COVID-19 FUTURE FORECASTING USING SUPERVISED MACHINE LEARNING MODELS

K. Sudha Devi, A. Gunasundari, N. Kavya Sri, T. Pavithra
Department of Information Technology, Vivekanandha college of Technology for Women,
Tiruchengode, Tamilnadu, India

ABSTRACT

The spread of COVID-19 in the whole world has put the humanity at risk. The resources of some of the largest economies are stressed out due to the large infectivity and transmissibility of this disease. The capability of ML models to forecast the number of upcoming patients affected by COVID-19 which is presently considered as a potential threat to mankind. In particular, four standard forecasting models, least absolute shrinkage and selection operator (LASSO) Support vector Machine (SVM) have been used in this study to forecast the threatening factors of COVID-19.Three types of predictions are made by each of the models, such as the number of newly infected cases, the number of deaths, and the number of recoveries But in the cannot predict the accurate result for the patients.

To overcome the issue, Proposed method using the long short-term Integrated Average (LSTIA) predict the number of COVID-19 cases in next 30 days ahead and effect of preventive measures like social isolation and lockdown on the spread of COVID-19.

INTRODUCTION

OVERVIEW OF COVID-19

COVID-19, the pandemic that is spreading worldwide, has revealed the vulnerability of human society to severe infectious diseases and the difficulty of solving this problem in a globally interconnected complex system. COVID-19 affected more than 100 countries in a span of weeks. As a result, the whole human race should not only collaborate to overcome the epidemic but also reasonably arrange to return to work and production according to the actual situation of each region and carry out geographical risk assessment. Many attempts have been conducted to find a suitable and fast way to detect infected patients in an early stage. After making chest CT scans of 21 patients infected with COVID19 in China, Guan et al found that CT scan analysis included bilateral pulmonary parenchymal ground-glass and consolidative pulmonary opacities, sometimes with a rounded morphology and a peripheral lung distribution. Consequently, COVID-19 diagnosis can be represented as an image segmentation problem to extract the main features of the disease. The disease caused by the novel coronavirus, or Coronavirus Disease 2019 (COVID-19) is quickly spreading globally. It has infected more than 1,436,000 people in more than 200 countries and territories as of April 9, 2020.

Coronavirus disease 2019 (COVID-19) is a contagious respiratory and vascular disease, caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). First identified in Wuhan, China, it is currently an ongoing pandemic. Common symptoms include fever, cough, fatigue, breathing difficulties, and loss of smell and taste Symptoms begin one to fourteen days after exposure to the virus. While most people have mild symptoms, some people develop acute respiratory distress syndrome (ARDS), which can be precipitated by cytokine storms, multi-organ failure, septic shock, and blood clots. Longer-term damage to organs (in particular, the lungs and heart) has been observed, and there is concern about a significant number of patients who have recovered from the acute phase of the disease but continue to experience a range of effects—known as long COVID—for months afterwards, including severe fatigue, memory loss and other cognitive issues, low grade fever, muscle weakness, and breathlessness.

EXPONENTIAL SMOOTHING

Exponential smoothing is a rule of thumb technique for smoothing time series data using the exponential window function. Whereas in the simple moving average the past observations are weighted equally, exponential functions are used to assign exponentially decreasing weights over time. It is an easily learned and easily applied procedure for making some determination based on prior assumptions by the user, such as seasonality. Exponential smoothing is often used for analysis of time-series data.

Exponential smoothing is one of many window functions commonly applied to smooth data in signal processing, acting as low-pass filters to remove high-frequency noise. This method is preceded by Poisson's use of recursive exponential window functions in convolutions from the 19th century, as well as Kolmogorov and Zurbenko's use of recursive moving averages from their studies of turbulence. There is no formally correct procedure for choosing {\display style \alpha}\alpha. Sometimes the statistician's judgment is used to choose an appropriate factor. Alternatively, a statistical technique may be used to optimize the value of {\display style \alpha}\alpha. For example, the method of least squares might be used to determine the value of {\display style \alpha }\alpha for which the sum of the quantities {\display style (s_{t}-x_{t+1})^{2}}{\display style (s_{t}-x_{t+1})^{2}} is minimized.

Unlike some other smoothing methods, such as the simple moving average, this technique does not require any minimum number of observations to be made before it begins to produce results. In practice, however, a "good average" will not be achieved until several samples have been averaged together; for example, a constant signal will take approximately {\display style 3/\alpha} {\display style 3/\alpha } stages to reach 95% of the actual value. To accurately reconstruct the original signal without information loss all stages of the exponential moving average must also be available, because older samples decay in weight exponentially. This is in contrast to a simple moving average, in which some samples can be skipped without as much loss of information due to the constant weighting of samples within the average. If a known number of samples will be missed, one can adjust a weighted average for this as well, by giving equal weight to the new sample and all those to be skipped. This simple form of exponential smoothing is also known as an exponentially weighted moving average (EWMA). Technically it can also be classified as an autoregressive integrated moving average (ARIMA) (0,1,1) model with no constant term

FUTURE FORECASTING

Forecasting is the process of making predictions of the future based on past and present data and most commonly by analysis of trends. A commonplace example might be estimation of some variable of interest at some specified future date. Prediction is a similar, but more general term. Both might refer to formal statistical methods employing time series, cross-sectional or longitudinal data, or alternatively to less formal judgmental methods. Usage can differ between areas of application: for example, in hydrology the terms "forecast" and "forecasting" are sometimes reserved for estimates of values at certain specific future times, while the term "prediction" is used for more general estimates, such as the number of times floods will occur over a long period. Risk and uncertainty are central to forecasting and prediction; it is generally considered good practice to indicate the degree of uncertainty attaching to forecasts. In any case, the data must be up to date in order for the forecast to be as accurate as possible. In some cases the data used to predict the variable of interest is itself forecast.

Qualitative forecasting techniques are subjective, based on the opinion and judgment of consumers and experts; they are appropriate when past data are not available. They are usually applied to intermediate- or long-range decisions. Examples of qualitative forecasting methods are informed opinion and judgment, the Delphi method, market research, and historical life-cycle analogy. Quantitative forecasting models are used to forecast future data as a function of past data. They are appropriate to use when past numerical data is available and when it is reasonable to assume that some of the patterns in the data are expected to continue into the future. These methods are usually applied to short- or intermediate-range decisions. Examples of quantitative forecasting methods are last period demand, simple and weighted N-Period moving averages, simple exponential smoothing, poisson process model based forecasting and multiplicative seasonal indexes. Previous research shows that different methods may lead to different level of forecasting accuracy. For example, GMDH neural network was found to have better forecasting performance than the classical forecasting algorithms such as Single Exponential Smooth, Double Exponential Smooth, and ARIMA and back-propagation neural network.

EXISTING SYSTEM

COVID 19 is currently considered a potential threat to humanity. In four standard prediction models, such as linear regression (left to right), at least complete summary and select operator, Support Vector Machine (SVM), have been used to predict COVID-19 threatening factors in this study. Predictions are made on each of the models, such as the number of new infections, the number of deaths, and the number of recurrences over the next 10 days. For the effects of the study it demonstrates a promising mechanism for the use of these methods in the current context of COVID 19 infection. Predictions are made on each of the models, such as the number of new infections, the number of deaths, and the number of recurrences over the next 10 days. For the effects of the study it demonstrates a promising mechanism for the use of these methods in the current context of COVID 19 infection.

COVID-19 does not seem to affect children severely; many pediatrics wards have been focused more on the emergency of COVID-19-related issues. For this reason, attention on many other acute and chronic diseases, especially those rarer, may be lacking. This scarcity of interest may cause, particularly in childhood, severe problems, or even death.

DRAWBACKS OF EXISTING SYSTEM

- COVID-19 problem cannot predict the exact result from the patients.
- Difficult to Monitor Performance - It is not easy for managers to monitor their staffs' progress and performance without them being in the same office space.
- This is especially escalated if the job role requires a lot of "background duties" that can't be monitored on a work's system.
- Financial burden on the world, Morbidity and mortality Social and mental distance between people.

PROPOSED SYSTEM

Machine learning methods proved to be effective for prediction due to automatically extracting relevant features from the training samples, feeding the activation from the previous time step as input for the current time step and networks self-connections. According to the results of the model analysis, we believe that the emergency intervention measures adopted in the early stage of the epidemic, such as blocking, restricting the flow of people, and increasing the support, had a crucial restraining effect on the original spread of the epidemic.

It is a very effective prevention and treatment method to continue to increase investment in various medical resources to ensure that suspected patients can be diagnosed and treated in a timely manner. The epidemic trends long short-term Integrated Average (LSTIA) of were first fitted and analyzed in order to prove the validity of the existing mathematical models. The results were then used to fit and analyze the situation of COVID-19. The prediction results of three different mathematical models are different for different parameters and in different regions. The prediction obtained by the proposed method of various components (number of positive cases recovered number of cases, etc.) will be accurate within a certain range and will be a beneficial tool for administrators and health officials.

ADVANTAGES OF PROPOSED SYSTEM

- Lifestyle modifications
- Health awareness
- Importance of health
- Advantages of real-time and fast, which can predict the incidence trend of infectious diseases as early as possible, and are suitable for data analysis of a large number of people.
- The sensitivity, spatial resolution and accuracy of its prediction result is improved

SOFTWARE DESCRIPTION

FRONT END: JAVA

The software requirement specification is created at the end of the analysis task. The function and performance allocated to software as part of system engineering are developed by establishing a complete information report as functional representation, a representation of system behavior, an indication of performance requirements and design constraints, appropriate validation criteria.

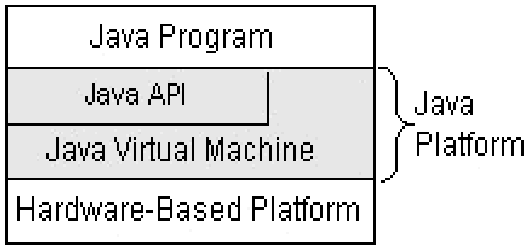
FEATURES OF JAVA

Java platform has two components:

- The *Java Virtual Machine* (Java VM)
- The *Java Application Programming Interface* (Java API)

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries (*packages*) of related components.

The following figure depicts a Java program, such as an application or applet, that's running on the Java platform. As the figure shows, the Java API and Virtual Machine insulates the Java program from hardware dependencies.



As a platform-independent environment, Java can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring Java's performance close to that of native code without threatening portability.

SOCKET OVERVIEW

A network socket is a lot like an electrical socket. Various plugs around the network have a standard way of delivering their payload. Anything that understands the standard protocol can "plug in" to the socket and communicate.

Internet protocol (IP) is a low-level routing protocol that breaks data into small packets and sends them to an address across a network, which does not guarantee to deliver said packets to the destination.

Transmission Control Protocol (TCP) is a higher-level protocol that manages to reliably transmit data. A third protocol, User Datagram Protocol (UDP), sits next to TCP and can be used directly to support fast, connectionless, unreliable transport of packets.

CLIENT/SERVER:

A server is anything that has some resource that can be shared. There are compute servers, which provide computing power; print servers, which manage a collection of printers; disk servers, which provide networked disk space; and web servers, which store web pages. A client is simply any other entity that wants to gain access to a particular server.

A server process is said to "listen" to a port until a client connects to it. A server is allowed to accept multiple clients connected to the same port number, although each session is unique. To manage multiple client connections, a server process must be multithreaded or have some other means of multiplexing the simultaneous I/O.

RESERVED SOCKETS

Once connected, a higher-level protocol ensues, which is dependent on which port user are using. TCP/IP reserves the lower, 1,024 ports for specific protocols. Port number 21 is for FTP, 23 is for Telnet, 25 is for e-mail, 79 is for finger, 80 is for HTTP, 119 is for Netnews-and the list goes on. It is up to each protocol to determine how a client should interact with the port.

JAVA AND THE NET

Java supports TCP/IP both by extending the already established stream I/O interface. Java supports both the TCP and UDP protocol families. TCP is used for reliable stream-based I/O across the network. UDP supports a simpler, hence faster, point-to-point datagram-oriented model.

INETADDRESS

The InetAddress class is used to encapsulate both the numerical IP address and the domain name for that address. User interacts with this class by using the name of an IP host, which is more convenient and understandable than its IP address. The InetAddress class hides the number inside. As of Java 2, version 1.4, InetAddress can handle both IPv4 and IPv6 addresses.

FACTORY METHODS

The InetAddress class has no visible constructors. To create an InetAddress object, user use one of the available factory methods. Factory methods are merely a convention whereby static methods in a class return an instance of that class. This is done in lieu of overloading a constructor with various parameter lists when having unique method names makes the results much clearer.

Three commonly used InetAddress factory methods are:

1. Static InetAddressgetLocalHost ( ) throws UnknownHostException
2. Static InetAddressgetByName (String hostName) throwsUnknowsHostException
3. Static InetAddress [ ] getAllByName (String hostName) throwsUnknownHostException

INSTANCE METHODS

The InetAddress class also has several other methods, which can be used on the objects returned by the methods just discussed. Here are some of the most commonly used.

Boolean equals (Object other)- Returns true if this object has the same Internet address as other.

1. byte [ ] get Address ( )-Returns a byte array that represents the object’s Internet address in network byte order.
2. String getHostAddress ( ) - Returns a string that represents the host address associated with the InetAddress object.
3. String get Hostname ( ) - Returns a string that represents the host name associated with the InetAddress object.
4. booleanisMulticastAddress ( )- Returns true if this Internet address is a multicast address. Otherwise, it returns false.

5. String toString ( ) - Returns a string that lists the host name and the IP address for convenience.

TCP/IP CLIENT SOCKETS

TCP/IP sockets are used to implement reliable, bidirectional, persistent, point-to-point and stream-based connections between hosts on the Internet. A socket can be used to connect Java's I/O system to other programs that may reside either on the local machine or on any other machine on the Internet.

There are two kinds of TCP sockets in Java. One is for servers, and the other is for clients. The Server Socket class is designed to be a "listener," which waits for clients to connect before doing anything. The Socket class is designed to connect to server sockets and initiate protocol exchanges.

The creation of a Socket object implicitly establishes a connection between the client and server. There are no methods or constructors that explicitly expose the details of establishing that connection. Here are two constructors used to create client sockets

Socket (String hostName, intport) - Creates a socket connecting the local host to the named host and port; can throw an UnknownHostException or anIOException.

Socket (InetAddressipAddress, intport) - Creates a socket using a preexistingInetAddressobject and a port; can throw an IOException.

A socket can be examined at any time for the address and port information associated with it, by use of the following methods:

- InetAddressgetInetAddress ( ) - Returns the InetAddress associated with the Socket object.
- IntgetPort ( ) - Returns the remote port to which this Socket object is connected.
- IntgetLocalPort ( ) - Returns the local port to which this Socket object is connected.

Once the Socket object has been created, it can also be examined to gain access to the input and output streams associated with it. Each of these methods can throw an IO Exception if the sockets have been invalidated by a loss of connection on the Net.

Input Streamget Input Stream ( ) - Returns the InputStream associated with the invoking socket.

Output Streamget Output Stream ( ) - Returns the OutputStream associated with the invoking socket.

TCP/IP SERVER SOCKETS

Java has a different socket class that must be used for creating server applications. The

ServerSocket class is used to create servers that listen for either local or remote client programs to connect to them on published ports. ServerSockets are quite different form normal Sockets.

When the user create a ServerSocket, it will register itself with the system as having an interest in client connections.

- ServerSocket(int port) - Creates server socket on the specified port with a queue length of 50.
- Serversocket(int port, int maxQueue) - Creates a server socket on the specified portwith a maximum queue length of maxQueue.
- ServerSocket(int port, int maxQueue, InetAddress localAddress)-Creates a server socket on the specified port with a maximum queue length of maxQueue. On a multihomed host, localAddress specifies the IP address to which this socket binds.
- ServerSocket has a method called accept( ) - which is a blocking call that will wait for a client to initiate communications, and then return with a normal Socket that is then used for communication with the client.

URL

The Web is a loose collection of higher-level protocols and file formats, all unified in a web browser. One of the most important aspects of the Web is that Tim Berners-Lee devised a saleable way to locate all of the resources of the Net. The Uniform Resource Locator (URL) is used to name anything and everything reliably.

The URL provides a reasonably intelligible form to uniquely identify or address information on the Internet. URLs are ubiquitous; every browser uses them to identify information on the Web.

PROJECT DESCRIPTION

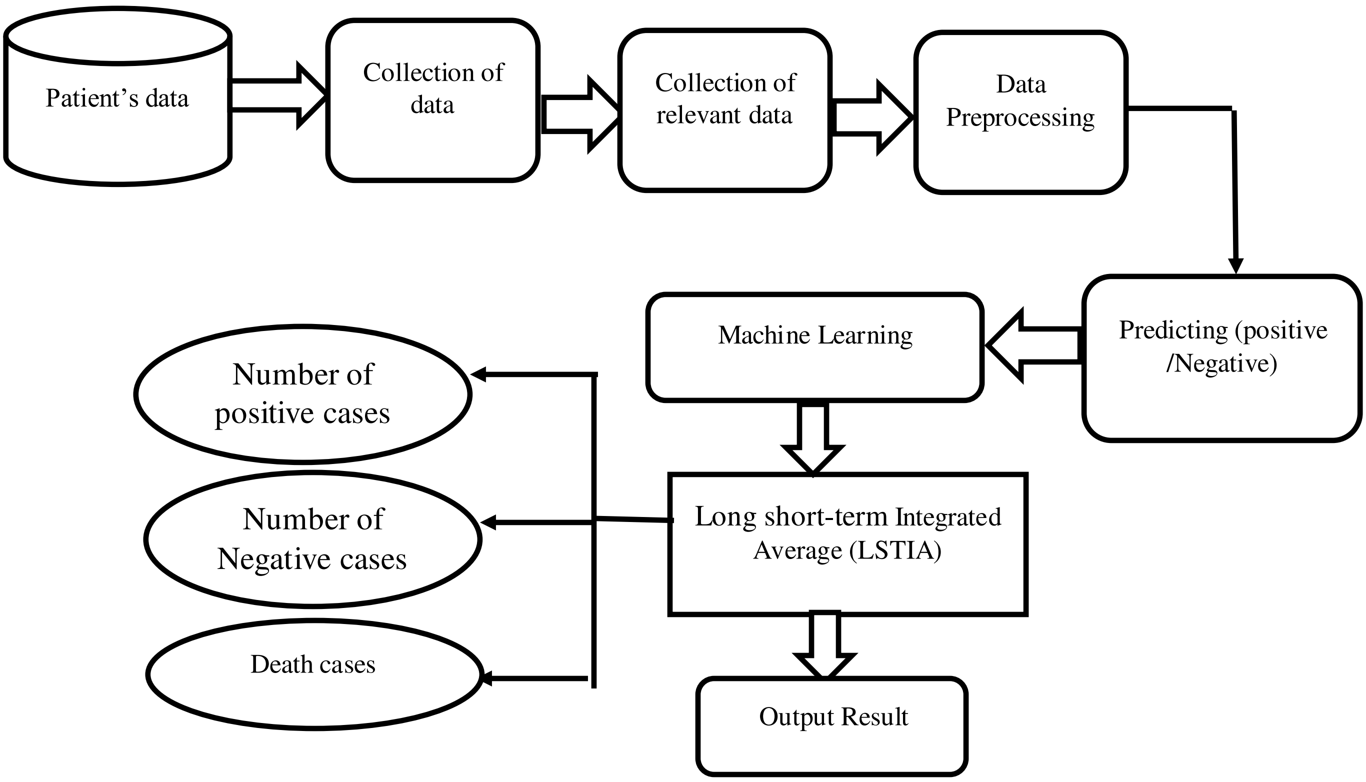
PROBLEM DEFINITION

COVID-19 mainly spreads through the air when people are near each other long enough,[a] primarily via small droplets or aerosols, as an infected person breathes, coughs, sneezes, sings, or speaks. Transmission via fomites (contaminated surfaces) has not been conclusively demonstrated. It can spread as early as two days before infected persons show symptoms (presymptomatic), and from asymptomatic (no symptoms) individuals. People remain infectious for up to ten days in moderate cases, and two weeks in severe cases. The standard diagnosis method is by real-time reverse transcription polymerase chain reaction (rRT-PCR) from a nasopharyngeal swab. Preventive measures include social distancing, quarantining, ventilation of indoor spaces, covering coughs and sneezes, hand washing, and keeping unwashed hands away from the face. The use of face masks or coverings has been recommended in public settings to minimize the risk of transmissions. There are no proven vaccines or specific treatments for COVID-19 yet, though several are in development. Management involves the treatment of symptoms, supportive care, isolation, and experimental measures.

OVERVIEW OF THE PROJECT

Due to the growing magnitude of number of cases and its subsequent stress on the administration and health professionals, some prediction methods would be required to predict the number of cases in future. The prediction of various parameters (number of positive cases, number of recovered cases, etc.) obtained by the proposed method is accurate within a certain range and will be a beneficial tool for administrators and health officials.A better understanding of such opposition ahead of a COVID-19 vaccine is therefore critical for scientists, public health practitioners, and governments. Advantages of real-time and fast, which can predict the incidence trend of infectious diseases as early as possible, and are suitable for data analysis of a large number of people.

SYSTEM FLOW DIAGRAM



CONCLUSION

A data-driven forecasting/estimation method has been used to estimate the possible number of positive cases of COVID-19 in India for the next 30 days. The number of recovered cases, long short-term Integrated Average (LSTIA) daily positive cases, and deceased cases has also been estimated by using and curve fitting. The effect of preventing measures as social isolation and lockdown has also been observed which shows that by these preventive measures, the spread of the virus can be reduced significantly. Although this method often requires sufficient data to support it, in the early stages of epidemic transmission, this method can still be used to more accurately predict the indicators of epidemic transmission in the short term, so as to provide intervention control at all levels of the departments and policy implementation provides short-term emergency prevention programs. The prediction results of three different mathematical models are different for different parameters and in different regions. In general, the fitting effect of Logistic model may be the best among the three models.

REFERENCES

[1] COVID, IHME, Forecasting Team, and Simon I. Hay. "COVID-19 scenarios for the United States." medRxiv (2020).

[2] Donders, Francesca, Risa Lonnée-Hoffmann, Aristotelis Tsiakalos, Werner Mendling, José Martinez de Oliveira, Philippe Judlin, Fengxia Xue, Gilbert GG Donders, ISIDOG COVID, and Guideline Workgroup. "ISIDOG recommendations concerning COVID-19 and pregnancy." Diagnostics 10, no. 4 (2020): 243.

[3] Covid, C. D. C., Response Team, CDC COVID, Response Team, CDC COVID, Response Team, Sherry L. Burrer et al. "Characteristics of health care personnel with COVID-19—United States, February 12–April 9, 2020." Morbidity and Mortality Weekly Report 69, no. 15 (2020): 477.

[4] Covid, C. D. C., Response Team, CDC COVID, Response Team, CDC COVID, Response Team, Stephanie Bialek et al. "Geographic differences in COVID-19 cases, deaths, and incidence—United States, February 12–April 7, 2020." Morbidity and Mortality Weekly Report 69, no. 15 (2020): 465.

[5] Fauci, Anthony S., H. Clifford Lane, and Robert R. Redfield. "Covid-19—navigating the uncharted." (2020): 1268-1269.