandemic, there has been a noticeable lack of unified national leadership and coordination, which has resulted in both the absence of a robust plan (or common goals) for local and state health departments and the dissemination of confusing mixed messages to the lay public. The resulting guidance allowed for the potential reopening of schools, restaurants, bars, and other institutions that were closed in many jurisdictions earlier in the pandemic, with limited specific direction for addressing sustained community transmission.

- Inadequate Testing Supply

  In addition to the lack of coordinated public health leadership, it has been surprising that, despite being a resource-rich nation, the United States still struggles to achieve adequate and consistent testing rates. In areas experiencing surges, there have been reports of long lines, test shortages, and over weeklong turnaround times, even though the past 5 months since the start of the epidemic should have provided ample time to increase supply chains for testing materials. It is a fundamental concept that health departments cannot trace cases that remain undetected. Yet even before the current surges, many putative cases, even those who were symptomatic, we were unable to obtain timely severe acute respiratory syndrome coronavirus 2 SARS-CoV-2 testing and results, and very few jurisdictions had implemented widespread, freely available public surveillance testing. This ineptitude in deploying a cohesive testing strategy stems from many organizational and national leadership barriers, including an underfunded public health outpatient testing infrastructure, regional insufficiencies in testing supplies reagents, and a lack of national guidance regarding the best strategy for

implementing surveillance testing.

COVI-Agent Sim is an agent-based compartmental model using Simpy, a process-based discrete-event simulation framework. Each agent tracks transitions through Susceptible, Exposed, Infectious, and Recovered (SEIR), pre-existing medical conditions, self-reported symptoms, and test results. Those data will enable the simulation of contact tracing apps. This research also compares Feature-Based Contact Tracing (FCT) with Binary Contact Tracing (BCT) and Digital Contact Tracing (DCT), and the research suggests that BCT and FCT methods can help to reduce disease's spread, while DCT methods can save lives even with lower adoption rates [17].

In Japan, COVID-19 Contact-Confirming Application (COCOA) was implemented to prioritize users' privacy from attackers, reduce the load of health care workers and systems, increase the responses to the pandemic, and reduce the population's mobility. COCOA's system architecture is shown below [18]:

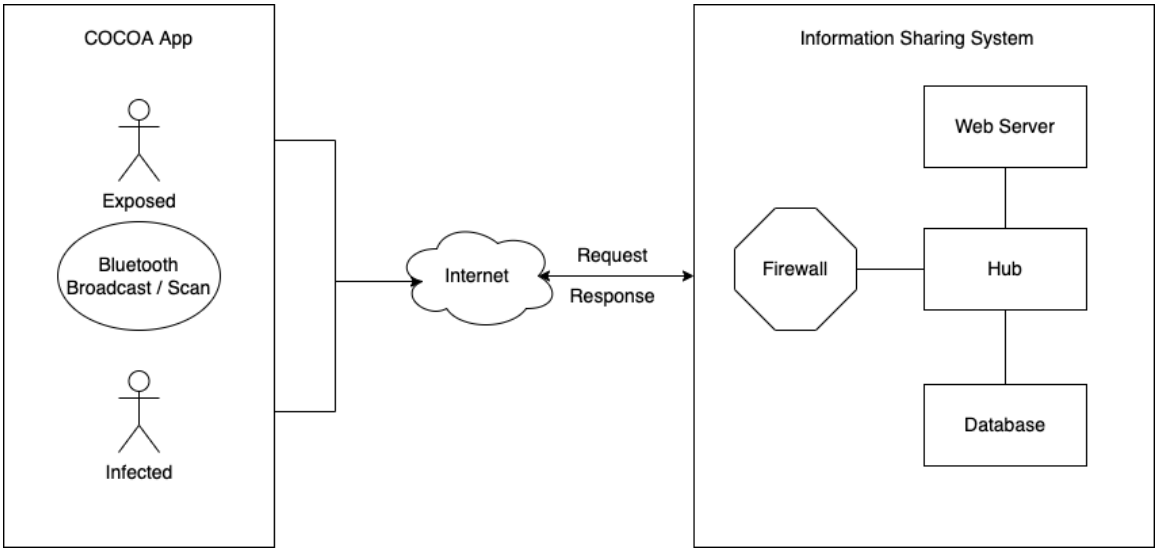


Figure 2.5: COCOA app workflow [18]

Meanwhile, in Guinea, the current system was using paper-based contact tracing. But that system faced limitations in its implementation. The contact tracers used paper forms (with internationally accepted forms), and that form was then sent to their supervisors for review and then data was entered through an Excel database. Figure 2.6 shows the flow of data from the paper-based contact tracing:

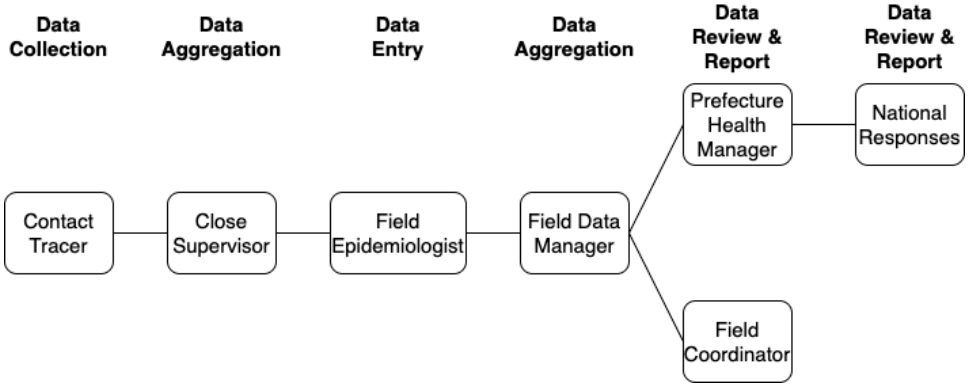


Figure 2.6: Data flow of Guinea's Paper-Based Contact Tracing [19]

But paper-based contact tracing has a couple of limitations, including delays between data collection and consumption, human error with data entry, and the efforts of data cleansing. That is why the development of mobile contact tracing was needed. It eliminated the limitations of paper-based contact tracing and provided dashboards that included data such as daily and weekly data [19].

In Singapore, BlueTrace was developed by the Singaporean government. It is a protocol for logging Bluetooth encounters between devices for the use of contact tracing while preserving the privacy of its users. When two devices encounter each other, both the devices exchange messages that contain temporary identifiers. The identifiers rotate regularly to prevent third parties from tracking users. The encounters are stored locally on the user's device, and the data can be accessed by the health authority. If a user is infected, they will be asked to share the encounter history with the health authority with the use of a PIN. The only health authority can decrypt the encounter history and can contact tracing the other infected users. BlueTrace includes the following privacy safeguard [20]:

- Limited collection of user's personal information, such as phone number, which is securely stored by the health authority
- Local storage of encounter history. The user's encounter history is stored locally on the device.
- Third-party apps cannot access the location of BlueTrace users. The device's temporary identifier rotates regularly, therefore preventing unauthorized access.

- Revocable consent. Users can opt-out of the use of their data. If they opt out, all data stored at the health authority is deleted, also the encounter history won't be linked to the user.

In April 2020, Google and Apple announced an exposure notification API that uses Bluetooth technology to help contact tracing even easier. Users didn't even need to download an app and they can opt out of the service. This technology cannot track a user's location and also doesn't share the identity with Google, Apple, and third-party apps. For every phone that opted-in, a randomly generated number was used to disguise the phone's identity that changed every few minutes. Then with Bluetooth technology, every time the phone is closed to other phones that also opted-in, they will exchange those randomly generated numbers. If in the future, someone is positive for COVID-19, they can report that positive result, and every phone that has been in contact for the last 14 days will also get a notification that they may have COVID-19 too [21].

In 2021, Covasim, an agent-based model of COVID-19 dynamics and interventions was introduced. It is a simulation of individual people, which largely focused on a single calculation: the probability of an agent at a given time will change from one state to another, including susceptible, exposed, infectious, recovered, or dead, and status based on presymptomatic (mild, severe, and critical) and asymptomatic [22]. Covasim's logical view is shown in Figure 2.7.

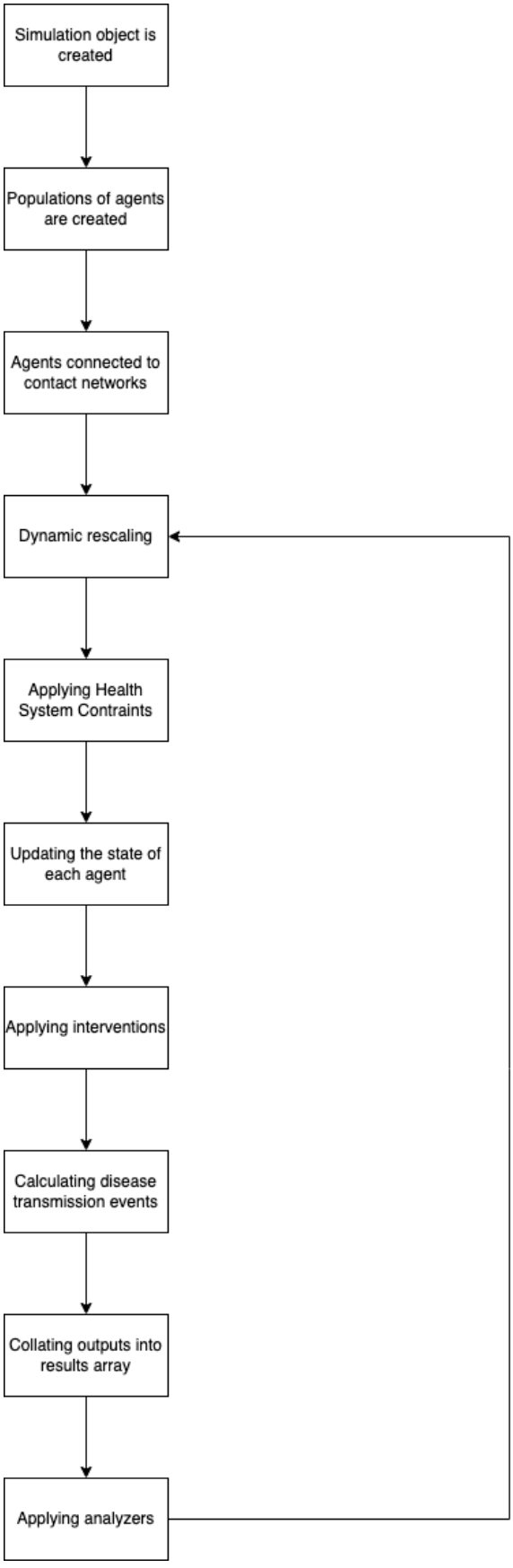


Figure 2.7: Logical view of Covasim [22]

Meanwhile, CoviChain uses blockchain technology to implement contact tracing. COVID-19 health records travel across from wearable devices to the intermediate edge, where blockchain is used to maintain the similarity of the health records while sharing the data. By implementing this technology, the time and cost of blockchain by using distributed storage have been reduced [23]. The architecture of CoviChain is shown in the figure below:

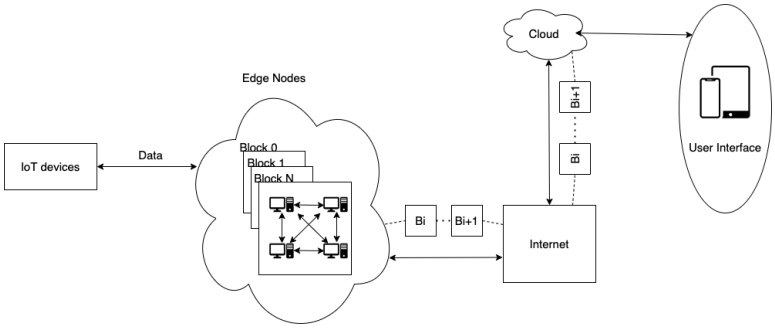


Figure 2.8: Architecture of CoviChain [23]

In 2020, The Bubblebox System was introduced as a device dedicated to performing contact tracing. The system is composed of a wearable device for example a wristband to log unsafe contacts, an app, to pair devices and identifies people wearing them, and an infrastructure to store data and make it available to health personnel [24].

South Korea's implementation of the Contact Tracing app is considered to be the most successful. Figure 2.9 describes the quarantine workflow in South Korea. To preserve users' privacy, this implementation used credit card data to determine the trajectory (time and location) of each user. The data were stored in encrypted form by using a functional encryption technique [25].

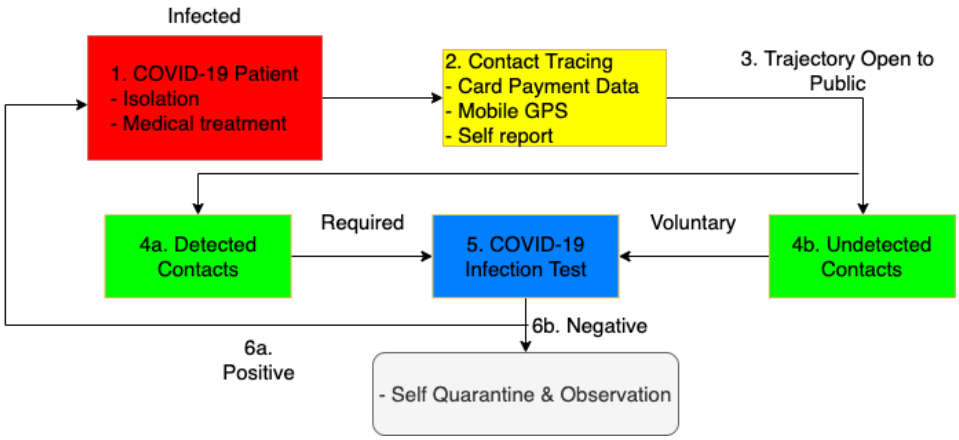


Figure 2.9: Quarantine Workflow of South Korea [25]

Chapter 3

Forward and Backward Contact Tracing

Forward tracing often targets 'downstream' individuals, who may have been infected by the index case (the original case in which the index case acquired the illness). Unlike forward contact tracing, backward contact tracing identifies and traces the source of the newly detected cases. This approach of tracing is particularly valuable when there is individual variation in the number of secondary transmissions (overdispersion). By using the branching process model, we explored the potential of the combination of backward contact tracing and forward contact tracing to control the COVID-19 spread. We estimated the cluster size by the application of backward contact tracing and simulated the effectiveness by combining backward contact tracing and forward contact tracing [26].

Figure 3.1 shows the comparison and the effectiveness of forward contact tracing and when we implement Forward and Backward Contact Tracing. The combination of the two is more effective because in that way we can trace the source of the infection and the transmission of the infection, unlike forward contact tracing which can only trace and prevent the next transmissions.

Figure 3.1: Comparison of Forward, Forward and Backward Contact Tracing [26]

Backward contact tracing can be implemented in all types of cases, especially when secondary cases are likely to be identified. Below are the situations in which Backward Contact Tracing can be applied, and some cases where Backward Contact Tracing is not necessarily applied [27]:

*3C = Closed spaces with poor ventilation, Crowded places with many people nearby, Close contact settings.

Situations in which Backward Contact Tracing may be applied:

- When there is no known exposure for the case (i.e., infected person does not know where the illness may have been acquired).
- When infected person identifies that they have the illness while attending one or more 3C's locations:
  - Family gathering.
  - Dinner party.
  - Group activity (musical or exercise).
  - Personal training or services.
  - Religious ceremonies or gatherings.

Situations in which Backward Contact Tracing is nor necessarily applied:

- Where the exposure to an infected person is known (e.g., when a household member is infected). However, Backward Contact Tracing could be applied to the initial infection of COVID-19 in the household.
- When there is an outbreak, for example in a monitored location such as hospitals.

- In monitored locations where health workers will already be employed to investigate and manage the risk of transmission, for example, schools, workplaces, child care centers, etc.
- In locations where the direct exposure to infection is generally not possible/feasible, for example, co-riders in public transport.
- Locations where 3Cs are not applicable, such as outdoors where physical distancing and masking measures have been maintained.

3.1 Equations

To find each type of contact tracing's effectiveness, here are the equations to calculate forward and backward tracing effectiveness:

Methods We computed the effectiveness of contact tracing as the proportion of generation 3 cases averted. Assuming a negative-binomial branching process with a mean and overdispersion parameter, The mean total number of generation 3 cases given an index case found by surveillance is

$$C_3 = R^2(1 + R(1 + \tfrac{1}{k}))$$

Forward Tracing The expected number of generation 1 cases excluding the initially found index case is $R(1+\frac{1}{k})$, of which proportion is independently detected by symptom-based surveillance. Therefore, the total number of generation 1 cases targeted by forward tracing is $1 + Rd(1 + \frac{1}{k})$. Each of them would result in R^2 generation 3 cases on average if not traced, however, proportion qc is averted by contact tracing (success rate q) and quarantine (relative reduction c in infectiousness). The number of generation 3 cases averted is thus given as

$$\Delta_F = R^2qc(1 + Rd(1 + \tfrac{1}{k})).$$

Forward and Backward Tracing We assume that backward contact tracing successfully identifies the primary case at probability b. Of the mean $R(1 + \frac{1}{k})$ generation 1 cases that are potentially under the scope of backward tracing, mean proportion (1-d)(1-bq) will remain undetected either by backward tracing or independent detection. The total number of generation 1 cases detected is $1 + (1 - (1 - d)(1 - bq))R(1 + \frac{1}{k})$, which gives the number of generation 3 cases averted as

$$\Delta_{F+B} = R^2qc(1 + (1 - (1 - d)(1 - bq))R(1 + \tfrac{1}{k})).$$

Effectiveness We computed the effectiveness of tracing as $\frac{\Delta_F}{C_3}$ and $\frac{\Delta_{F+B}}{C_3}$.

Chapter 4

Clustering

Clustering is a statistical method for processing data. It works by organizing items into groups, or clusters, based on how closely associated they are. Clustering is an unsupervised learning algorithm and is typically used when there is no assumption made about the relationships within the data. It provides about where the associations and patterns in the data, but not what the data might be or what they mean [28]. Some of the applications of clustering are mentioned below:

4.1 Density-Based Clustering

Density-based clustering is a non-parametric algorithm, given a set of points in some space, it groups together points that are closely packed together (points with many nearby neighbors), marking as outliers points that lie alone in low-density regions (whose nearest neighbors are too far away). DBSCAN (Density-based spatial clustering of applications with noise) is the most applied clustering algorithm [29]. It uses the concept of density reachability and density connectivity [30].

- Density Reachability

  A point p is said to be density reachable from a point q if point p is within ϵ distance from point q and q has a sufficient number of points in its neighbors which are within distance ϵ.

- Density Connectivity

  A point p and q are said to be density connected if there exists a point r that has a sufficient number of points in its neighbors and both the points p and q are within the ϵ distance. This is the chaining process. So, if q is neighbor of r, r is neighbor of s, s is neighbor of t which in turn is neighbor of p implies that q is neighbor of p.

Algorithmic steps for DBSCAN clustering:

Let X = x1, x2, x3, ..., xn be the set of data points. DBSCAN requires two parameters: ϵ (eps) and the minimum number of points required to form a cluster (minPts).

- Start with an arbitrary starting point that has not been visited.
- Extract the neighborhood of this point using ϵ (All points which are within the ϵ distance are neighborhood).
- If there are sufficient neighborhoods around this point then the clustering process starts and the point is marked as visited else this point is labeled as noise (Later this point can become part of the cluster).
- If a point is found to be a part of the cluster then its ϵ neighborhood is also part of the cluster and the above procedure from step 2 is repeated for all ϵ neighborhoods points. This is repeated until all points in the cluster are determined.
- A new unvisited point is retrieved and processed, leading to the discovery of a further cluster or noise.
- This process continues until all points are marked as visited.

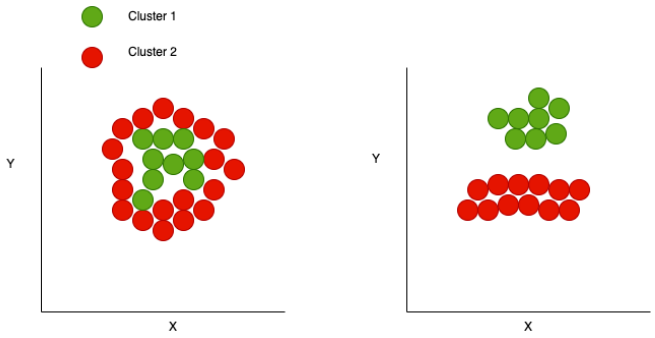


Figure 4.1: DBSCAN [31]

4.2 Hierarchical-Based Clustering

Hierarchical-based clustering or hierarchical cluster analysis is an algorithm that involves creating clusters from top to bottom. This clustering method has two types: Agglomerative Hierarchical Clustering and Divisive Hierarchical Clustering. In this research, the focus will be on Agglomerative Hierarchical Clustering which has a bottom-up approach (starts with its cluster, and pairs of clusters are merged as one moves up the hierarchy). This is how the algorithm works [32]:

- A single point cluster is made - forms N cluster
- Take two of the closest points and make a cluster - forms N-1 cluster
- Take the other two of the closest points and make another cluster - forms N-2 cluster
- Repeat step 3 until there is only one cluster left.

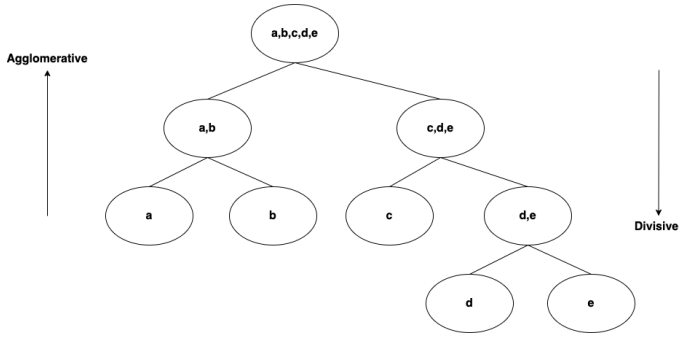


Figure 4.2: Agglomerative and Divisive Algorithm [32]

4.3 K-Means Clustering

K-Means segregates the unlabeled data into various groups, called clusters, based on having similar features, and common patterns.

K-Means Algorithm is an Iterative algorithm that divides a group of n datasets into k subgroups /clusters based on the similarity and their mean distance from the centroid of that particular subgroup/ formed. K is a pre-defined number of clusters formed by an algorithm. If K=2, that means the number of clusters to be formed is 2. Below are the steps of the K-Means algorithm [33]:

- Define the value of K, to determine the number of clusters that will be formed.
- Select random K points that will act as centroids.
- Assign each data point, from the distance of the randomly selected centroid to the nearest centroid that will form a predefined cluster.
- Place a new centroid of each cluster.
- Repeat step number 3.
- If any reassignment occurs, skip to step 4, else algorithm is finished.

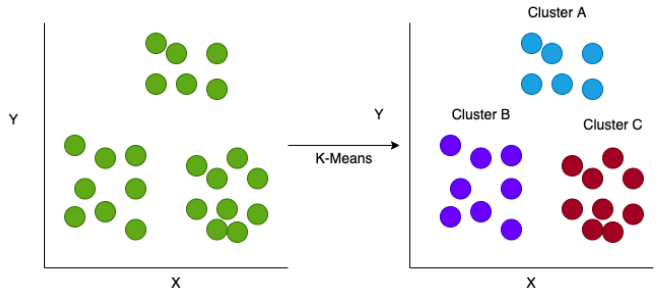


Figure 4.3: K-Means Clustering [33]

Chapter 5

Epidemic Modeling

Epidemic modeling is aimed to build a framework to include network structure. And it is focused on mathematical and conceptual details underlying the epidemic model on what is relevant to the current COVID-19 pandemic [34]. This chapter will describe how contact tracing works based on the social network and how can a person be a super spreader and spread the disease to other people.

5.1 Social Network

It shows how each individual is connected to his/her friends. A social network can be described as a graph, where the nodes are individuals, and edges represent social connections.

*M = Male, F= Female, (*) = node

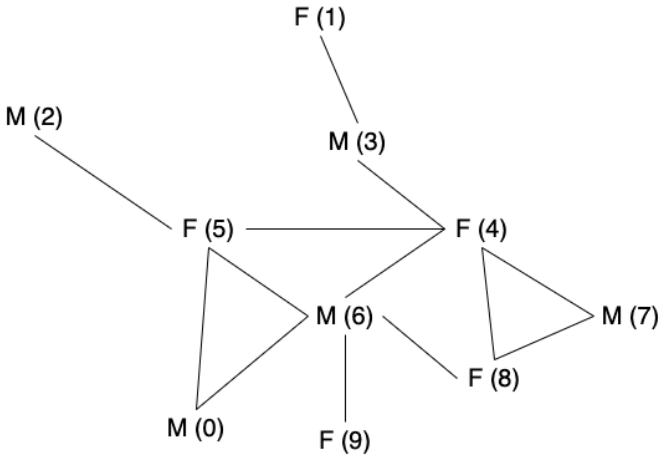


Figure 5.1: Social Network

Each edge represents interaction from one individual to another and how one can infect the other. In another approach, nodes can also be characterized by any feature

that could impact their likelihood of becoming infected (age, gender, etc).

As nodes can infect other nodes that which they are connected through edges, the position of an infected node can have a huge impact. For example, node 9 has only one connection which is node 6. On the other hand, nodes 4, 5, and 6 have more possibility to spread the infection.

5.2 Super-Spreaders

A direct consequence of this variability is the fact that specific people are capable of infecting many others. In the epidemic modeling, they are known as Super-Spreaders.

For example, let's say that Node 6 is infected. After some time the network might look like this:

*Blue = Susceptible, Yellow = Exposed, Purple = Infectious, Green = Recovered

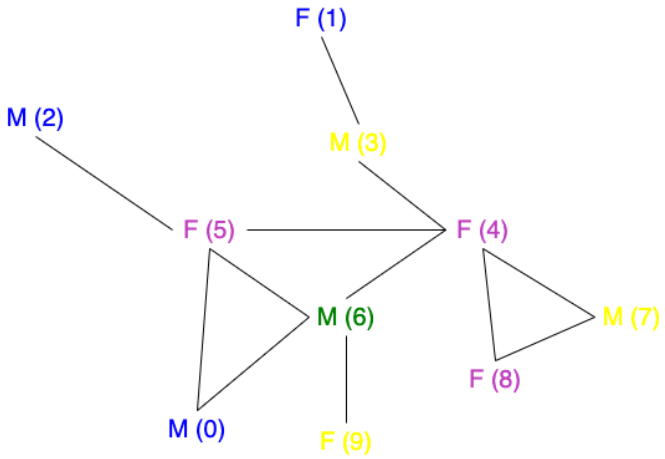


Figure 5.2: Super-Spreaders

5.3 Contact Tracing

One way to contain the epidemic and limit the number of cases is through contact tracing. Every time an individual is infected, he/she has to be located and quarantined immediately. While that individual is being tested, other individuals that have contact with the infected individual, need to be traced and tested. If none of them are infected, it can be certain that the outbreak is contained.

Chapter 6

Agent-Based Modeling

Using Mesa Agent-Based Modeling as a framework, the application of the SIR (Susceptible, Infected, Recovered) model theory for disease spread. Two classes (CovidModel and CovidAgent) are created and going to interact with each other. And based on the Model and the Agent, we can trace each Agent's movement and status.

6.1 Create COVID Model Environment

Environment stores a couple of global parameters, including:

- Infection probability
- Incubation period of disease
- Treatment period
- Death probability

The environment is created to provide a model with several agents. Agents are going to be able to move, as well as the limits of the environment. The init method also initializes the parameters below:

- Creates a set of possible destinations for Agents. Agents can interact with each other and also the environment.
- Stores the status and position of each Agent. It stores the status of Susceptible (S), Infected (I), Recovered (R), and Death (D) Agents.

6.2 Simulating COVID Agents

The second part includes the simulation of the model by the Agents, which are:

- Initial position, where the Agents are staying, for example, hotel or home.
- Target position, destinations where Agents are going before going back to the initial position.
- Infection time, the moment Agent is infected.
- Set of possible destinations, which allow Agents not to go to the same place.

In this simulation, the environment will check the status of Agents, the interaction between Agents that are in the same position, and apply the set of movement r

Status checks of Agents are provided below:

- If an Agent is infected, the death of the agent is based on the death probability parameter. In case of death, the Agent is removed from the environment, and the count of deaths in the environment is increased.
- If the infected Agent has survived for the entire treatment period, the status is changed to RECOVERED.

Also, the environment is set to move Agents, with rules below:

- If the Agent is at home/hotel, it will leave and go to other possible destinations.
- If Agent has arrived at a destination, it will head back home.
- But if the Agent is infected, it will stay home, or else go back to its initial destination.

6.3 Modeling COVID Infection

The environment will then model the interactions between Agents, as described below:

- The model asks the environment for a list of Agents that are at the same position at the same time.
- Checks the current Agent if it is currently infected. For each uninfected Agent, the probability of the infection is checked. In case of infection, the status of the Agent is updated.

Chapter 7

Exposure Notification

In 2020, Google and Apple teamed up to create an exposure notification API to help the government and citizens track and notify COVID-19 cases. This chapter will discuss more how the API works, how the app exchanges keys between people, how to trace an infected person, and finally how to determine infected persons.

7.1 Crypto Components

This specification is to create a 3-tier system of secure keys that can be exchanged selectively [35]:

- Handset Tracing Key is a unique 32-byte encryption key that will never leave the device.
- Daily Trace Key (DTK) is a derivative encryption key that is created using HMAC Key Derivation Function (HKDF) every 24 hours. This key will leave the user's device if the user is confirmed to be positive. If a positive infection is reported, the last 14 days of data will be transmitted to a system that is maintained by an app developer or agency. This is done so that the system can distribute the key to all handsets to trace potential infections.
- Rotating Proximity Indicator (RPI) is a 16-byte digital signature that is derived from DTK (a single DTK can have up to 144 unique RPIs. HMAC keys are used to compute this and part of the message of HMAC is the nearest 10-minute interval of the day:
  - 00:00–00:10 = interval 0
  - 00:10–00:20 = interval 1
  - ...

– 23:50–24:00 = interval 143

7.2 Proximity Exchange

A single handset will have:

- One unique Tracing Key
- One active Daily Tracing Key
- Historical list of Daily Tracing Key
- One active Rotating Proximity Indicator

When 2 handsets come into each device's proximity, both handsets will exchange their current RPI.

7.3 Determining Risky Contacts

When the handset owner receives a positive result for COVID-19, the owner will have an option to self-identify using the application. The application will:

- Gather the Daily Tracing Keys data for the last 14 days.
- Send DTKs to server.

7.4 Application Overview

Below are the overview on how the process works:

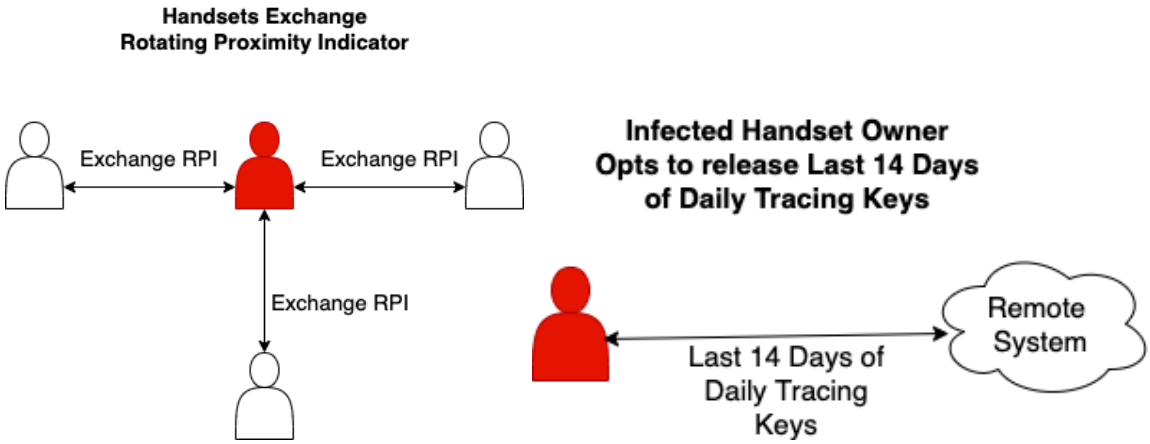


Figure 7.1: Application Overview

Chapter 8

K-Core Superspreading

K-Core of a network is the maximum subgraph in which each node has k connections to other nodes, despite how many links they have outside the subgraph [36]. The first accepted k-core concept was proposed by Seidman called the k-core pruning process to obtain the k-core of a given network, which is to remove the nodes that have a degree less than k recursively [37]. The concept is shown in the figure below 8.1:

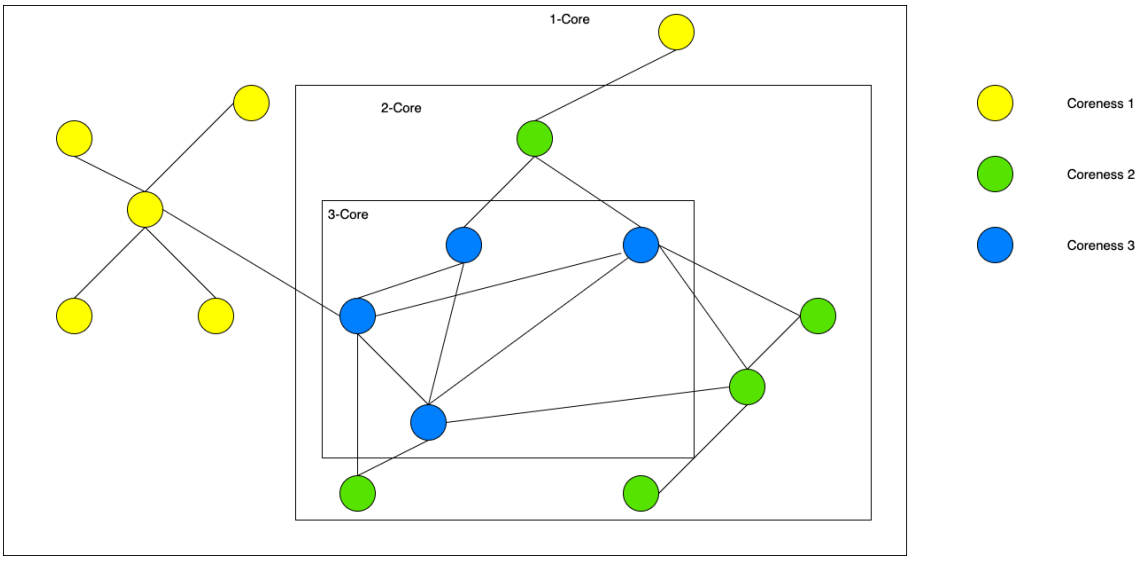


Figure 8.1: K-Core concept

8.1 COVID-19 Model

The COVID-19 spreading model is represented by Susceptible-Exposed-Infectious-Recovered (SEIR) process. In this model, susceptible individuals get infected by the virus and progress to exposure rate, and then the virus incubates for 7-14 days (infected), and individuals recover from the virus after 14 days. And in that timeline,

contacts that were close to the individuals were traced [38].

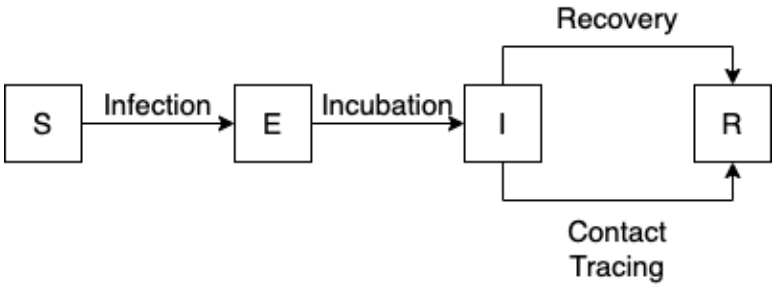


Figure 8.2: SEIR Model [38]

8.2 Contact Model

By using SEIR Model, the GPS geolocation of the trajectories of both infected and susceptible people is used to trace several layers of contacts in the transmission network using the following model.

To incorporate this model, the probability for infection is calculated with the equations below:

$$\text{Pi[n]} = \text{pi[n]}(1 - \text{Pi[n - 1]}) + \text{Pi[n - 1]}.$$

Table 8.1: Model parameter

Parameter	Notation
Probability of infection for a series of repeated contacts	Pi[n]
Probability of infection	pi[n]

The iteration between source and target, P1[n], generates a higher probability of infection than single contact pi[n]. This means there is a difference between a short single contact between two people and short repeated contact between the same people.

Chapter 9

Results and Discussions

As discussed in the previous chapters, Forward and Backward Contact Tracing, Clustering (Density, Hierarchical, and K-Means Clustering), Graph-Based Epidemic Modeling, Agent-Based Modeling, and Exposure Notification are used for this research work. Each of the data processing and the results are shown in this chapter.

9.1 Forward + Backward Contact Tracing

The comparison between forward and backward contact tracing is based on the parameters below:

Table 9.1: Comparison parameter

Parameter	Notation
Reproduction number	R
Overdispersion parameter	k
Relative reduction in transmission due to quarantine	c
Probability of identifying the primary (G0) case by backward tracing	b
Probability of identifying each offspring of an already identified case	q
Probability of a G1 case identified by surveillance independently of contact tracing	d

And based on the parameters above, we are going to compare the effectiveness of forward contact tracing only, and what if forward and backward contact tracing is implemented. The comparison of the implementation of contact tracing mentioned before is shown in Figure 9.1:

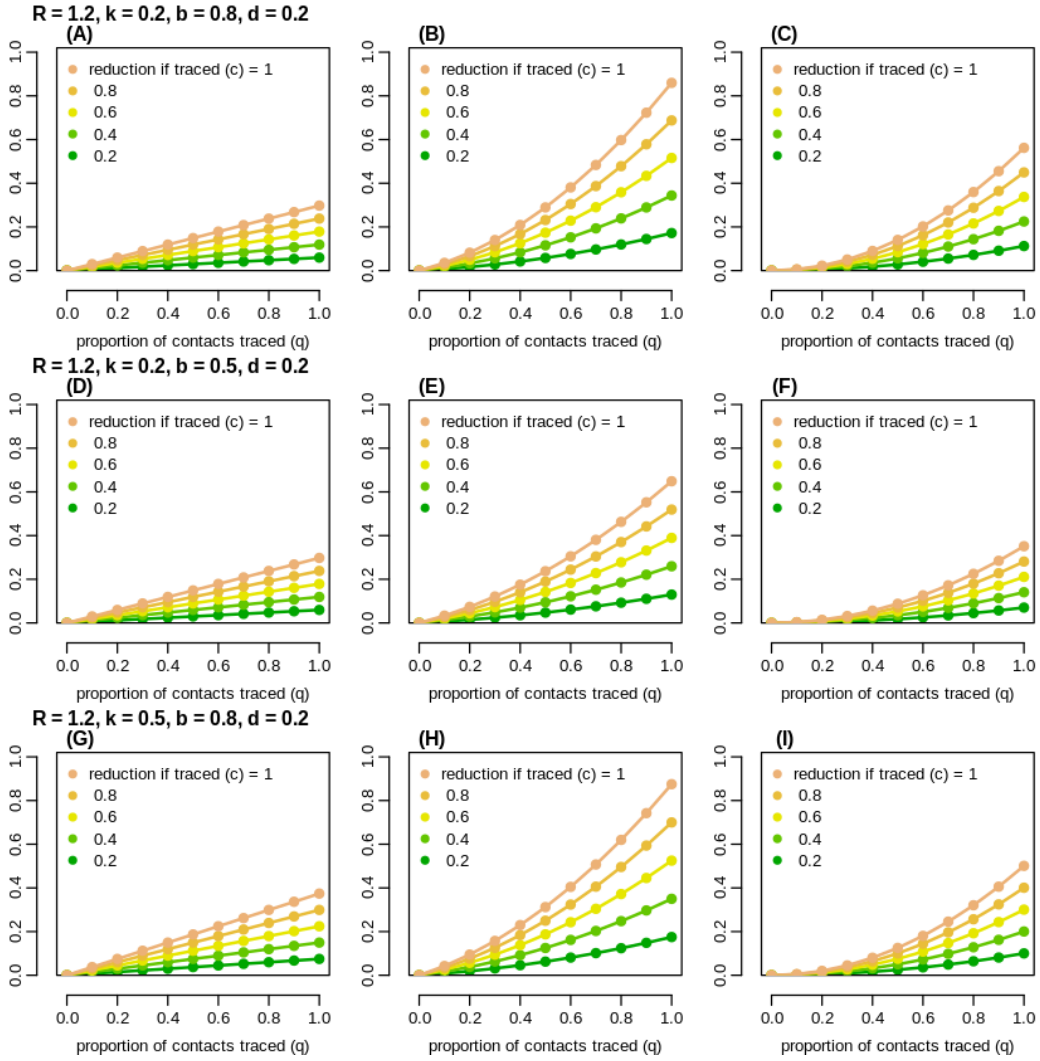


Figure 9.1: Effectiveness of contact tracing

From the result, we can see that after the implementation of forward and backward contact tracing, the proportion of contacts traced improved significantly, although no contact tracing method is perfect, this is an improvement.

9.2 Clustering

The dataset used for the Clustering algorithm wasn't collected from government sites, since the data included privacy information. A mock data was created with 4 fields, including id, timestamp (dates and hours), longitude, and latitude. The data contain 100 entries, with unique IDs Person1, Person2, Person3, Person4, ..., Person10. It shows the position of a person at certain times. The below figure shows the first 5 records of the mock data.

	id	timestamp	longitude	latitude
0	Person1	2020-08-27 17:33:33	60.077519	13.988041
1	Person2	2020-08-27 20:13:18	60.029391	13.903152
2	Person1	2020-08-27 18:22:23	60.078368	13.933152
3	Person2	2020-08-27 03:38:36	60.002145	13.967506
4	Person3	2020-08-27 01:11:35	60.040521	13.966431

Figure 9.2: Data used

9.2.1 Density-Based Clustering

EDA (Exploratory data analysis)

Processed data would be analyzed using a scatter plot showing the IDs with longitudes and latitudes on the x-axis and y-axis respectively.

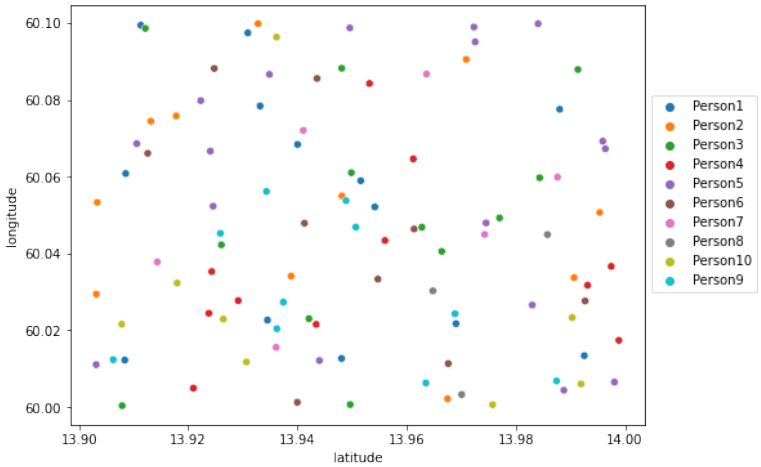


Figure 9.3: Scatter plot

The same dataset was analyzed using a joint plot as shown below.

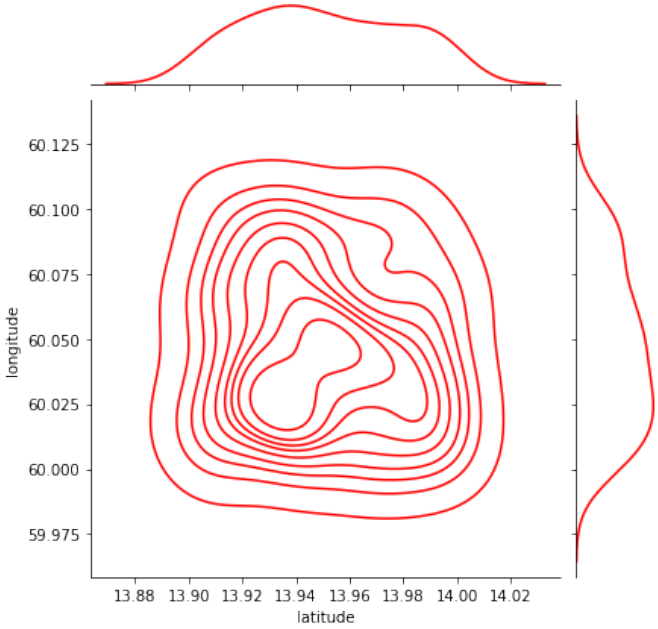
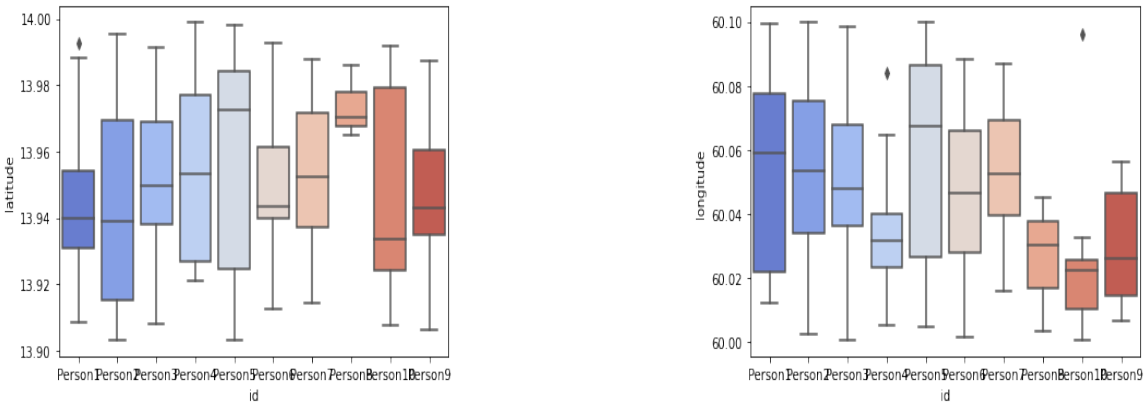


Figure 9.4: Joint plot

Below is the visualization of IDs, longitude, and latitude with a box plot.



(a) Latitude (b) Longitude

Figure 9.5: Box Plot visualization of IDs, longitude and latitude

Identifying infected people

Clusters are formed based on the person's position (longitude and latitude), and if each person is in the range of 6 feet, a cluster will be generated. The process will be repeated until all clusters are generated. For this dataset, 16 clusters were able to be generated.

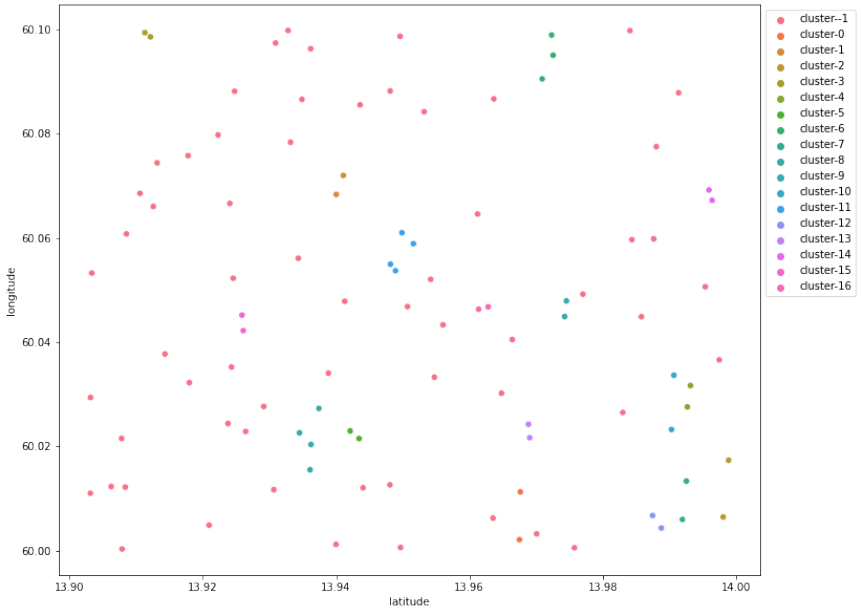


Figure 9.6: Clusters

We would be able to get the potential infections based on the ID that we input. For example, in this case, I would like to find the potential infections for Person10. The algorithm will then find IDs based on the clusters made from the process before, and if a cluster was found, results from that cluster would be shown.

```
get_infected_names("Person10")
```

```
['Person1', 'Person2']
```

Figure 9.7: Potential infection

9.2.2 Hierarchical-Based Clustering

For Hierarchical clustering, the same dataset from the previous results will also be used here. The first step of this clustering was to generate a dendrogram, as shown in the process below.

Generate a Dendogram

Based on the data, a dendrogram was generated with an Agglomerative Algorithm where the complete data was generated bottom-up, where the observation started from its clusters, and clusters were merged as it moved up the hierarchy.

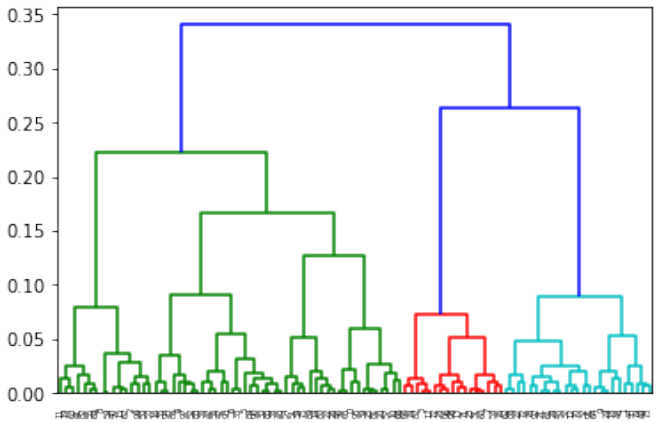


Figure 9.8: Hierarchical Clustering Dendogram

Clustering

With a Hierarchical algorithm, 5 clusters were generated within the first level. As the level went on, there were 2 clusters on level 4.

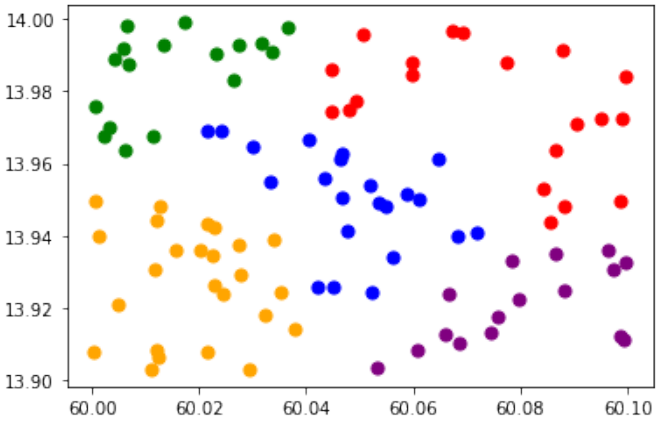


Figure 9.9: Level 1 Clustering

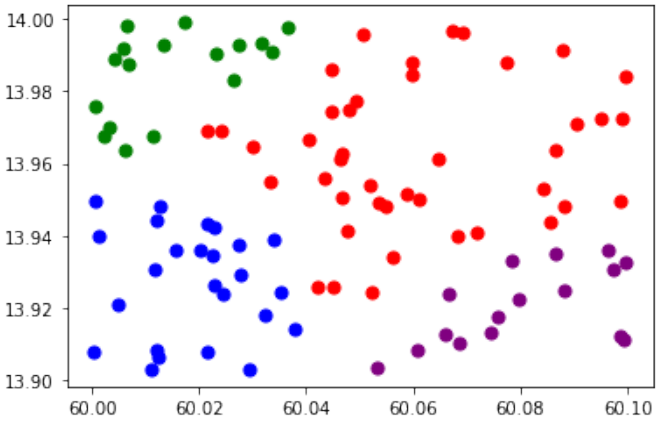


Figure 9.10: Level 2 Clustering

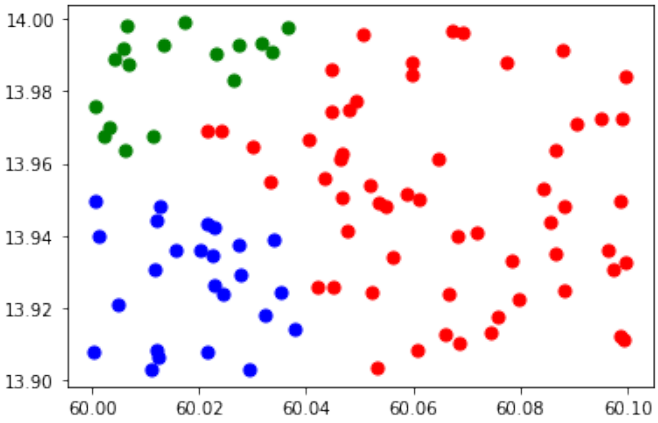


Figure 9.11: Level 3 Clustering

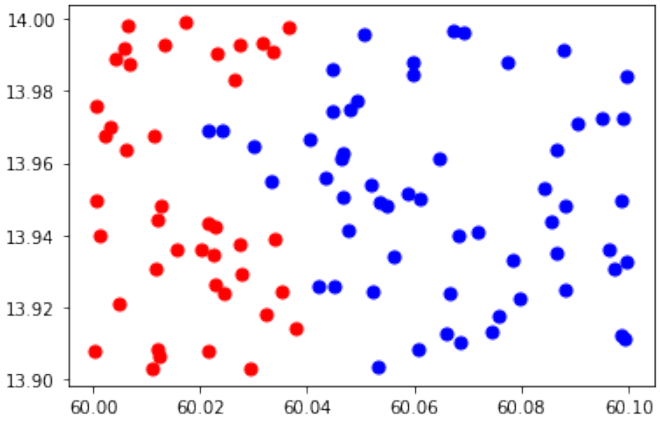


Figure 9.12: Level 4 Clustering

9.2.3 K-Means Clustering

Data Processing

The same dataset was used for K-Means clustering. The difference was, that only longitude and latitude data was needed for this clustering, as shown below in figure 9.13.

	longitude	latitude
0	60.077519	13.988041
1	60.029391	13.903152
2	60.078368	13.933152
3	60.002145	13.967506
4	60.040521	13.966431
...	...	...
95	60.087861	13.991382
96	60.031707	13.993157
97	60.043339	13.956026
98	60.026532	13.983008
99	60.086588	13.934874

Figure 9.13: K-Means data

Determine Cluster

The difference between K-Means, Density, and Hierarchical Clustering is in K-Means we have to set how many clusters to generate. In our experiment, we tried to generate 3 clusters as shown below.

```
kmeans = KMeans(3)
kmeans.fit(x)

KMeans(n_clusters=3)
```

Figure 9.14: Determine Clusters

Generate Cluster

A graph of the cluster has been generated after some clusters have been set earlier, in this case, it was 3 clusters, and were represented with 3 colors, which are red, purple, and light blue.

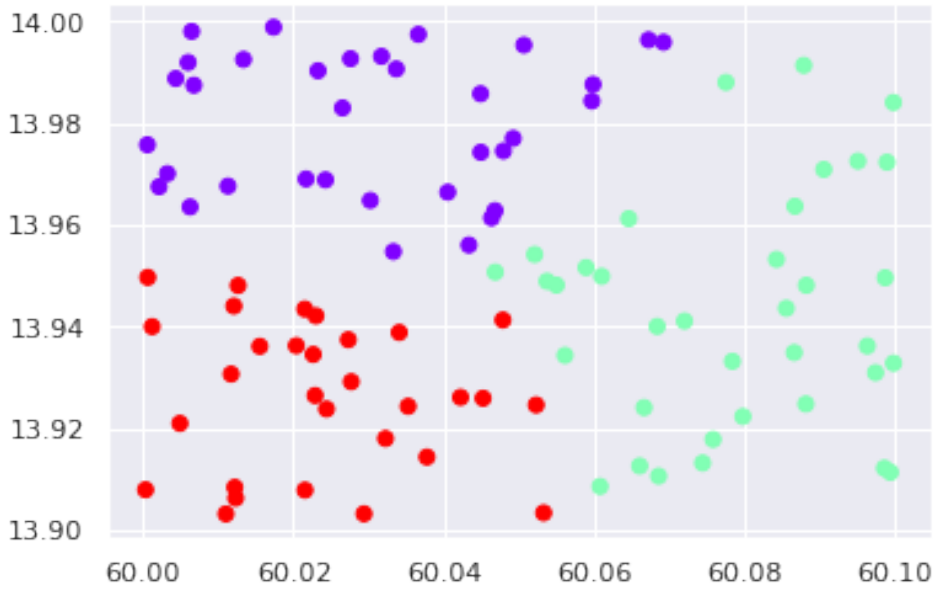


Figure 9.15: K-Means Clusters

9.3 Epidemic Modeling

In this Epidemic Modeling, the first thing is to set up a network, by deciding which node has which connections to other nodes. And then we can assume that a node is infected and we can find out which nodes are infected or at risk.

9.3.1 Setting Up Network

Firstly, we need to set up which node has a connection to other nodes. For example, Node 1 only has a connection to Node 3, Node 7 has connections to Node 4 and Node 8, etc as shown in 9.16. And after the connections have been made, we can check how many connections a node has as shown in 9.17.

```
G = nx.Graph()
G.add_edges_from([
    (1, 3),
    (3, 4),
    (4, 5),
    (4, 6),
    (4, 7),
    (4, 8),
    (8, 7),
    (8, 6),
    (6, 5),
    (6, 0),
    (6, 9),
    (5, 0),
    (5, 2)
])
```

Figure 9.16: Network

```
DegreeView({1: 1, 3: 2, 4: 5, 5: 4, 6: 5, 7: 2, 8: 3, 0: 2, 9: 1, 2: 1})
```

Figure 9.17: Connections

Next, we can set up the gender of each node sequentially that will be added to the table:

```
nx.set_node_attributes(G, dict(zip(range(10), ('M', 'F', 'M', 'M', 'F', 'F', 'M', 'M', 'F', 'F'))), 'Gender')
nx.set_node_attributes(G, dict(G.degree()), 'Degree')
```

Figure 9.18: Gender

After the network and gender have been set up, and the degree for each node has been checked, we can summarize the data as below:

Node	Gender	Degree
0	M	2
1	F	1
2	M	1
3	M	2
4	F	5
5	F	4
6	M	5
7	M	2
8	F	3
9	F	1

Figure 9.19: Data

9.3.2 Contact Tracing

If we assume Node 6 is infected, we can find who is at risk by checking his contacts:

```
[4, 8, 5, 0, 9]
```

Figure 9.20: Contact

And if we want to take the contact tracing a step further, we can also get the contacts of these nodes:

```
{0, 2, 3, 4, 5, 6, 7, 8, 9}
```

Figure 9.21: Full Contact

9.4 Agent-Based Modeling

9.4.1 Creation of Model and Agents

To create agent-based modeling, first, we need to create the model itself. Each Agent has their status and it's possible to change their status if they are in a position with another Agent. We can see the plot of the agent movement in Figure 9.22:

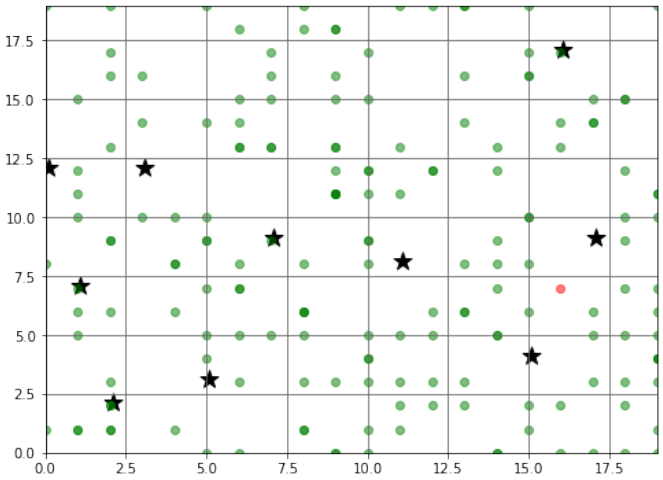


Figure 9.22: Agent Movement

9.4.2 Draw the network

After we created the model and we know the position of each Agent, we can depict the model with the spiral network, as shown below in Figure 9.23:

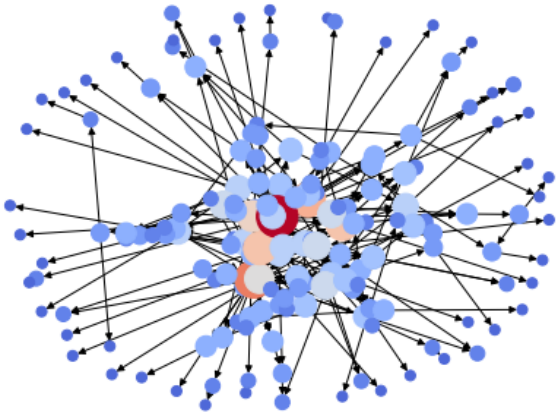


Figure 9.23: Model's Spiral Network

And after each Agent's movement has been determined, a simulation of how the Agents move is initiated, where some rules are applied to the simulation:

- If the Agent is at home, the Agent will go to a random destination.
- If the Agent is at the destination, the Agent will go home.

Some rules are also applied for Agent's status update:

- If the Agent is infected and currently at home, the Agent won't go to a random destination.
- If the Agent is infected and currently at a random destination, the Agent will go home.

The final simulation of this network is shown in Figure 9.24. The video real-time movement of the Agent can be viewed here.

* $\star$ = Destination, Green = Susceptible, Red = Infected, Cyan = Recovered

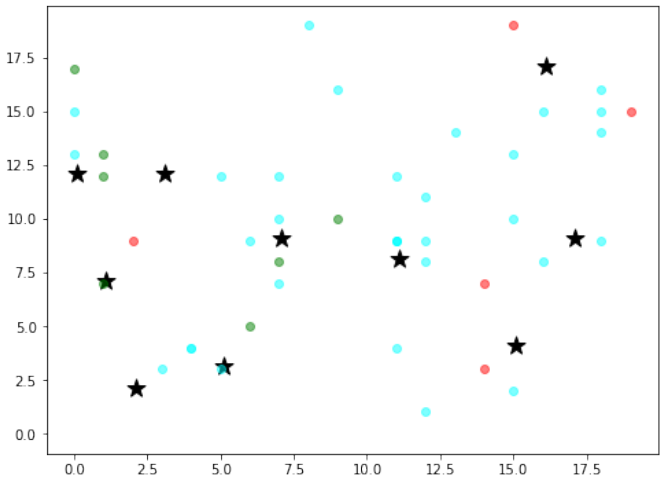


Figure 9.24: Simulation Result

9.5 Exposure Notification

For the Exposure Notification algorithm, unique handset ID and status were used to process and calculate contact periods that are needed for contact tracing.

9.5.1 Handset's data generation

With the use of current time, DTK and RPI data were generated from handset.py code and that data was then used and exchanged with other handsets within the simulation.

```
@dataclass
class TEK:
    seed_epoch: InitVar[Number] = None
    enin: int = None
    key: bytes = None

    def __post_init__(self, seed_epoch: Number):
        if not self.enin:
            self.enin = get_enin(seed_epoch)
        if not self.key:
            self.key = get_random_bytes(16)


class RPIK:

    def __init__(self, tek: TEK):
        self.key = HKDF(
                    tek.key,
                    b'',
                    SHA256,
                    num_keys=1,
                    context=('EN-RPIK').encode()
                )
        self.tek = tek
        self.cipher = AES.new(self.key, AES.MODE_ECB)

    def get_proximity_id(self, enin):
        data = ('EN-RPI' + str(enin)).encode()
        enc = self.cipher.encrypt(pad(data, AES.block_size))
        self.cipher = AES.new(self.key, AES.MODE_ECB)
        return enc.hex()

    def enumerate_proximity_ids(self):
        # map each RPI by the ENIN, this way we can
        # easily find which slice of time an RPI
        # was most likely observed in
        pid_map = {}
        enin = self.tek.enin
        for _ in range(0, 144):
            pid_map[self.get_proximity_id(enin)] = enin
            enin += 6
        return pid_map
```

Figure 9.25: DTK and RPI data generation

9.5.2 Simulate Handset's Activity

For this simulation purpose, other handsets were created as part of the initial handsets activity and grouped with family, friends, coworkers, and others.

```
class Life:

    def __init__(self, start_time: int):
        self.family = get_handsets(random.randrange(2, 8), 'family')
        self.friends = get_handsets(random.randrange(10, 20), 'friend')
        self.coworkers = get_handsets(random.randrange(15, 40), 'coworker')
        self.others = get_handsets(random.randrange(40, 100), 'other')

        self.all_handsets = self.family + self.friends + self.coworkers + self.others  # noqa
```

Figure 9.26: Other Handsets grouping

After other handsets were generated, an activity was created for the initial handset to have contact with the groupings that have been set before. For this example, the simulation was set on weekdays activity, which included breakfast, going to work, after party, and dinner.

```
def weekday(self):
        day_start = self.time  # save the first hour of our day

        # starting a new day, generate the TEK
        # for each handset

        self.subject.create_tek(self.time)

        for h in self.all_handsets:
            h.create_tek(self.time)

        # spend a couple hours in the morning with family
        self.hour('family')
        self.hour('family')

        # stop for some breakfast / coffee on the way to work?
        if random.choice([1, 2, 3]) == 1:
            self.hour('others')

        # work, work!
        self.hour('coworker')
        self.hour('coworker')
        self.hour('coworker')

        # lunch / gym?
        if random.choice([1, 2, 3]) == 1:
            self.hour('others')

        # moar work
        self.hour('coworker')
        self.hour('coworker')

        # happy hour?
        if random.choice([1, 2, 3, 4, 5]) == 1:
            self.hour('friends')

        # back home
        self.hour('family')
        self.hour('family')

        # fast forward to the next morning
        self.time = day_start + ONE_DAY
```

Figure 9.27: Simulation

9.5.3 Results

The results for this simulation were a txt file, where the data included a total count of grouping, handset ID that had contact with the initial handset, the relationship with the initial handset, and contact periods. A sample of the result is shown below.

```
Simulation Start Time: 2020-04-13T07:00:00

Family Count: 7
Friend Count: 13
Coworker Count: 31
Other Count: 50

--------------------
Handset ID: a37c920b3aac4cd5b58d3e4c4846277c
Relation to subject: family [SIMULATION DATA ONLY, would not be revealed real-world]
Contact periods:
        2020-04-17T08:00:00
        2020-04-18T10:00:00
        2020-04-20T08:00:00
        2020-04-20T16:00:00
        2020-04-21T16:00:00
        2020-04-21T17:00:00
        2020-04-24T16:00:00
        2020-04-25T10:00:00
        2020-04-26T08:00:00
        2020-04-26T09:00:00
--------------------
Handset ID: 7ca289d1b54c4cac9dc31847bffb3813
Relation to subject: family [SIMULATION DATA ONLY, would not be revealed real-world]
Contact periods:
        2020-04-13T18:00:00
        2020-04-16T09:00:00
        2020-04-17T17:00:00
        2020-04-18T08:00:00
        2020-04-19T10:00:00
        2020-04-22T08:00:00
        2020-04-22T09:00:00
        2020-04-24T17:00:00
        2020-04-25T08:00:00
        2020-04-25T14:00:00
--------------------
```

Figure 9.28: Results

9.6 K-Core Superspreading

To understand the effect of contact networks in lockdown, we need to understand the term bond percolation. The network connectivity in bond percolation is reduced by removing a small fraction of links between nodes, and the result is monitored by studying the size of the giant connected component [39].

9.6.1 Data Description

For the K-Core Superspreading algorithm, we used two different datasets, daily-cases.csv, and rmsdaily.csv. Below is the data sample from both of the datasets:

A
daily_cases
5
5
1
3
6
3
6
7
9
25
24
39
50
45
74
105
115
146
137

(a) Daily Cases

	date	rms
0	2020-03-01	780.877281
1	2020-03-02	887.25148
2	2020-03-03	868.409881
3	2020-03-04	897.718234
4	2020-03-05	887.240501
5	2020-03-06	940.151816
6	2020-03-07	875.304251
7	2020-03-08	891.484378
8	2020-03-09	961.899426
9	2020-03-10	906.47022
10	2020-03-11	897.407617
11	2020-03-12	906.182613
12	2020-03-13	988.211735
13	2020-03-14	890.609455
14	2020-03-15	940.575235
15	2020-03-16	968.147545
16	2020-03-17	813.68202
17	2020-03-18	779.030426
18	2020-03-19	641.298163
19	2020-03-20	728.903868
20	2020-03-21	647.890104

(b) RMS Daily

Figure 9.29: K-Core Superspreading Sample Data

9.6.2 Results

We find a large decrease in the GCC size within the 6-day lockdown on March 19 by 89.6% and the cumulative cases kept growing at a lower rate.

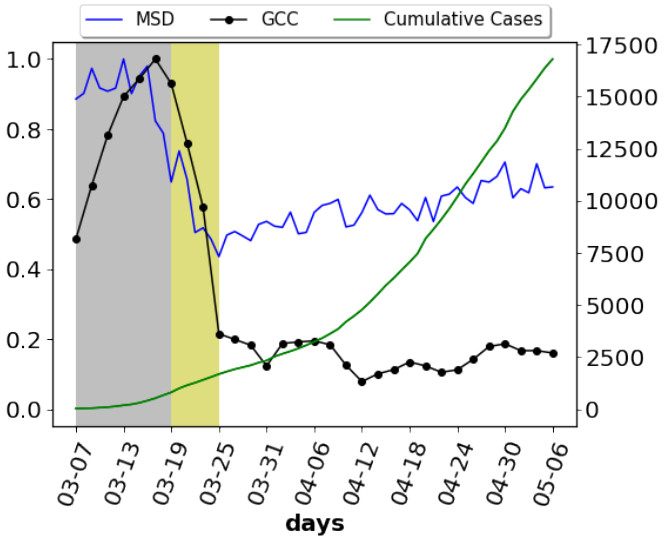


Figure 9.30: Structural components of transmission net- works across the lockdown (GCC)

The plot below shows the 0.5-kcore size (red), and the 0.5-kshell size (cyan) all normalized before lockdown. The size of 0.5-kshell is reduced during the lockdown, the 0.5-kcore wasn't reduced as much and keeps increasing.

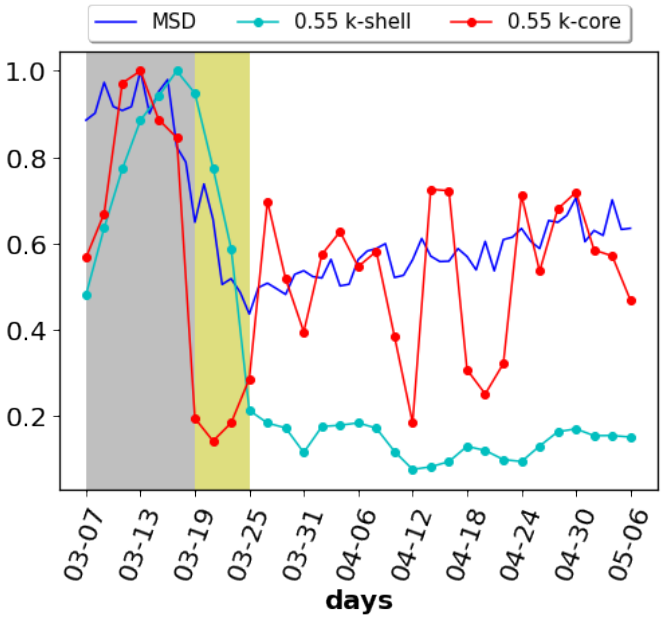


Figure 9.31: Structural components of transmission net- works across the lockdown (K-core)

Chapter 10

Conclusion and Future Works

Technology in contact tracing methods has evolved a long way, from paper-contact tracing to the recently Bluetooth-able contact tracing technology. Each method has its advantages and disadvantages. During the pandemic, many countries and companies tried to implement contact tracing, but some apps don't protect users' privacy. With the help of machine learning, different models were designed. We concluded that Google and Apple's Exposure Notification API works best to protect users' privacy, where users can opt-in to be able to use the app, and the app doesn't track users' location activity. On the other hand, the Clustering algorithm is the most effective in terms of tracking and tracing infected cases, but it has to sacrifice users' privacy since this algorithm needs users' locations for precise results.

Meanwhile, a Graph-based Epidemic Modeling algorithm could help further contact tracing and find the possible infections that occur from the first contact tracing. Also, the combination of forward and contact tracing shows that the proportion of contacts traced improves significantly, which helps health workers to trace more positive cases.

Table 10.1: Algorithms' findings

Algorithm	Findings
Forward and Backward Contact Tracing	Improves total contacts traced
Clustering	Effective to trace infected cases, but doesn't preserve users' privacy
Graph-Based Epidemic Modeling	Could help trace further contact tracing (with graph visualization)
Agent-Based Modeling	Helps to trace Contact Tracing with the creation of Agents and Models
Exposure Notification API	Protects users' privacy, since the app doesn't track users' location history
K-core Superspreading	GCC size decreases during lockdown. Kshell reduces during lockdown, but kcore keeps increasing

The trend for COVID-19 and contact tracing kept fluctuating in December 2021 and has decreased since 2022. Also, we can see that not all countries were keen to implement the Contact Tracing app for their citizens. One of the reasons is the concern for security and privacy. Meanwhile, some countries like South Korea managed to implement the Contact Tracing app in the country, preserved users' data privacy, and successfully slowed down COVID-19's spread.

10.1 Future Work

We hope that this research could assist medical professionals to compare and decide which methods are more suitable to track COVID-19 or other infectious diseases in the future. The research is still in progress as we keep analyzing other methods of contact tracing and which methods will benefit both medical professionals and users. This research is still in progress and we aim to keep analyzing different methods so it brings the most effective solution to Contact Tracing.

Bibliography

[1] *Covid-19 case*, 2022. [Online]. Available: https://gisanddata.maps.arcgis.com/apps/dashboards/bda7594740fd40299423467b48e9ecf6.

[2] WHO, *Coronavirus disease (covid-19): Contact tracing*, 2021. [Online]. Available: https://www.who.int/news-room/questions-and-answers/item/coronavirus-disease-covid-19-contact-tracing#:~:text=Contact%5C%20tracing%5C%20can%5C%20break%5C%20the,transmission%5C%20can%5C%20be%5C%20stopped..

[3] A. B. Dar, A. H. Lone, S. Zahoor, A. A. Khan, and R. Naaz, "Applicability of mobile contact tracing in fighting pandemic (covid-19): Issues, challenges and solutions," *Computer Science Review*, vol. 38, p. 100 307, 2020.

[4] *Main reasons why adults in the united states are not using a covid-19 (coronavirus) contact tracing app on their mobile phone as of december 2020*, 2020. [Online]. Available: https://www.statista.com/statistics/1197325/adults-us-reasons-not-using-contact-tracing-app-coronavirus/.

[5] L. Ceci, *Adoption of government endorsed covid-19 contact tracing apps in selected countries as of july 2020*, 2021. [Online]. Available: https://www.statista.com/statistics/1134669/share-populations-adopted-covid-contact-tracing-apps-countries/.

[6] WHO, "Contact tracing in the context of covid-19.," *Contact Tracing*, 2021.

[7] R. Ernszt, *The pros and cons of contact tracing apps*, 2020. [Online]. Available: https://www.digitalbulletin.com/ThoughtLeaders/Technology/2020/November/proprivacy/the-pros-and-cons-of-contact-tracing-apps/.

[8] A. L. Greiner, K. M. Angelo, A. M. McCollum, K. Mirkovic, R. Arthur, and F. J. Angulo, "Addressing contact tracing challenges—critical to halting ebola virus disease transmission," *International Journal of Infectious Diseases*, vol. 41, pp. 53–55, 2015.

[9] B. Sowmiya, V. S. Abhijith, S. Sudersan, R. Sakthi Jaya Sundar, M. Thangavel, and P. Varalakshmi, "A survey on security and privacy issues in contact tracing application of covid-19," *SN Computer Science*, vol. 2, no. 3, p. 136, 2021.

[10] S. Canada, *Willingness of canadians to use a contact tracing application*, 2020. [Online]. Available: https://www150.statcan.gc.ca/n1/pub/45-28-0001/2020001/article/00059-eng.htm.

[11] WHO, *Ethical considerations to guide the use of digital proximity tracking technologies for covid-19 contact tracing*, 2020. [Online]. Available: https://www.who.int/publications/i/item/WHO-2019-nCoV-Ethics_Contact_tracing_apps-2020.1.

[12] *Comparing centralized decentralized contact-tracing approaches*, 2021. [Online]. Available: `https://sites.sanford.duke.edu/techpolicy/2021/02/21/centralizedvsdecentralized/`.

[13] D. Storm van Leeuwen, A. Ahmed, C. Watterson, and N. Baghaei, "Contact tracing: Ensuring privacy and security," *Applied Sciences*, vol. 11, no. 21, 2021.

[14] J. Almagor and S. Picascia, "Exploring the effectiveness of a covid-19 contact tracing app using an agent-based model," *Scientific Reports*, vol. 10, no. 1, p. 22 235, 2020.

[15] P. Koetter, M. Pelton, J. Gonzalo, *et al.*, "Implementation and process of a covid-19 contact tracing initiative: Leveraging health professional students to extend the workforce during a pandemic," *American Journal of Infection Control*, vol. 48, no. 12, pp. 1451–1456, 2020.

[16] E. Clark, E. Y. Chiao, and E. S. Amirian, "Why contact tracing efforts have failed to curb coronavirus disease 2019 (covid-19) transmission in much of the united states," *Clinical Infectious Diseases*, vol. 72, no. 9, e415–e419, Aug. 2020.

[17] P. Gupta, T. Maharaj, M. Weiss, *et al.*, *Covi-agentsim: An agent-based model for evaluating methods of digital contact tracing*, 2020. DOI: `10.48550/ARXIV.2010.16004`. [Online]. Available: `https://arxiv.org/abs/2010.16004`.

[18] I. Nakamoto, M. Jiang, J. Zhang, *et al.*, "Evaluation of the design and implementation of a peer-to-peer covid-19 contact tracing mobile app (cocoa) in japan," *JMIR Mhealth Uhealth*, vol. 8, no. 12, e22098, Dec. 2020.

[19] J. A. Sacks, E. Zehe, C. Redick, *et al.*, "Introduction of mobile health tools to support ebola surveillance and contact tracing in guinea," *Global Health: Science and Practice*, vol. 3, no. 4, pp. 646–659, 2015.

[20] R. Sun, W. Wang, M. Xue, G. Tyson, and D. C. Ranasinghe, "Venuetrace: A privacy-by-design covid-19 digital contact tracing solution: Poster abstract," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 790–791, ISBN: 9781450375900. [Online]. Available: `https://doi.org/10.1145/3384419.3430615`.

[21] G.
bibinitperiod Apple, *Exposure notifications: Using technology to help public health authorities fight covid-19*, 2020. [Online]. Available: `https://www.google.com/covid19/exposurenotifications/`.

[22] C. C. Kerr, R. M. Stuart, D. Mistry, *et al.*, "Covasim: An agent-based model of covid-19 dynamics and interventions," *PLoS computational biology*, vol. 17, no. 7, e1009149–e1009149, Jul. 2021.

[23] S. L. T. Vangipuram, S. P. Mohanty, and E. Kougianos, "Covichain: A blockchain based framework for nonrepudiable contact tracing in healthcare cyber-physical systems during pandemic outbreaks," *SN Computer Science*, vol. 2, no. 5, p. 346, 2021.

[24] A. Polenta, P. Rignanese, P. Sernani, *et al.*, "An internet of things approach to contact tracing—the bubblebox system," *Information*, vol. 11, no. 7, 2020.

[25] W. Kim, H. Lee, and Y. D. Chung, "Safe contact tracing for covid-19: A method without privacy breach using functional encryption techniques based-on spatio-temporal trajectory data," *medRxiv*, 2020.

[26] A. Endo, Q. J. Leclerc, G. M. Knight, *et al.*, "Implication of backward contact tracing in the presence of overdispersed transmission in covid-19 outbreaks.," *Wellcome Open Res*, vol. 5, p. 239, 2020.

[27] P. H. Ontario, *Backward contact tracing*, 2021. [Online]. Available: https://www.publichealthontario.ca/-/media/documents/ncov/phm/2021/05/covid-19-backward-contact-tracing.pdf?la=en#:~:text=Backward%5C%20contact%5C%20tracing%5C%20(BCT)%5C%20is,than%5C%20forward%5C%20contact%5C%20tracing%5C%20alone..

[28] *What is cluster analysis? when should you use it for your survey results?* [Online]. Available: https://www.qualtrics.com/experience-management/research/cluster-analysis/.

[29] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *KDD*, 1996.

[30] K. Sarvakar, "Density based methods to discover clusters with arbitrary shape in weka," May 2013. DOI: 10.13140/2.1.2642.3686.

[31] A. Chauhan, *Fully explained dbscan clustering algorithm with python.* [Online]. Available: https://towardsai.net/p/machine-learning/fully-explained-dbscan-clustering-algorithm-with-python.

[32] N. S. Chauhan, *What is hierarchical clustering?* 2019. [Online]. Available: https://www.kdnuggets.com/2019/09/hierarchical-clustering.html.

[33] P. Sharma, *K means clustering simplified in python*, 2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/04/k-means-clustering-simplified-in-python/.

[34] B. Goncalves, *Network structure, super-spreaders and contact tracing*, 2020. [Online]. Available: https://medium.data4sci.com/epidemiology-201-network-structure-superspreaders-and-contact-tracing-336754e14e9a.

[35] J. Myers, *Contact tracing: Deep dive simulation*, 2020. [Online]. Available: https://gretel.ai/blog/contact-tracing-deep-dive-simulation.

[36] Y.-X. Kong, G.-Y. Shi, R.-J. Wu, and Y.-C. Zhang, "K-core: Theories and applications," *Physics Reports*, vol. 832, pp. 1–32, 2019.

[37] S. B. Seidman, "Network structure and minimum degree," *Social Networks*, vol. 5, no. 3, pp. 269–287, 1983.

[38] A. Siraj, A. Worku, K. Berhane, *et al.*, "Early estimates of covid-19 infections in small, medium and large population clusters," *BMJ Global Health*, vol. 5, no. 9, 2020.

[39] M. Serafino, H. Monteiro, S. Luo, *et al.*, "Superspreading k-cores at the center of covid-19 pandemic persistence," Aug. 2020. DOI: 10.1101/2020.08.12.20173476.