Utilizing GPS Data To Track Potential COVID19 Patients Using Cloud Based Document Database And Preventing Further Contagion

Aditya Kumar
Dept of Computer Science and Engineering
Birsa Institute of Technology, Sindri

Abstract- *COVID19 is a highly infectious disease caused by the most recently discovered Corona Virus. Corona viruses are a large family of viruses which may cause illness in animals or humans. In humans, several coronaviruses are known to cause respiratory infections ranging from the common cold to more severe diseases such as Middle East Respiratory Syndrome (MERS) and Severe Acute Respiratory Syndrome (SARS). The new virus called SARS-CoV-2 and disease were unknown before the outbreak began in Wuhan, China, in December 2019. According to WHO and Ministry of Health and Family Welfare (Govt. of India), people who have been infected by the new Corona virus can spread the virus when he/she coughs or exhales. The droplets from the infected person's nose or mouth can land of surfaces and objects. The viruses on the surfaces can remain upto 72 hours and can infect other people who touch the surface, then touching their eyes, nose or mouth. People can also catch COVID-19 if they breathe in droplets from a person with COVID-19 who coughs out or exhales droplets.Studies to date suggest that the virus that causes COVID-19 is mainly transmitted through contact with respiratory droplets rather than through the air.The risk of catching COVID-19 from someone with no symptoms at all is very low. However, many people with COVID-19 experience only mild symptoms. This is particularly true at the early stages of the disease. It is therefore possible to catch COVID-19 from someone who has, for example, just a mild cough and does not feel ill. It is very important to prevent the spread of this highly infectious virus. However, the prevention of contagion is possible only when all the potential patients can be identified and quarantined for a period of 14-21 days. Identifying every potential patient is a very difficult task. This paper aims at proposing techniques that allow the effective utilisation of an individual's smartphone GPS data to identify the number of people who were in the close proximity. People who were in close proximity of an identified patient can be tested and then quarantined if found positive. This will allow the proper use of testing kits that are not abundant at the moment. This paper also aims at making the jobs of health care professionals easier, who are tracing the potential infected subjects by introducing an idea for an application that allows the comparison of the GPS data of certain in infected individuals and individuals who were in a close proximity of the infected ones within a specified time period. There are now places around the world (cities or areas) where the disease is spreading. For people living in, or visiting, these areas the risk of catching COVID-19 is higher. The GPS data of people who visited these hotspots or live there can be monitored regularly. NoSQL databases such as MongoDB can be utilised to store the GPS data as these type of data can be massive unstructured datasets. The NoSQL databases is chosen because of the sharing feature, which allows parallel storage and processing.*

Keywords- GPS, COVID19, NoSQL database, Contagion, Corona Virus, Path Query, Cloud based databases, MongoDB, processing, querying, distributed users, location data tracking.

I. INTRODUCTION

Today, almost 3.5 billion people use a smartphone[5]. This number is expected to reach 3.8 billion in 2021. Mobile devices have become quite popular in recent years and the use of GPS is also increasingly common. Almost every smartphone uses the GPS for location based activities. The smartphones utilises transmission through GSM/HSPA/LTE and GPS to locate. This location data can be utilised to identify the potential COVID19 patients. The people who were in the close proximity of the infected individual can be tracked down and tested easily. The contagion can be slowed down drastically. The locations and cities where the risk of COVID19 is higher can benefit by utilising the location data. The location data of people who visited these hotspots or live there can be monitored regularly. A government monitored application/web portal can utilise the proposed technique to monitor a large number of smartphones (specially in the hotspots) to contain the spread of COVID19. This system is also future proof as this type of pandemic can occur in future and an exsisting system to track the infected individuals can definitely be a great help.

II. STORAGE SYSTEM

We have to consider a large user base. Any cloud based application faces large traffic flow, and must be able to withstand sudden and huge surges while ensuring low latency, high stability, and real-time scalability in the future. We also need to consider that the amount of data stored is relatively large (as large number of smartphone need to be monitored to prevent the contagion), and the data size, scenarios, and size of the user base are closely related. COVID19 seems to spread at an exponential rate [7]. Thus, the application/web portal can experience an explosive growth period so the system needs to be quickly scalable.The data is time-sensitive. Users may need to query a range by time, or query the ride path with the starting time and ending time unknown. We also need a low cost system to utilise this technique as the funds for health care services and food are more important at this point of time in the pandemic. Support for high concurrency writes is also very important.

The characteristics of the storage system can be thus can be summarised as: Scalability, Low cost, high stability, support for high concurrency writes, support for range of queries and large storage capacity in a single table.

III. CLOUD BASED NOSQL DATABASES AND DOCUMENT DATABASES

According to the specified characteristics of the storage system, we prefer the cloud based NoSQL databases such as MongoDB, CouchDB, Simple DB. The databases such as MongoDB are document databases. Document databases were designed to handle the storage and the management of large scale documents [4]. These type of database assign a key value to each document. Documents may contain multiple key-value pairs, or key-array pairs, or even nested documents. Documents are encoded in a standard data exchange format such as XML, JSON (Javascript Option Notation) or BSON (Binary JSON). Document databases are recognized as a powerful, flexible and agile tool to store Big Data.

Document databases allow users to search for data based on the content of documents. A query can be made either by keys, values or examples. The encoded documents contain metadata objects, so it is possible to query data by example [8]. Thus, these type of databases can be used to provide the flexibility that our use case needs. For queries, either a query language or a programming API can be employed.

CouchDB and MongoDB are open source document databases. Data is stored in documents with self-contained records. MongoDB is preferred as it stores data the Memory-Mapped Storage Engine and a socket-based wire protocol with BSON. MongoDB also manages replication using a form of asynchronous master/slave replication called Replica Sets. Furthermore, MongoDB is suitable for dynamic queries and ensures a better performance on Big databases.

IV. PROPOSED SYSTEM AND STORAGE SCHEME

The key idea is to store the meta data during an individual's movements in the form of tables.

A. Mobile Phone Client

After the user starts the application, the mobile phone sends a piece of data every five seconds to the application server. This message includes the current GPS data and meta values (longitude, latitude, speed, distance and so on).

B. Server End

After receiving the message from the client, the application server first calibrates the GPS data in the message and then save the message to the GPS path table in MongoDB. The primary key saves the user ID, the task ID, the timestamp, and the attribute column saves the longitude, the latitude, the time, the distance travelled and so on. The data at each time is stored in one row.

After the user data is sent to the user's app server, the app server can first calibrate the data and then write the data to MongoDB. But sometimes we need to consider that the write process is the critical path, while the cleaning and calibration are a non-critical path, so the write and calibration processes are required to be asynchronous.

C. GPS Path Query

It is possible to query the full path history by using range query in MongoDB for example $range(aggregation). On the smartphone end, a range query is required i.e. to query the GPS data from the earliest time of movement to the current time. On the server end, after receiving the query request from the mobile end, the request will be converted into a range query and path history can be queried. For previous movements also, range query can be used to get the location history.

D. Query of GPS path for tracking location history of infected individual

The GPS path data is stored in the specific user ID only and the officials need to pull the data to read it. Another table, records the shared user ID, task ID and other information.ID and other values designed for the sharing process need to be encrypted to prevent malicious impersonation. Any encryption algorithm such as RSA can be used for this purpose.

On the infected individual's end, user_id and the task_id are sent to the web portal where officials can monitor the location history. After the user_id and task_id of infected individual is known, officials can use the range query to get the location data and path data of the infected individual.
The server stores these data in a table and then send a notification to the tracking web portal including a special ID value (encrypted user_id and task_id).

E. Storage Scheme

The storage scheme involves five steps:

Step 1: The smartphone requests from the application server to write data into MongoDB.
Step 2: After the app server approves the request and agrees to grant write permission to this user, a temporary write permission and authorisation time for MongoDB is given by STS service.
Step 3: After receiving the request, the STS service will generate a temporary token, including the temporary AccessKeyId, the temporary AccessKeySecret, and the temporary token, and return them to the app server.
Step 4: After the app server receives the STS result, it will return it to the app on the smartphone end.
Step 5: After the app on the mobile phone receives the temporary token, it can start to write data to MongoDB. The temporary token will expire after a while.

This type of storage scheme is often termed as Direct Storage Scheme. In addition to reducing the pressure on the application server, the above method can decouple the most critical GPS write process from the app server and make the critical path independent, which is its biggest advantage. As a result, even if the app server suffers a fault and becomes unavailable, the user's GPS write process will not be affected.

The storage scheme is illustrated using the flow diagram fig 1 and the proposed system using fig 2.

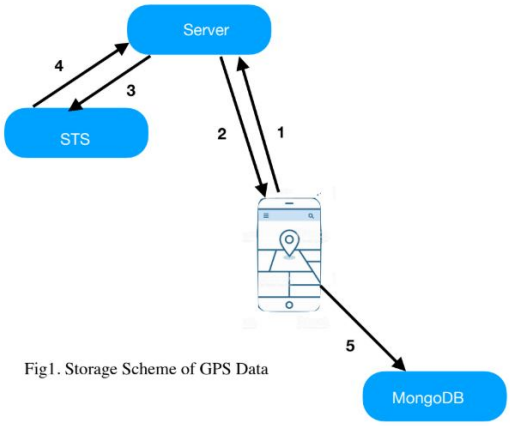

Fig1. Storage Scheme of GPS Data

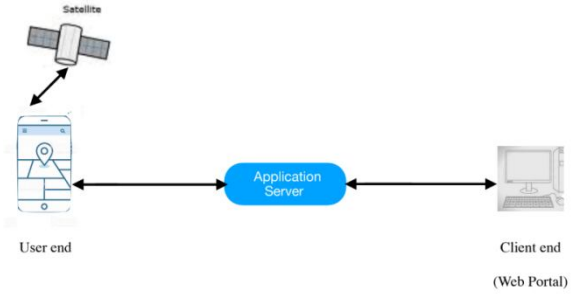

Fig 2 . The proposed System

F. Comparison of Location data of Infected and Potentially Infected Individuals

The final phase after collecting the location data is to compare the location data of the infected individuals with others. If the location data shows that certain individual was in a close proximity of the infected individual, then that individual becomes a suspect and a test is necessary.

For this purpose, the client side can be equipped with Studio3T for MongoDB. The Data Compare and Sync of Studio3T can be used for this purpose. Simply drag the source database on top of the target database, and voilà! All collections with the same name in the source and target databases are automatically selected for comparison [11]. Studio 3T's Data Compare and Sync feature also lets you configure data comparison settings, which can come in handy in case you only want to compare certain fields, filter based on certain criteria, etc [11]. After this an analysis can be run by clicking on the Run Comparison. When the run is completed, a tab will open for each pair of collections and the comparison data can be compared on various aspects.

For a more convenient data analysis and easier sharing across teams, the comparison results can be exported as CSV file. MongoDB documents, fields and data can be synced.

V. FURTHER WORKS

Although most experts believe another influenza pandemic will occur, it is difficult to predict when or where it will appear or how severe it will be.It is currently impossible to predict the emergence of a future pandemic other than to strongly suspect. The data gathered during the COVID19 can be exported as CSV file. Data gives the potential to learn. This data can used as training data sets for a system that can predict future pandemics. The predictive system can be designed using various Machine Learning concepts and training algorithms. Furthermore, to improve the ability to predict influenza pandemics, it is necessary to increase knowledge of the basic biology and ecology underlying host–switching events [13]. The interdisciplinary field of Machine Learning and biology can be beneficial to predict and stop any future pandemics.

Thus, it is essential that while carefully monitoring current and identifiable risks, pandemic prevention strategies must also be based on expecting the unexpected and being capable of reacting accordingly.

VI. CONCLUSION

COVID19 is a highly infectious disease and is mainly transmitted through contact with respiratory droplets rather than through the air. This paper suggested a novel method to track the potential patients using the GPS data by storing it on a cloud based document database (which is a NoSQL database) and then comparing the location data of the infected individuals with others. NoSQL databases provides many advantages to deal with Big Data storage, processing and querying. Also, these databases ensure better sharding and real-time data replication at a lower cost while optimizing system resources. Direct Storage scheme is favoured to store the location data. This type of storage scheme reduces the pressure on the application server and can decouple the most critical GPS write process from the app server making critical path independent.

Furthermore, the paper also gives an insight on use of data collected in the current pandemic and using this data as training set to Machine Learning models to predict a future pandemic. Biology and ML can together help to prevent any such pandemics in the future.

VII. ACKNOWLEDGEMENTS

I express my sincere thanks to Dr Diksha Kumari for her constant support, motivation, enlightenment and for providing her valuable insight and expertise on COVID19, Corona Viruses and pandemics that greatly assisted the research.

REFERENCES

[1] K. Kaur and R. Rani, “Modeling and querying data in NoSQL databases,” in *IEEE International Conference on Big Data*, Oct 2013, pp. 1–7.
[2] R. Cattell, “Scalable sql and nosql data stores,” *ACM SIGMOD Record*, vol. 39, no. 4, pp. 12–27, 2011.
[3] C. Strauch, U.-L. S. Sites, and W. Kriha, “Nosql databases,” *Lecture Notes, Stuttgart Media University*, 2011.
[4] A. B. M. Moniruzzaman and S. A. Hossain, “Nosql database: New era of databases for big data analytics - classification, characteristics and comparison.” *International Journal of Database Theory and Application*, vol. 06, no. 04, 2013.
[5] https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/ accessed on 08/03/2020
[6] https://www.mohfw.gov.in/ accessed on 21/03/2020
[7] https://www.who.int/emergencies/diseases/novel-coronavirus-2019 accessed on 21/03/2020
[8] D. Loshin, Big data analytics: from strategic planning to enterprise integration with tools, techniques, NoSQL, and graph. Elsevier, 2013.
[9] Thuong Le-Tien, Vu Phung, "Routing and Tracking System for Mobile Vehicles in Large Area", The Dept. of Electrical Electronics Engineering, HCM University of Technology, Vietnam.
[10] https://docs.mongodb.com/ accessed on 13/03/2020
[11] https://studio3t.com/knowledge-base/articles/compare-mongodb-collections/ accessed on 19/03/2020
[12] S. Beis, S. Papadopoulos, and Y. Kompatsiaris, “Benchmarking graph databases on the problem of community detection,” in *New Trends in Database and Information Systems II*. Springer, 2015, pp. 3–14.
[13] Taubenberger J.K., Morens D.M., Fauci A.S. The next influenza pandemic: Can it be predicted? JAMA. 2007;297:2025–2027. doi: 10.1001/jama.297.18.2025