UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

FORECASTING THE COVID-19 PANDEMIC IN THE UNITED STATES AND PERU USING ARIMA, LSTM, GRU, CNN, AND A HYBRID APPROACH

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE

By

MARY HOOVER
Norman, Oklahoma
2023

FORECASTING THE COVID-19 PANDEMIC IN THE UNITED STATES AND PERU USING ARIMA, LSTM, GRU, CNN, AND A HYBRID APPROACH

A THESIS APPROVED FOR THE
GALLOGLY COLLEGE OF ENGINEERING

BY THE COMMITTEE CONSISTING OF

Dr. Charles Nicholson, Chair

Dr. Talayeh Razzaghi

Dr. Andrés D. González

ACKNOWLEDGEMENTS

I would like to express my gratitude to my committee chair for his guidance and support throughout this process. I sincerely appreciate my committee members for their invaluable feedback and willingness to help. It was a privilege to work with them. Without their support, this work would not have been possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abstract

The COVID-19 outbreak spread swiftly and infected many individuals resulting in overwhelmed and overfilled hospitals causing an immense loss of life globally. Identifying the number of infected individuals preemptively provides critical time for governmental and health officials to implement a strategy to respond to the pandemic such as requiring masking, reducing public gatherings, closing restaurants, as well as additional time to prepare hospitals and medical staff for surges in infections. The work explores implementing convolutional neural network models (CNN), long short-term network models (LSTM), gated recurrent unit models (GRU), the combination of encoding CNN layers and decoding LSTM and/or GRU layers in a hybrid model, and Auto-Regressive Integrated Moving Average (ARIMA) models to predict COVID-19 case count in the United States and Peru for 7, 15 or 30 days in the future using 30 days of case counts. The study evaluates predictions from January 23, 2020 through March 9, 2023 for the United States and March 6, 2020 through April 2, 2023 for Peru. For each model, the forecasting results are displayed visually and presented statistically using RMSE and MAPE. The hybrid model performed as well as or better than any other model when predicting 7 days, 15 days, or 30 days into the future. These results demonstrate models that potentially assist healthcare providers and policymakers' response to the spread of COVID-19.

Chapter 1: Introduction

The COVID-19 disease was first recorded in late December 2019 in Wuhan China before spreading around the globe. The respiratory illness is caused by the SARS-CoV-2 virus and has brought widespread healthcare challenges to much of the world. The disease is mild to moderate in many of those infected, but for some it is serious and can be deadly. Those at higher risk are the elderly, those with cardiovascular disease, diabetes, respiratory diseases, or cancer [1]. The disease is particularly challenging as it is highly infectious and easily transmitted through human interaction. The virus spreads from an infected person's mouth or nose through liquid from the size of aerosols to droplets [1]. The high rate of infection has resulted in an unprecedented demand on the healthcare system resulting in overwhelmed staff, space issues, and lack of supplies in many cases.

As of March 10 2023, in the United States there have been 103 million confirmed cases and 1.1 million deaths and in Peru there have been 4.4 million confirmed cases and 219,000 deaths [2]. Therefore, the deaths per confirmed case in Peru are more than 4.5 times than that of the United States. Overall, Peru had the highest mortality rate of any country in the world due to the virus[3]. Both countries had an immense loss of life with hospitals quickly being overcome as the disease progressed shockingly quickly. Peru's healthcare system was impacted particularly hard as it was already near its limits before the COVID-19 pandemic occurred [4]. Additionally, there was lower investment in healthcare in the country resulting in less new facilities being built in the years before the pandemic which resulted in greater impact to areas of higher population growth disproportionately [4]. As cases increased, both governments implemented social distancing mandates, travel bans, and other interventions in an effort to minimize the spread of the disease resulting in variability of the rate of transmission of the disease. Additionally, the behavior of the disease changed as new variants appeared throughout both countries and some individuals achieved natural immunity for some time after infection. These variables affected the rate of spread of the disease resulting in dynamic behavior over the course of the pandemic in both countries.

The spread of a disease exhibits specific patterns that can be identified and predicted using time series modeling. Modeling and forecasting the spread of the pandemic can help provide critical time for governments and health officials to strategize and implement a response to the disease. Implementing predictive analysis results in additional response time allowing for governmental interventions to slow the spread of the disease and additional time for medical professionals to prepare for surges in hospitalizations. If a large enough daily case count is predicted, hospitals can adjust staffing and equipment accordingly or request additional assistance from other areas. Additionally, an emergency response can occur such as requiring masking, reducing public gatherings, closing restaurants, and other restrictions that reduce the rate of spread of the disease [5].

The goal of the research is to explore models that can accurately forecast the daily case counts of COVID-19 in the United States and Peru. Prediction of the pandemic is challenging due to the dynamic behavior of the disease and varying accuracy of the recorded case counts in each country. The study leverages time series modeling with the deep learning feature identification of convolutional neural network models (CNN), deep temporal learning of long short-term network models (LSTM) and gated recurrent unit models (GRU), the combination of encoding CNN layers and decoding LSTM and/or GRU layers in a hybrid model, and the statistical predictive capabilities of the Auto-Regressive Integrated Moving Average (ARIMA) model. The CNN, LSTM, GRU, hybrid, and ARIMA models are evaluated to gauge their performance at predicting the daily case counts 7, 15, and 30 days in the future utilizing 30 day input sequences for both the United States and Peru pandemic. For each model, the forecasting results are visualized, and the statistical results are presented for the performance of the models.

The following sections are organized such that Chapter 2 provides descriptions of the different modeling approaches. Chapter 3 reviews other work related to forecasting the pandemic. Chapter 4 explores the behavior of the data as well as limitations in data collection accuracy. It also contains the methodology behind each model’s setup and application. Chapter 5 presents the empirical results and analysis. The conclusion occurs in Chapter 6 with a discussion of the findings and opportunities for additional work.

Chapter 2: Background

The techniques utilized in this work to model the spread of the COVID-19 pandemic require the use of time series data. The number of positive COVID-19 tests were recorded daily throughout the pandemic. The daily case count is related to the previous days case counts as individuals spread the disease to others in their vicinity with an incubation period of two to fourteen days [6]. Time series data differs from cross sectional data because it focuses on how data moves over time instead of variables at a specific point in time. Because of this difference, time series data can have additional characteristics such as trend, seasonality, cyclical variability, and other irregularities. These characteristics introduce challenges in forecasting time series data.

CNN models specialize in identifying pattern features in data. The convolutional layers accomplish pattern recognition using kernels. In one dimensional CNNs, the kernel is slid across the time series data, and the result of the convolution is the product of the kernel and the signal. Additionally, each convolutional layer has filters that control the number of outputs that occur after convolution. The output of the convolutional layer is the intermediate results added together with the learned bias. A nonlinear activation function such as the rectified linear unit (ReLU) function can be applied to the results if it improves the pattern recognition. ReLU functions convert any negative values to 0 and return positive or zero values only such as in Equation 1.

$$ReLu(x) = \max(0, x) \tag{1}$$

Pooling layers are generally applied after convolution to decrease the dimensions of the matrix. These layers speed up the computation process for subsequent layers. For max pooling layers, a pool size is determined such that for each pool, only the largest value is kept as the output value. Next, a flattening layer is applied to force the output into a one-dimensional matrix. After flattening, dense networks can be utilized to connect all layers of the feature map. These dense networks can have an activation function such as the ReLU function to add nonlinearity to the result of the model. The final dense network's output is the predicted sequence. Figure 1 presents a possible structure for a convolutional neural network model.

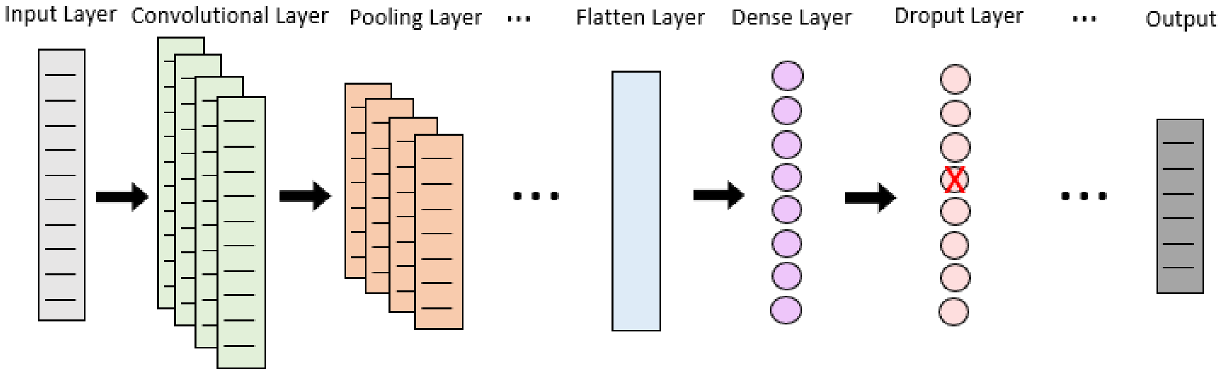

Figure 1: Example of a Convolutional Neural Network

LSTM networks are a type of recurrent neural network that are good at analyzing time series data due to the fact that they incorporate feedback connections as well as feedforward connections. Each unit has three incoming vectors which are the memory, input, and the hidden state vectors. The information from the three incoming vectors is selected based on the three gates internal to an LSTM unit. The structure of an LSTM unit contains a forget gate, input gate, and output gate as shown in Figure 2. LSTM networks utilize these units to learn long term dependencies within the data.

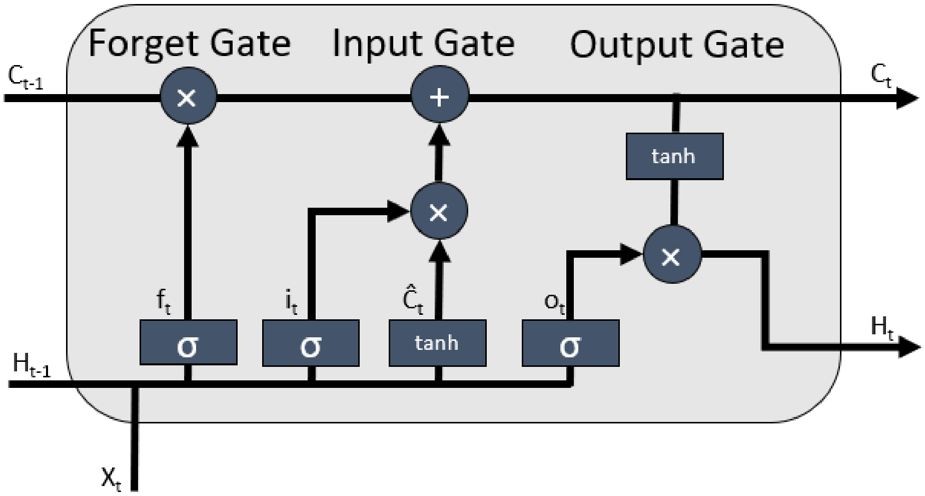

Figure 2: Structure of a LSTM cell

LSTM cells pass data from one timestamp to the next depending on its significance. The mechanism that determines if the data is passed in is the forget gate. The value of the forget gate f_t is determined based on the input at time t x_t and the previous output hidden vector H_{t-1} being fed into the gate and then multiplied with weight matrices W_f and U_f for the gate with an added bias factor B_f. A sigmoid activation function (see Equation 2) is applied to the results (see Equation 3). The sigmoid function is a nonlinear function that transforms the inputs into values between 0 and 1, where the larger inputs are assigned values closer to 1 and smaller valued inputs are assigned values near 0 as in Figure 3.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2}$$

$$f_t = \sigma\left(W_f x_t + U_f H_{t-1} + B_f\right) \tag{3 }$$

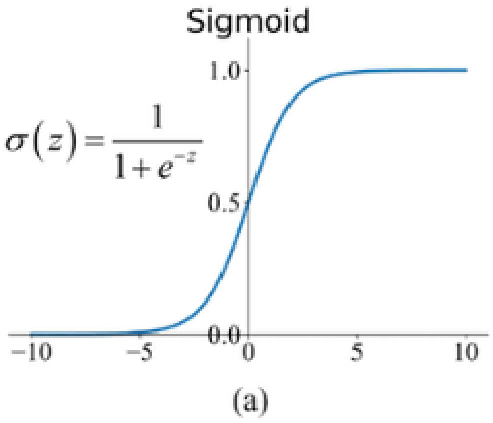

Figure 3: Sigmoid Function [7]

Next, new information is added to the network by an input gate i_t. The input vector is calculated as the input at time t x_t and the previous hidden output H_{t-1} being fed into the gate and then multiplied with weight matrices W_t and U_t with an added bias factor B_i which is then passed through a sigmoid function as in Equation 4. A memory cell candidate vector $\hat{c}_t$ is created by applying the hyperbolic tangent (tanh) function to the input at time t x_t multiplied with weight matrices W_c and the previous output H_{t-1} times the weight matrix U_c plus a bias factor B_c as in Equation 5. The tanh function maps data as a hyperbolic tangent between -1 and 1 such as in Equation 6. An application of the function is displayed in Figure 4.

$$i_t = \sigma(W_i x_t + U_i H_{t-1} + B_i) \tag{4}$$

$$\hat{c}_t = tanh(W_c x_t + U_c H_{t-1} + B_c) \tag{5}$$

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{6}$$

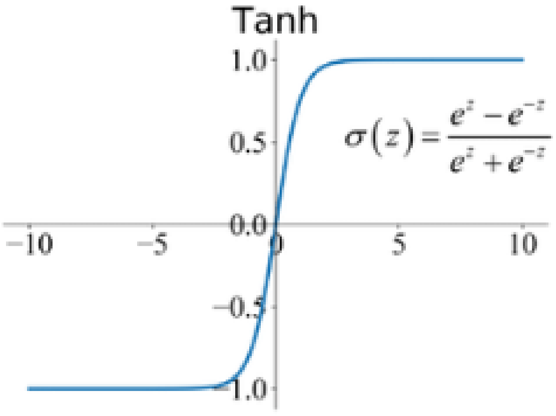

Figure 4: Hyperbolic Tangent Function [7]

The final gate is the output gate o_t which determines the hidden state vector and the final output if it is the last unit in the network. A selector vector is created using a sigmoid function on the input of time t x_t multiplied with the weight matrix W_o and the previous hidden output H_{t-1} fed into the gate times the weight matrix U_o with an added bias B_o as in Equation 7. The memory cell internal state vector C_t is calculated by multiplying the input gate result i_t with the candidate vector $\hat{c}_t$ plus the forget gate result f_t multiplied with the candidate vector previous memory cell state vector C_{t-1} as in Equation 8. The final hidden state vector is the result of the tanh function applied to the memory vector C_t multiplied with the output gate result o_t as shown in Equation 9. The final output is the hidden vector H_t.

$$o_t = \sigma(W_o x_t + U_o H_{t-1} + B_o) \tag{7}$$

$$C_t = (i_t \circ \hat{c}_t + f_t \circ C_{t-1}) \tag{8}$$

$$H_t = \tanh(C_t) \circ o_t \tag{9}$$

An LSTM network will have a sequence fed in by an input layer. Next, some number of LSTM layers will occur. Each LSTM layer can have multiple LSTM cells in it. A nonlinear activation function such as the ReLU, Tanh, or Sigmoid function can be used on the output of these layers. Afterwards, a dropout layer can occur to prevent overtraining of the network. Finally, there is a dense output layer that produces the sequence at the end of the network. An example of one such network is in Figure 5.

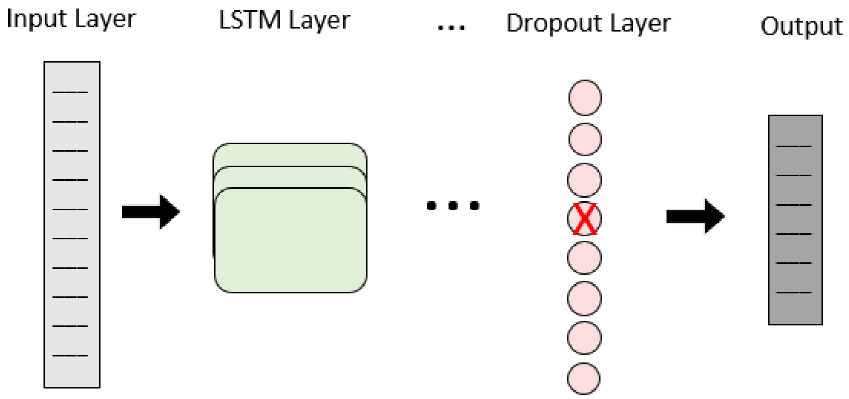

Figure 5: Example of a LSTM Neural Network

GRU networks are a type of neural network that are similar to LSTM networks. Both networks use gating to update the flow of information, but GRU networks are simpler. GRU networks do not have a separate memory vector and have two gates instead of three. They utilize the hidden layer only without needing a memory vector. The structure of a GRU unit contains a reset gate and update gate as shown in Figure 6. GRU networks are faster to use and less prone to overfitting than LSTM networks but can be less accurate.

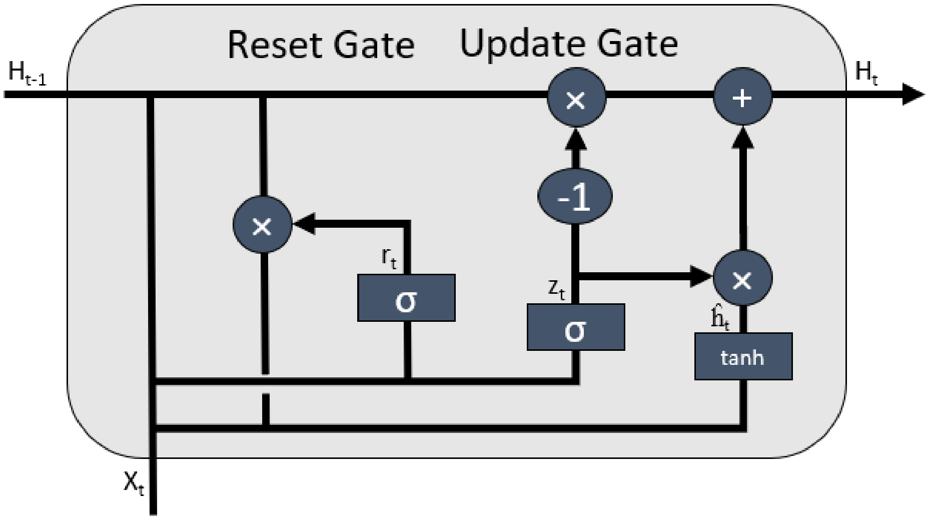

Figure 6: Structure of a GRU cell

GRU cells regulate how much information is passed from the previous cell using the reset gate. The value of the reset gate r_t is calculated using the value at time t x_t and the previous output hidden vector H_{t-1} which is fed into the gate and then multiplied with weight matrices W_r and U_r for the gate with an added bias factor B_r. The reset gate utilizes a sigmoid function to scale the output as shown in Equation 10. The reset gate controls the short-term memory of the network.

$$r_t = \sigma(W_r x_t + U_r H_{t-1} + B_r) \tag{10}$$

The update gate controls the long-term memory of the network. This gate allows the cell to determine how much of the previous information is utilized and how much is updated. The value of the update gate z_t utilizes the value at time t x_t and the previous output hidden vector H_{t-1} that was fed into the gate and then multiplied by weight matrices W_z and U_z for the gate with an added bias factor B_z. The reset gate utilizes a sigmoid function to scale the data between 0 and 1 as shown in Equation 11.

$$z_t = \sigma(W_z x_t + U_z H_{t-1} + B_z) \tag{11}$$

The candidate hidden state vector is the result of the tanh function applied to the quantity of the time at value t x_t times the weight parameter W_x plus the elementwise multiplication of the reset gate output r_t and the previous hidden state H_{t-1} times a weight parameter U_z plus a bias factor B_h as in Equation 12. The final hidden state vector is the update gate result u_t multiplied with the previous hidden state vector H_{t-1} plus one minus the update gate result u_t times the candidate hidden vector $\hat{h}_t$ as shown in Equation 13.

$$\hat{h}_t = tanh(W_x x_t + U_z(r_t \circ H_{t-1}) + B_z) \tag{12}$$

$$H_t = u_t \circ H_{t-1} + (1 - u_t) \circ \hat{h}_t \tag{13}$$

The GRU network is setup in a similar manner to the LSTM model. There is an input layer followed by some number of GRU layers. For the GRU layers, there will be a number of GRU cells. After the GRU layers, an activation function may appear such as ReLU, Tanh, or Sigmoid functions. A dropout layer may occur to prevent overtraining. Lastly, there is a dense output layer that creates the prediction sequence at the end of the network. An example of a GRU network is in Figure 7.

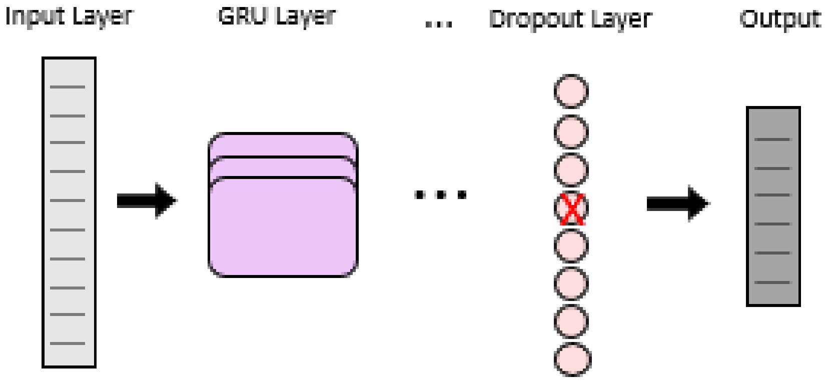

Figure 7: Example of a GRU Neural Network

Arima models are another common way to forecast time series data using previous time steps. The model is created using three regression type expressions. The first expression incorporates autoregression with a lag order parameter p. The autoregressive portion of the model incorporates the previous values with a linear function. An autoregressive model for p lag terms is given in Equation 14 with noise ε_t and a parameter φ that is multiplied with the lagged variable y_{t-p}. In the moving average model, the time series is regressed with an order of past observations denoted as parameter q. The model utilizes a regression with previous errors ε_{t-q} multiplied with a parameter θ plus a noise parameter ε_t as in Equation 15.

$$y_t = \varepsilon_t + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \dots + \varphi_p y_{t-p} \tag{14}$$

$$y_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_p \varepsilon_{t-q} \tag{15}$$

Finally, differencing is incorporated into the model using parameter d which stabilizes the model by differencing the consecutive time series data if necessary. A general ARIMA model incorporates autocorrelation, differencing, and moving average as in Equation 16. The general ARIMA equation can be simplified utilizing a backshift operator. The backshift operator indicates the shifting of the time series back by some number of days as in Equation 17 with differencing of y_t indicated as y_t'. The simplified ARIMA model utilizes the backshift as in Equation 18.

$$y_t' = \varepsilon_t + \varphi y_{t-1}' + \ldots + \varphi_p y_{t-p}{}' + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} \tag{16}$$

$$\mathrm{B} y_t = y_{t-1} \tag{17}$$

$$\left(1 - \varphi_1 B - \cdots - \varphi_p\, B^p\ \right)(1 - B)^d y_t = (1 + \theta_1 B + \cdots + \theta_q B^q)\varepsilon_t \tag{18}$$

Arima models can be extended to model seasonal datasets as well. The models utilize the original ARIMA parameters as well as additional seasonal parameters. Seasonal specific variables are for autocorrelation P with parameter ϕ, differencing D, moving average $Q$with parameter Θ, and a length of seasonality (m). The seasonal portion of the model is made similarly to the nonseasonal model, but it incorporates back shifting based on the set seasonal period. The generalized Seasonal ARIMA model is presented in Equation 19.

$$\left(1 - \varphi_1 B - \cdots - \varphi_p\, B^p\ \right)(1 - \phi_1 B^m - \cdots - \phi_P\, B^{P*m}\ )(1 - B^m)^D (1 - B)^d y_t$$

$$= (1 + \theta_1 B + \cdots + \theta_q B^q)(1 + \Theta_1 B^m + \cdots + \Theta_Q B^{Q*m})\varepsilon_t \tag{19}$$

Chapter 3: Related Work

Due to the devastating effects and global impact of the pandemic, extensive research has occurred around COVID-19 worldwide. The research spans a variety of areas such as diagnosis with radiology imaging, disease tracking, predicting patient health outcomes, computational biology, prediction of protein structures, drug discovery, and internet social control [8]. This section will concentrate on research with the goal of forecasting the COVID-19 spread in the United States and Peru. Disease forecasting research has spanned over a variety of techniques such as compartmental, statistical, and deep learning modeling.

Compartmental modeling is a common approach in epidemiology that separates the entire population into distinct compartments. The simplest model uses the compartments of *susceptible* (the subset of the population that is not currently infected, but is not immune to the disease), *infected* (the subset of the population that is currently infected), and *recovered* (the subset of the population that has recovered from the infection and is now immune from reinfection. This model paradigm is commonly referred to as a SIR model. Compartmental models are often expanded to include additional compartments, e.g., the SEIR model includes an *exposed* compartment for individuals who have been exposed to the disease, but who have not yet become infected. Other models may include a *death* compartment which includes individuals who do not recover from the infection, e.g. a SIRD model reflects the possibility that an individual recover from infection (I → R) or the disease is fatal (I → D). Additional complexity can be introduced by allowing the population to move from the *recovered* compartment back to the *susceptible* compartment (e.g., SIRS models) to model the potential of temporary immunity after infection.

Compartmental models have been used successfully for a variety of diseases such as measles, dengue fever, influenza, HIV, SARS, H1N1, and Ebola [**9**]. The equations involved in compartmental models are relatively simple – they relate to the rate at which members of the population move from one compartment to the next. The equations require parameter values to reflect these various rates. That said, one important limitation is that compartmental models do not include other potentially informative parameters relating to individual characteristics or social influence. While they are explainable model that incorporate biological realism in their parameters, they become less practical when modeling disease spread impacted by extremely dynamic social and public health responses such as those that occurred during the COVID-19 pandemic.

SIR models focus on the number of susceptible individuals, infected individuals, and recovered individuals with transmission rate and rate of removal variables. Vega et al. [10] incorporate machine learning to dynamically set the parameters of a SIR model to forecast the new COVID-19 infections in the US and Canada. The parameters were varied over time to capture the changes in trend due to public health responses. The study utilized data from January 1, 2020 to July 25, 2020 to predict up to 4 weeks into the future. Kreck and Scholz [11] use a SIR model and data from March 7, 2020 through September 12, 2020 in Peru to predict recovery of individuals 52 weeks into the future through September 11, 2021. They found that 88.5 people will recover out of every 100 and that the peaks take on average 12 weeks. Jiménez and Merma [12] utilize a modified SIR model with the addition of a quarantined population and isolated population variable to simulate the effects of vaccinations and quarantine in Peru. They found that a variable isolation and quarantine rate could reduce the deaths from 280 thousand deaths to 200 thousand deaths compared to relaxed restrictions.

SEIR models are based on the number of susceptible individuals, asymptomatic individuals, infected individuals, and recovered individuals with an infection rate, protection rate, the inverse of average latent time, the inverse of average quarantine time, coefficients for cure rate, and coefficients for time dependent mortality rate. Al-Raeei et al. [13] apply the Runge-Kutta method with a SEIR model to forecast new cases of the disease with data available up to December 29, 2020. The study spans the United States, Russia, the United Kingdom, France, Brazil, and India. They predicted the infection peak to occur in March or April 2021 in the United States. Reiner and Barber [14] incorporate both the case and mortality data from February 1, 2020 to July 21, 2020 to determine the disease trajectory and the effects of non-pharmaceutical intervention. They determine that 95% mask usage would ameliorate the worst effects of the epidemic in the United States. Unterbrink et al. [15] utilized SEIR models in Peru to predict data from August 2002 to January 2023 with good results when predicted case counts changed slowly, but over estimation after spikes in data.

SIRD modeling utilizes the number of susceptible individuals, infected individuals, and removed individuals. Removed individuals are those that have recovered or passed away. The model incorporates additional parameters such as coefficient of transmission, rate of recovery and rate of deaths. Al-Raeei [16] applied the SIRD model using COVID-19 data up to March 30, 2020 in China, the United States, Russia, and the Syrian Arab Republic to forecast the number of infected cases, recovered cases, and deceased cased. They found the coefficient of infection, recovery, and mortality for each of the countries in the studied and applied the coefficient to the Syrian Arab Republic to forecast the variables for all of 2020. Mishra et al. [17] developed a SIR model to evaluate the significance of key parameters to the COVID-19 model. These parameters were political action, socioeconomic risk factors, and health factors where the study parameters related to prevention were the most important.

Overall, the compartmental models were predominantly used to determine generalizations about infection peak, proportion recovered, isolation impact, quarantine impact, and the effect of non-pharmaceutical intervention rather than prediction of infections on specific dates. The studies that did forecast case count some number of weeks into the future did so utilizing smaller ranges of data that would have limited the social and public health interventions that occurred during the larger time span incorporated into this study. The compartmental forecasting models tended to respond slowly to quick changes in the data which resulted in inaccuracies when spikes in cases occurred.

Statistical modeling such as ARIMA and SARIMA models was explored by some researchers. ARIMA based models have been utilized to forecast diseases in the past such as hepatitis B [18], measles [19], dengue fever [20], hemorrhagic fever [21], and hand, foot, and mouth disease [22]. One such study focused on Peru and utilized an ARIMA model with data from March 6, 2020 to June 11, 2020 to predict the next 30 days of data with the MAPE of the forecast being 7.8% [23]. They found that while the ARIMA model did not exactly forecast the observed case count, all of the observed cases were within the 95% confidence interval of the forecast. Singh et al. [24] predicted confirmed cases, deaths, and recoveries for the top 15 affected countries around the globe utilizing ARIMA models. The study used data up through April 24, 2020 and predicted the variables from April 25, 2020 to July 7, 2020. They found a fast spread of the disease in the United States, the United Kingdom, Turkey, China, and Russia; however, the ARIMA model results suffered as it struggled to incorporate volatility or turning points in the predictions. Abolmaali and Shirzaei [25] compared ARIMA and SARIMA models for four states (Alabama, Washington, California, and Massachusetts) to forecast the number of COVID-19 cases over 90 days. The ARIMA models outperformed the SARIMA models in all cases, but both models performed less well than the Holt-Winters Double Exponential Smoothing Additive model. The research is limited by the span of dates in the dataset as it only covers 16 months, and the performance of the ARIMA and SARIMA models are heavily influenced by the manual selection of parameters incorporated into the models.

The ARIMA and SARIMA studies for both the United States and Peru suffered from limited data with a maximum of 16 months of data in a study. Each of the model's parameters were selected by manual iteration and comparison to AIC resulting in variation in selected parameters from one study to another. The selected parameters to create the optimum model depend highly upon the behavior of the data in the dataset, so it is unlikely that the models built on data from 2020 would have the same behavior as the data spanning 2020-2023 due to the dynamic nature of the pandemic.

Deep learning techniques incorporating LSTM, GRU, and CNN models encompass a number of studies. These models have been applied for time series modeling of other diseases such as hepatitis [26] and influenza [27]. Xia et al. [28] explore LSTM to model the pandemic in Russia, Peru, and Iran. The study uses data from January 22, 2020 to July 7, 2020 under the assumption that the COVID-19 policy did not change in that period. They predict Peru's case count for July 8-11 with -2.57%, 5.48%, -12.81%, and -2.5% error. The study assumes that people cooperated with the public measures in place. ArunKumar et al. [29] compare GRU and LSTM models to predict cumulative and recovered cases in the United States, Brazil, India, Russia, South Africa, Mexico, Peru, Chile, the United Kingdom, and Iran. They forecasted 60 days into the future with training dates up to October 1, 2020. The LSTM outperformed the GRU model for both Peru and the United States. ArunKumar et al. [30] utilize GRU, LSTM, ARIMA, and SARIMA models to forecast COVID-19 in the USA, Brazil, India, Russia, South Africa, Mexico, Peru, Chile, the United Kingdom, and Iran. When predicting confirmed cases, the SARIMA model performed best for Peru, and the GRU model performed best for the United States. Aguilar and Ibáñez-Reluz [31] utilize 10 features for a multivariate CNN model over data from March 6, 2020 through February 21, 2021. The model predicted the next 15 days of cases with an average Root Mean Squared Log Error (RMSLE) of 0.471 for the Peruvian coast.

The limitations in the deep learning studies were based around the minimal data incorporated into each model. Deep learning techniques like GRU, LSTM, and CNN require a lot of data to accurately select model architectures, determine hyperparameters, and train the models. In addition to lacking in data to train and select model parameters, the model's datasets did not have the range of social and public health interventions that occurred during the larger time span incorporated into this study.

The novelties of the study will be to apply and compare the ARIMA, LSTM, GRU, CNN, and a hybrid model to a much wider range of pandemic data. The range will incorporate multiple variants, social interventions, public health interventions, and medical interventions to allow for an understanding and comparison of the models' performance over the dynamic conditions of the pandemic. Additionally, a novel hybrid model is constructed with the capabilities of a CNN encoding layer and LSTM and/or GRU decoding layers.

Chapter 4: Methodology

4.1 The Data

Data regarding the spread of COVID-19 in the United States is available from John Hopkins University [32]. The data contains information on positive COVID-19 cases by region every day from January 23, 2020 to the present. At the time of the present study, the data was extracted through March 9, 2023 and is comprised of 3,616,714 observations. The data utilized in the modeling is limited to the entire United States and not any data related to specific states. A misreported negative confirmed case count in the dataset is replaced with 0. The Peru dataset comes from the official MINSA website and contains 1,124 rows of data [33]. The data from MINSA for Peru is similar to that extracted from John Hopkins University for the US. It consists of daily positive case count. Additionally, the data from MINSA also includes the number of COVID-19 related deaths reported per day. The data used for the present study for the Peru analysis comprises the dates from March 6, 2020 through April 2, 2023. For both Peru and the US, variables of interest are the dates and the daily positive COVID-19 case counts summarized at the country level.

Data quality is essential for accurate forecasting of the disease spread. Data for the COVID-19 pandemic faces issues such as inaccuracies in recorded data from missing data, lack of accurate testing, and asymptomatic cases. In the United States, the capacity for COVID-19 testing varied throughout the pandemic with less than 1,000 specimens tested daily until March

4, 2020 [34]. Home tests were issued an emergency use authorization by the U.S. Food and Drug Administration on November 17, 2020 in the US with variable availability during the pandemic. COVID-19 testing is not perfect, and there is a disparity in accuracy between the two major types of testing: molecular tests, such as polymerase chain reaction (PCR) and rapid antigen tests used for at-home testing. The PCR tests are more accurate but take more time to process and require lab technicians and specialized equipment. One study estimated that the PCR test has sensitivities of 80% and specificity of 98 to 99% in clinical settings [35]. This implies that about 20% of patients with the disease will have test results that incorrectly report a negative result. The rapid antigen tests, while maintaining a high specificity (99.9%), have a noticeably lower sensitivity around 65% [36]. This is exacerbated when considering asymptomatic patients. The sensitivity falls for rapid antigen testing of asymptomatic individuals is only 41.2% [37]. Therefore, many false negatives can occur, especially with at home testing, which may result in the spread of the disease without that case being officially counted. Additionally, home test results may not be reported for data collection. Another challenge is that those with asymptomatic cases may never be tested and accounted for but could account for 32% of overall cases [38].

A study found the daily case count consistently had lesser new cases reported on Saturdays and Sundays [39]. This is consistent with the data extracted for both the US and Peru in the present study. The 7-day seasonality is likely due to less testing being completed or reported on the weekend rather than a facet of the disease transmission.

Peru faced additional challenges in its ability to record accurate daily case counts. Due to Peru's inadequate laboratory capacity, there was limited molecular testing in the country for the pandemic. They instead used serological tests resulting in lesser test sensitivity [4]. Serological tests are antibody tests that indicate if a person has been infected at some point in time, but not necessarily that they are currently sick with the disease [39]. Despite the government's goal to expand PCR testing, before January 9, 2021, the tests were only 23.5% of the total [40]. The country also suffered a shortage of home tests [41]. These issues likely resulted in a number of positive cases going untested and unrecorded.

Another difficulty is that Peru's healthcare infrastructure is decentralized with public and private entities which limits a comprehensive response to the pandemic. Healthcare is administered through five organizations which are The Ministry of Health (MINSA), Armed Forced (FFFA), National Police (PNP), EsSAlud, and the private sector [42]. Data sharing was a challenge for Peru as each sector kept separate records of cases with some doing so manually adding to likely inaccuracies in the total number of positive cases. Additionally, Peru's healthcare workers suffered with a lack of appropriate personal protective equipment, resulting in the country having the largest portion of infected healthcare workers [40]. The sick healthcare staff likely factored into data-entry backlogs at the hospitals [41]. Due to these issues, the case count recorded does not necessarily match the actual count of positive cases each day for either the United States or Peru. The reported COVID-19 daily case count data for the United States and Peru is presented in Figures 8 and 9.

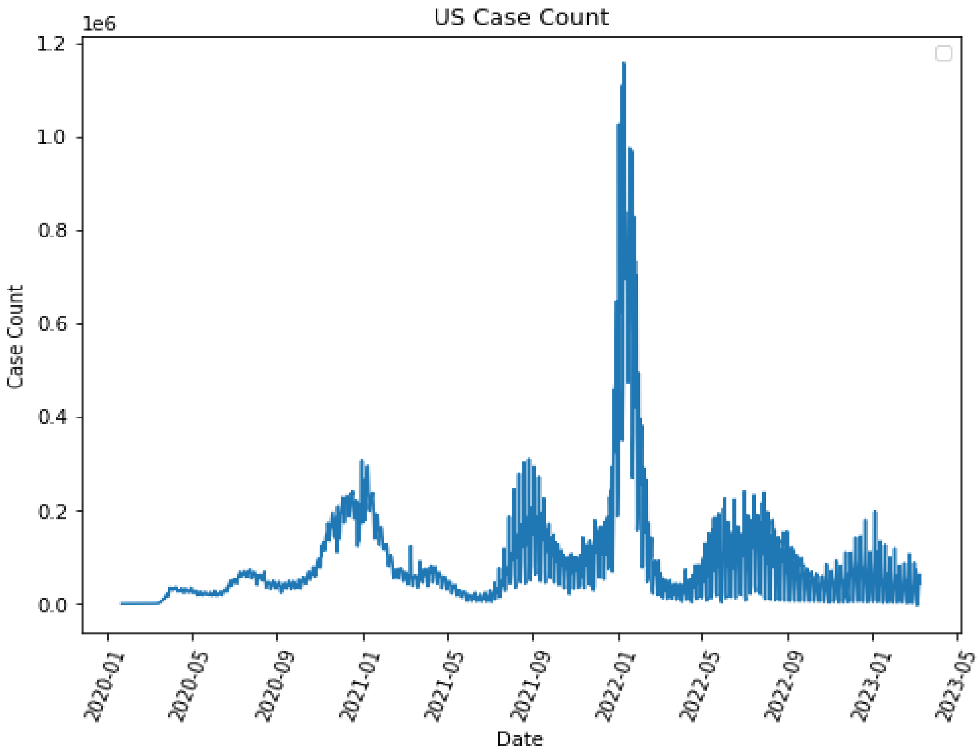

Figure 8. Confirmed Case Count in the United States

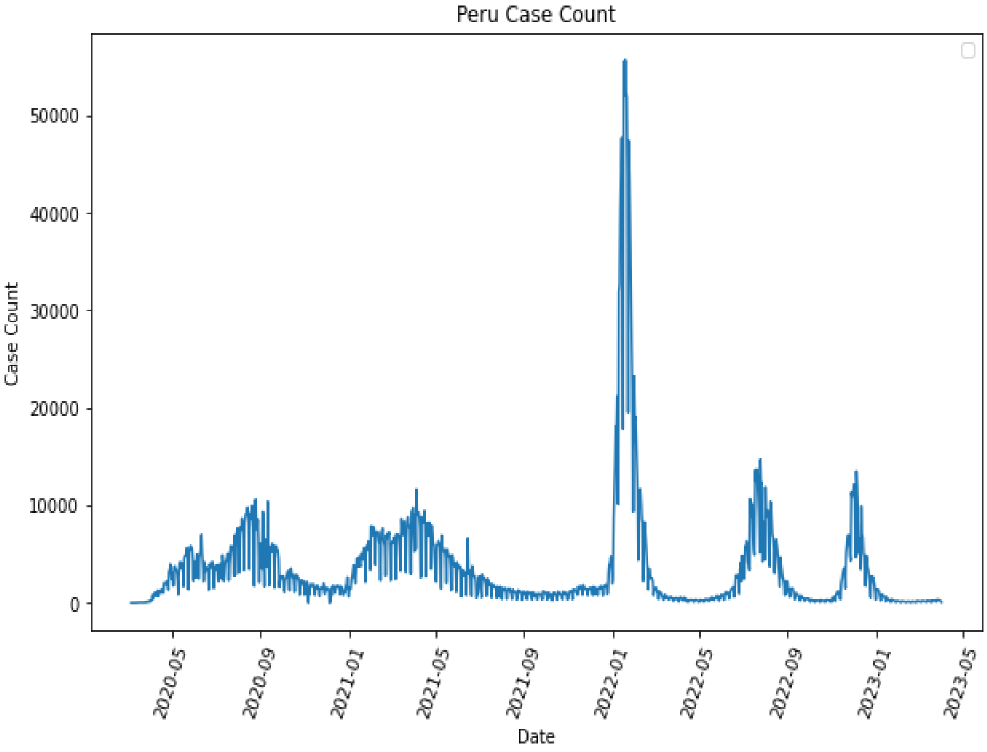

Figure 9: Confirmed Case Count in Peru

Figure 10 displays the case counts per capita for the United States and Peru. The United States tends to have more cases per capita when compared to Peru. There are 5 segments in the United States data with much higher case counts per capita occurring December 2020 through January 2021, August through September 2021, January through February 2022, May through September 2022, and December through January 2023. The segments of Peru data with large case counts per capita correspond to those in the United States on January 2022, July 2022, and December 2022. However, the patterns in case count before January 2022 are not mirrored between the two countries.

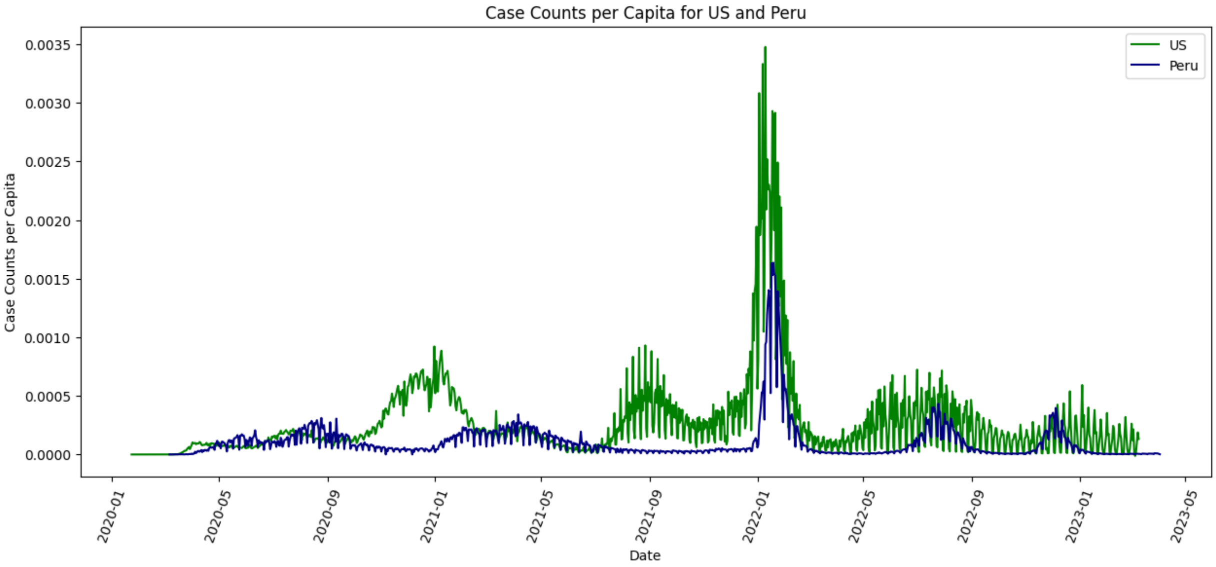

Figure 10: Case Counts per Capita in the United States and Peru

Evaluating the properties of the time series datasets is vital when determining how and which models are applied for forecasting. ARIMA models require that datasets are stationary or can be converted to a stationary dataset. For accurate forecasting, there should be no substantial cyclical variability or other irregularity such that it affects the trend or variance over the time-period studied. Data is considered to trend if the mean is continuously increasing or decreasing. Seasonality can occur if periodic occurrences result in peaks or valleys in the trend. One example of this could be an uptick in COVID-19 cases after large gatherings on holidays over multiple years. Cyclical events are like seasonal variability but occur without a fixed period. One such example could be changing mandates around social distancing or inconsistent rate of spread as different variants become dominant. Lastly, other irregularities can cause data to be erratic such as short-term noise.

Quantifying the stationarity of data is often done using the Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test which is a type of unit root test. The test evaluates if the data is stationary by utilizing linear regression with a deterministic trend component, random walk component, and stationary error. The KPSS test is performed on the United States and Peru datasets with the result that both are stationary with 95% confidence.

Manual evaluation of the consistency of the mean and variance over the dataset is also performed. The 7-day rolling mean and variance for the United States and Peru are presented in Figures 11 and 12. Markedly, there are many segments that appear cyclical in nature with the case count rising to a peak then falling with variable period lengths. These segments have visual differences in mean and variance from other portions of the dataset with a segment between January 2022 and March 2022 being the most unlike the rest of the data in both countries.

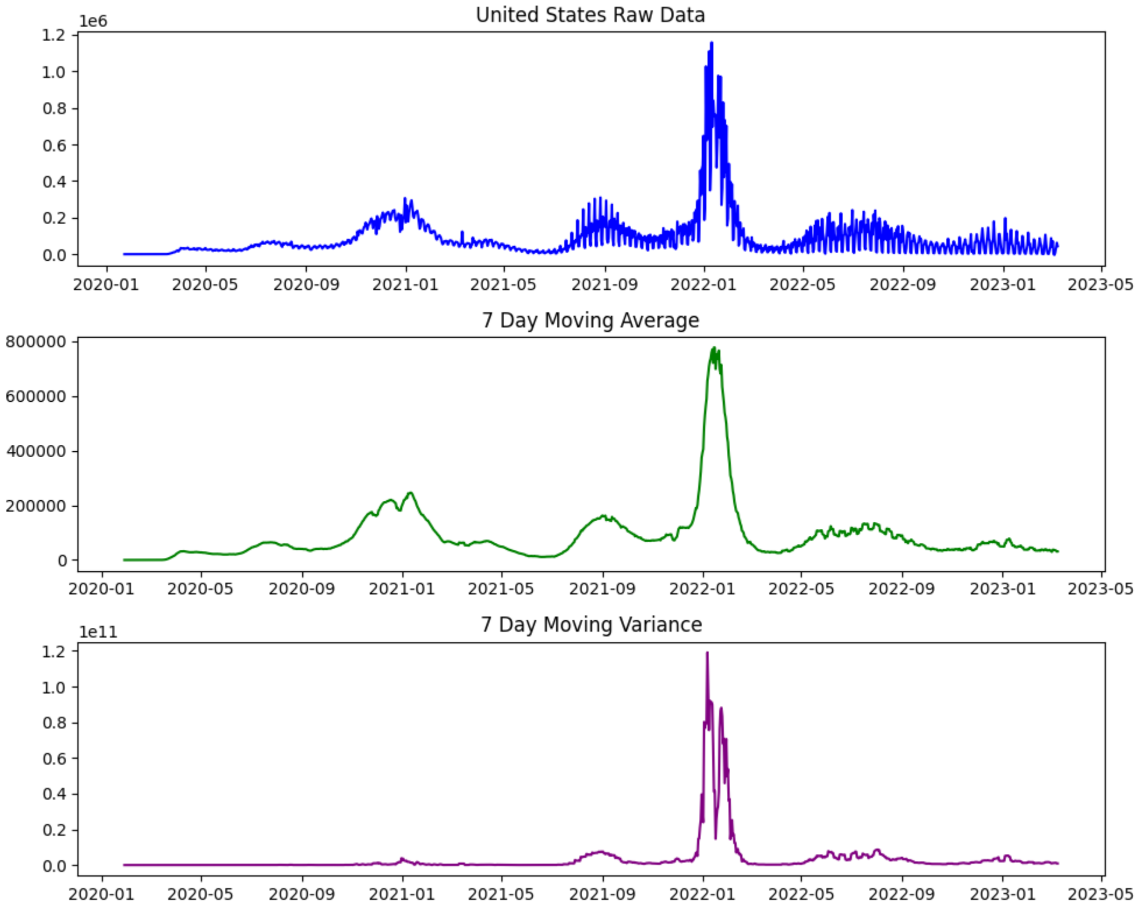

Figure 11: 7 day rolling average and variance of United States Case Count

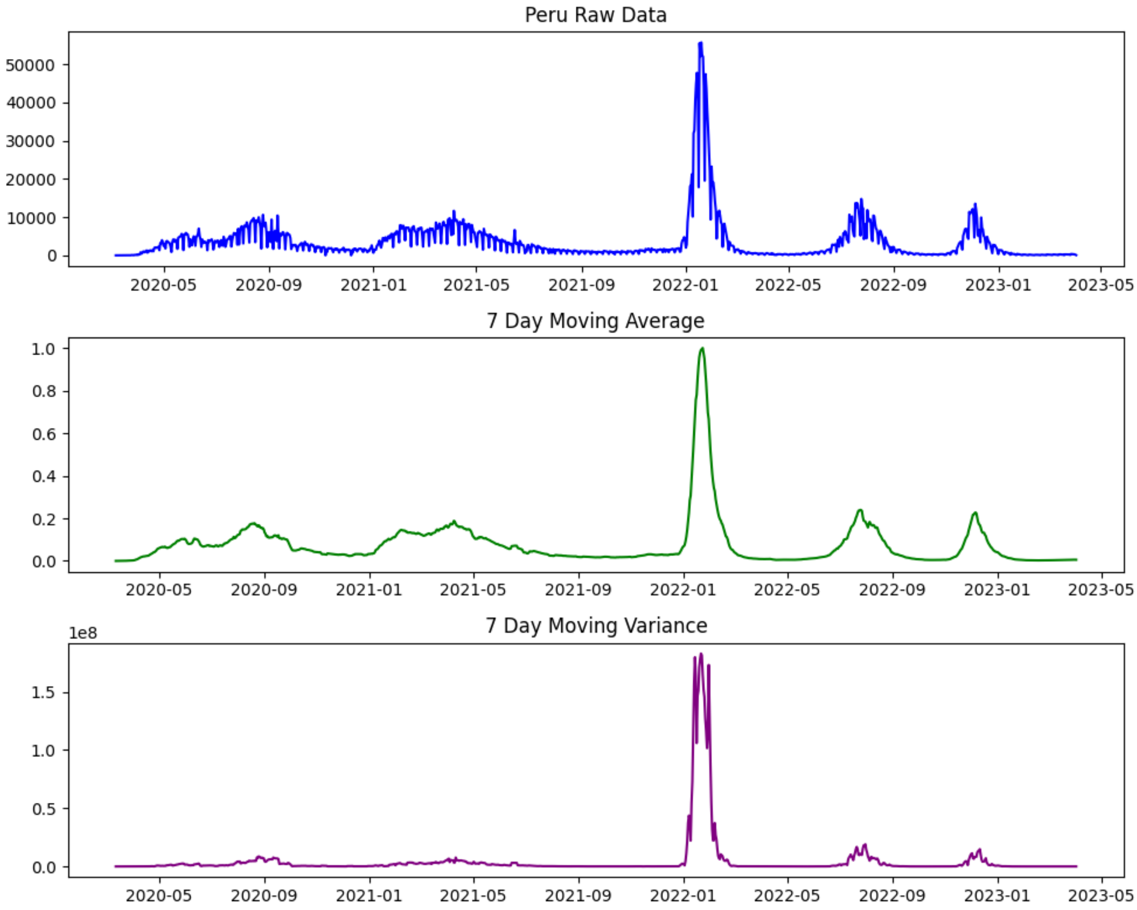

Figure 12: 7 day rolling average and variance of Peru Case Count

To quantify how the data changes over time, the data is segmented into three equal sections. The mean and variance of the sections are presented in Table 1 and 2. The three segments are substantial containing 380 days in each section of the US dataset and 375 days in each section of the Peru dataset. Between the first and second section, the mean of the United States case count increases by 90% and the variance decreases by 32%. Conversely, between the second and third segment the mean decreases by 47% and the variance decreases by 14%. Between the first and second section of the Peru dataset, the mean increases by 68% and the variance increases by 1360%. On the other hand, between the second and third segment the mean decreases by 59% and the variance decreases by 88%.

Date Range	Mean	Variance
January 23, 2020- February 6, 2021	69,724	5,094,875,225
February 7, 2021-February 22, 2022	132,185	3,448,970,413
February 21, 2022- March 9, 2023	62,987	2,951,420,566

Table 1: Mean and Variance over United States data

Date Range	Mean	Variance
March 1, 2020-March 11, 2022	3,550	6,050,983
March 12, 2021-March 22, 2022	5,237	88,398,591
March 23, 2022-April 2, 2023	2,157	10,361,703

Table 2: Mean and Variance over Peru data

Histograms for the United States and Peru segments depict the frequency of the distribution of the daily case count in each data segment. Figures 13 and 14 show overlayed histograms of the first, second, and third segments for the datasets. For both countries, all 3 segments have skewed distributions. This result is expected as many days of the pandemic had lower case counts with the highest frequency bins at the lower numbers of cases per day. The first and third segments in the datasets had a similar range of bins. The second segments in both datasets had a much larger range of bins with discernable frequency in the high case ranges due to the case spikes withing the second segments of the datasets. The extremely right skew of the second segments of both datasets is a product of much of the time period in the segment having lower more consistent case count before the large spike in cases between January 2022 and March 2022.

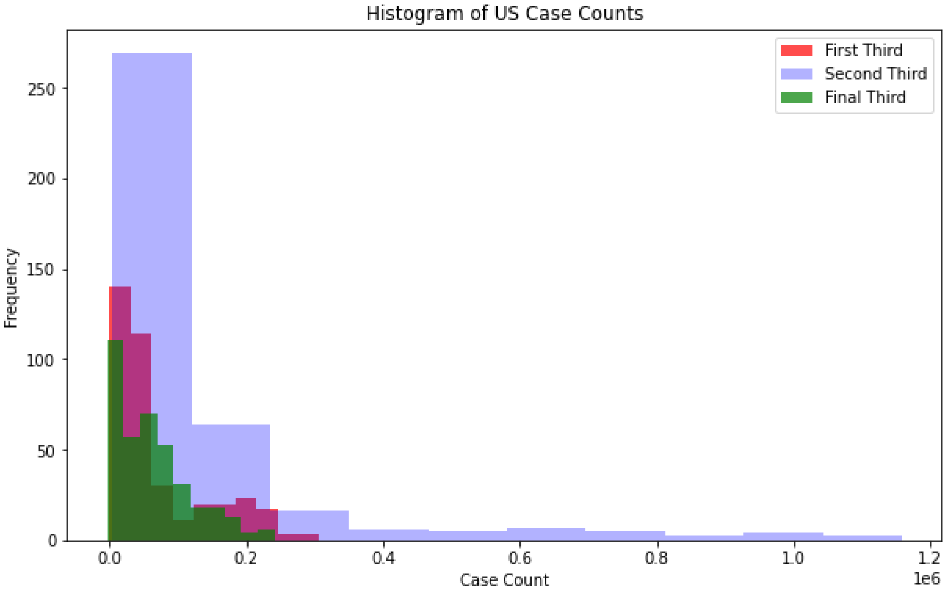

Figure 13: Histogram of United States segmented case counts

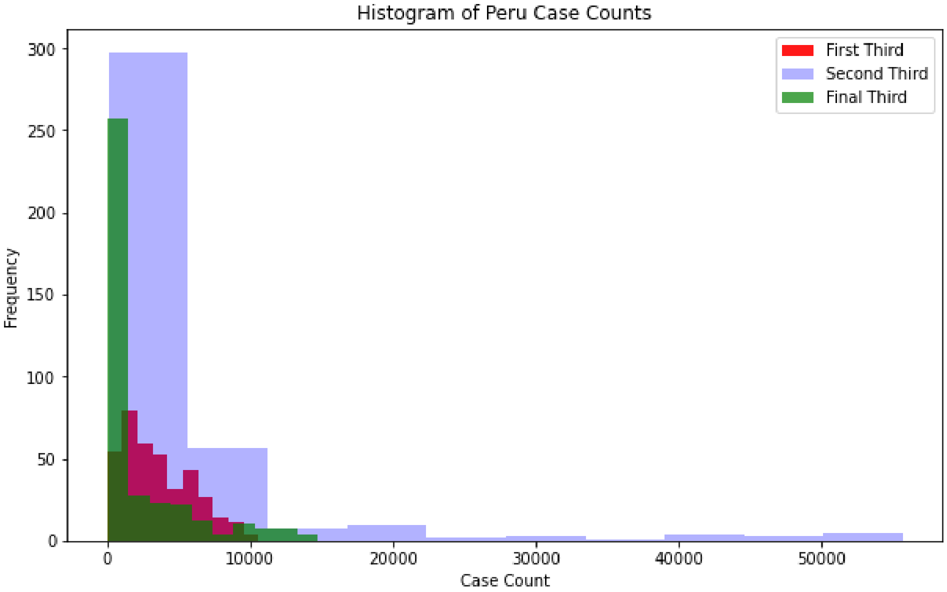

Figure 14: Histogram of Peru segmented case counts

Autocorrelation is utilized to quantify if significant seasonality is occurring. In these plots, the correlation of a signal is compared with a lagged signal after it. There is a high correlation in very low lag days as one day's case count depends on the days or weeks of cases before it. This is due to the spread of the disease being based on interactions with those that were already sick in the area. Additionally, there is a lag in displaying symptoms, taking a COVID-19 test, and receiving a positive test after initial exposure from an infected individual which can allow for the spread of the disease over multiple days. Figure 15 highlights the seasonal 7-day pattern in the lower lag days in the United States and displays the higher lag days with spikes around 140 days and the year mark. Figure 16 illustrates the seasonal 7-day pattern in the lower lag days for the Peru dataset and demonstrates the seasonality at approximately 190 days, 320 days, and 520 days. These spikes imply seasonality in both datasets.

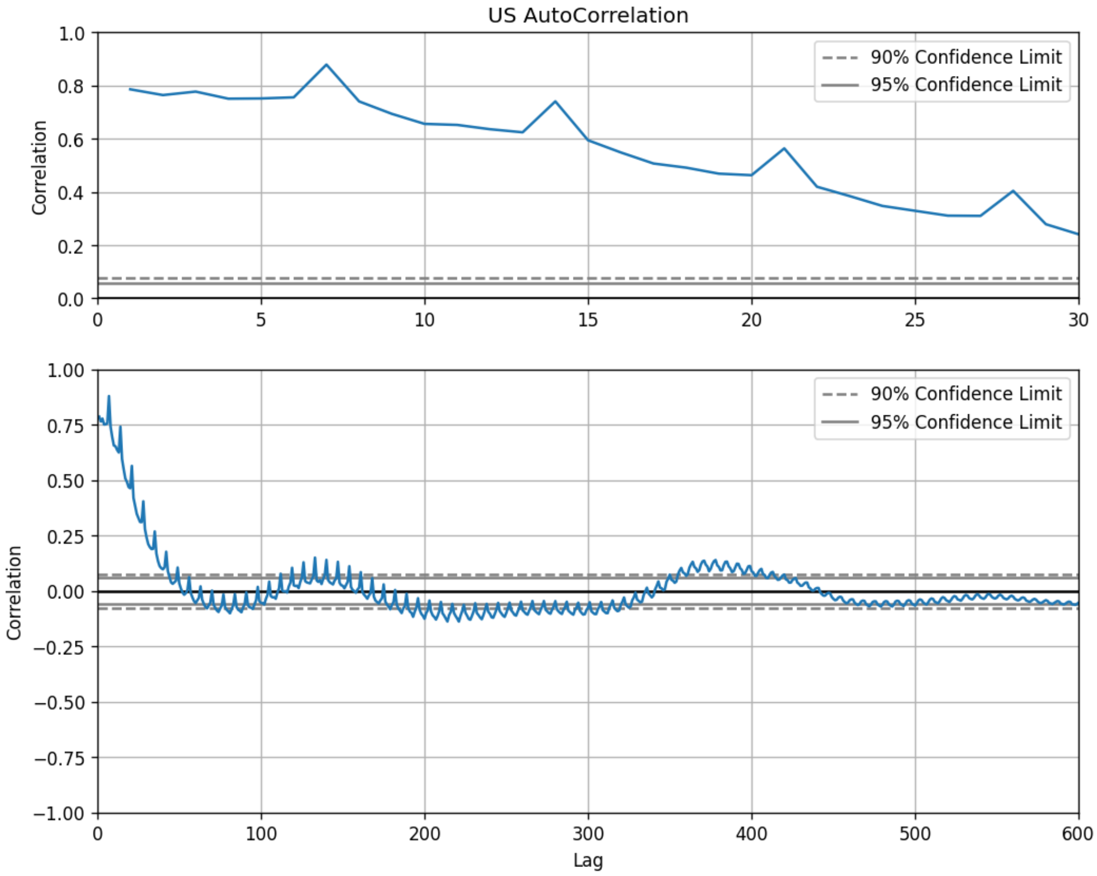

Figure 15: Autocorrelation of United States Case Counts

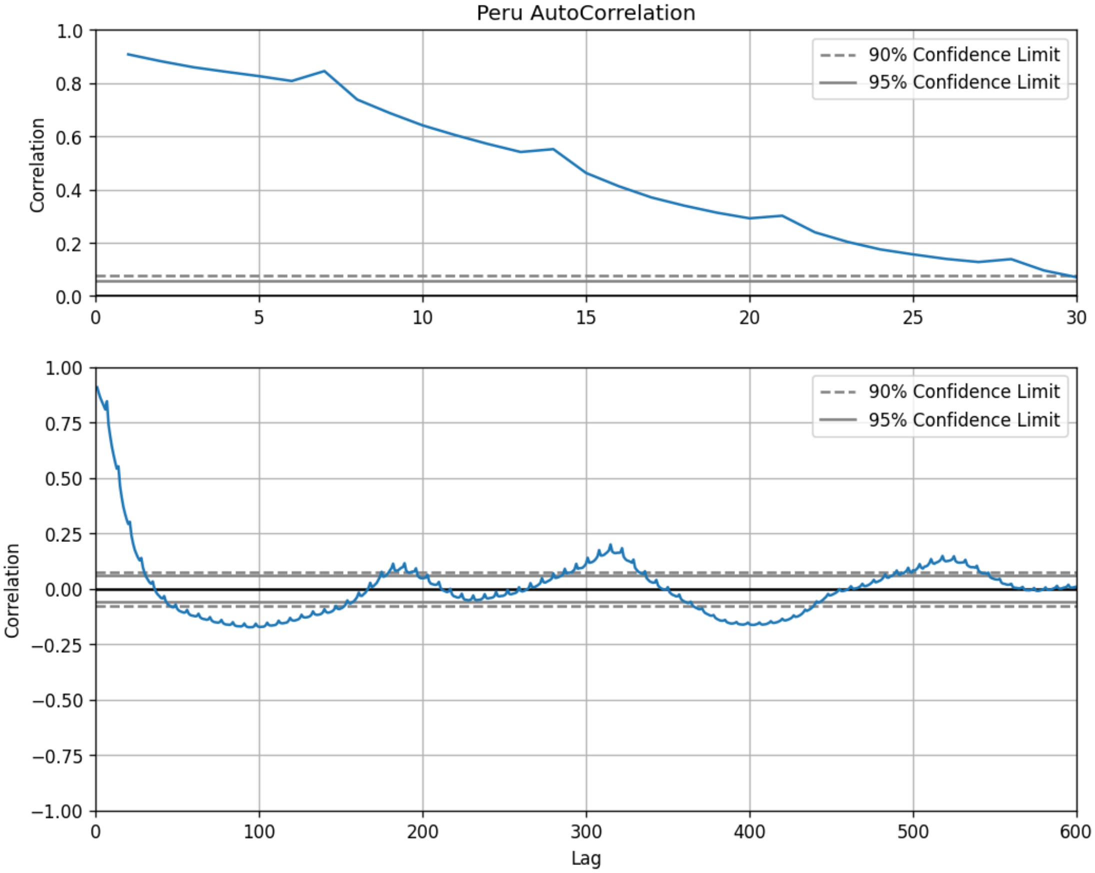

Figure 16: Autocorrelation of Peru Case Counts

To minimize the noise in the data, the COVID-19 case count is smoothed using a 7-day moving average before the models are trained and forecasting is performed. The case counts are Min-Max scaled as in Equation 17 where the new value x' is calculated using the current value x, minimum value of the dataset x_{min}, and maximum value of the dataset x_{max}.

$$x' = \frac{(x - x_{min})}{(x_{max} - x_{min})} \tag{17}$$

The final step to prepare the data to be modeled is to create the input and output sequences. The time series data is segregated using rolling windows to create 30-day input sequences with 7, 15, or 30-day output sequences. To demonstrate how the window rolls over the dataset, three consecutive 30-day input sequences in yellow and 30-day output sequences in orange are shown in Figure 17. Other options to sequence data are splitting the data at a fixed time point or utilizing an expanding window. These methods are not optimal for the dataset due to its dynamic nature. Both options would train on a larger range of dates that would likely contain segments of time that have different behavior which would reduce the prediction accuracy of the output series.

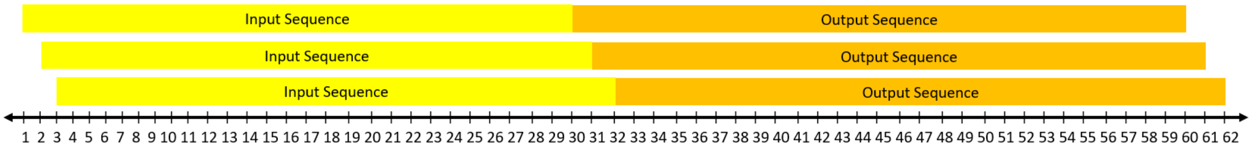

Figure 17: Input and Output rolling window sequences

4.2 Modeling Setup

Each model is run three times to forecast 7, 15, and 30 day output sequences. The range in output sequences allows for an exploration of how the model's performance varies over different prediction lengths. The longer sequences of 30 days would be optimal to allow for the most time to make actionable interventions. Each model forecasts all rolling window output sequences using dates from January 23, 2020 to February 7, 2023 for the United States data and March 6, 2020 to March 3, 2023 for the Peru dataset. The average results of the predictions are presented as a metric for model performance over the full dataset in the results section. Additionally, each model will be utilized to predict a 7, 15, and 30 day output sequence after February 7, 2023 for the United States and March 3, 2023 for the Peru Dataset.

To create models useful for predicting the daily case count, the CNN, LSTM, GRU, hybrid, and ARIMA model's architecture and/or hyperparameters first are determined. Hyperband tuning is utilized over data from January 23, 2020 to February 7, 2023 for the United States data and March 6, 2020 to March 3, 2023 for the Peru dataset to optimize the model parameters and determine the architecture for the CNN, LSTM, GRU, and hybrid models. Hyperband tuning uses explore and exploit theory to converge on an accurate solution with faster processing time [43]. Each of these models use the ADAM optimizer to minimize the mean squared error loss function. The learning rate of the optimizer was varies between 0.01, 0.001, and 0.0001. Each model is allowed 2000 max epochs for training and fit with 2000 epochs with a patience of 50 epochs.

4.2.1 CNN Modeling Setup

All CNN architectures are required to have at least one convolutional layer but could have up to 5 layers. Each convolutional layer has a filter size and kernel size that vary as in Table 3 and can have ReLU activation or no activation. After each convolutional layer, a pooling layer can occur with max pooling of pool size 2, 3 or 4. Next, a flatten layer occurs. There can be up to 5 dense layers with varying neuron size as shown in Table 3 each of which could have ReLU activation or no activation. After the dense layers, a dropout layer can be selected with a dropout rate of 0.1, 0.3, or 0.5. Finally, there is a dense output layer that is the size of the prediction sequence.

Layer	**Parameters**
Input Layer	-
1 to 5 Convolutional Layers	Filter size: 2, 4, 8, 16, 32, 64, 128, or 256 Kernels size: 1, 3, 5, 7, 9, 11, 13, or 15 ReLU activation or no activation
Up to 1 Max pooling layer per CNN Layer	Pool size: 2,3,4
Flatten Layer	-
Up to 5 Dense Layers	Neurons: 25, 50, 75, or 100 ReLU activation or no activation
Up to 1 Dropout Layer	Dropout rate: 0.1, 0.3, or 0.5
Dense Output Layer	-

Table 3: CNN Architecture

4.2.2 LSTM Modeling Setup

The LSTM model’s architecture is required to have at least one LSTM layer but could have up to 3 layers. Each LSTM layer has some unit size as in Table 4 and can have ReLU, tanh, sigmoid, or no activation. After the LSTM layers, a dropout layer can be selected with a dropout rate of 0.1, 0.3, or 0.5. Finally, there is a dense output layer the size of the prediction sequence.

Layer	Parameters
Input Layer	-
1 to 3 LSTM Layers	Units: 2, 4, 8, 16, 32, 64, 128, or 256 ReLU, tanh, sigmoid, or no activation
Up to 1 Dropout Layer	Dropout rate: 0.1, 0.3, or 0.5
Dense Output Layer	-

Table 4: LSTM Architecture

4.2.3 GRU Modeling Setup

The GRU model's architecture is required to have at least one GRU layer but could have up to 3 layers. Each GRU layer has some unit size as in Table 5 and can have ReLU, tanh, sigmoid, or no activation. After the GRU layers, a dropout layer can be selected with a dropout rate of 0.1, 0.3, or 0.5. Finally, there is a dense output layer the size of the prediction sequence.

Layer	Parameters
Input Layer	-
1 to 3 GRU Layers	Units: 2, 4, 8, 16, 32, 64, 128, or 256 ReLU, tanh, sigmoid, or no activation
Up to 1 Dropout Layer	Dropout rate: 0.1, 0.3, or 0.5
Dense Output Layer	-

Table 5: GRU Architecture

4.2.4 Hybrid Modeling Setup

Lastly, the hybrid model utilizes the CNN architecture options from the input layer through the flatten layer. The CNN architecture is the encoder to learn patterns from the dataset and pass it on to the LSTM and/or GRU layers. Then the LSTM and GRU architecture options are allowed to the optimizer. The LSTM and/or GRU architecture works to decode and model the temporal associations. Next, up to 5 dense layers and a dropout layer may occur with parameters as in Table 6. Finally, a dense output layer the size of the prediction sequence is implemented.

Layer	Parameters
Input Layer	-
1 to 5 Convolutional Layers	Filter size: 2, 4, 8, 16, 32, 64, 128, or 256 Kernels size: 1, 3, 5, 7, 9, 11, 13, or 15 ReLU activation or no activation
Up to 1 Max pooling layer per CNN Layer	Pool size: 2,3,4
Flatten Layer	-
Up to 3 LSTM Layers	Units: 2, 4, 8, 16, 32, 64, 128, or 256 ReLU, tanh, sigmoid, or no activation
Up to 3 GRU Layers	Units: 2, 4, 8, 16, 32, 64, 128, or 256 ReLU, tanh, sigmoid, or no activation
Up to 5 Dense Layers	Neurons: 25, 50, 75, or 100 ReLU activation or no activation
Up to 1 Dropout Layer	Dropout rate: 0.1, 0.3, or 0.5
Dense Output Layer	-

Table 6: Hybrid Architecture

4.2.5 ARIMA Modeling Setup

Seasonal ARIMA modeling requires a prespecified seasonal length. Both datasets have multiple seasonal peaks as can be seen in the autocorrelation plots of the preprocessed datasets in Figures 18 and 19. The short-term seasonality occurring every 7 days in both datasets is smoothed due to the 7-day moving average preprocessing step. The red dashed line indicates the peak occurring at lag 135 and the green dashed line marks the peak at 372 for the United States data in Figure 18. In Figure 19 the red dashed line indicates the peak at 185, the green dashed line at peak 316, and the purple line at lag 521 in the Peru dataset. Incorporating such large seasonalities result in failure because the application runs out of memory due to the high number of parameters to be estimated, so a nonseasonal ARIMA model will be utilized for these datasets.

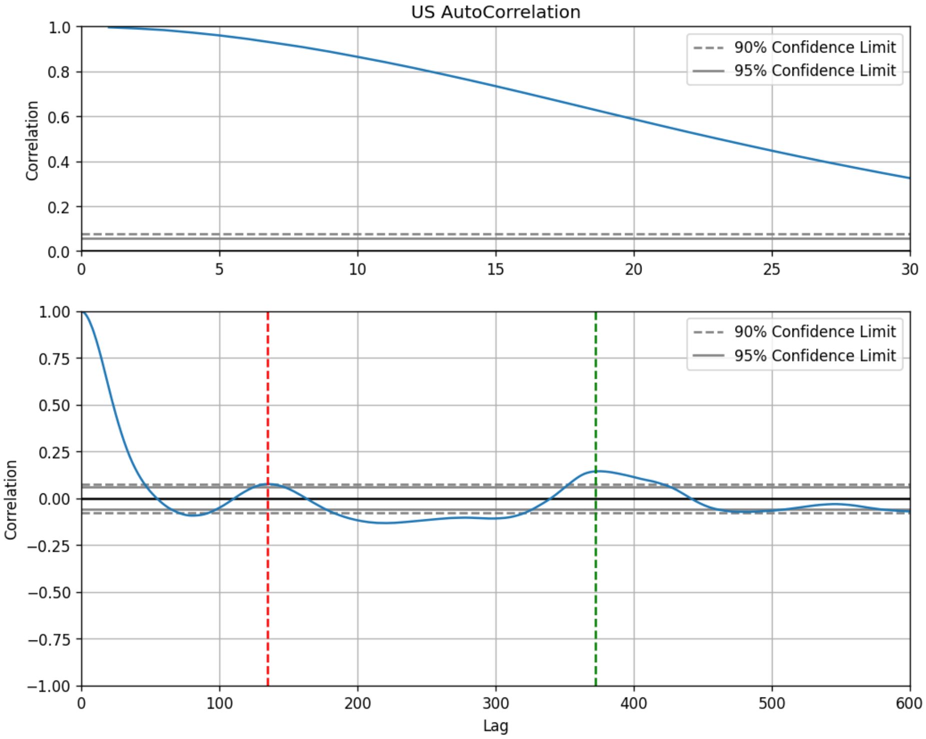

Figure 18. Autocorrelation plots of the United States Preprocessed Data

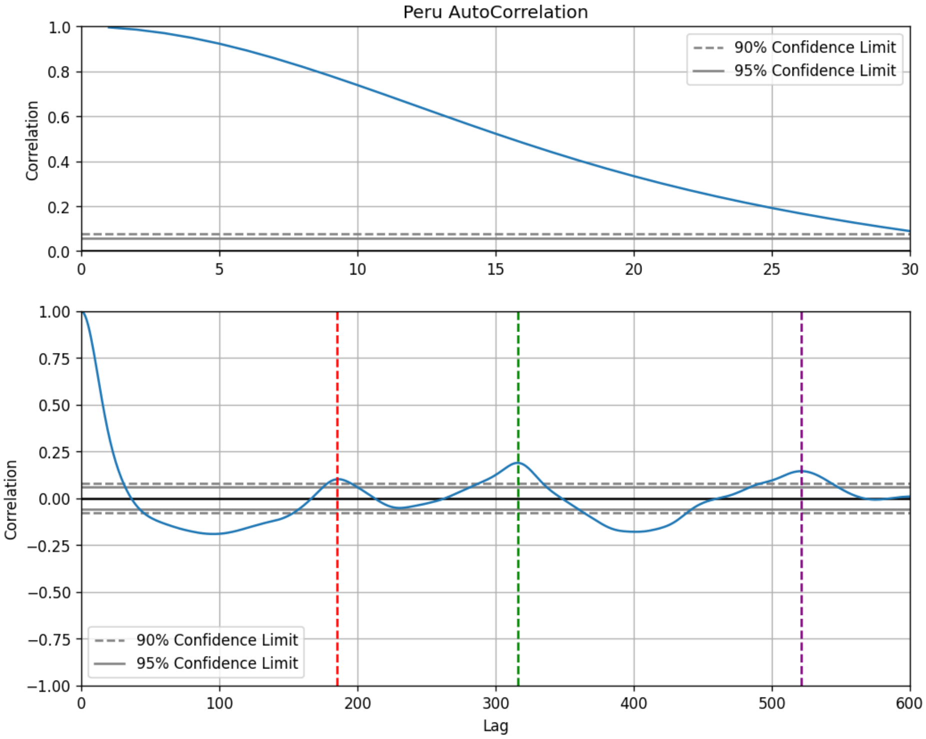

Figure 19. Autocorrelation plots of the Peru Preprocessed Data

The KPSS test previously determined that differencing is not necessary for the datasets, therefore the estimated ARIMA modeling parameter d is set to 0 for no differencing. The ARIMA modelling parameters for autocorrelation (p) and moving average (q) can be estimated utilizing the autocorrelation and partial autocorrelation plots if the data follows ARIMA(p,d,0) or ARIMA(0,d,q) [44]. If both p and q are nonzero, then the plots do not allow for the parameter estimation. The partial autocorrelation plot allows conclusions to be drawn about the autoregressive portion of the ARIMA model. The parameter p estimation is the number of lags outside of the shaded significance region in the partial autocorrelation plots for the United States in Figure 21 and Peru in Figure 23. Both plots have more than 10 consecutive and nonconsecutive lags outside of the significance region making estimation of the parameter impossible. The autocorrelative parameter q can sometimes be estimated using the autocorrelation plots. When evaluating the autocorrelation plots with no differencing in Figure 20 for the United States and Figure 22 for Peru, both datasets have many consecutive high positive values in the autocorrelation of the dataset with no obvious changes in trend making the estimation impossible. The difficulty in assigning p and q values from the plots make it likely that the ARIMA model is not ARIMA(p, d, 0) or ARIMA (0, d, q).

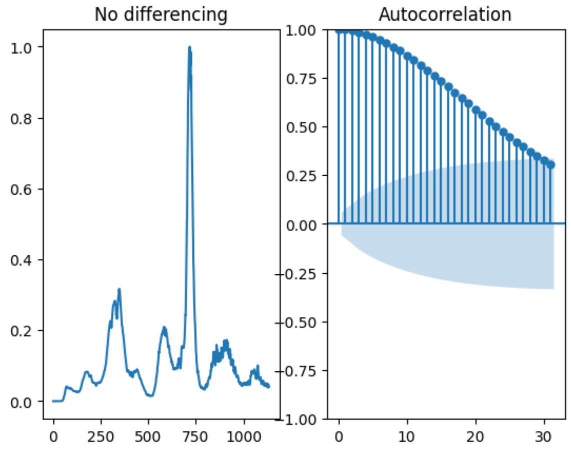

Figure 20: US Autocorrelation with no differencing

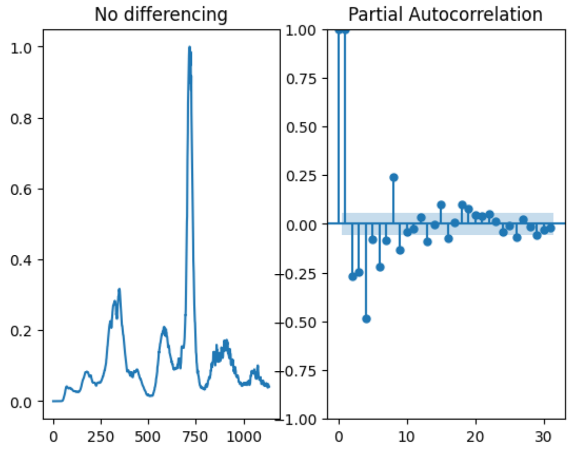

Figure 21: US Partial Autocorrelation with no differencing

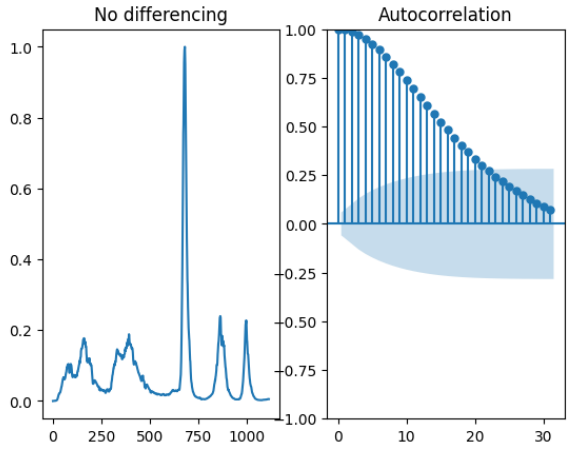

Figure 22: Peru Autocorrelation with no differencing

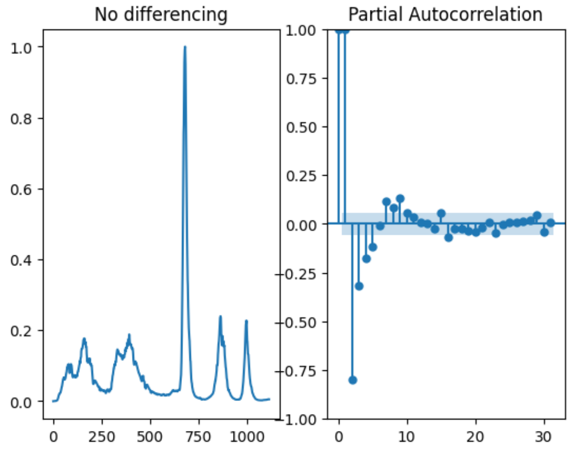

Figure 23: Peru Partial Autocorrelation with no differencing

Due to the issues in finding the autoregressive and moving average term by hand, these estimations will be performed using the *auto_arima* function from *pmdarima* in Python [45]. The function fits models using a range of p, d, and q values. The best parameters are returned that minimize the Akaike information criterion (AIC). The AIC is an estimation of the model's ability to fit the dataset calculated using the maximum likelihood estimate and number of parameters in the model. The *auto_arima* function can find local minimum instead of global minimum, so the parameters will be varied manually and then fit to the dataset to confirm the model parameters that return the lowest AIC value.

4.3 Metrics

Performance of the models are presented using root mean squared error (RMSE) and mean absolute percentage error (MAPE). The metrics are calculated in Equations 15 and 16 where n is the number of observations in the dataset, y_i is the true value, and $\hat{y}_i$ is the value predicted by the model.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \tag{15}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i} * 100\% \tag{16}$$

Chapter 5: Results and Analysis

5.1 Modeling Results

5.1.1 CNN Model Results

Table 7 presents the optimal architecture and modeling parameters found during hyperparameter tuning of the CNN models for the 7 day, 15 day, and 30 day prediction sequences for the United States and Peru. The optimal architecture for 7 day predictions of both datasets have the least layers with two convolutional layers each. The United States models utilize two additional dense layers, whereas the Peru dataset includes a max pooling layer after the first convolution. The 15 day and 30 day optimal architectures are deeper containing additional layers compared to the models for the 7 day predictions. The architecture for 15 day prediction sequences in the United States has 3 convolutional layers, a max pooling layer, and 3 additional dense layers. The Peru architecture for 15 day predictions has 5 convolutional layers, a max pooling layer, and 3 additional dense layers. For the 30 day predictions, the United States model utilizes 4 convolutional layers, a max pooling layer, and 4 additional dense layers. The Peru architecture for 30 day predictions contains 3 convolutional layers, a max pooling layer, and an additional dense layer.

Dataset	7 Day Prediction Model	15 Day Prediction Model	30 Day Prediction Model
United States	Learning rate 0.0001	Learning rate 0.001	Learning rate 0.001
	Input	Input	Input
	Convolutional layer with filter size 32 and kernel size 11 with ReLU activation	Convolutional layer with filter size 32 and kernel size 1	Convolutional layer with filter size 8 and kernel size 7 with ReLU activation
	Convolutional layer with filter size 256 and kernel size 7 with ReLU activation	Convolutional layer with filter size 256 and kernel size 7	Convolutional layer with filter size 8 and kernel size 7
	Flatten	Max pooling layer with pool size of 3	Convolutional layer with filter size 2 and kernel size 15
	Dense Layer with 75 units	Convolutional layer with filter size 64 and kernel size 1 with ReLU activation	Max pooling layer with pool size of 2
	Dense Layer with 25 units	Convolutional layer with filter size 16 and kernel size 5	Convolutional layer with filter size 64 and kernel size 1 with ReLU activation
	Dense Output Layer	Convolutional layer with filter size 32 and kernel size 3	Flatten
		Flatten	Dense Layer with 50 units with ReLU activation
		Dense Layer with 100 units with ReLU activation	Dense Layer with 75 units
		Dense Layer with 25 units	Dense Layer with 75 units with ReLU activation
		Dense Layer with 25 units	Dense Layer with 50 units with ReLU activation
		Dense Output Layer	Dense Output Layer
Peru	Learning rate 0.001	Learning rate 0.001	Learning rate 0.001
	Input	Input	Input
	Convolutional layer with filter size 256 and kernel size 7 with ReLU activation	Convolutional layer with filter size 32 and kernel size 5 with ReLU activation	Convolutional layer with filter size 4 and kernel size 11 with ReLU activation
	Max pooling layer with pool size of 2	Convolutional layer with filter size 8 and kernel size 11	Max pooling layer with pool size of 2
	Convolutional layer with filter size 128 and kernel size 3 with ReLU activation	Max pooling layer with pool size of 4	Convolutional layer with filter size 64 and kernel size 5 with ReLU activation
	Flatten	Convolutional layer with filter size 32 and kernel size 9 with ReLU activation	Convolutional layer with filter size 128 and kernel size 5
	Dense Output Layer	Flatten	Flatten
		Dense Layer with 25 units	Dense Layer with 50 units
		Dense Layer with 75 units	Dense Output Layer
		Dense Layer with 100 units with ReLU activation	
		Dense Output Layer	

Table 7: CNN Optimal Hyperparameters

The CNN model's prediction performance is presented in Table 8 for both the United States and Peru data. Each CNN model predicts the daily case counts with low error for all prediction lengths. The lowest RMSE prediction for both datasets occur when predicting 7 day case counts, and the highest results occur when predicting the 30 day case counts. The RMSE for the United States prediction sequences are lower than those for Peru with average values of

0.018 versus 0.025. The MAPE for the United States prediction sequences are higher than those for Peru with an average MAPE of 170% versus 30%.

Dataset	Prediction Length (Days)	RMSE	MAPE
United States	7	0.007	183%
	15	0.015	213%
	30	0.033	106%
Peru	7	0.014	17%
	15	0.025	37%
	30	0.035	29%

Table 8: CNN Results

The 7 day, 15 day, and 30 day forecasted values for the United States and Peru datasets are depicted graphically in Figure 24 and Figure 25. For each day, the average of the forecast value for each prediction sequence is plotted versus the actual daily case count. Overall, the predictions fit closely to the actual case counts for both data sets except for a few date ranges. In the United States dataset, the 30 day prediction model struggles to forecast the data accurately with predictions consistently lower than the actual case count up to 60 days into the dataset. Additionally, the 30 day prediction overestimates the actual case count between 870-960 days in the dataset. The 15 day and 30 day forecast values underestimate the actual case counts for the first 60 and 140 days of the dataset respectively. The 30 day prediction struggles to accurately predict the plateau after the last two peaks in the dataset with underestimation of the actual case counts from 900-940 days into the dataset and after 995 days of the dataset.

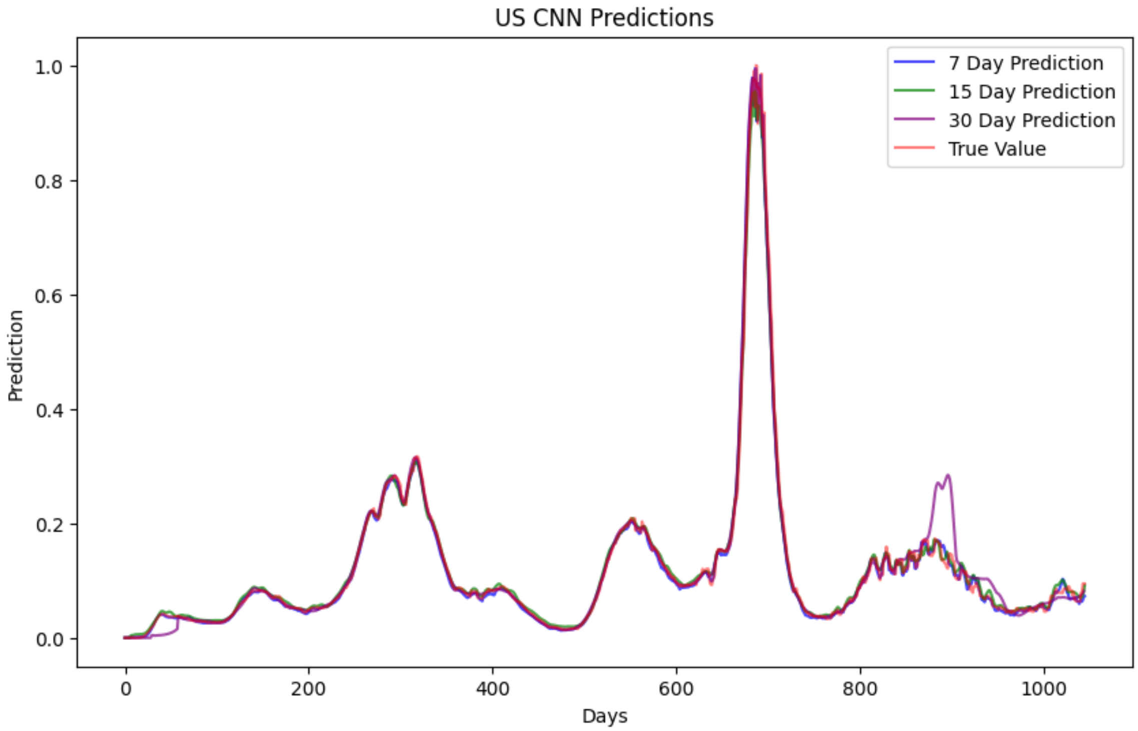

Figure 24: United States CNN prediction sequences

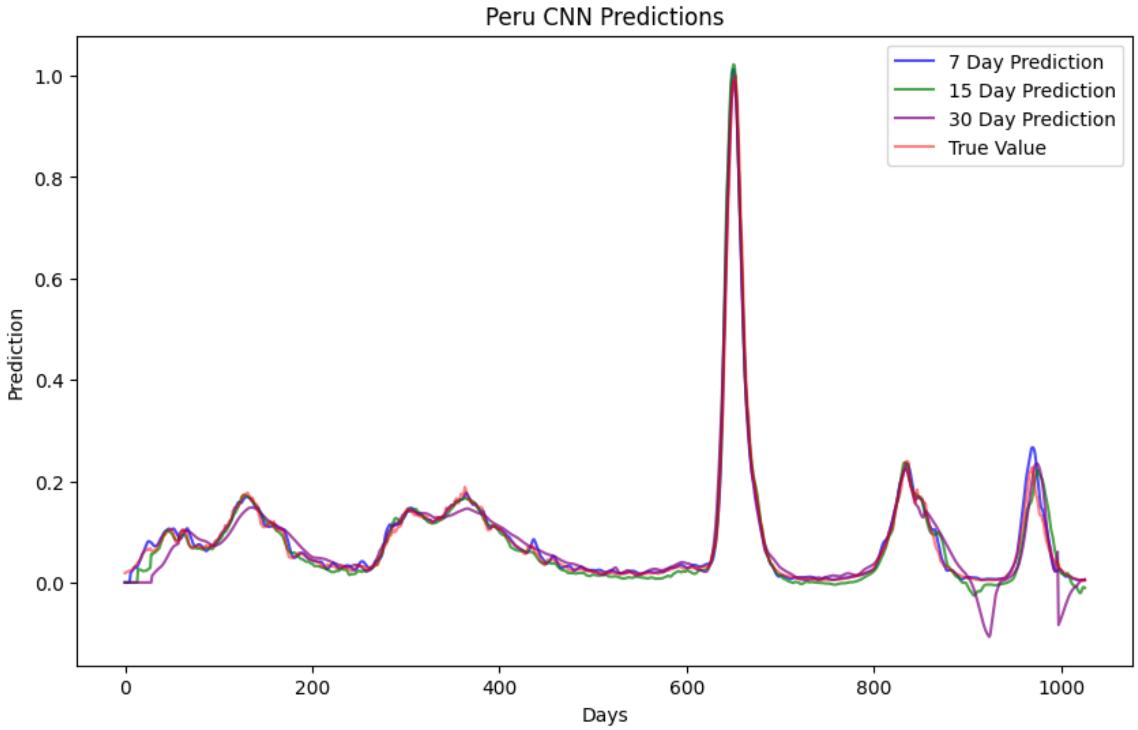

Figure 25: Peru CNN prediction sequences

5.1.2 LSTM Model Results

The optimal architecture and modeling parameters found using hyperparameter tuning for the 7 day, 15 day, and 30 day predictions are presented in Table 9 for the LSTM models. All models utilize between 2 and 3 LSTM layers. The architecture for the 7 day prediction of the United States dataset contains the largest learning rate of 0.01 versus the learning rate of 0.001 that is found to be optimal for all other LSTM predictions. The predictions for the 7 day and 15 day sequences utilize models containing 2 LSTM layers for the United States data and 3 LSTM layers for the Peru dataset. For the 30 day predictions, the United States model incorporates 3 LSTM layers, but the Peru model integrates 2 LSTM layers. Each model integrates between 1 and 3 activation functions.

Dataset	7 Day Prediction Model	15 Day Prediction Model	30 Day Prediction Model
United States	Learning rate 0.01	Learning rate 0.001	Learning rate 0.001
	Input	Input	Input
	LSTM Layer with 32 units	LSTM Layer with 4 units with tanh activation	LSTM Layer with 256 units
	LSTM Layer with 256 units with ReLU activation	LSTM Layer with 256 units with tanh activation	LSTM Layer with 2 units with ReLU activation
	Dense Output Layer	Dense Output Layer	LSTM Layer with 256 units with tanh activation
			Dense Output Layer
Peru	Learning rate 0.001	Learning rate 0.001	Learning rate 0.001
	Input	Input	Input
	LSTM Layer with 128 units with ReLU activation	LSTM Layer with 64 units with ReLU activation	LSTM Layer with 128 units with ReLU activation
	LSTM Layer with 128 units with ReLU activation	LSTM Layer with 256 units with ReLU activation	LSTM Layer with 256 units with ReLU activation
	LSTM Layer with 256 units with tanh activation	LSTM Layer with 128 units with sigmoid activation	Dense Output Layer
	Dense Output Layer	Dense Output Layer	

Table 9: LSTM Optimal Hyperparameters

Table 10 reflects the LSTM model's performance using the metrics RMSE and MAPE. For both datasets, the forecast of the 7 day sequences result in the lowest RMSE, and the 30 day prediction lengths result in the highest RMSE. The United States predictions have lower RMSE overall with an average value of 0.018 versus the 0.022 average value of the Peru dataset. The United States MAPE was higher than in the Peru dataset with average MAPEs of 221% versus 39%.

Dataset	Prediction Length (Days)	RMSE	MAPE
United States	7	0.007	204%
	15	0.014	260%
	30	0.034	200%
Peru	7	0.011	15%
	15	0.025	20%
	30	0.030	52%

Table 10: LSTM Results

Figure 26 and Figure 27 visualize the prediction results for 7 day, 15 day, and 30 day forecasting sequences in the United States and Peru. Each plot contains the actual daily case count and the average forecast value on each day for each prediction sequence. The United States predictions closely follow the actual value, but those for the Peru dataset deviate often for the 15 day and 30 day prediction lengths. The United States 30 day predictions underestimate the actual value until 60 days into the dataset and overestimate it into a false peak between 875 and 915 days. The 15 day predictions in Peru underestimate the actual value the first 30 days of the dataset and overestimate the spike occurring between 970 and 985 days into the dataset. The

Peru 30 day prediction sequence has multiple overestimations and underestimation of the data with the worst occurring at the largest peak in the Peru dataset between 635 and 680 days.

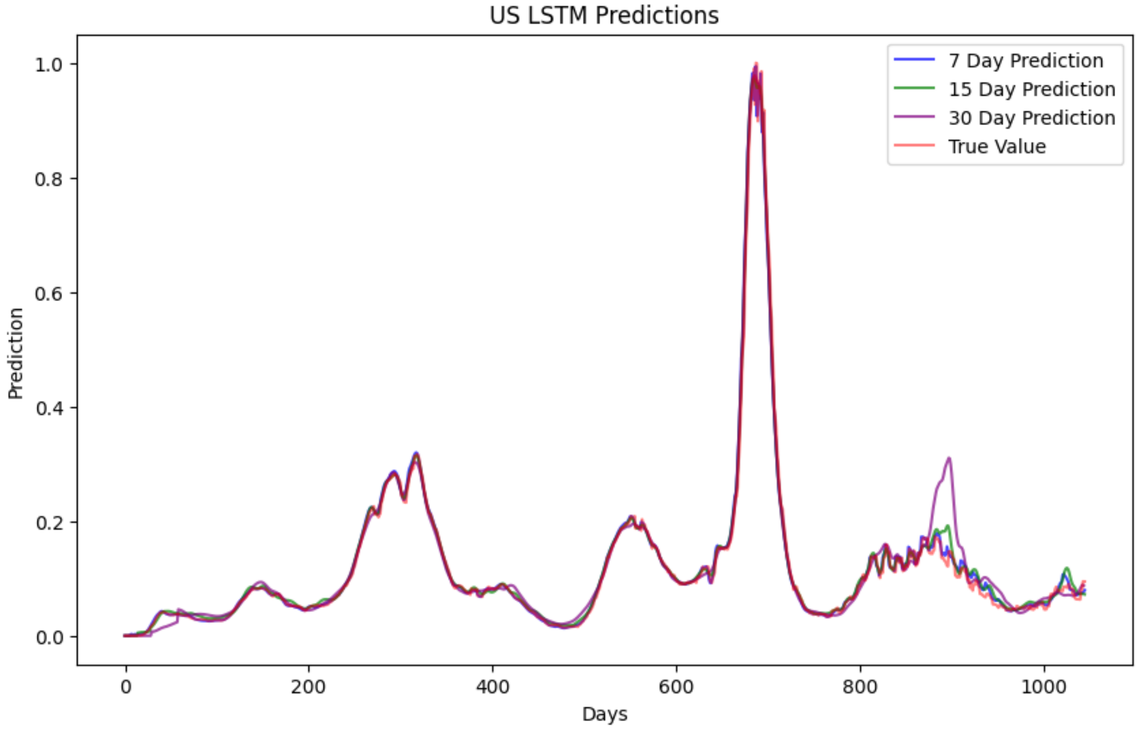

Figure 26: United States LSTM prediction sequences

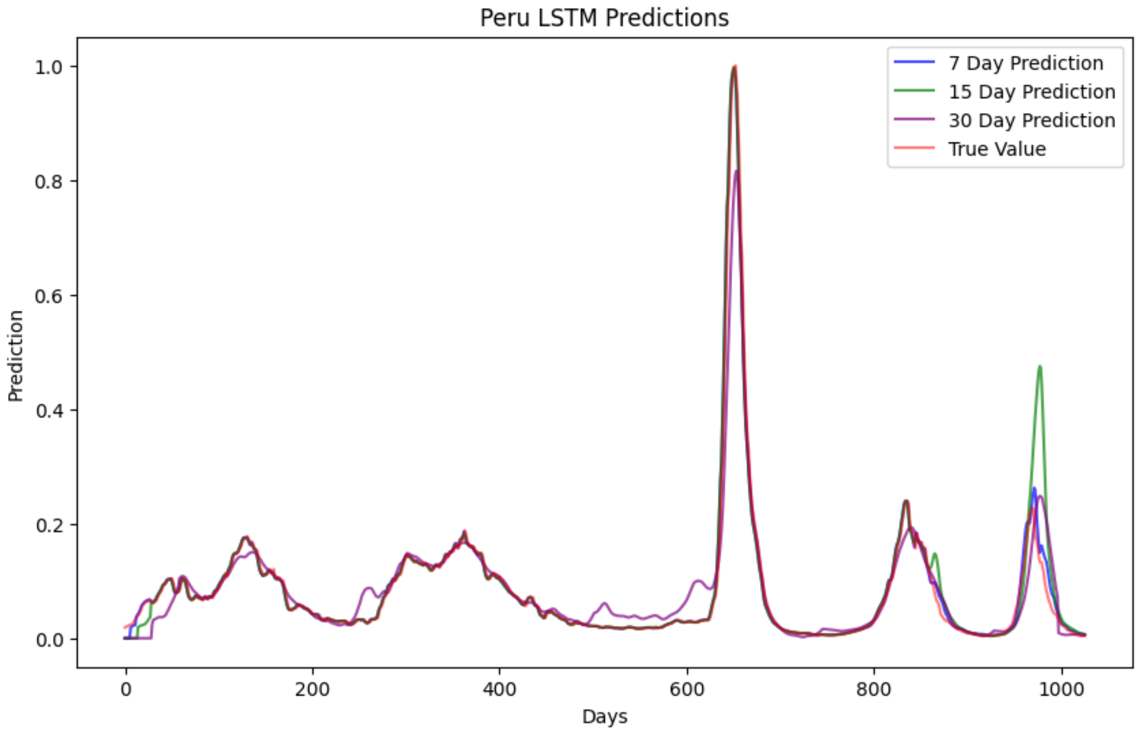

Figure 27: Peru LSTM prediction sequences

5.1.3 GRU Model Results

The results of the GRU architecture and hyperparameter tuning for the 7 day, 15 day, and 30 day prediction sequences are available in Table 11 for both the United States and Peru. Each model uses at least 2 GRU layers and between 1 and 2 activation functions. The United States models contain 2 GRU layers for 7 day predictions, 3 GRU layers for 15 day predictions, and 2 GRU layers for 2 day predictions. The Peru models incorporates 3 GRU layers for all prediction lengths. Only the Peru model forecasting 7 day case counts features a dropout layer.

Dataset	7 Day Prediction Model	15 Day Prediction Model	30 Day Prediction Model
United States	Learning rate 0.001	Learning rate 0.001	Learning rate 0.001
	Input	Input	Input
	GRU Layer with 32 units with ReLU activation	GRU Layer with 128 units with ReLU activation	GRU Layer with 128 units
	GRU Layer with 256 units with sigmoid activation	GRU Layer with 8 units	GRU Layer with 256 units with ReLU activation
	Dense Output Layer	GRU Layer with 128 units with tanh activation	Dense Output Layer
		Dense Output Layer	
Peru	Learning rate 0.001	Learning rate 0.01	Learning rate 0.001
	Input	Input	Input
	GRU Layer with 8 units	GRU Layer with 8 units	GRU Layer with 256 units
	GRU Layer with 64 units with tanh activation	GRU Layer with 32 units with ReLU activation	GRU Layer with 64 units with ReLU activation
	GRU Layer with 128 units with ReLU activation	GRU Layer with 128 units with tanh activation	GRU Layer with 32 units with ReLU activation
	Dropout block with a dropout rate of 0.1	Dense Output Layer	Dense Output Layer
	Dense Output Layer		

Table 11: GRU Optimal Hyperparameters

The RMSE and MAPE for the 7 day, 15 day, and 30 day predictions are available in Table 12 for the GRU models. The RMSE of the United States data was less than half of that for the Peru data for each prediction length. The Peru RMSE is the largest for the 15 day and 30 day prediction sequences with values of 0.042 and 0.054 respectively. The 7 day predictions for both datasets were better with RMSEs of 0.007 for the United States and 0.014 for Peru. The MAPE for the 7 day prediction sequence in the United States was the highest at 157%. The Peru 15 day sequences have the highest MAPE of 113%.

Dataset	Prediction Length (Days)	RMSE	MAPE
United States	7	0.007	157%
	15	0.013	26%
	30	0.036	38%
Peru	7	0.014	14%
	15	0.042	113%
	30	0.054	34%

Table 12: GRU Results

Figure 28 displays the results for the GRU model's predictions in the United States, and Figure 29 presents the same for Peru. A majority of both sets of data predict accurately regardless of prediction length. Inaccuracies occur for the United States 30 day prediction sequence over the first 60 days. The United States 15 day prediction overestimates the data from 880 to 890 days in the dataset. The 15 day prediction in the Peru dataset underestimates the data before day 30, and the 30 day prediction underestimates the data before day 60. All of the prediction sequences overestimate the final peak in the Peru data occurring between 950 and 1000 days into the dataset.

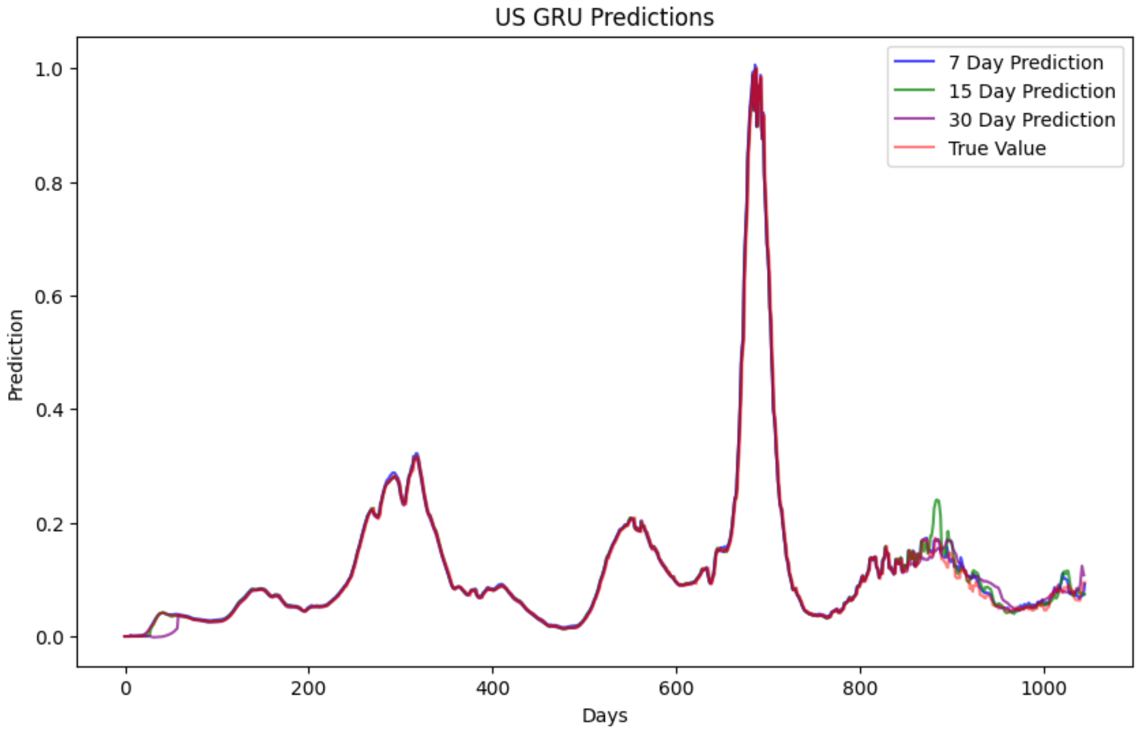

Figure 28: United States GRU prediction sequences

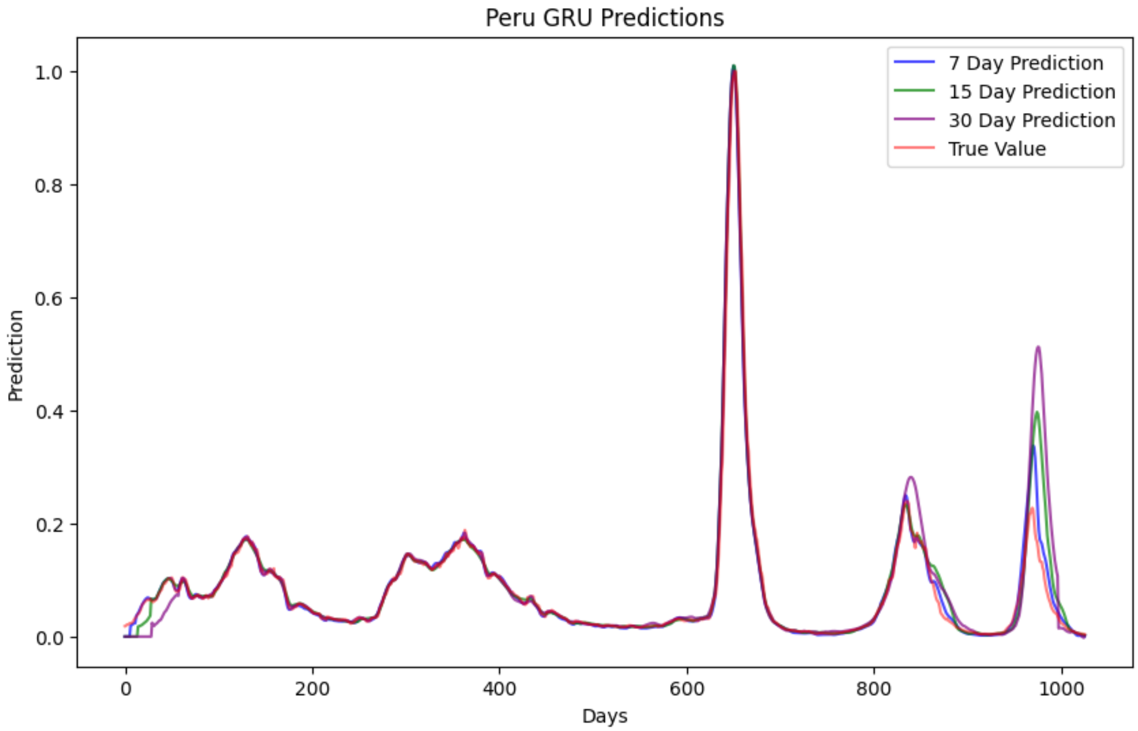

Figure 29: Peru GRU prediction sequences

5.1.4 Hybrid Model Results

Table 13 contains the hybrid model's optimal architecture and hyperparameters for the 7 day, 15 day, and 30 day prediction sequences for the United States and Peru. The optimal models vary between 1 to 3 convolutional layers, 1 to 2 LSTM layers, 0 to 3 GRU layers, and 0 to 3 additional dense layers. The models for the 7 day prediction sequences in the United States contain 2 convolutional layers, 2 LSTM layers, 2 GRU layers, and 3 additional dense layers. The Peru 7 day prediction model contain a convolutional layer, a max pooling layer, and 2 LSTM layers. The model for 15 day predictions in the United States contain 2 convolutional layers, a max pooling layer, 2 LSTM layers, a GRU layer, and 3 dense layers. The model for 15 day predictions in Peru incorporate a convolutional layer, a max pooling layer, 2 LSTM layers, a GRU layer, and 3 additional dense layers. The 30 day prediction models were the deepest for both sets of data. The United States model incorporates 3 convolutional layers, 2 LSTM layers, 3 GRU layers, and 4 additional dense layers. The Peru model contains 1 convolutional layer, 1 LSTM layer, 2 GRU layers, and 3 additional dense layers.

Dataset	7 Day Prediction Model	15 Day Prediction Model	30 Day Prediction Model
United States	Input	Input	Input
	Learning rate 0.0001	Learning rate 0.001	Learning rate 0.001
	Convolutional layer with filter size 256 and kernel size 1	Convolutional layer with filter size 64 and kernel size 15	Convolutional layer with filter size 32 and kernel size 7
	Convolutional layer with filter size 2 and kernel size 1	Convolutional layer with filter size 8 and kernel size 7 with ReLU activation	Convolutional layer with filter size 8 and kernel size 11 with ReLU activation
	Flatten	Max pooling layer with pool size of 4	Convolutional layer with filter size 8 and kernel size 15
	LSTM Layer with 256 units	Flatten	Flatten
	LSTM Layer with 128 units with sigmoid activation	LSTM Layer with 256 units with ReLU activation	LSTM Layer with 128 units with ReLU activation
	GRU Layer with 4 units	LSTM Layer with 8 units	LSTM Layer with 4 units with tanh activation
	GRU Layer with 256 units with sigmoid activation	GRU Layer with 24 units with ReLU activation	GRU Layer with 128 units with sigmoid activation
	Dense Layer with 25 units with ReLU activation	Dense Layer with 100 units	GRU Layer with 4 4units with ReLU activation
	Dense Layer with 75 units	Dense Layer with 50 units with ReLU activation	GRU Layer with 256 units
	Dense Layer with 100 units	Dense Layer with 50 units with ReLU activation	Dense Layer with 75 units with ReLU activation
	Dense Output Layer	Dense Output Layer	Dense Layer with 50 units
			Dense Layer with 75 units
			Dense Layer with 100 units
			Dense Output Layer
Peru	Input	Input	Input
	Learning rate 0.0001	Learning rate 0.001	Learning rate 0.001
	Convolutional layer with filter size 256 and kernel size 5 with tanh activation	Convolutional layer with filter size 8 and kernel size 5	Convolutional layer with filter size 4 and kernel size 1
	Max pooling layer with pool size of 2	Flatten	Flatten
	Flatten	LSTM Layer with 256 units with ReLU activation	LSTM Layer with 8 units with ReLU activation
	LSTM Layer with 128 units with tanh activation	Dense Layer with 100 units with ReLU activation	GRU Layer with 128 units with ReLU activation Dense Output Layer
	LSTM Layer with 64 units with ReLU activation	Dense Layer with 50 units	GRU Layer with 128 units with ReLU activation
	Dense Output Layer	Dense Layer with 25 units	Dense Layer with 25 units with ReLU activation
		Dense Output Layer	Dense Layer with 50 units
			Dense Layer with 25 units
			Dense Output Layer

Table 13: Hybrid Optimal Hyperparameters

The hybrid models predict 7 day, 15 day, and 30 days of case counts well for both the United States and Peru data sets as shown in Table 14. The 7 day sequences have the lowest RMSE, and the 30 day sequences have the highest RMSE for both datasets. The average RMSEs of both datasets is very low with United States having an average value of 0.016 and Peru having an average value of 0.018. The average MAPE of the datasets are 83% and 21% for the United States and Peru respectively.

Dataset	Prediction Length (Days)	RMSE	MAPE
United States	7	0.008	93%
	15	0.010	43%
	30	0.029	114%
Peru	7	0.013	21%
	15	0.018	15%
	30	0.024	28%

Table 14: Hybrid Results

The predictions for daily case count in the United States and Peru are displayed in Figure 30 and Figure 31. The 30 day prediction for the United States and Peru dataset underestimate the actual value for the first 60 days, and the 15 day prediction for the Peru dataset underestimates the first 30 days. The 30 day prediction overestimates the actual value between 875 and 970 days, and the 15 day prediction overestimates the actual value between 910 and 960 days for the United States. The 7 day, 15 day, and 30 day predictions did not accurately predict the peak between 950 and 1000 days in the Peru dataset.

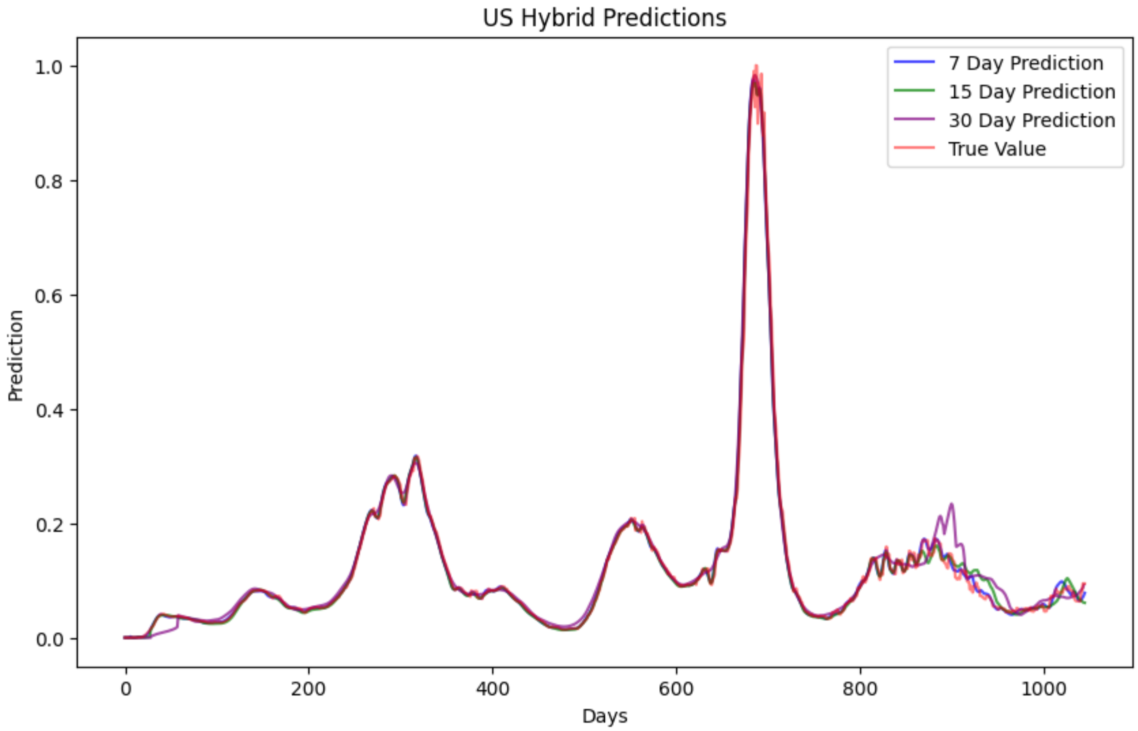

Figure 30: United States Hybrid prediction sequences

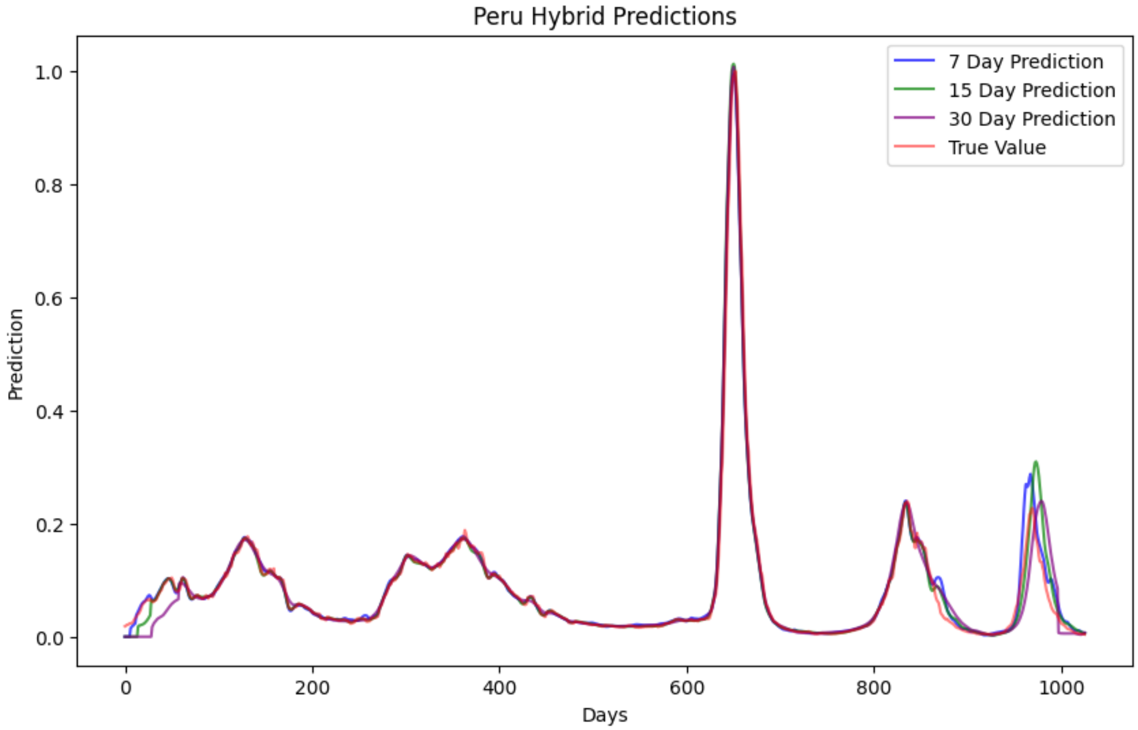

Figure 31: Peru Hybrid prediction sequences

5.1.5 ARIMA Model Results

The *auto_arima* function from the pmdarima package produces optimal parameters of ARIMA(5, 1, 5) for the United States data and ARIMA(2, 0, 1) for the Peru data. These values are varied manually to evaluate if the results are a local minimum or a global minimum. The evaluation occurs by running an ARIMA model of the data for the parameters found using the *auto_arima* function. Next the parameters shift up or down one parameter at a time until a minimum AIC is found such that manually varying the parameters does not produce a lower value. The ARIMA parameters and resulting AIC are displayed in Table 15 for the United States and Table 16 for Peru. The lowest AIC occurs at ARIMA (6, 0, 9) for the United States data and ARIMA (4, 0, 6) for the Peru data, so all forecasting is performed with these values.

Model	AIC
ARIMA(5, 1, 5)	-7789.132
ARIMA(5, 1, 6)	-7851.372
ARIMA(5, 1, 7)	-7878.372
ARIMA(5, 1, 8)	-7888.707
ARIMA(5, 1, 9)	-7883.789
ARIMA(5, 1, 4)	-7791.027
ARIMA(5, 1, 3)	-7393.292
ARIMA(5, 0, 8)	-7899.735
ARIMA(5, 2, 8)	-7847.870
ARIMA(5, 0, 9)	-7910.680
ARIMA(5, 0, 10)	-7910.616
ARIMA(5, 0, 11)	-7905.726
ARIMA(4, 0, 9)	-7912.015
ARIMA(3, 0, 9)	-7791.565
ARIMA(6, 0, 9)	-7912.118
ARIMA(7, 0, 9)	-7908.369
ARIMA(6, 1, 9)	-7893.571
ARIMA(6, 0, 10)	-7910.435
ARIMA(6, 0, 8)	-7897.609

Table 15: United States ARIMA Parameter Exploration

Model	AIC
ARIMA(2,0,1)	-8805.452
ARIMA(2,0,0)	-8725.217
ARIMA(2,0,2)	-8805.764
ARIMA(2,0,3)	-8808.278
ARIMA(2,0,4)	-8855.458
ARIMA(2,0,5)	-8881.491
ARIMA(2,0,6)	-8879.170
ARIMA(2,0,7)	-8883.979
ARIMA(2,0,8)	-8889.841
ARIMA(2,0,9)	-8902.577
ARIMA(2,0,10)	-8896.884
ARIMA(2,0,11)	-8897.219
ARIMA(2,1,9)	-8858.349
ARIMA(2,2,9)	-8783.037
ARIMA(1,0,9)	-8798.359
ARIMA(3,0,9)	-8890.566
ARIMA(4,0,9)	-8890.566
ARIMA(5,0,9)	-8890.824
ARIMA(2,0,11)	-8897.219
ARIMA(2,0,12)	-8896.242
ARIMA(4,0,6)	-8905.130
ARIMA(4,0,7)	-8891.121

Table 16: Peru ARIMA Parameter Exploration

The results of the 7 day, 15 day, and 30 day predictions using ARIMA models on the United States and Peru dataset are displayed in Table 17. The RMSE for both datasets are the lowest for 7 day predictions and the highest for the 30 day predictions. The average RMSE's for the dataset are 0.068 for the United States and 0.081 for Peru. The overall MAPEs were low with an average of 21% for the United States and 36% for Peru.

Dataset	Prediction Length (Days)	RMSE	MAPE
United States	7	0.028	12%
	15	0.058	18%
	30	0.119	34%
Peru	7	0.031	14%
	15	0.073	27%
	30	0.138	66%

Table 17: ARIMA Results

Figure 32 and Figure 33 display the forecasted and actual case count for the United States and Peru datasets. For both figures, the predicted data for any sequence length is very close to the actual value when the data is stationary. All three sequences predict poorly when data spikes or drops swiftly resulting in higher error. The model underestimates and lags behind the actual value when the data changes swiftly. The 30 day prediction models for both datasets underestimate the first 60 days of data. The 15 day prediction model for Peru underestimates the first 30 days of data.

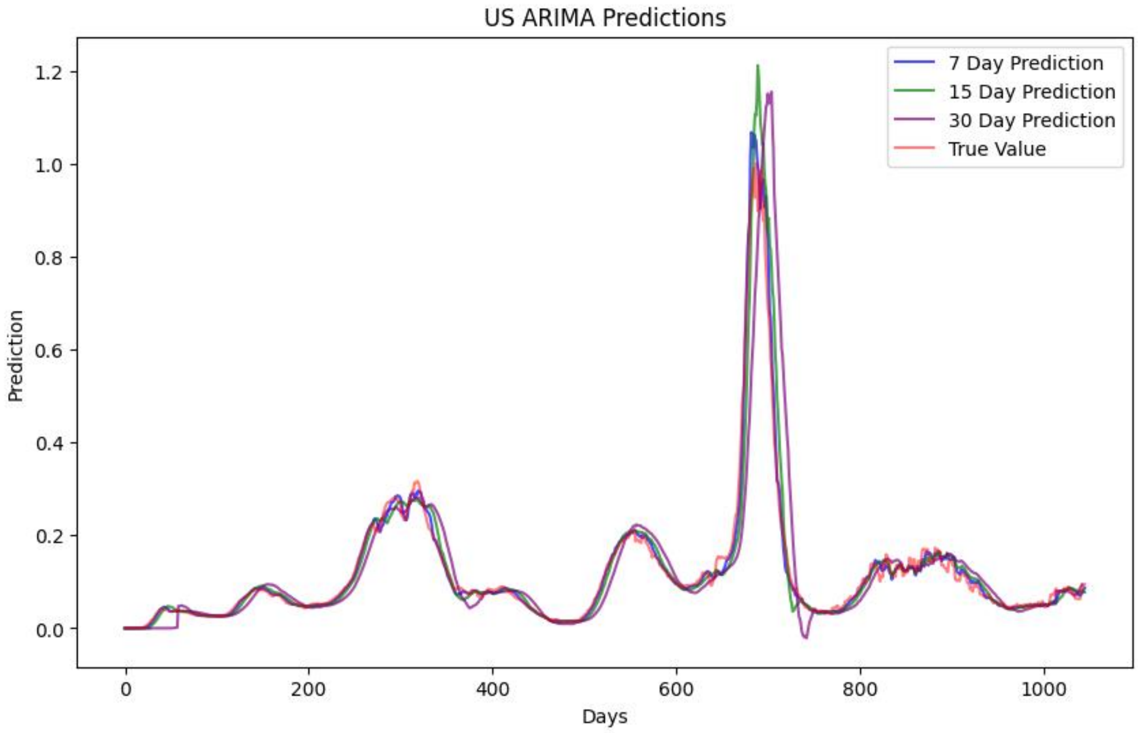

Figure 32: United States ARIMA prediction sequences

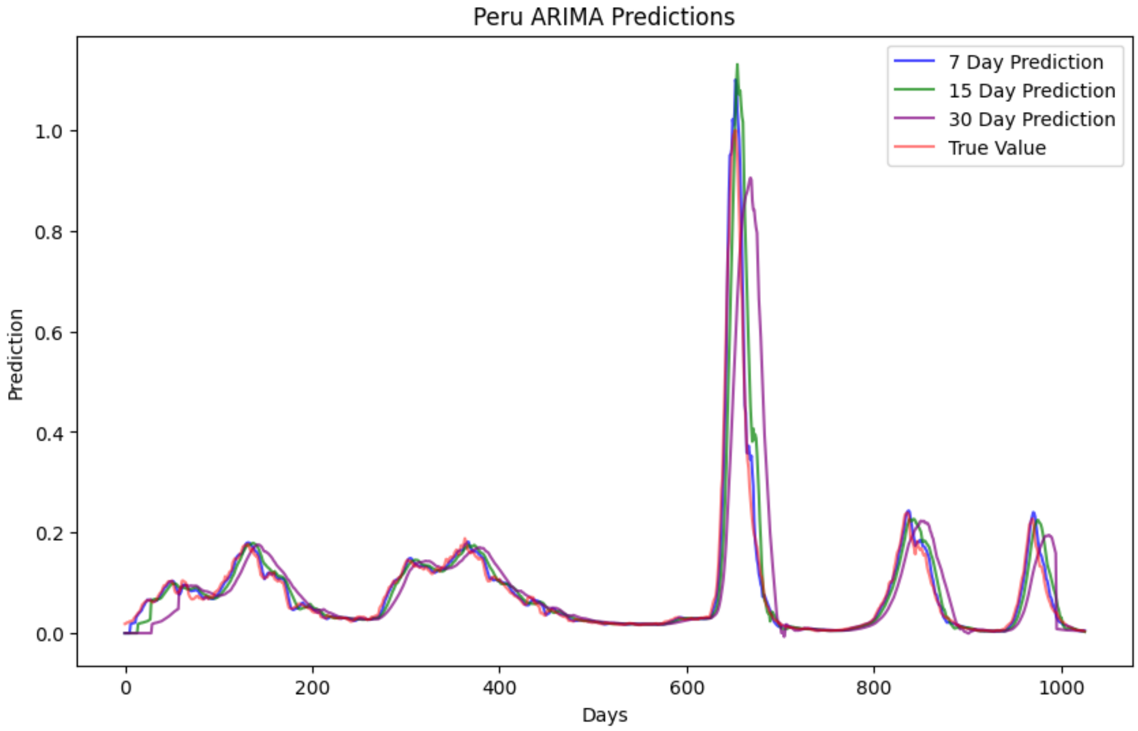

Figure 33: Peru ARIMA prediction sequences

5.1.6 Model Comparisons

Figure 34 and Figure 35 display the RMSE of the predictions for all sequences of models over the United States and Peru data. The CNN, LSTM, and GRU models predict the daily case count well but are indistinguishable for 7 day, 15 day, or 30 day sequences in the United States data. The hybrid United States model performs similarly to the CNN, LSTM, and GRU models for the 7 day prediction sequence, but showcases a lowering in RMSE of between 0.0026 and 0.005 for the15 day sequence and between 0.004 and 0.007 for the 30 day prediction sequence. The ARIMA model's predictions are the least accurate with larger RMSE values for all prediction sequences.

The CNN, LSTM, GRU, and hybrid models forecast the daily case count well for the 7 day prediction sequence, but not for the 15 day and 30 day sequences for the Peru data. The ARIMA model performs worse when predicting 7 day sequences of case count with an increase in RMSE of 0.019 in comparison to the other models. For the 15 day prediction sequence, the GRU model results in an increase of 0.003 RMSE, but the hybrid model performs better with a decrease of 0.007 in RMSE in comparison to the CNN and LSTM models. The Arima model forecasts worse than the other models when predicting 15 day sequences with an increase in RMSE of 0.055 in comparison to the GRU model. The 30 day predicting hybrid model forecasts the best with a RMSE of 0.024 in comparison to the LSTM model with a RMSE of 0.03, the CNN model with a RMSE of 0.035, the GRU model with a RMSE of 0.054, and the ARIMA model with a RMSE of 0.138.

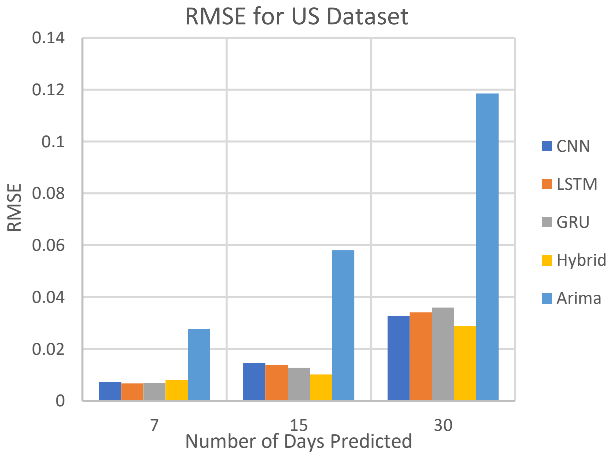

Figure 34: United States Model RMSE

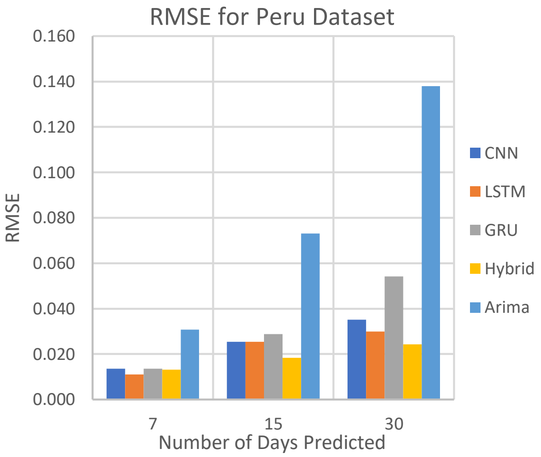

Figure 35: Peru Model RMSE

5.2 Future Application

The 7 days, 15 days, and 30 days immediately following February 7, 2023 for the United States and March 3, 2023 for the Peru data are predicted using each model to confirm their viability in data that was not included in the training set.

5.2.1 7 Day Forecast

Table 18 contains the results of the 7 day prediction from February 8, 2023 through February 14, 2023 for the United States data and from March 4, 2023 to March 10, 2023 for the Peru data. The results are presented graphically along with the actual daily case count in Figure 36 for the United States prediction and Figure 37 for the Peru predictions. All models predict the 7 days sequence well with low error when in comparison to the actual case count in the United States data set. The ARIMA and GRU models tend to overestimate the actual case count, but the

CNN, LSTM, and hybrid models result in better predictions with limited underestimation. The ARIMA and GRU models have the highest error at a RMSE of 0.007, whereas the hybrid model predict the data with an RMSE of 0.004. In contrast, the ARIMA and GRU models have the lowest RMSE of 0.001 for the Peru dataset and MAPEs of 12% and 15%. The hybrid model predictions have the highest error with a RMSE of 0.007 and MAPE of 9%. The CNN, LSTM, hybrid, and GRU models overestimate the actual case count of the Peru dataset whereas the ARIMA model underestimate the value.

Dataset	Model	RMSE	MAPE
United States	CNN	0.006	12%
	LSTM	0.006	11%
	GRU	0.007	15%
	Hybrid	0.004	9%
	ARIMA	0.007	17%
Peru	CNN	0.003	39%
	LSTM	0.004	68%
	GRU	0.001	15%
	Hybrid	0.007	90%
	ARIMA	0.001	12%

Table 18: 7-Day Forecasting Results

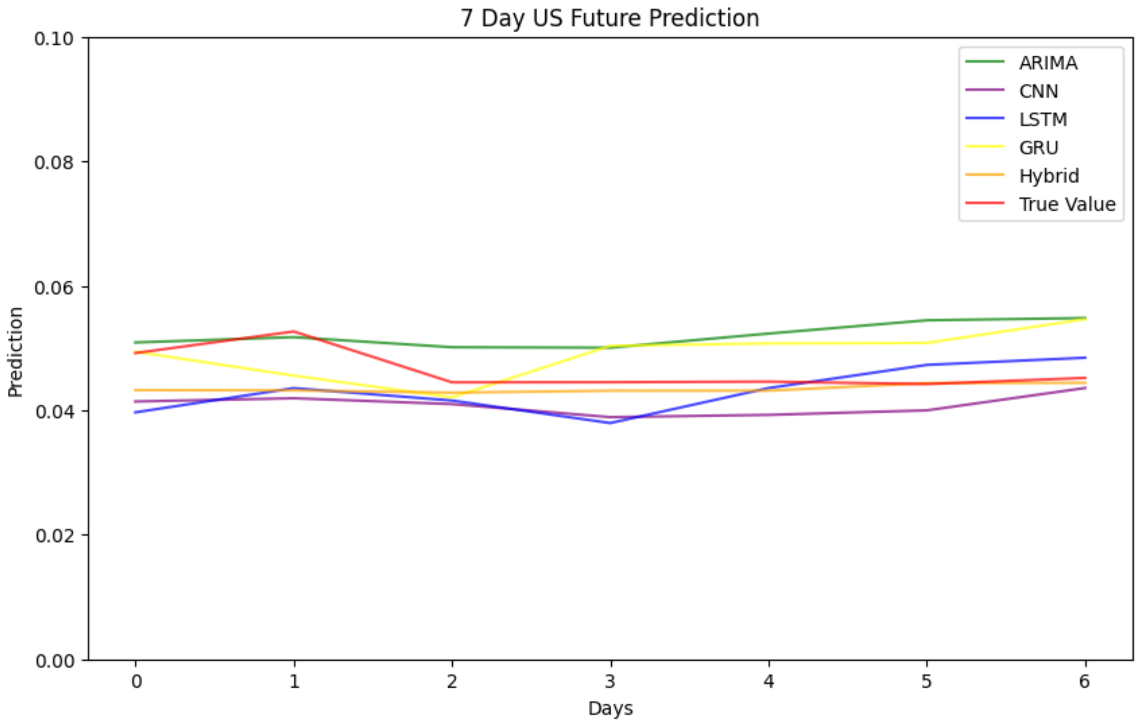

Figure 36: United States 7-Day Forecasting Predictions

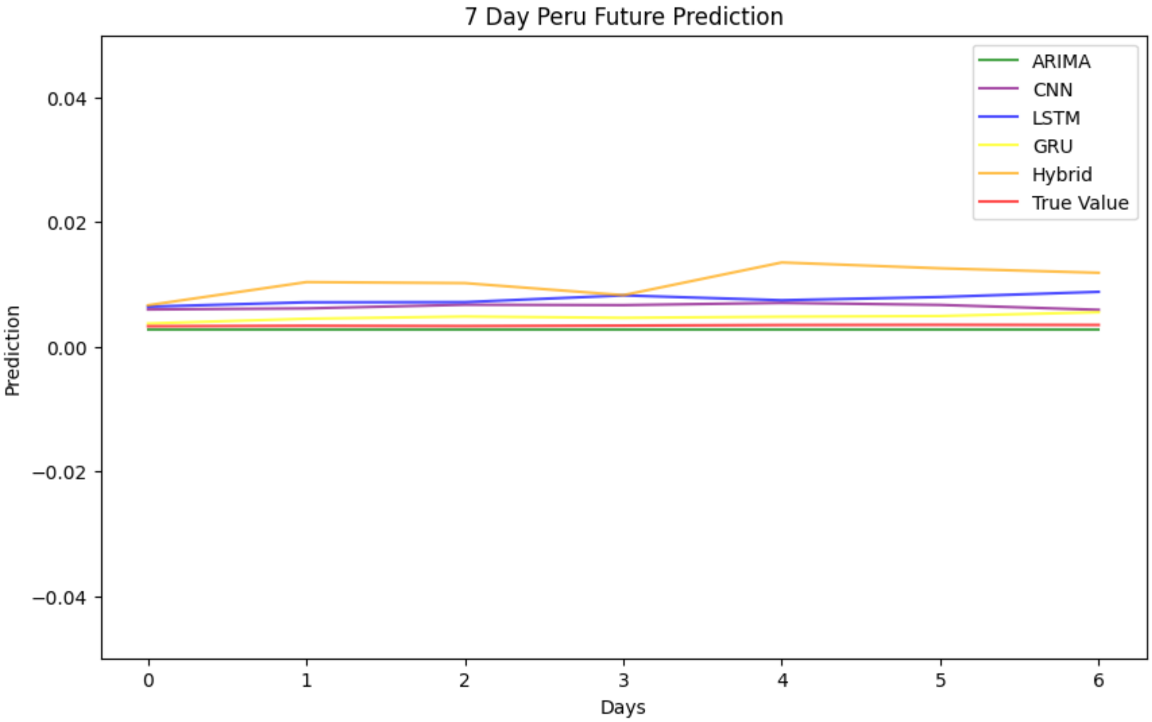

Figure 37: Peru 7-Day Forecasting Predictions

5.2.2 15 Day Forecast

The results of the 15 day prediction immediately following the data utilized in training is displayed in Table 19 where the United States results span February 8, 2023 to February 22, 2023 and the Peru results span March 4, 2023 through March 18, 2023. Figure 38 contains a visualization of the predicted daily case counts and actual case count for the United States. Figure 39 contains the same for Peru.

The CNN and LSTM models result in the worst predictions with overestimation of the daily case count for the 15 day sequence in the United States with RMSE of 0.010 and MAPE of 27%. The GRU and Arima model perform similarly well with RMSEs of 0.009 and 0.007 respectively and MAPEs of 19%. The hybrid model results in the best prediction with a RMSE of 0.003 and MAPE of 7%. Alternatively, the ARIMA model forecasts a case count the closest to the actual case count for the Peru dataset with a RMSE of 0.001 and MAPE of 25%. The hybrid model works well with an RMSE of 0.002 and MAPE of 31%. However, the GRU, LSTM, and CNN models execute less well with RMSEs between 0.004 and 0.008. The LSTM overestimates the actual value, the GRU underestimates the actual value, and the CNN model starts with an overestimation and drops to an underestimation.

Dataset	Model	RMSE	MAPE
United States	CNN	0.010	27%
	LSTM	0.010	27%
	GRU	0.009	19%
	Hybrid	0.003	7%
	ARIMA	0.007	19%
Peru	CNN	0.008	120%
	LSTM	0.004	64%
	GRU	0.004	79%
	Hybrid	0.002	31%
	ARIMA	0.001	25%

Table 19: 15-Day Forecasting Results

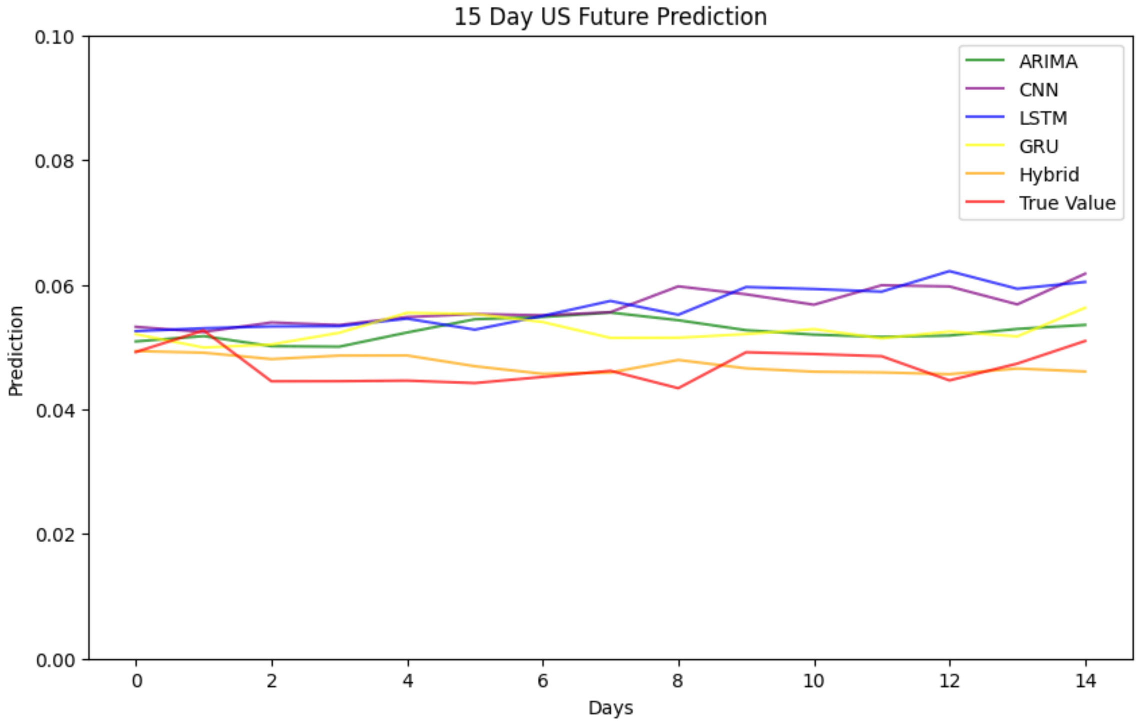

Figure 38: United States 15-Day Forecasting Predictions

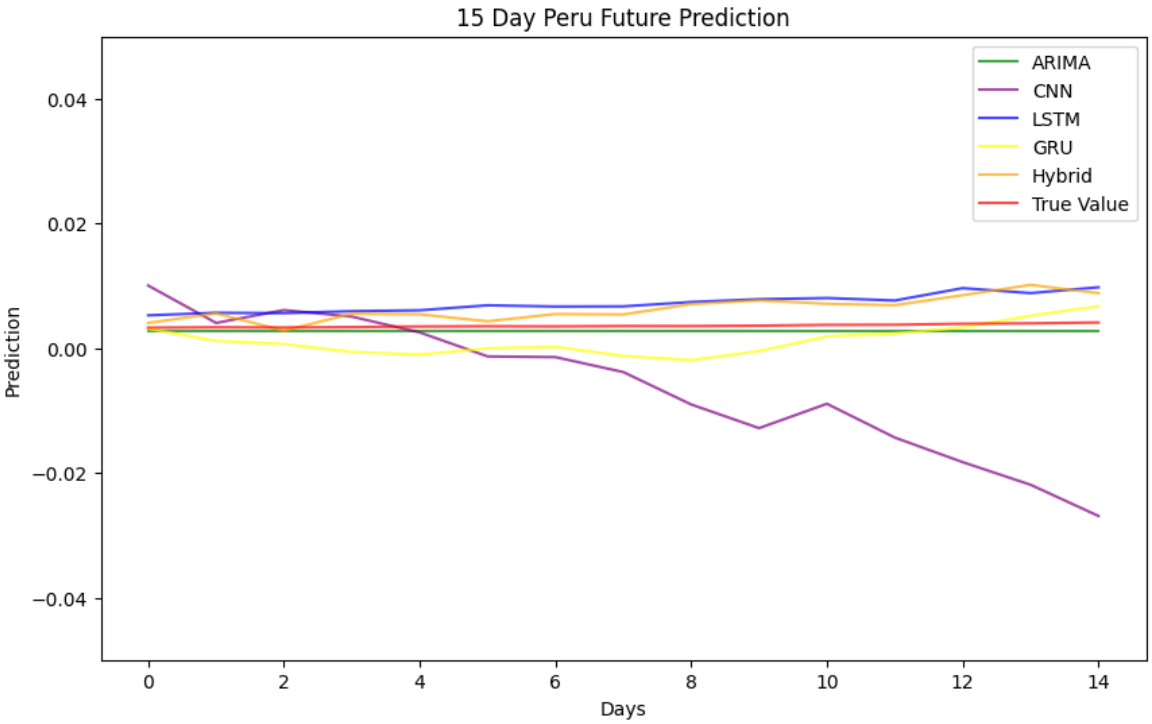

Figure 39: Peru 15-Day Forecasting Predictions

5.2.3 30 Day Forecast

Table 20 presents the results of the 30 day predictions. The United States prediction spans February 8, 2023 through March 9, 2023, and the Peru prediction spans March 4, 2023 to April 2, 2023. Visualizations are created for the United States' and Peru's 30 day predictions and actual case count in Figure 40 and Figure 41 respectively.

The GRU, CNN, and LSTM models produce the poorest results with RMSEs of 0.020, 0.017, and 0.014 with MAPEs of 24%, 24%, and 33%. The ARIMA model performs well with a RMSE of 0.008 and MAPE of 16%. The hybrid model predicted the 30 day sequence with the lowest error with a RMSE of 0.005 and MAPE of 8%. The GRU, CNN, and ARIMA models tend to overestimate the dataset. The LSTM model overestimates the data after 12 prediction days.

In comparison, the ARIMA model predicts the 30 day Peru sequence the best with a RMSE of 0.002 and MAPE of 35%. The CNN, LSTM, GRU, and hybrid models behave similarly with RMSEs ranging from 0.006 to 0.012 and MAPEs ranging from171% to 619%. The LSTM and hybrid model predict an increase in case count resulting in an overestimation of the daily case count.

Dataset	Model	RMSE	MAPE
United States	CNN	0.017	33%
	LSTM	0.014	24%
	GRU	0.020	24%
	Hybrid	0.005	8%
	ARIMA	0.008	16%
Peru	CNN	0.006	171%
	LSTM	0.010	619%
	GRU	0.008	202%
	Hybrid	0.012	249%
	ARIMA	0.002	35%

Table 20: 30-Day Forecasting Results

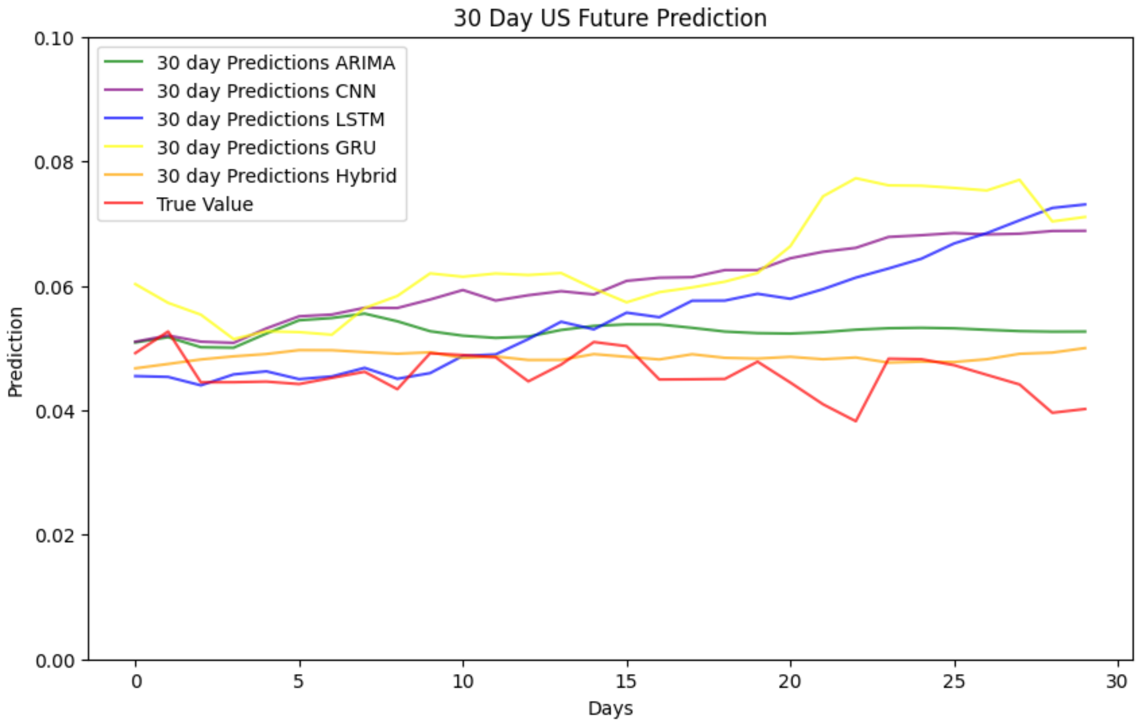

Figure 40: United States 30- Day Forecasting Predictions

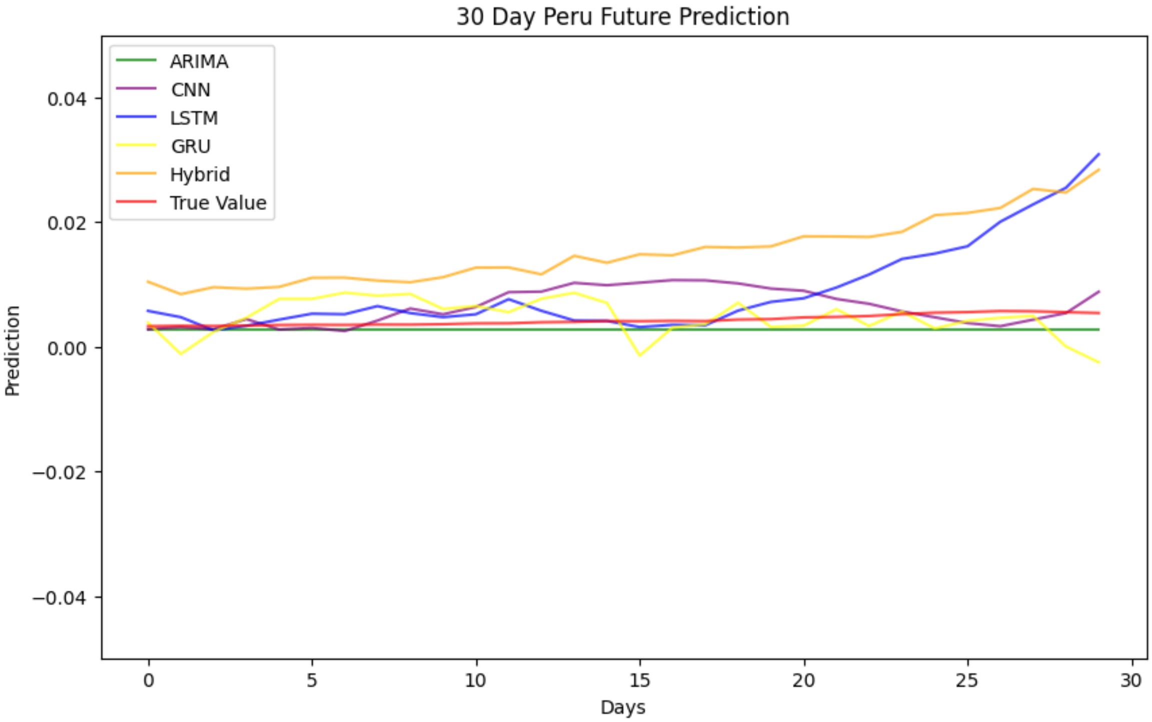

Figure 41: Peru 30- Day Forecasting Predictions

5.2 Discussion

The United States data is modeled accurately over a majority of the dataset, but there are two areas where predictions struggle. All 30-day predictions underestimate the first 60 days of case counts in the United States. The models predicted a flatter case count instead of the increase in cases that was seen. The poor prediction may be the result of poor data quality as there was a lack of testing early in the pandemic so positive cases may have been occurring in the area but were not officially counted. Additionally, the peak in the United States data occurring between May and September 2022 is overestimated by the 30 day CNN, 15 day LSTM, 30 day LSTM, 15 day GRU, and 30 day hybrid predictions. The cause of the overestimation is not obvious, but deeper study into additional variables such as social distancing, vaccinations, and the dominant variant's transmission rate may result in a cause for the smaller and flatter peak in case counts.

Similarly, to the United States results, there were two areas that increased the error of the predictions in the Peru dataset. All of the 15 day models underestimate the first 30 days of data and the 30 day models struggle to predict the first 60 days of data. These underestimations may have a similar cause as discussed for the United States dataset such as lack of accurate testing. The 15 day CNN, 30 day CNN, 15 day LSTM, 30 day LSTM, 15 day GRU, and 30 day GRU models struggled to accurately predict the peaks and plateaus of the data from July 2022 to January 2023. The models either overestimate the peaks or underestimate the plateau value. There may be an underlying change in the behavior of the virus's transmission rate. More exploration needs to occur to find a root cause of the poor model performance in the date range. For both sets of data, the ARIMA models predicts stationary data well but fails to provide the same performance as the other models on the full range of dates studied.

Creating a hybrid model results in improvements in predicting 15 day, and 30 day sequences compared to traditional CNN, LSTM, GRU, or ARIMA models. The hybrid models utilizes CNN as an encoding layer to identify one dimensional patterns in the data. Incorporating LSTM and/or GRU layers allow for temporal learning to be integrated using feedback and feedforward connections. GRU layers are faster than LSTM layers and are less prone to overfitting the data; however, the LSTM layers can be more accurate with longer data sequences. Four of the six hybrid architectures utilized CNN, LSTM, and GRU Layers, whereas the other two hybrid architectures did not utilize GRU layers at all.

Chapter 6: Conclusion

The work explores CNN, LSTM, GRU, hybrid, and ARIMA models to predict COVID-19 case count in the United States and Peru. The models are evaluated for 7 day, 15 day, and 30 day predictions utilizing 30 day case count input sequences. For each model, the forecasting results are displayed visually and presented statistically using RMSE and MAPE. The hybrid models performed equally well or better when forecasting 7, 15, and 30 day case counts for both the United States and Peru. The approach was evaluated on the dataset over which the architecture and hyperparameters were determined and the models were trained over, as well as the 7 days, 15 days, and 30 days that immediately follow.

Even though the recorded case count was imperfect due to limited testing, some manual recording, and variations in the disease spread due to social, health, and governmental interventions, the study demonstrates the ability to predict the case count of the pandemic over the span of January 23, 2020 to March 9, 2023 for the United States and March 6, 2020 to April 2, 2023 for Peru. The ability to accurately model and forecast up to 30 days in the future, provides critical time for governmental and health officials to implement a strategy to respond to the disease by allowing for the implementation of interventions such as requiring masking, reducing public gatherings, closing restaurants, and other restrictions to slow the spread of the disease and providing additional time to prepare hospitals and medical staff for surges in infections.

The forecasting results could be further improved by incorporating additional variables related to the spread of COVID-19 such as social distancing mandates, travel bans, masking, public gathering reduction, and businesses closing. The prediction length can be increased to explore how accurately the models can forecast longer time periods that would allow for additional response time. The parameters for the models can be expanded as the LSTM and GRU models reached the maximum number of layers allowed in their architecture optimization.

References

[1] “Coronavirus disease (covid-19),” World Health Organization, https://www.who.int/news-room/fact-sheets/detail/coronavirus-disease-(covid-19) (accessed Sep. 1, 2023).

[2] “Covid-19 map,” Johns Hopkins Coronavirus Resource Center, https://coronavirus.jhu.edu/map.html (accessed Oct. 13, 2023).

[3] O. Dyer, “Covid-19: Peru’s official death toll triples to become world’s highest,” *BMJ,* 2021. doi:10.1136/bmj.n1442

[4] A. Schwalb and C. Seas, “The covid-19 pandemic in Peru: What went wrong?,” The American Journal of Tropical Medicine and Hygiene, vol. 104, no. 4, pp. 1176–1178, 2021. doi:10.4269/ajtmh.20-1323

[5] D. Pradhan, P. Biswasroy, P. Kumar Naik, G. Ghosh, and G. Rath, “A review of current interventions for COVID-19 prevention,” Archives of Medical Research, vol. 51, no. 5, pp. 363–374, 2020. doi:10.1016/j.arcmed.2020.04.020

[6] “Transmission of covid-19,” European Centre for Disease Prevention and Control, https://www.ecdc.europa.eu/en/infectious-disease-topics/z-disease-list/covid-19/facts/transmission-covid-19#:~:text=Transmissibility%2C%20incubation%20period%2C%20and%20infectivity,six%20days%20for%20earlier%20strains (accessed Sep. 15, 2023).

[7] J. Feng, X. He, Q. Teng, C. Ren, H. Chen, and Y. Li, “Reconstruction of porousmedia from extremely limited information using conditional generative adversarialnetworks,” Physical Review E, vol. 100, no. 3, p. 033308, 2019.

[8] A. Kumar, P. K. Gupta, and A. Srivastava, “A review of modern technologies for tackling COVID-19 pandemic,” *Diabetes &amp; Metabolic Syndrome: Clinical Research &amp; Reviews*, vol. 14, no. 4, pp. 569–573, 2020. doi:10.1016/j.dsx.2020.05.008

[9] V. Iranzo and S. Pérez-González, “Epidemiological models and COVID-19: A comparative view,” *History and Philosophy of the Life Sciences*, vol. 43, no. 3, 2021. doi:10.1007/s40656-021-00457-9

[10] R. Vega, L. Flores, and R. Greiner, “SIMLR: Machine learning inside the SIR model for covid-19 forecasting,” *Forecasting*, vol. 4, no. 1, pp. 72–94, 2022. doi:10.3390/forecast4010005

[11] M. Kreck and E. Scholz, “Back to the roots: A discrete Kermack–McKendrick model adapted to covid-19,” *Bulletin of Mathematical Biology,* vol. 84, no. 4, 2022. doi:10.1007/s11538-022-00994-9

[12] C. Jiménez and M. Merma, “Numerical modelling of coronavirus pandemic in Peru,” *Epidemiologic Methods,* vol. 11, no. s1, 2022. doi:10.1515/em-2020-0026

[13] M. Al-Raeei, M. S. El-Daher, and O. Solieva, “Applying Seir model without vaccination for covid-19 in case of the United States, Russia, the United Kingdom, Brazil, France, and India,” *Epidemiologic Methods,* vol. 10, no. s1, 2021. doi:10.1515/em-2020-0036

[14] R. C. Reiner and R. M. Barber, “Modeling covid-19 scenarios for the United States,” *Nature Medicine,* vol. 27, no. 1, pp. 94–105, 2020. doi:10.1038/s41591-020-1132-9

[15] J. Unterbrink, C. Nicholson, T. Razzaghi, A. Gonzalez, and B. Huamani, “IISE Annual Conference & Expo 2023,” in *Proceedings of the IISE Annual Conference & Expo 2023*

[16] M. Al-Raeei, “The forecasting of covid-19 with mortality using SIRD epidemic model for the United States, Russia, China, and the Syrian Arab Republic,” *AIP Advances,* vol. 10, no. 6, 2020. doi:10.1063/5.0014275

[17] B. K. Mishra *et al.*, “Mathematical model, forecast and analysis on the spread of covid-19,” *Chaos, Solitons &amp; Fractals,* vol. 147, p. 110995, 2021. doi:10.1016/j.chaos.2021.110995

[18] D. Benvenuto, M. Giovanetti, L. Vassallo, S. Angeletti, and M. Ciccozzi, “Application of the Arima model on the COVID-2019 Epidemic Dataset,” *Data in Brief,* vol. 29, p. 105340, 2020. doi:10.1016/j.dib.2020.105340

[19] Z. H. Peng, “The applied research of the time series analysis in the forecasting and early warning of infectious diseases,” *Chinese Journal of Health Statistics,* vol. 27, pp. 459–463.

[20] M. S. D. P. Nayak and K. Narayan, “Forecasting Dengue Fever Incidence Using ARIMA Analysis,” *International Journal of Collaborative Research on Internal Medicine & Public Health,* vol. 11, no. 3, pp. 924–932, 2019.

[21] Q. Liu, X. Liu, B. Jiang, and W. Yang, “Forecasting incidence of hemorrhagic fever with renal syndrome in China using Arima model,” *BMC Infectious Diseases,* vol. 11, no. 1, 2011. doi:10.1186/1471-2334-11-218

[22] L. LIU, R. S. LUAN, F. YIN, X. P. ZHU, and Q. LÜ, “Predicting the incidence of hand, foot and mouth disease in Sichuan province, China using the ARIMA model,” *Epidemiology & Infection,* vol. 144, no. 1, pp. 144–151, 2016.

[23] D. A. Cordova Sotomayor and F. B. Santa Maria Carlos, “Application of the integrated autoregressive method of moving averages for the analysis of series of cases of covid-19 in

Peru," *Revista de la Facultad de Medicina Humana*, vol. 21, no. 1, pp. 65–74, 2021. doi:10.25176/rfmh.v21i1.3307

[24] R. K. Singh *et al.*, "Prediction of the COVID-19 pandemic for the top 15 affected countries: Advanced Autoregressive Integrated moving average (ARIMA) model," *JMIR Public Health and Surveillance*, vol. 6, no. 2, 2020. doi:10.2196/19115

[25] S. Abolmaali and S. Shirzaei, *Forecasting covid-19 number of cases by implementing Arima and Sarima with grid search in United States*, 2021. doi:10.1101/2021.05.29.21258041

[26] Z. Xia, L. Qin, Z. Ning, and X. Zhang, "Deep learning time series prediction models in surveillance data of hepatitis incidence in China," *PLOS ONE*, vol. 17, no. 4, 2022. doi:10.1371/journal.pone.0265660

[27] Y. Wu, Y. Yang, H. Nishiura, and M. Saitoh, "Deep learning for epidemiological predictions," *The 41st International ACM SIGIR Conference on Research &amp; Development in Information Retrieval*, 2018. doi:10.1145/3209978.3210077

[28] P. Wang, X. Zheng, G. Ai, D. Liu, and B. Zhu, "Time series prediction for the epidemic trends of covid-19 using the improved LSTM Deep Learning Method: Case studies in Russia, Peru and Iran," *Chaos, Solitons &amp; Fractals*, vol. 140, p. 110214, 2020. doi:10.1016/j.chaos.2020.110214

[29] K. E. ArunKumar, D. V. Kalaga, Ch. M. Kumar, M. Kawaji, and T. M. Brenza, "Forecasting of COVID-19 using deep layer recurrent neural networks (RNNS) with gated recurrent units (grus) and long short-term memory (LSTM) cells," *Chaos, Solitons &amp; Fractals*, vol. 146, p. 110861, 2021. doi:10.1016/j.chaos.2021.110861

[30] K. E. ArunKumar, D. V. Kalaga, Ch. Mohan Sai Kumar, M. Kawaji, and T. M. Brenza, "Comparative analysis of gated recurrent units (GRU), long short-term memory (LSTM) cells, autoregressive integrated moving average (ARIMA), Seasonal Autoregressive Integrated moving average (SARIMA) for forecasting COVID-19 trends," *Alexandria Engineering Journal*, vol. 61, no. 10, pp. 7585–7603, 2022. doi:10.1016/j.aej.2022.01.011

[31] L. Aguilar I., M. Ibáñez-Reluz, J. C. Z. Aguilar, E. W. Zavaleta-Aguilar, and L. A. Aguilar, "Forecasting sars-COV-2 in the Peruvian regions: A deep learning approach using temporal convolutional neural networks," *Selecciones Matemáticas*, vol. 8, no. 1, pp. 12–26, 2021. doi:10.17268/sel.mat.2021.01.02

[32] "Coronavirus disease (covid-19)," World Health Organization, https://www.who.int/news-room/fact-sheets/detail/coronavirus-disease-(covid-19) (accessed Sep. 1, 2023).

[33] Covid 19 en el perú - ministerio del salud, https://covid19.minsa.gob.pe/sala_situacional.asp (accessed Sep. 1, 2023).

[34] Z. Yu, P. Keskinocak, L. N. Steimle, and I. Yildirim, “The impact of testing capacity and compliance with isolation on covid-19: A mathematical modeling study,” AJPM Focus, vol. 1, no. 1, p. 100006, 2022. doi:10.1016/j.focus.2022.100006

[35] He JL, Luo L, Luo ZD, et al. Diagnostic performance between CT and initial real-time RT-PCR for clinically suspected 2019 coronavirus disease (COVID-19) patients outside Wuhan, China. Respir Med. 2020;168:105980. doi:10.1016/j.rmed.2020.105980

[36] Jegerlehner S, Suter-Riniker F, Jent P, Bittel P, Nagler M. Diagnostic accuracy of a SARS-CoV-2 rapid antigen test in real-life clinical settings. Int J Infect Dis. 2021 Aug;109:118-122. doi: 10.1016/j.ijid.2021.07.010. Epub 2021 Jul 7. PMID: 34242764; PMCID: PMC8260496.

[37] Pray IW, Ford L, Cole D, et al. Performance of an Antigen-Based Test for Asymptomatic and Symptomatic SARS-CoV-2 Testing at Two University Campuses — Wisconsin, September–October 2020. MMWR Morb Mortal Wkly Rep 2021;69:1642–1647. DOI: http://dx.doi.org/10.15585/mmwr.mm695152a3

[38] B. Fraser, “Covid-19 strains remote regions of Peru,” The Lancet, vol. 395, no. 10238, p. 1684, 2020. doi:10.1016/s0140-6736(20)31236-8

[39] Shang, W., Kang, L., Cao, G., Wang, Y., Gao, P., Liu, J., & Liu, M. (2022). Percentage of asymptomatic infections among SARS-COV-2 omicron variant-positive individuals: A systematic review and meta-analysis. Vaccines, 10(7), 1049. https://doi.org/10.3390/vaccines10071049

[40] P. Herrera-Añazco et al., “Some lessons that Peru did not learn before the second wave of Covid-19,” The International Journal of Health Planning and Management, vol. 36, no. 3, pp. 995–998, 2021. doi:10.1002/hpm.3135

[41] B. Fraser, “Covid-19 strains remote regions of Peru,” The Lancet, vol. 395, no. 10238, p. 1684, 2020. doi:10.1016/s0140-6736(20)31236-8

[42] K. Thelwell, “6 facts about Peru’s healthcare system,” The Borgen Project, https://borgenproject.org/6-facts-about-perus-healthcare-system/ (accessed Nov. 12, 2023).

[43] Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., & Talwalkar, A. (2017). Hyperband: A novel bandit-based approach to hyperparameter optimization. The journal of machine learning research, 18(1), 6765-6816.

[44] Hyndman, R.J., & Athanasopoulos, G. (2018) Forecasting: principles and practice, 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2. Accessed on Sep. 8, 2023

[45] “PMDARIMA,” PyPI, https://pypi.org/project/pmdarima/ (accessed Oct. 19, 2023).