PyDOSS: An open-source Python-based communicable disease outbreak surveillance system designed and used in a COVID-19 community care facility

Andrew Fang, Jonathan Kia-Sheng Phua, Terrence Chiew, Daniel De-Liang Loh, Lincoln Ming Han Liow, Weien Chow, Charles Xian-Yang Goh, Hian Liang Huang

Submitted to: JMIR Public Health and Surveillance
on: June 24, 2020

Table of Contents

PyDOSS: An open-source Python-based communicable disease outbreak surveillance system designed and used in a COVID-19 community care facility

Andrew FangMBBS, MTech, MMED, ; Jonathan Kia-Sheng PhuaMBBS, ; Terrence Chiew; Daniel De-Liang LohMBBS, MRCS, ; Lincoln Ming Han LiowMBBS, DWD(CAW), MRCS, MMED, FRCSEd, FAMS, ; Weien Chow; Charles Xian-Yang Goh; Hian Liang HuangMBChB, MRCP,

Corresponding Author:
Andrew FangMBBS, MTech, MMED,
Phone: +6593690001
Email: andrew.fang.h.s@singhealth.com.sg

Abstract

Background: During the Coronavirus Disease 2019 (COVID-19) outbreak, community care facilities (CCF) were set up as temporary out-of-hospital isolation facilities to contain the surge of cases in Singapore. Confined living spaces within CCFs posed an increased risk of communicable disease spread among residents.

Objective: This inspired our healthcare team managing a CCF operation to design a low-cost communicable disease outbreak surveillance system (CDOSS).

Methods: Our CDOSS was designed with the following considerations: (1) comprehensiveness, (2) efficiency through passive reconnoitering from electronic medical record (EMR) data, (3) ability to provide spatiotemporal insights, (4) low-cost and (5) ease of use. We used Python to develop a lightweight application – Python-based Communicable Disease Outbreak Surveillance System (PyDOSS) – that was able perform syndromic surveillance and fever monitoring. With minimal user actions, its data pipeline would generate daily control charts and geospatial heat maps of cases from raw EMR data and logged vital signs. PyDOSS was successfully implemented as part of our CCF workflow. We also simulated a gastroenteritis (GE) outbreak to test the effectiveness of the system.

Results: PyDOSS was used throughout the entire duration of operation; the output was reviewed daily by senior management. No disease outbreaks were identified during our medical operation. In the simulated GE outbreak, PyDOSS was able to effectively detect an outbreak within 24 hours and provided information about cluster progression which could aid in contact tracing. The code for a stock version of PyDOSS has been made publicly available.

Conclusions: PyDOSS is an effective surveillance system which was successfully implemented in a real-life medical operation. With the system developed using open-source technology and the code made freely available, it significantly reduces the cost of developing and operating CDOSS and may be useful for similar temporary medical operations, or in resource-limited settings.

(JMIR Preprints 24/06/2020:21760)
DOI: https://doi.org/10.2196/preprints.21760

Preprint Settings

1) Would you like to publish your submitted manuscript as preprint?
✔ **Please make my preprint PDF available to anyone at any time (recommended).**
Please make my preprint PDF available only to logged-in users; I understand that my title and abstract will remain visible to all users.
Only make the preprint title and abstract visible.
No, I do not wish to publish my submitted manuscript as a preprint.
2) If accepted for publication in a JMIR journal, would you like the PDF to be visible to the public?
✔ **Yes, please make my accepted manuscript PDF available to anyone at any time (Recommended).**
Yes, but please make my accepted manuscript PDF available only to logged-in users; I understand that the title and abstract will remain v
Yes, but only make the title and abstract visible (see Important note, above). I understand that if I later pay to participate in <a href="http

Original Manuscript

PyDOSS: An open-source Python-based communicable disease outbreak surveillance system designed and used in a COVID-19 community care facility

Authors:

Authorship	Full Name	Appointment/Role	Institutional address
First Author	Dr Fang Hao Sen Andrew	Family Physician, SingHealth Polyclinics/ Medical Informatics Officer, SingHealth	167 Jalan Bukit Merah, Connection One (Tower 5), #15-10, Singapore 150167
Second Author	Dr Jonathan Kia-Sheng Phua	Medical Officer, Department of Diagnostic Radiology, Singapore General Hospital	1 Outram Road, Singapore 169608
Third Author	Dr Terrence Chiew	Clinical Associate, Singapore National Eye Centre	11 Third Hospital Ave, Singapore 168751
Fourth Author	Dr Daniel Loh De-Liang	Medical Officer, Department of Neurosurgery, Singapore General Hospital	1 Outram Road, Singapore 169608
Fifth Author	Dr Ming Han Lincoln Liow	Associate Consultant, Department of Orthopedic Surgery, Singapore General Hospital	1 Outram Road, Singapore 169608
Sixth Author	Dr Chow Weien	Consultant, Department of Cardiology, Changi General Hospital	2 Simei Street 3, Singapore 529889
Seventh Author	Dr Goh Xian-Yang, Charles	Consultant, Department of Nuclear Medicine and Molecular Imaging, Singapore General Hospital	1 Outram Road, Singapore 169608
Last Author	Dr Hian Liang Huang	Consultant, Department of Nuclear Medicine and Molecular Imaging, Singapore General Hospital	1 Outram Road, Singapore 169608

Corresponding Author:

Dr Fang Hao Sen Andrew
Address: 167 Jalan Bukit Merah, Connection One (Tower 5), #15-10, Singapore 150167
SingHealth Polyclinics Head Office
Telephone: +65-93690001
Email: andrew.fang.h.s@singhealth.com.sg

Abstract

Background:

During the Coronavirus Disease 2019 (COVID-19) outbreak, community care facilities (CCF) were set up as temporary out-of-hospital isolation facilities to contain the surge of cases in Singapore. Confined living spaces within CCFs posed an increased risk of communicable disease spread among residents. This inspired our healthcare team managing a CCF operation to design a low-cost communicable disease outbreak surveillance system (CDOSS).

Methods:

Our CDOSS was designed with the following considerations: (1) comprehensiveness, (2) efficiency through passive reconnoitering from electronic medical record (EMR) data, (3) ability to provide spatiotemporal insights, (4) low-cost and (5) ease of use. We used Python to develop a lightweight application – Python-based Communicable Disease Outbreak Surveillance System (PyDOSS) – that was able perform syndromic surveillance and fever monitoring. With minimal user actions, its data pipeline would generate daily control charts and geospatial heat maps of cases from raw EMR data and logged vital signs. PyDOSS was successfully implemented as part of our CCF workflow. We also simulated a gastroenteritis (GE) outbreak to test the effectiveness of the system.

Results:

PyDOSS was used throughout the entire duration of operation; the output was reviewed daily by senior management. No disease outbreaks were identified during our medical operation. In the simulated GE outbreak, PyDOSS was able to effectively detect an outbreak within 24 hours and provided information about cluster progression which could aid in contact tracing. The code for a stock version of PyDOSS has been made publicly available.

Conclusions:

PyDOSS is an effective surveillance system which was successfully implemented in a real-life medical operation. With the system developed using open-source technology and the code made freely available, it significantly reduces the cost of developing and operating CDOSS and may be useful for similar temporary medical operations, or in resource-limited settings.

Funding:

None

Keywords:

Disease surveillance, COVID-19, open-source

Introduction

On March 11, 2020, the World Health Organization (WHO) declared the Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2) outbreak as a pandemic.(1) Singapore, a city-state and international travel hub reported its first imported case on January 23, 2020 and was one of the first countries to be affected by Coronavirus Disease 2019 (COVID-19).(2) Since then, there have been 42,313 confirmed cases and 26 deaths in Singapore, as of this writing on June 23, 2020.(3)

In order to contain the surge of COVID-19 infection and preserve hospital capacity for critically ill patients, Singapore's Ministry of Health (MOH) set up several out-of-hospital isolation facilities.(4) Community care facilities (CCFs) were designed to accommodate COVID-19 patients with mild symptoms and low risk factors.(5) These CCFs provided capacity to hold up to 20,000 patients for the duration of the isolation period.

Closely shared living spaces within CCFs posed an increased risk of communicable disease spread among residents. This drove the need to develop a disease outbreak surveillance system to provide early detection of potential outbreaks within the CCF. As CCF operations were meant to be temporary, investing in proprietary commercial surveillance software or developing a full-suite disease surveillance system were economically non-feasible.(6,7) While free software tools for disease surveillance were available, they had several drawbacks. Namely, they were incompatible with our electronic medical record (EMR) system, relied on labour-intensive manual data entry and posed security concerns with regard to personal data protection.(8,9)

Our team developed a bespoke, effective, low-cost communicable disease outbreak surveillance system (CDOSS) for CCF operations. This paper will describe the design, post-implementation review and outbreak simulation outcomes of our CDOSS – Python-based Disease Outbreak Surveillance System (PyDOSS). To our knowledge, this is the first fully Python-based spatiotemporal CDOSS that has been made open-source.

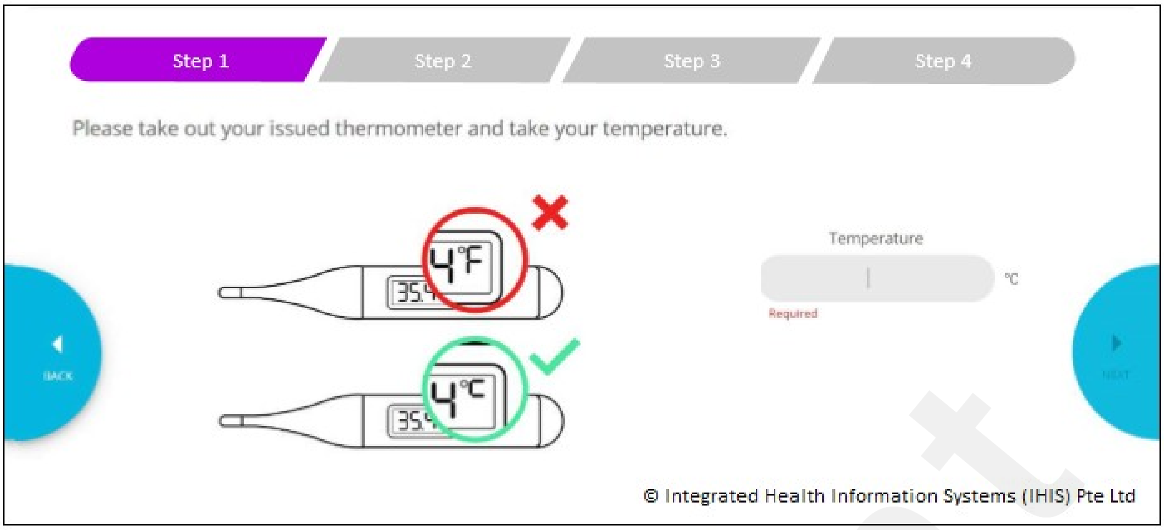
Methods

Setting

Singapore EXPO, an exhibition hall venue was repurposed into a 8,000 bed CCF (CCF@EXPO).(11,12) Singhealth was tasked by MOH to manage the healthcare needs of 3,200 residents housed in four halls. The halls were organized into rows and rooms. Each room was shared by two residents. Residents were young to middle-aged adult male migrant workers who had tested positive for COVID-19 infection. The majority were clinically well or had mild symptoms. Singapore's policy was to isolate COVID-19 positive individuals for twenty-one days from the day of symptom onset or diagnosis, whichever earlier, before they could be discharged. This was twice the duration after which patients were no longer infectious, as suggested by local data.(13)

Primary healthcare teams provided basic medical screening at point of admission and primary-care services for CCF@EXPO residents throughout their stay. Physicians utilised GP Connect (GPC), which was a primary-care clinic based EMR system developed by MOH (14) to perform clinical documentation, medication management and results review for CCF residents. Residents were required to measure their own vital signs, including temperature, and input their results into an electronic data capture system (Health Discovery) at least once a day. (Figure 1) The compliance to vital signs monitoring was consistently above 99.5% for our CCF residents. Integrated Health Information Systems (IHIS), the agency which managed GPC and Health Discovery in CCF@EXPO aided in the extraction of EMR and vital signs data which were used for this project.

Figure 1. Screen capture of temperature self-input in Health Discovery



The healthcare management team convened daily to discuss outstanding issues in the CCF. Daily updates included admission, discharge, occupancy and primary care statistics. Bearing in mind the increased risk of communicable disease spread within the CCF, the management team requested clarity on the spectrum of primary care cases seen within the CCF, highlighting the need to detect potential communicable disease outbreaks. This led to the development of PyDOSS.

Design of PyDOSS

Our disease surveillance system was designed with the following considerations:

1. Comprehensive and early detection of communicable diseases prevalent in CCF@EXPO
2. Efficient passive reporting by leveraging EMR data
3. Able to provide intuitive insights through spatiotemporal information
4. Low-cost of development
5. Ease of use

To ensure that the system was targeted in its coverage of prevailing communicable diseases, we consulted an Infectious Disease (ID) specialist whom advised us to monitor for gastroenteritis, chickenpox, measles, mumps, rubella, dengue and scabies outbreaks. It was recommended that influenza be excluded as all residents were COVID-19 positive and were expected to have acute respiratory symptoms. We decided to implement syndromic surveillance, in addition to passive surveillance. Syndromic surveillance enabled early detection of outbreaks before confirmed diagnoses were made.(15,16) Two disease syndromes – (1) acute diarrhoeal illness and (2) potentially infectious rash were identified for syndromic surveillance.

For syndromic surveillance, we leveraged EMR data to analyze (1) visit dates, (2) patient location and (3) finalized diagnoses as input into the disease surveillance system. From a restricted list of 1,306 Systematized Nomenclature of Medicine – Clinical Terms (SNOMED-CT) primary care diagnoses available in GPC, we studied recent records and adopted a consensus-driven approach to carefully select a list of diagnoses as indicators for the two disease syndromes of interest. Table 1 was the final list of diagnoses that were used as indicators for syndromic surveillance.

Table 1. List of diagnoses from EMR for syndromic surveillance

Syndrome	**Acute diarrhoeal illness**	**Potentially infectious rash**
Diagnosis	• Acute diarrhoea	• Dengue (38362002)

(SNOMED-CT Concept ID)	(409966000) • Diarrhoea (62315008) • Enteritis (64613007) • Gastroenteritis (25374005) • Vomiting (422400008)	• Disorder of skin (95320005) • Eruption (271807003) • Herpes zoster (4740000) • Herpes zoster without complication (111859007) • Infestation by Sarcoptes scabiei var hominis (128869009) • Itching (418290006) • Itching of skin (418363000)Measles (14189004) • Measles without complication (111873003) • Rubella (36653000) • Varicella (38907003)

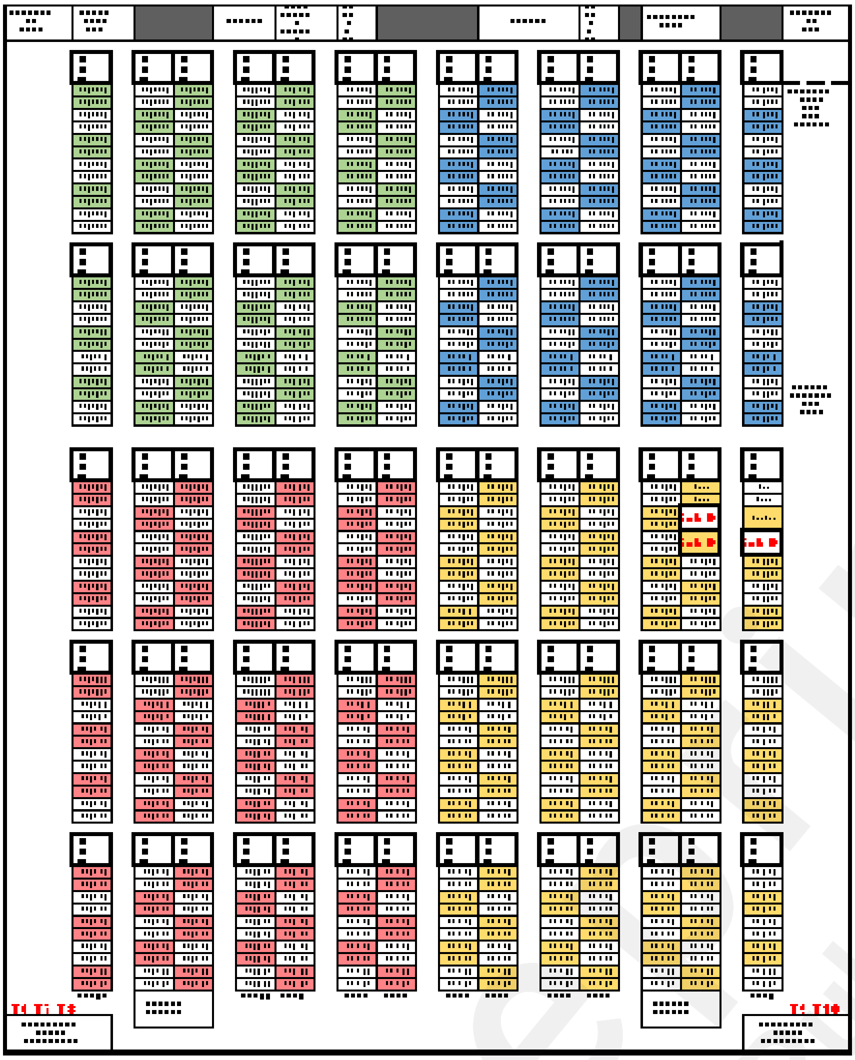
In addition to syndromic surveillance, we also monitored for cases of fever, which we defined as temperature ≥37.5 degree Celsius. We used vital signs data from Health Discovery for the fever monitoring. Fever was selected as it is known to be a common and early objective indicator of infection, manifesting before other vital signs would turn abnormal.

With the knowledge that communicable diseases were likely to cluster in space, adding spatial information to traditional time-based methods would provide additional power and efficacy in detecting outbreaks.(17) A recent review found about a third of public health surveillance algorithms made use of spatial information.(18) We thus designed the system to generate two outputs: (1) a control chart to identify time-based aberrations and (2) a geospatial map of cases to highlight any physical clustering of cases.

For the control charts, we analyzed historical data and computed the mean and standard deviation of the daily number of cases for each of the two disease syndromes. An initial threshold of two standard deviation from the daily mean was chosen as a reasonable balance between sensitivity and specificity. As the daily numbers followed a normal distribution curve, we expected a trigger rate of about 2.5%. The absolute numbers were also tracked prospectively to evaluate the need to adjust the threshold.

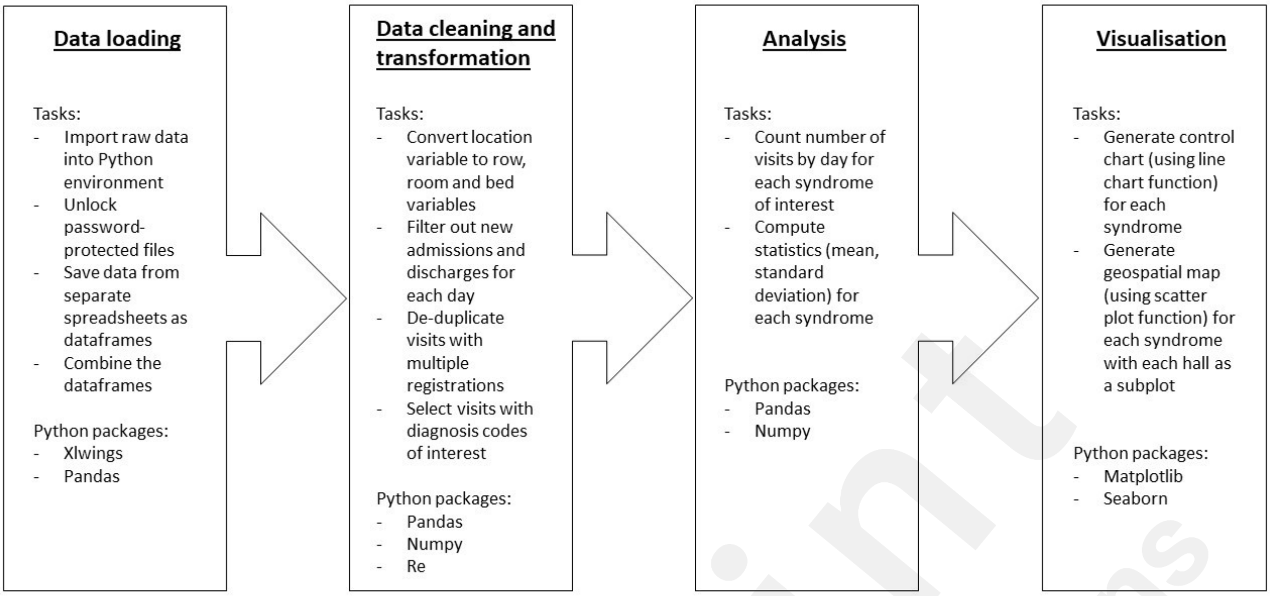
For the geospatial map, we used the CCF@EXPO hall bed layouts and highlighted positive cases within windows of the past five and three days for syndromic surveillance and fever surveillance respectively (Figure 2) Unique maps were generated for each disease. Diseases were tagged to bed location and flagged as coloured triangles to allow cluster detection by visual inspection. The reason for the look-back window was based on the rationale that cases of transmissible diseases tend to disseminate and develop over a period of time, instead of all within the same day.

Figure 2. Overview of bed layout in a CCF@EXPO hall



To address the consideration of cost, we developed the system with Python 3.7, which is an open-source programming language.(19) In addition, we used the following Python packages: “xlwings” package for reading the EMR data, the “numpy”, “pandas” and “re” packages for data cleaning, transformation and analysis, and the “matplotlib” and “seaborn” packages for data visualization.(20–25) Figure 3 illustrates the data pipeline we developed for PyDOSS.

Figure 3. PyDOSS data pipeline



Finally, to achieve ease-of-use, we followed a user one-click philosophy. With minimal user actions, PyDOSS should be able to seamlessly load and flow the raw EMR data through its data pipeline to generate the desired outputs.

About PyDOSS

PyDOSS was a lightweight application. It consisted of a single Python file, less than 100 kilobytes in size. It only required users to open the file, select the path to the raw EMR data file on the local computer and then click to run. The application would generate the control charts and geospatial maps. (Figures 4, 5 and 6) The code for the stock version of PyDOSS and an example dataset have been made available on GitHub.(26)

Figure 4. Example control chart of daily report sick for acute diarrhoeal illness syndrome

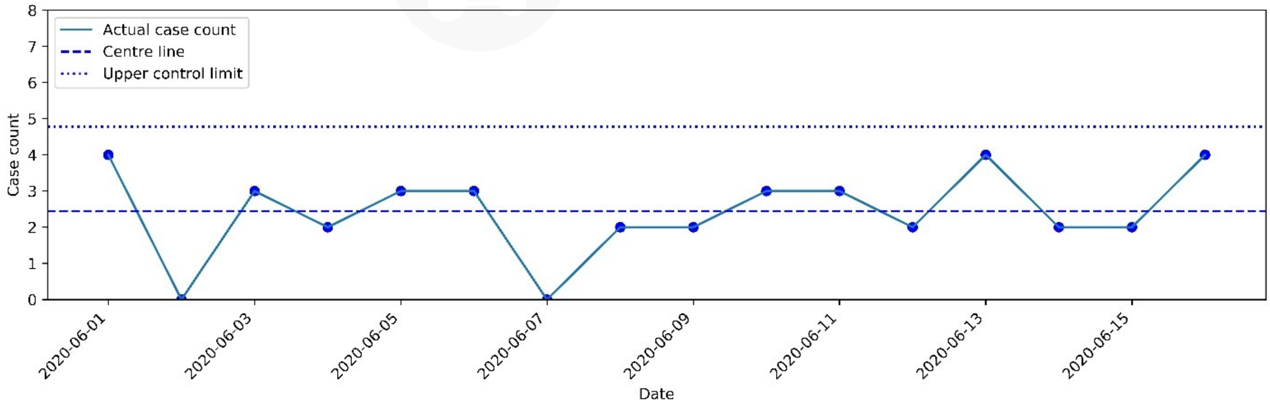


Figure 5. Example geospatial map of residents who reported sick for potentially infectious rash syndrome[1]

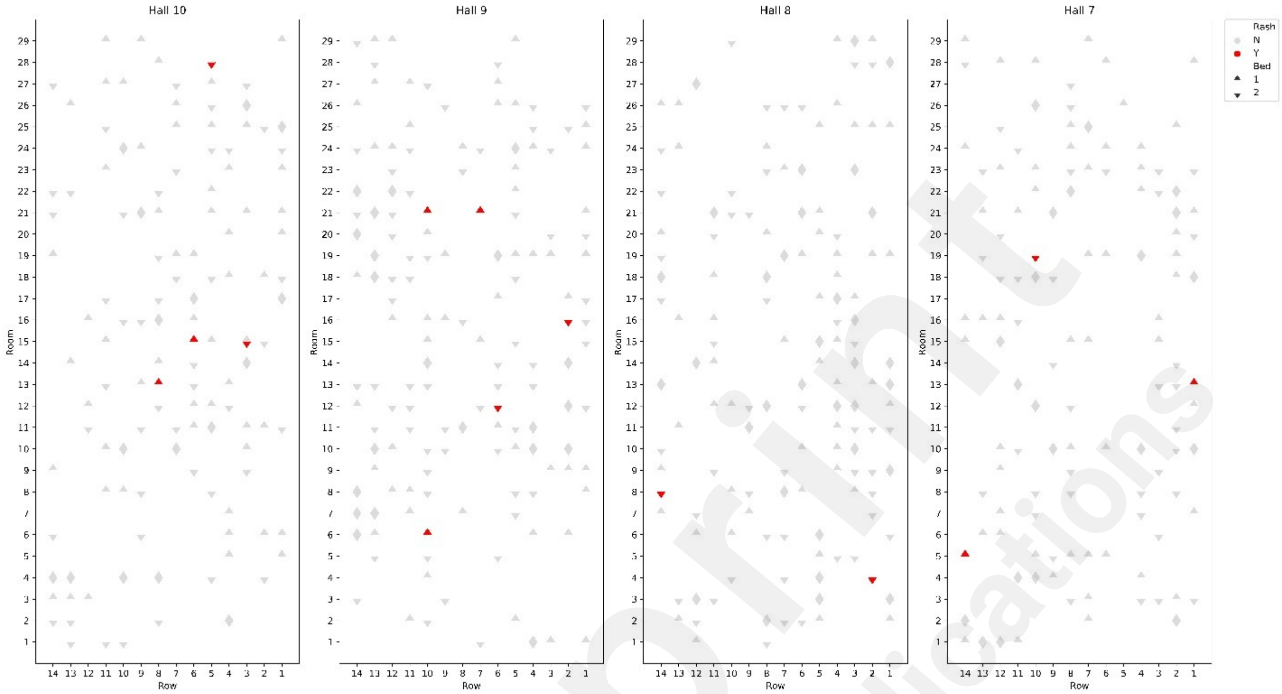
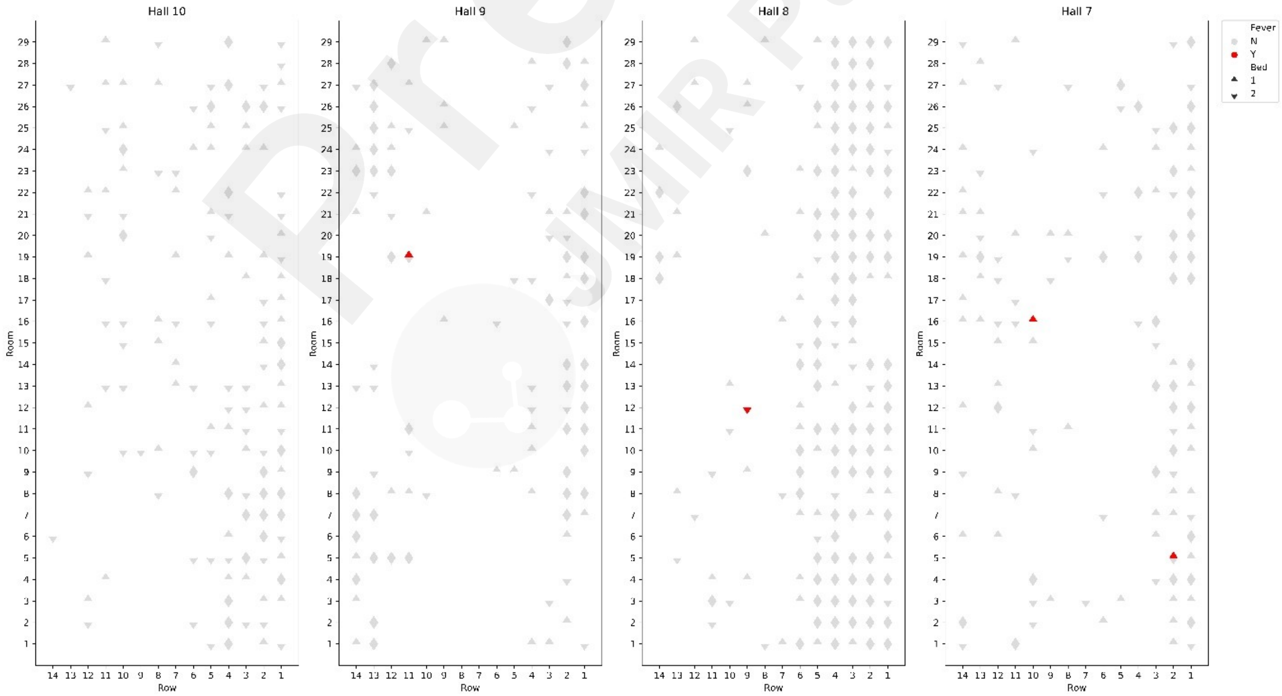


Figure 6. Example geospatial map of residents who recorded fever on vital signs monitoring



[1] Each hall could contain up to 800 residents. The halls were arranged side by side. Within each hall, there were 14 rows and 29 rooms per row. Each room could accommodate 2 beds. Some rooms were designated as a sickbay where residents could report sick and be monitored closely.

Results

Implementation of the disease surveillance system

PyDOSS was implemented after refinement and validation of the data pipeline. As part of a daily workflow, IHIS would provide our team with a password-protected file containing EMR data of primary care visits from the previous day. Our team would use PyDOSS to parse the EMR data which generated the control chart and geospatial maps. The process of parsing the data was instantaneous. The control charts and geospatial maps were then sent to an ID specialist for review. If it was deemed that there was potential outbreak, the medical and infection prevention and epidemiology teams would be alerted immediately for follow-up action. The control chart and geospatial maps were also compiled into the daily updates to the healthcare management team.

Testing of the disease surveillance system

During the initial two weeks, there were no triggers from PyDOSS of potential disease outbreaks. In order to test PyDOSS performance, we simulated a gastroenteritis outbreak. We conducted the simulation by first defining a time and location of the outbreak. Next, we inserted dummy cases of gastroenteritis into the EMR data within the vicinity of the cluster over a five day period. We then serially ran the PyDOSS script on this modified dataset over the days prior, during and after the simulated outbreak period. The results from the test showed that PyDOSS was able to detect the outbreak within a day. (Figure 7) It was also able to track the progression of the cluster which would have helped in contact tracing and investigation of the source. (Figure 8)

Figure 7. Control chart of daily sick reports for a simulated acute diarrhoeal illness outbreak

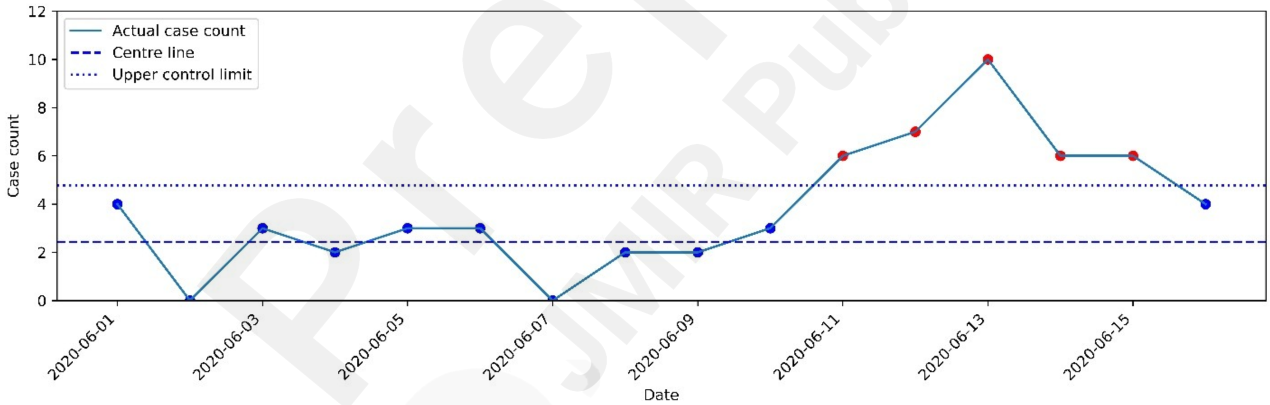
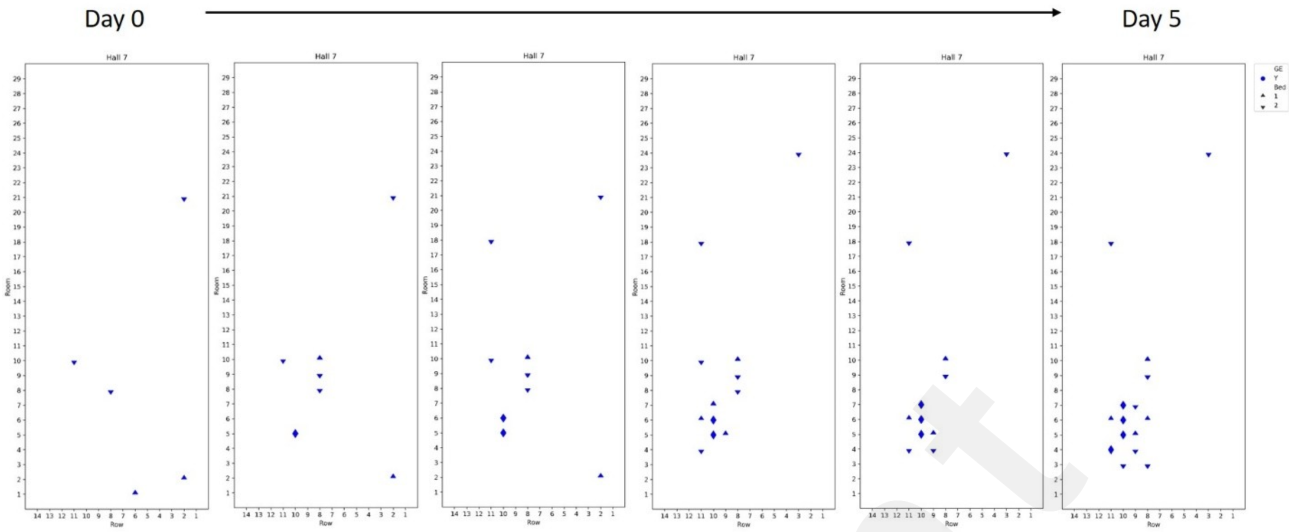


Figure 8. Serial geospatial maps of residents who reported sick for a simulated acute diarrhoeal illness outbreak



Discussion

We described the design of an easy-to-use, low-cost disease outbreak surveillance system that leverages EMR data to provide spatiotemporal information of recent cases. It was successfully implemented as part of our healthcare operations in managing COVID-19 patients within a CCF, and was demonstrated to be capable of detecting potential outbreaks early in a simulated model.

This method of designing and implementing low-cost disease surveillance systems which leverage EMR data and open-source software can potentially be used in similar temporary healthcare operations, such as humanitarian missions and military medical operations, or in resource-limited settings. Whilst several open-source tools for disease surveillance already exist, the strength of a CDOSS like PyDOSS is that it leverages raw extracted EMR data to reduce inefficiency of manual data entry or cleaning, is EMR agnostic and can be run on an internet separated computer if a connection is unavailable or to maximize personal data protection. Table 2 shows a comparison of PyDOSS with two other open-source disease surveillance software tools, OpenESSENCE and Epi Info™.(10,27)

Table 2. Comparison of PyDOSS with other open-source disease surveillance systems – OpenESSENCE and Epi Info™

	PyDOSS	**OpenESSENCE**	**Epi Info™ Community Edition**
Software installation	Not required	Installation required	Installation required
Cost of use	Free-of-charge	Free-of-charge	Free-of-charge
Ease of use	One-click data loading and reporting	User training advised	User training advised
Able to leverage EMR data	Yes. Able to work with raw extracted EMR data.	Only integrated with specific EMR systems such as OpenMRS. Only	Yes, but only able to read cleaned spreadsheet data.

		able to read cleaned spreadsheet data.	
Able to provide spatiotemporal information	Yes	Yes	Yes
Able to modify reporting and diseases of interest	Yes, with basic knowledge of Python	Only if additional modules are available	Only if additional modules are available

We chose to use Python over alternative programming software such as R and Matlab for building data pipelines as it consistently ranked as one of the top programming languages over the past few years.(28,29) Compared to R, Python was technically superior in that it was designed to be general purpose and for object-oriented programming, whereas R was more specialized in handling statistics. Another advantage Python had over R was its popularity and much wider developer base.(30) This would naturally allow for easier adoption of programs written in Python. We did not consider Matlab as it required a paid license to use.

From the geospatial maps, it can be seen that PyDOSS works well when the population under surveillance was evenly spread out and individuals occupied exclusive spaces. Such a grid-like arrangement would also be typical in other healthcare institutions like hospitals and domiciliary care institutions. Considering its use in settings where dense and sparse occupancy areas are inter-mixed, we would still expect PyDOSS to function well. Moreover, with the ability to adjust marker transparency, cases (coloured markers) that overlapped would be given greater prominence, producing what would be an impressive heatmap visualization.

PyDOSS currently relies on heuristics like visual inspection to detect outbreaks from spatiotemporal information. In this area, research has been done to develop and validate alternative statistical methods for spatiotemporal disease surveillance. Such methods include cumulative sum, scan statistics and model-based algorithms.(31–33) These methods could be incorporated to further systematize and improve PyDOSS' performance.

The interpretation of PyDOSS' outputs needs to be done in the context in which it is applied. In our CCF operations, PyDOSS used residents' bed location as a proxy for their whereabouts, while in reality residents were free to move around within their designated hall. The residents also shared amenities and equipment, such as toilets and vital signs monitoring devices. The fluid, dynamic mixing and interaction among residents could potentially stymie the early predictive capability of the geospatial maps in cluster detection. Nevertheless, from the control charts, PyDOSS still preserved the capability to detect any intra-hall aberration in daily case numbers to mitigate this limitation.

Moving forward, we have shared PyDOSS with IHIS and provided the source code to evaluate integration with GPC and Health Discovery so as to further improve our workflow efficiency. Additionally, we have made open-source a stock template of PyDOSS code for others who may wish to modify it for their own purposes.

Conclusion

In managing the medical operations of a COVID-19 isolation facility, we were motivated to develop a low-cost CDOSS. This paper presents the design of PyDOSS, illustrated its implementation in a real-life medical operation and demonstrated its effectiveness in detecting outbreaks through simulation. It leverages EMR data to provide near real-time spatiotemporal syndromic surveillance, allowing for early detection of outbreaks and providing better situational awareness and support for contact tracing. In the spirit of the open science movement, we have made PyDOSS open-source so that others, especially those in similar temporary medical operations or resource-limited settings, may be able to modify it to develop their own CDOSS.(34)

Contributors

FHSA and GXC conceptualized the PyDOSS system. FHSA, PKJ, CT and HHL developed and implemented the system. PKJ and LDD tested the system. FHSA and PKJ wrote the first draft of the paper. LDD, LMHL, GXC and CW provided comments for the first draft. All authors reviewed and approved the final manuscript.

Conflicts of interests

None declared.

Acknowledgements

We acknowledge IHIS which helped to provide the data for this work. We also thank SingEx and Resorts World Sentosa for providing the logistical support during our CCF operations.

References

1. WHO Director-General’s opening remarks at the media briefing on COVID-19 - 11 March 2020 [Internet]. [cited 2020 Jun 18]. Available from: https://www.who.int/dg/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---11-march-2020

2. Wong JEL, Leo YS, Tan CC. COVID-19 in Singapore—Current Experience: Critical Global Issues That Require Attention and Action. JAMA. 2020 Apr 7;323(13):1243–4.

3. Coronavirus Disease (COVID-19) Situation Reports [Internet]. [cited 2020 Jun 13]. Available from: https://www.who.int/emergencies/diseases/novel-coronavirus-2019/situation-reports

4. hermesauto. Patients who are well but still testing positive for Covid-19 to be moved to community isolation facility to preserve hospital capacity [Internet]. The Straits Times. 2020 [cited 2020 Jun 13]. Available from: https://www.straitstimes.com/singapore/patients-who-are-well-but-still-testing-positive-for-covid-19-to-be-moved-to-community

5. hermesauto. Community care facilities for Covid-19 patients with mild symptoms will double bed capacity to 20,000 by end-June [Internet]. The Straits Times. 2020 [cited 2020 Jun 13]. Available from: https://www.straitstimes.com/singapore/coronavirus-community-care-facilities-for-patients-with-mild-symptoms-will-have-bed

6. Infectious Disease Surveillance System | Outbreak Management [Internet]. Conduent. [cited 2020 Jun 13]. Available from: https://www.conduent.com/solution/public-health-technology/disease-surveillance-system/

7. WorldCare | Sunquest [Internet]. 2018 [cited 2020 Jun 13]. Available from: https://www.sunquestinfo.com/software-and-services/world-care/

8. Disease Surveillance System [Internet]. [cited 2020 Jun 13]. Available from: https://www.ihis.com.sg/Awards/Pages/2018%20NHITEA/Cat-B-Disease-Surveillance-System.aspx

9. Campbell TC, Hodanics CJ, Babin SM, Poku AM, Wojcik RA, Skora JF, et al. Developing open source, self-contained disease surveillance software applications for use in resource-limited settings. BMC Med Inform Decis Mak. 2012 Sep 6;12(1):99.

10. Nieves E, Jones J. Epi Info™: Now an Open-source application that continues a long and productive “life” through CDC support and funding. Pan Afr Med J [Internet]. 2009 Apr 30 [cited 2020 Jun 13];2. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2984269/

11. Singapore Expo’s Covid-19 care facility: How Surbana Jurong set up 2 halls in 3 days [Internet]. TODAYonline. [cited 2020 Jun 13]. Available from: https://www.todayonline.com/singapore/singapore-expo-covid-19-care-facility-how-surbana-jurong-set-up-2-halls-in-3-days

12. A look into building and managing a COVID-19 community care facility | Video [Internet]. CNA. [cited 2020 Jun 13]. Available from: https://www.channelnewsasia.com/news/singapore/a-look-into-building-and-managing-a-covid-19-community-care-12739516

13. MIN COW. Position Statement from the National Centre for Infectious Diseases and the Chapter of Infectious Disease Physicians, Academy of Medicine, Singapore: Period of Infectivity to Inform Strategies for De-isolation for COVID-19 Patients. [Internet]. Elements. 2020 [cited 2020 Jun 22]. Available from: https://scholarbank.nus.edu.sg/handle/10635/168938

14. GPConnect [Internet]. [cited 2020 Jun 13]. Available from: https://www.ihis.com.sg/Project_Showcase/Healthcare_Systems/Pages/GPConnect.aspx

15. Pavlin JA. Investigation of disease outbreaks detected by “syndromic” surveillance systems. J Urban Health. 2003 Mar 1;80(1):i107–14.

16. May L, Chretien J-P, Pavlin JA. Beyond traditional surveillance: applying syndromic surveillance to developing settings – opportunities and challenges. BMC Public Health. 2009 Jul 16;9(1):242.

17. Sparks R, Bolt S, Okugami C. Spatio-Temporal Disease Surveillance. In 2012.

18. Yuan M, Boston-Fisher N, Luo Y, Verma A, Buckeridge DL. A systematic review of aberration detection algorithms used in public health surveillance. J Biomed Inform. 2019 Jun 1;94:103181.

19. Python.org [Internet]. Python.org. [cited 2020 Jun 13]. Available from: https://www.python.org/

20. Python for Excel [Internet]. xlwings. [cited 2020 Jun 13]. Available from: https://www.xlwings.org/

21. NumPy [Internet]. [cited 2020 Jun 13]. Available from: https://numpy.org/

22. pandas - Python Data Analysis Library [Internet]. [cited 2020 Jun 13]. Available from: https://pandas.pydata.org/

23. re — Regular expression operations — Python 3.8.3 documentation [Internet]. [cited 2020 Jun 19]. Available from: https://docs.python.org/3/library/re.html

24. Matplotlib: Python plotting — Matplotlib 3.2.1 documentation [Internet]. [cited 2020 Jun 13]. Available from: https://matplotlib.org/

25. seaborn: statistical data visualization — seaborn 0.10.1 documentation [Internet]. [cited 2020 Jun 13]. Available from: https://seaborn.pydata.org/

26. Fang A. andrewfanghs/pydoss [Internet]. 2020 [cited 2020 Jun 24]. Available from: https://github.com/andrewfanghs/pydoss

27. Hodanics C. OpenESSENCE: Disease Surveillance through Medical Record System Integration with OpenMRS. Online J Public Health Inform [Internet]. 2014 Apr 29 [cited 2020 Jun 14];6(1). Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4050757/

28. Cass S. The 2018 Top Programming Languages. IEEE Spectr. 2018;2.

29. Ozgur C, University V. MatLab vs. Python vs. R. :19.

30. Guo P. Python Is Now the Most Popular Introductory Teaching Language at Top U.S. Universities [Internet]. [cited 2020 Jun 19]. Available from: https://cacm.acm.org/blogs/blog-cacm/176450-python-is-now-the-most-popular-introductory-teaching-language-at-top-u-s-universities/fulltext

31. Robertson C, Nelson TA, MacNab YC, Lawson AB. Review of methods for space–time disease surveillance. Spat Spatio-Temporal Epidemiol. 2010 Jul;1(2):105–16.

32. Hl Z, Sj L, Zj L, Yj L, Wz Y. [Application of cumulative sum control chart algorithm in the detection of infectious disease outbreaks]. Zhonghua Liu Xing Bing Xue Za Zhi Zhonghua Liuxingbingxue Zazhi. 2010 Dec 1;31(12):1406–9.

33. Kulldorff M, Heffernan R, Hartman J, Assunção R, Mostashari F. A Space–Time Permutation Scan Statistic for Disease Outbreak Detection. Blower SM, editor. PLoS Med. 2005 Feb 15;2(3):e59.

34. Open science is a research accelerator | Nature Chemistry [Internet]. [cited 2020 Jun 18]. Available from: https://www.nature.com/articles/nchem.1149

Abbreviations

CCF: Community care facility
CCF@EXPO: Community care facility at the Singapore EXPO

CDOSS: Communicable disease outbreak surveillance system
EMR: Electronic medical records
GE: Gastroenteritis
GPC: GP Connect
ID: Infectious Diseases
IHIS: Integrated Health Information Systems
PyDOSS: Python-based Disease Outbreak Surveillance System
SNOMED-CT: Systematized Nomenclature of Medicine – Clinical Terms
WHO: World Health Organization