FORECASTING COVID-19 CASE COUNTS USING DEEP LEARNING

by

Rinda Venkata Krishna Hemanth Digamarthi

Thesis
submitted in partial fulfillment of the requirements for
the Degree of Master of Science(Computer Science)

Acadia University
Spring Convocation 2022

This thesis by Rinda Venkata Krishna Hemanth Digamarthi was defended successfully in an examination on April 14, 2022.

The examining committee for the thesis defense was:

Dr. Michael Robertson, Chair

Dr. Pawan Lingras, External Examiner

Dr. Andrew McIntyre, Internal Examiner

Dr. Daniel L. Silver, Supervisor

Dr. Darcy Benoit, Head

Contents

List of Tables

List of Figures

Abstract

Provincial health care systems are in need of forecasting models to combat viral infectious diseases such as COVID-19, which can estimate changes in the number of cases in advance. This information can be used to alert policymakers and health care management to understand the changes in a pandemic and to estimate the future demand for health services. This study uses machine learning and deep learning methods to determine the most important variables responsible for the transmission of the virus and to predict daily or 7-day moving average case counts up to one week in advance.

Data was collected and prepared into daily records containing the number of new COVID case counts, demographic data on those found positive with COVID, outdoor weather variables, indoor environmental variables; and human movement data based on cellular mobility, and public health care restriction data. Inductive Decision Tree (IDT) models were developed and tested to analyze variable importance and to determine the typical manner in which these variables interacted to predict daily case counts. Indoor relative humidity was found to be an important factor not typically considered in such studies.

Predictive models were then developed using two deep neural network approaches: Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) neural networks. A 5-fold chronological cross-validation approach was used to develop these models. The best LSTM models forecasted tomorrow's daily COVID case counts with 90.7% accuracy, and the 7-day rolling average COVID case counts with 98.1% accuracy using the test set data. We also developed LSTM models to forecast the next 7 days of daily COVID case counts. The best models were able to predict the Daily case counts of the test set with a mean accuracy of 90.5% over all days with the lowest accuracy of 91.4% on the first day predict. Models forecasting the 7-day rolling average case counts had a mean accuracy of 92.5%, and 94.3%

on first day (D+1) on the same test set.

Definitions of Abbreviations and Symbols

- COVID-19 - Corona Virus Disease 2019
- SARS-CoV-2 - Severe Acute Respiratory Syndrome - Corona Virus - 2
- IDT - Inductive Decision Trees
- ARIMA - Auto Regressive Integrated Moving Averages
- ANN - Artificial Neural Networks
- RNN - Recurrent Neural Networks
- LSTM - Long Short Term Memory Networks
- CNN - Convolutional Neural Networks
- RH - Relative Humidity
- IRH - Indoor Relative Humidity
- IAT - Indoor Air Temperature
- STL - Single Task Learning
- MTL - Multi Task Learning
- MAE - Mean Absolute Error

- MAPE - Mean Absolute Percentage Error
- API - Application Programming Interface
- WEKA - Waikato Environment for Knowledge Analysis
- CUDA - Compute Unified Device Architecture
- cuDNN - CUDA Deep Neural Network Library
- ReLU - Rectified Linear Unit

Acknowledgements

First and foremost, I would like to thank my supervisor, Dr. Daniel Silver, for giving me the opportunity to work on this interesting and ongoing topic, and for showing extraordinary patience with me during the writing of this thesis. He motivated me with his immense guidance throughout my masters degree and in completion of this research. His expertise and insightful feedback was invaluable in formulating the research questions and methodology, which pushed me to sharpen my thinking and brought my work to a higher level. I am proud of, and grateful for, my time working with Dr.Silver.

I would like to thank Dr. Yigit Aydede of Saint Mary's University for his leadership and statistical guidance as Principal Investigator; Mr. Ray MacNeil from CLARI for bringing the team from multiple universities together and for his counsel and meeting administration; Dr. Mutlu Yuksel of Dalhousie University for his feedback and collegial interactions over the course of this project. Additionally, I would like to thank NS COVID 19 Health Research Coalition for providing the financial support for this project.

I also would like to thank Acadia University and staff of Jodrey School of Computer Science for their administrative and in-kind support.

I also wish to acknowledge the support of my friends and family, in particular, my mother, Lakshmi Andal; my father, Srinivas; my sister, Harika; my uncle, Koteswar Rao; my grandmother, Parvathi Devi; and my grandfather, Sankar Narayana. They were the main source of encouragement and emotional support to keep me moving forward.

Chapter 1

Introduction

This research was completed as part of joint work by Acadia University, St. Mary's University, and Dalhousie University. The larger project entitled "The role of environmental determinants and social mobility in viral infection transmission in Halifax" was managed by Dr. Yigit Aydede of Saint Mary's University (SMU) and funded by the Nova Scotia COVID-19 Health Research Coalition, awarded on April 29, 2020. The researchers from SMU and Dalhousie pursued advanced statistical approaches to understand the relationships between various independent factors and the dependent COVID-19 case count variable. We develop a forecasting model to predict the number of COVID-19 case counts or an average number of case counts up to 7 days into the future given the weather, mobility, and calendar data to determine important variables for COVID-19 transmission in the regions of Ontario.

1.1 Problem

Epidemiological forecasting is an important tool for provincial health care systems to combat viral infectious diseases such as COVID-19. These models can estimate the changes in the number of COVID-19 cases in advance to alert policymakers and health care management. This information can be used to understand changes in a pandemic and to estimate the future demand for health services.

The original intention was for the joint team to work with COVID-19 data from the Province of Nova Scotia captured by 811 Call Center system and COVID-19 testing triage

system. Unfortunately, this data was not available three months following the award of funding, and negotiations with the province to obtain such continued for several more months. Subsequently, the joint team agreed to proceed with data from 32 counties in Ontario. The joint team acquired Ontario COVID 19 data and associated weather and mobility data from Facebook. The SMU and Dalhousie researchers finally accessed Province of Nova Scotia Data in the spring of 2021, but Acadia completed its work with the Ontario data. This proved to be beneficial to forecasting as the Ontario data was a much larger and richer dataset than Nova Scotia.

1.2 Objective

The main objective of this research is to develop machine learning models that can predict the number of COVID-19 cases per day and one to seven days into the future. To make this possible, recent weather and mobility data are needed, which are helpful to determine the important factors that affect the transmission of the COVID-19 virus. Our approach is as follows:

- Collect and prepare data on daily COVID cases, outdoor environmental conditions, indoor environmental conditions, and human movement based on cellular mobility and public health restrictions.
- Use descriptive statistics and statistical modeling methods to determine the most important independent variables for the transmission of the virus; and
- Use deep learning techniques such as LSTMs and CNNs to forecast the next seven days of daily COVID-19 case counts.

1.3 Scope

We will be using data from the Toronto area of Ontario, Canada, specifically in four districts of Toronto, Peel, York, and Durham for this research. From March 1st, 2020, to December 31st, 2020, these four Ontario districts have the highest number of cases compared to other

districts, with approximately two-thirds of all the cases. This data is merged with the weather, mobility, and calendar data from the same time period.

1.4 Thesis Structure

This thesis report consists of six chapters. Chapter 1 introduces the problem and its scope, along with the objectives. Chapter 2 presents the background knowledge, such as the related studies on epidemiological forecasting and the traditional methods used. Furthermore, the decision trees, ARIMA, artificial neural networks, convolutional neural networks, and the long-short term memory networks are discussed to understand the proposed theory and experiments in the research. Chapter 3 illustrates the theory and the approaches used for data collection and preparation. It also provides information about the implementation of model development, including the software libraries and the hardware used. Chapter 4 describes the data analysis of all the variables, and we select the variables for model development. Chapter 5 presents empirical studies of all the experiments along with the result summaries and discussion. Chapter 6 summarizes the findings and discusses the contributions of this research and the future work.

Chapter 2

Background

In this chapter, relevant theories and concepts are provided as background for developing our theory and approach. This includes the fundamentals of epidemiology and prior work on forecasting viral infections using statistical and machine learning-based algorithms. Section 2.1 discusses related work on traditional methods of forecasting viral infections. Section 2.2 introduces Machine Learning, Inductive Decision Trees which we will use for variable selection and artificial neural networks. Section 2.3 discusses time series modeling methods such as ARIMA, Recurrent Neural Networks, Long Short-Term Memory networks and Convolutional Neural Networks. Section 2.4 describes the evaluation metrics used in this research.

2.1 Forecasting Viral Infections

Epidemiological modeling and forecasting are important activities used in strategic planning to draft health policies and to fight against the virus. Statistical methods such as Autoregressive Integrated Moving Average (ARIMA), Least Absolute Shrinkage and Selection Operators (LASSO), Multivariate Adaptive Regression Splines (MARS) have been applied to epidemiological forecasting given time-series data from Dallas County, Texas, USA [1]. The data used in their research consists of three parts:

1. Influenza case data, which contains the number of cases of Influenza-like illnesses (ILI) of all the hospitals present in Dallas.

2. Google search data, which consists of weekly data on normalized search frequencies about influenza of the following 7 terms: cold, cough, fever, flu, H3N2, influenza, and sore throat.

3. Weather data, which consists of variables such as temperature, dew point, humidity, and pressure to forecast Influenza-like illnesses. The data ranges from the first week of October 2011 to the last week of April 2018 [1].

For an epidemiological forecasting problem, the daily or 7-day moving average case count is provided as the target value along with input attributes such as weather, mobility, calendar and public health restrictions. To evaluate the performance of the predictive model, the predicted case counts are compared with the actual case counts using Mean Absolute Error (MAE). The best deep learning models which predict influenza case data using google search frequencies and weather data have an MAE of 4.14 and 4.89 for one week ahead and two weeks ahead of forecasting, respectively [1]. The experimental results show that the deep learning models are more accurate than the traditional statistical approaches.

Similarly, deep learning and conventional statistical approaches have been used to forecast influenza virus transmission in Hong Kong [2]. Unlike in the USA, Hong Kong is an area where influenza has no constant seasonal trend. Three types of data have been used in this research [2]:

1. Influenza data taken from the Hong Kong Center for Health Protection which consists of the case data for influenza-like illnesses from 50 outpatient clinics.

2. Google search data consisting of the normalized frequencies of 13 search terms (cough, cold, flu, H3N2, H7N9, Avian flu, fever, and the remaining 6 terms in the Chinese language).

3. Weather data including daily air pressure, absolute minimum temperature, mean dew point, mean relative humidity, mean amount of cloud, rainfall, and hours of sunlight.

The above data was collected from the Hong Kong Observatory. The data covers from the first week of 2011 to the third week of 2016. The period from week 1 of 2011 to week 3 of 2014 consisting of a 104-week window was considered as the training data and the

remaining data from week 4 of 2014 to week 1 of 2016 are used as test and validation sets. Four individual models such as Generalized Linear Model (GLM), Least Absolute Shrinkage and Selection Operator (LASSO), Autoregressive Integrated Moving Average (ARIMA), and Deep Learning (DL) with Feedforward Neural Networks (FNN) are used to forecast Influenza like illnesses both one week and two weeks in advance.

A statistical method known as Bayesian Model Averaging (BMA) is used to integrate multiple forecast scenarios from all the developed models. The results show that for one-week ahead forecasting, the fused model using BMA has an MAE of 1.23 and for two-week ahead forecasting, the deep learning model has an MAE of 26.

2.1.1 Forecasting Viral Infections using Weather data

The correlation between weather and COVID-19 pandemic is analyzed for Jakarta, Indonesia, and reported in [3]. COVID-19 case data and weather data have been obtained from the health Ministry of Jakarta and the Meteorological department of Jakarta. Weather data such as temperature, humidity, and rainfall are used. The Spearman-rank correlation has been used to compute correlations of the data. The results conclude that only average temperature significantly correlates with the transmission of the virus.

For understanding the impact of weather on the transmission of the virus, the study uses infection data from 3,739 distinct locations. The data has been augmented with case counts data reported by China, Australia, Canada, and the United States [4]. The weather data consists of daily data for maximum and minimum temperatures, humidity, precipitation, snowfall, moon illumination, sunlight hours, ultraviolet index, cloud cover, wind speed and direction, pressure as well as the air pollutants including ozone(O3), nitrogen dioxide, sulphur dioxide and particulate matter. The data ranges from Dec 2019 to Apr 2020.

To understand the pandemic spread level, the effective reproductive number R_0 is estimated in this research [4]. The reproduction number (R_0) is one of the most fundamental and often used metrics for the study of the way a disease spreads. It is the number of cases directly caused by an infected individual throughout his infectious period. The epidemiologists calculate R_0 value using contact tracing data obtained at the onset of the pandemic. A pandemic grows if the value of R_0 is greater than one, and it ends if the R_0 value becomes

less than 1. The current R_0 value of COVID-19 is estimated to be between 2.5 to 3.0. The results conclude a negative relationship between the temperature and the humidity variables to the transmission of the virus.

Along with temperature, humidity also has a profound effect on COVID-19 transmission [5]. The daily number of new cases, cured cases, and deaths have been manually extracted from publicly released World Health Organization COVID-19 reports from the official health departments of countries covering 1,112 sub-national regions in 57 countries and 124 countries as of May 31, 2020. Statistical analysis such as the Multivariate regression model has been used to explore the weather conditions' effect on the transmission. The meteorological data containing daily average temperature and relative humidity have been taken from European Center for Medium level forecasts. Other socioeconomic variables such as Gross Regional Product (GRP), elevation data, labor age, and school-age were taken from the World Development Index Database, Altimeter Corrected Elevations dataset, and Gridded Population of the world (GPW), respectively. The results showed that higher temperature and higher humidity could significantly reduce daily new cases. The average daily temperature is significantly and negatively correlated with daily new cases. For an average incubation period of six days of the virus, every one degree Celsius increase in the daily average temperature results in a 2.88% decrease in new daily cases. Also, every one percent increase in relative humidity with a 6-day lag causes a 0.19% decrease in new daily cases [5].

2.1.2 Forecasting Viral Infections using Mobility data

Mobility data can be considered as the measure of public compliance to government-induced policies and can act as a mediator between the policies and the spread of the disease. A recent study investigated the impact of seasonal changes, weather abnormalities and public health interventions such as closures of public places, schools, and private gatherings on COVID-19 case counts [6]. The data is from 144 geopolitical areas worldwide, extracted from online count dashboards of Johns Hopkins University. The atmospheric attributes used are temperature and relative humidity, taken from publicly available meteorological websites. The statistical method used in this study is Weighted Random Effects Regression which determines the association between the log rate ratio of the COVID-19 virus [6]. It

was concluded that only the area-wide public health interventions consistently correlated with the changes in COVID case counts. It is found that seasonality plays only a minor role.

Information about the COVID-19 prevalence data of 135 countries from the European Center for Disease Prevention and Control (ECDC) is taken to better understand how the lockdown policies affect the daily incidence of the COVID-19 and mobility patterns. Mobility data was taken from Google Community Mobility reports [7]. Non-pharmaceutical intervention data was collected from the Oxford COVID-19 government response tracker for most countries and consists of public policies corresponding to mitigating the propagation of the virus. The paper concludes that cancellation of public events and enforcing restrictions on the gatherings has the most significant effect on reducing the pandemic [7]. School closures, as well as the stay-at-home requirements, help in reducing the incidence of new infections.

2.1.3 Impact of Indoor Environmental Conditions on Viral Transmission

The transmission routes of COVID-19 are still debated. However, recent evidence strongly suggests that COVID-19 could be transmitted via air in poorly ventilated places. Relative indoor humidity has a profound impact on the airborne transmission of the COVID-19 virus [8].

Dry indoor air has a significant role in disease transmission and resident health. In cold winters, drawing outdoor air inside and heating it up to a comfortable temperature makes the indoor air less humid. Relative humidity between 40% to 60% is considered optimal. Relative humidity levels below 40% have been found to improve the lifetime of some viruses on hard surfaces and are more likely to be transmitted in the air. Infectious droplets from coughing and sneezing are transmitted to others by being suspended in the air. At low humidity, water will quickly evaporate from the droplets, making them lighter and easier to remain in suspension. If the relative humidity is high, the droplets are more likely to fall to the ground instead of being inhaled by another host. Therefore, to reduce the risk of infection, it is suggested that relative humidity should be maintained above 40% for indoor environments such as hospitals, offices, and public transports to minimize the airborne spread of COVID-19 [9].

2.2 Machine Learning

Machine Learning is a subset of Artificial Intelligence. It is a study of mathematical algorithms that computers use to optimize a given algorithm which improves automatically without any human intervention by relying on example data [10]. The models built using the Machine Learning algorithms make the decisions based on the training data, by identifying relevant patterns in it rather than being explicitly programmed. Recently, with the increase in fields such as big data, large sources of data and computational resources are readily available to apply machine learning in many use cases such as time series forecasting, image recognition, speech recognition, medical imaging, and recommendation systems.

Machine Learning can be divided into three main categories: (1) Supervised learning; (2) Unsupervised learning; and (3) Reinforcement learning.

Supervised Learning: Supervised learning is a type of machine learning in which the algorithms are made to model the relationships between the target outputs and input features [11]. The model is trained until it can understand the underlying patterns in the given training data to accurately predict unseen data in the test set.

Unsupervised Learning: Unsupervised learning is a type of machine learning in which it uses machine learning algorithms to determine the hidden patterns in unlabeled datasets. Unsupervised learning models are utilized for three main tasks: clustering, association, and dimensionality reduction.

Reinforcement Learning: Reinforcement learning is the training of machine learning models to make a sequence of decisions, based on a policy. The agent learns the policy to maximize its total reward in an uncertain and potentially complex environment [12]. In reinforcement learning, an agent faces a game-like situation. The computer employs trial and error to develop the policy for decision making. To get the machine to do what the programmer wants, the agent receives either rewards or penalties for the actions it performs.

2.2.1 Supervised Learning

Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs. It infers a function from labeled training data consisting of a set of training examples. A supervised learning algorithm analyzes

the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for the algorithm to correctly determine the class labels for unseen examples [47].

In supervised learning, the training set contains pairs of input variables and output variables, the goal is to learn a mapping function *f(x)* between input space X to target output space Y,

$$f(x) : X \rightarrow Y \tag{2.1}$$

where, *h(x)* is a hypothesis function that approximates the true mapping function *f(x)*. A supervised learning algorithm attempts to develop an *h(x)* $\approx$ *f(x)*.

When a supervised learning model is trained, it will look for a underlying relationship between the input x and target output y to predict the value of y, given the knowledge of input value x. This algorithm uses the error between the function and target function to determine the quality of the developed model, also known as the generalization error. The difference between *h(x)* and *f(x)* is called the generalization error, e which is used to determine the quality of the developed model.

$$e = l(f(x) \text{ - } h(x)) \tag{2.2}$$

where, l is the loss function, and has the MSE = $(f(x) - h(x))^2$

Supervised learning can be further divided into two types, classification and regression. Classification uses an algorithm to accurately assign test data into specific categories. It recognizes specific entities within the dataset and attempts to draw some conclusions on how those entities should be labeled or defined. Regression is used to understand the relationship between dependent and independent variables. It is typically used for forecasting problems, where the output is real valued. Linear regression is a type of regression which uis used to find the relationship between dependent and independent variable, illustrated by equation 2.3.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + ... + \beta_n n_1 + \epsilon_i \tag{2.3}$$

where, β_0 is called intercept, β_1 is the coefficient of x_1 and β_2 is the coefficient of x_2.

2.2.2 IDT for data modeling and attribute selection

Inductive Decision Trees (IDTs) are non-parametric supervised learning methods used for classification and regression [13]. The main intention of using IDTs in this research is to understand which input variables are most important in the transmission of COVID-19. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. The M5P trees are used in this research for regression modeling.

M5P Trees

Wang and Witten proposed the M5P trees in 1997. They reimplement Quinlan's M5 algorithm with modifications to generate more compact and accurate models. The M5P algorithm creates an unpruned decision tree by recursively splitting the instance space to reduce the subset variation to develop a decision tree for regression. The data is preprocessed so that all the splits in the M5P tree are binary. After the tree is developed, linear regression models are built from the data at every leaf node. The M5P turns inner nodes into a leaf node with regression planes if the expected error of a subtree is more significant than the estimated error for the linear model at its root. It is also known as post pruning the tree. The M5P tree uses the linear regressions at the leaves to estimate an output value for new instances. Thus, M5P trees develop piece with models consisiting of a series of linear models.

2.2.3 Artificial Neural Networks

Artificial Neural Networks (ANNs) are a type of machine learning algorithm and are at the heart of deep learning. Their name and structure are inspired by the human nervous system, mimicking how biological neurons process information [14]. The main building block of an ANN is the artificial neuron. Figure 2.1 shows the typical structure of an artificial neuron, with input variables x_1, x_2, x_3, ... x_n each with corresponding weights w_{1j}, w_{2j}, w_{3j}, ... w_{nj}. Every connection has a weight which can be either positive or negative. The neuron also has a bias value x_0 or $b = 1$, associated with a weight of w_0. An activation function determines the output y by computing the sum of products of the weights and its input node. The output of the summation of all the received neuron signals multiplied with each

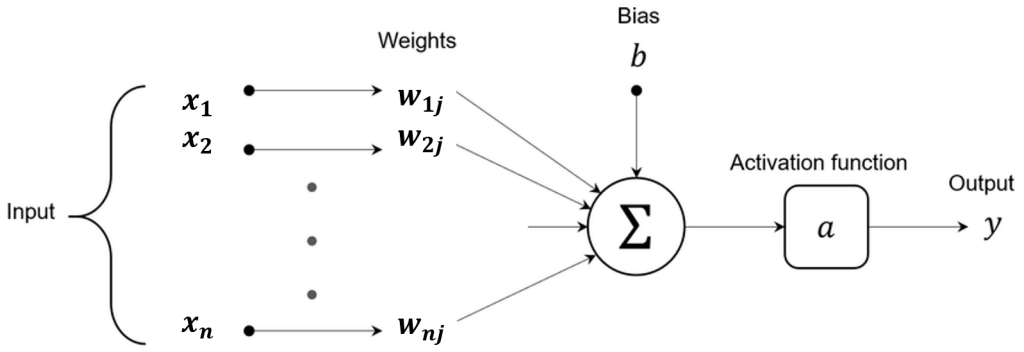


Figure 2.1: Structure of a Artificial Neuron [15].

of their associated weights is pass through an activation function to obtain the final output y.

$$y_j = \sigma(\sum_{i=1}^{N} w_{ij}x_i + b_j) \tag{2.4}$$

The above equation 2.4 shows the calculation of the output neuron y_j, x_i and b_j are input and bias respectively, and w_{ij} represents the weights between the nodes i and j. The function σ is called the activation function.

2.2.4 Multi-Layer Perceptron

A Multi-layer perceptron (MLP) is a type of Artificial Neural Network in which there are multiple layers of perceptrons. Figure 2.2 shows an architecture of an MLP. An MLP consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron which uses a non-linear activation function. MLP becomes more capable of learning more complex problems as the number of hidden nodes and hidden layers are increased [16].

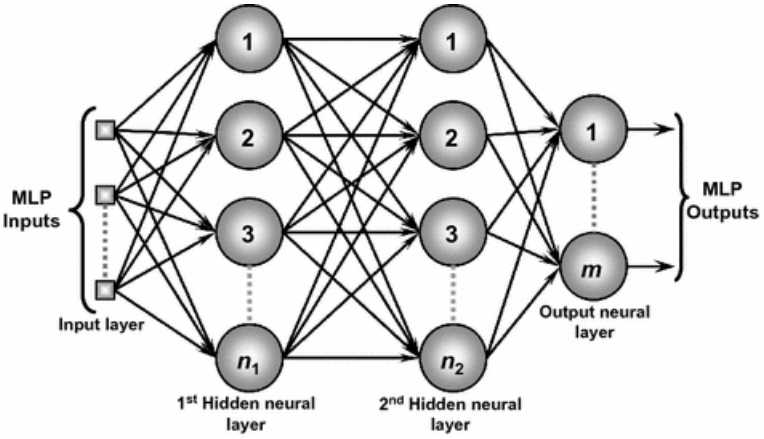


Figure 2.2: Structure of a Multi-layer perceptron.

2.2.5 Back-propagation

Back-propagation is a widely used method for calculating derivatives inside deep feedforward neural networks. It forms an important part of a number of supervised learning algorithms for training feedforward neural networks, such as stochastic gradient descent. When a neural network is trained using gradient descent, a loss function is calculated, which represents how far the network's predictions are from the true labels [17].

Backpropagation allows us to calculate the gradient of the loss function with respect to each of the weights of the network. This enables every weight to be updated individually to gradually reduce the loss function over many training iterations. It uses the error between target output values and predicted values to update the weights and eventually minimizes the error as close to zero as possible. The error is sent back through the neural network to update all weights by computing the gradient of error with respect to each weight. It does so by using the chain rule of calculus and passes backwards one layer from the previous layer starting from the output layer.

When a neural network has a target output t_j and a predicted output y_j, the total error of the neural network, and the predicted output y_j can be measured using the following MSE equation:

$$Error = \frac{1}{2}\sum_{j}(t_j - y_j)^2 \tag{2.5}$$

$$y_j = \sigma(\sum_{i=0}^{n} w_{ij} x_i) \tag{2.6}$$

where x_i is the input, σ, is the activation function and w_{ij} is the corresponding weight.

The back-propagation algorithm uses the gradient descent method to find the optimal set of weights that minimize the error. It calculates the derivative of the error with respect to its weight. It can be defined as:

$$\Delta w_{ij} = -\eta \frac{\delta Error}{\delta w_{ij}} \tag{2.7}$$

2.2.6 Activation Functions

Activation functions are functions used in a neural network to compute the weighted sum of inputs and biases. It is used to decide whether a neuron can be activated. It manipulates the presented data and produces an output for the neural network that contains the parameters in the data. Without the activation functions, the neural networks are just a linear regression model. These functions are the ones where the non-linearity computations are carried out, making the network capable of learning and performing more complex tasks [48].

Linear

It is a straight-line function where the activation is proportional to the input, i.e., the weighted sum from the neurons. Figure 2.3 depicts a linear function.

$$f(x) = x \tag{2.8}$$

Sigmoid

Sigmoid is a nonlinear activation function. It is also known as the logistic function. Its curve looks like a S-shape, as shown in Figure 2.4 as it exists between 0 to 1. The sigmoid activation function is majorly used in regression and classifcation problems. It is denoted by ϕ.

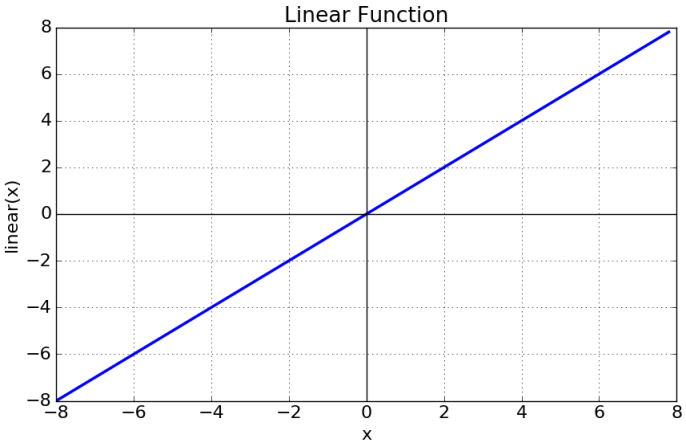


Figure 2.3: A Linear activation function.

$$\phi(z) = \frac{1}{1 + e^{-z}} \tag{2.9}$$

Tanh

Tanh is a nonlinear activation function. It is very similar to the sigmoid activation function. As shown in Figure 2.5, It takes any real value as input and outputs values in the range of -1 to 1. The more larger the input, the more closer the output value will be to 1.0, whereas smaller the input, the closer the output will be to -1.0.

$$\sigma(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} \tag{2.10}$$

Rectified Linear Unit (ReLU)

Rectified Linear Unit is a type of activation function most commonly used in deep learning models. As shown in Figure 2.6, the function returns 0 if it receives any negative input, and

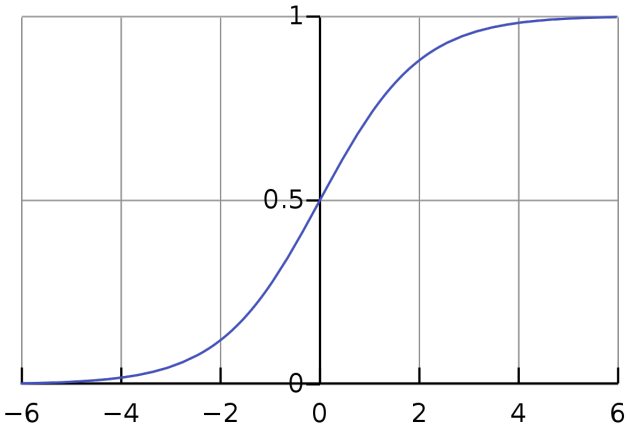


Figure 2.4: A Sigmoid activation function.

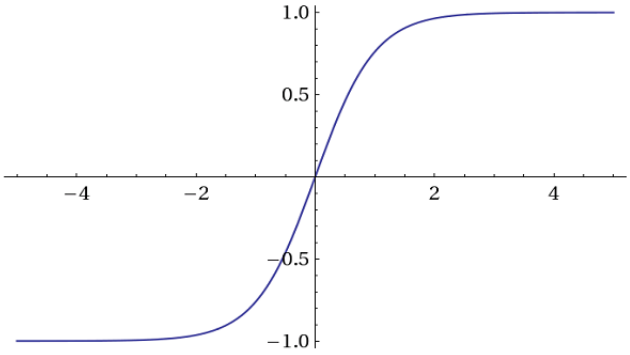


Figure 2.5: A tanh activation function.

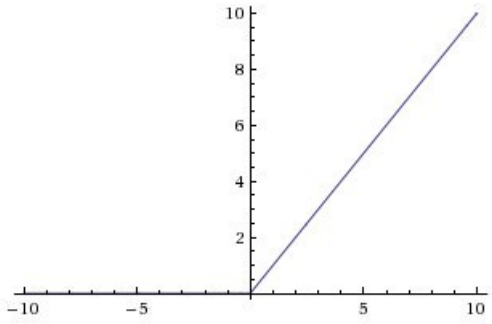


Figure 2.6: A ReLU activation function.

for any positive value, it returns the same value.

$$R(z) = max(0, z) \tag{2.11}$$

2.3 Time Series Modeling Methods

A time-series can be defined as a sequence where the metrics are recorded over regular time intervals. Time-series data can be categorized depending on the frequency considered. It can be yearly, quarterly, monthly, weekly, daily, hourly, in minutes, or in seconds.

Time series data can be found in economics, social sciences, finance, epidemiology, and the physical sciences. The time-series data adds one or more variables, providing time-related information to the examples and subsequently to the predictive models.

2.3.1 ARIMA and Persistence Models

Auto-Regressive Integrated Moving Average (ARIMA) is a statistical method that can capture a suite of different standard temporal structures in time series data and can be used for forecasting a target variable into the future [18]. ARIMA stands for:

- **Auto Regression:** It is a linear regression model which uses the dependent relationship between a given observation and a number of lagged observations.

- **Integrated:** In order to make the time series stationary, the differencing of the raw observations is used, i.e., subtracting one observation from an observation that is present at the previous time step. Differencing is transforming a non-stationary time series into a stationary one. This is an important step in preparing data for the ARIMA model. The best way to determine if the series is sufficiently differenced is to plot the differenced series and check for a consistent mean and variance over the time series.

- **Moving Average:** It incorporates the dependency between an observation and a residual error from a moving average model applied to lagged observations.

The ARIMA model uses 3 terms: p, d, and q

- p is the order of the Auto Regressive term. It refers to the number of lags of the output variable to be used as predictors. The output at time t is a linear combination of past outputs.

$$y_t = c + \epsilon_t + \sum_{i=1}^{P} \varphi_{t-i} y_{t-i} \tag{2.12}$$

- q is the order of the Moving Average term. It refers to the number of lagged forecast errors that can go into the ARIMA model. The output at time t is a linear combination of past forecast errors.

$$y_t = c + \epsilon_t + \sum_{i=1}^{q} \theta_{t-i} \epsilon_{t-i} \tag{2.13}$$

- d is the value of the differencing that is required to make the time series stationary. To determine if a given time series data is sufficiently differenced or not, the differenced series should be plotted and constant mean and variance should be checked.

- The values of AR and MA can be combined to form the general Autoregressive Integrated Moving Average (ARIMA) model:

$$d_t = c + \epsilon_t + \sum_{i=1}^{P} \varphi_{t-i} y_{t-i} + \sum_{i=1}^{q} \theta_{t-i} \epsilon_{t-i} \tag{2.14}$$

Although ARIMA investigates the linear relationship between variables from past observations, it assumes that there is a linear correlation structure in a time series [49]. Most

of the real-world data examples like time series forecasting are often non-linear in nature. To overcome the limitation of ARIMA not dealing with time-series data with non-linear patterns, we will use more sophisticated neural network methods for making the prediction. Recently, recurrent neural networks have been shown to provide state-of-the-art results in forecasting problems with little or no feature engineering. The recurrent neural networks have the capability to deal with non-linear time series data and support multivariate input and output data as well.

2.3.2 Recurrent Neural Networks (RNN)

A recurrent neural network (RNN) is a type of artificial neural network which uses sequential or time-series data. These deep learning algorithms are commonly used for ordinal or temporal problems, such as time series forecasting, language translation, and speech recognition. They are distinguished by their memory as they take information from prior inputs to influence the current input and output [19].

Unlike traditional deep neural networks, where the inputs and outputs are independent of each other, the output of recurrent neural networks depends on the prior elements within the sequence. The information from the previous timesteps is returned which is used as an additional input. Figure 2.7 shows a unrolled RNN with a input, hidden and output layer. At a given time step t, recurrent neural network receives the input x_t and the output h_{t-1} from the previous time step and generates an output value $h_t = \mathrm{A}(x_t,h_{t-1})$. This sequence like architecture makes recurrent neural networks suitable for time series data.

In theory, recurrent neural networks can handle long sequences, while in fact they fail to connect information from a previous step to the current step when the gap between them is too large. The backward propagated error over time in a RNN becomes very small to affect the appropriate change in the weight of the network. Therefore, the RNN discards the information from the earlier time steps when moving through the later steps, resulting in the loss of important information. This loss of information is generally known as the vanishing gradient problem [20].

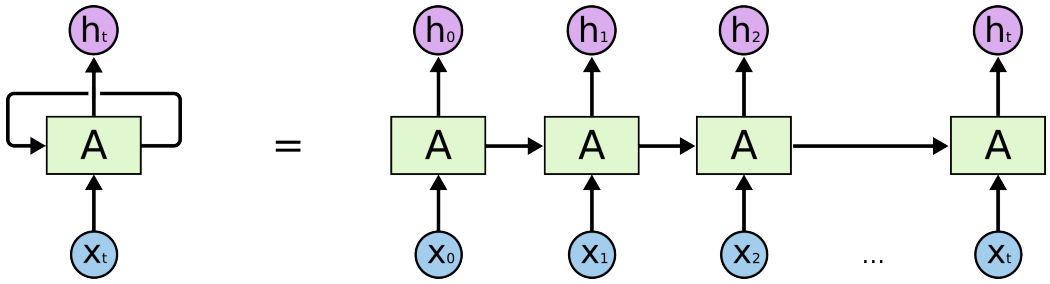


Figure 2.7: An unrolled recurrent neural network representing all the nodes [21].

2.3.3 Long Short-Term Memory Networks (LSTM)

Long Short-Term Memory networks are a special kind of RNN, which are explicitly designed to avoid the problem of long-term dependencies [21]. Each LSTM node is a complex neuron called a memory cell, as shown in Figure 2.8. The memory cell is responsible for removing information from the cell state and adding new information to the cell state. This is done through three mechanisms, known as the forget gate, input gate and output gate.

- **Step 1:** In the first step, a sigmoid layer called the forget gate decides which information to remove from the cell state. Using the values of h_{t-1} and x_t , the value of the forget gate output is calculated as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2.15}$$

  Where, h_{t-1} is the output from the previous time step, x_t is the current input, b_f is bias, and W_f is the weight.

- **Step 2:** In the next step, the input gate decides which information will be used to update the cell state. A sigmoid layer is used to decide which values to update and a tanh layer known as $\tilde{C}_t$ is created to be added to the state.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2.16}$$

$$\tilde{C}_t = tanh(w_C \cdot [h_{t-1}, x_t] + b_C) \tag{2.17}$$

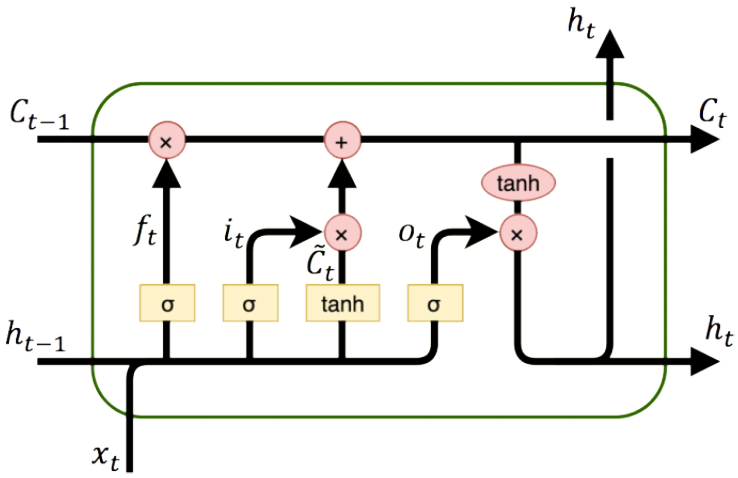


Figure 2.8: LSTM architecture [22].

.

- **Step 3:** To update the cell state, the forget gate is applied to the previous state C_{t-1} and a new term $i_t * \tilde{C}_t$ is added. The new cell state C_t is:

$$C_t = C_{t-1} * f_t + i_t * \tilde{C}_t \tag{2.18}$$

- **Step 4:** Lastly, the output gate decides the output value h_t based on the new cell state C_t. An output gate can be defined as a function o_t shown in equation 2.19. The new state C_t is passed through a *tanh* layer and multiplied by the output gate to get the output value h_t.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{2.19}$$

$$h_t = o_t * tanh(C_t) \tag{2.20}$$

2.3.4 Convolutional Neural Networks (CNN)

Convolutional neural networks are one of the most successful deep learning algorithms. They have proven effective on challenging computer vision problems such as identifying and localizing objects in images and automatically describing the content of images. Although

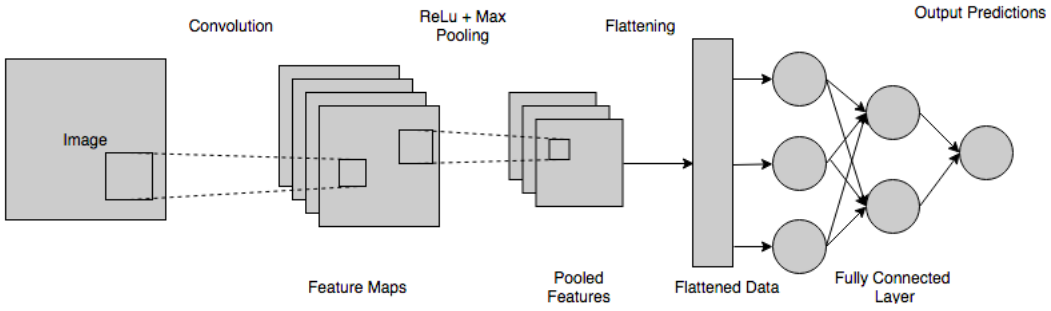


Figure 2.9: Convolutional Neural Network.

originally developed for two-dimensional image data, CNNs can be used to solve multivariate and multistep time series forecasting problems. Multivariate time series are datasets with a temporal ordering comprised of more than one observed variables for each time step. The model must learn to predict the next value in a sequence based on a series of recent observations. As shown in the Figure 2.9, a CNN can be defined as a combination of basic building blocks known as a convolutional layer, a pooling layer, dropout layer, and a fully connected layer.

Convolution Layer

This layer is the essential component of feature engineering. The convolutional layer reads an input, such as a 2D image or a 1D signal using a kernel of filter that reads in small segments and steps across the entire input field. Each kernel read results in an interpretation of the input projected onto a feature map and represents an interpretation of the input.

The kernel can be defined as a matrix that moves over the input data performs the convolutional operation. Typically for the image data, the kernel extracts the features from the images. In the case of multivariate time series data, a one-dimensional convolution filter slides vertically over an input matrix of size n x d, where n is the number of variables and d is the number of days in the time series sequence. Figure 2.10 shows the input time series data on the left as a matrix of size n x d. The red represents a kernel that is considered a 1 x d value window that moves through each day and it takes the sequence of d days to predict the output value for the next day. The convolutional kernel has the same width as the number of variables. And, the length of the kernel is the number of training examples it considers before

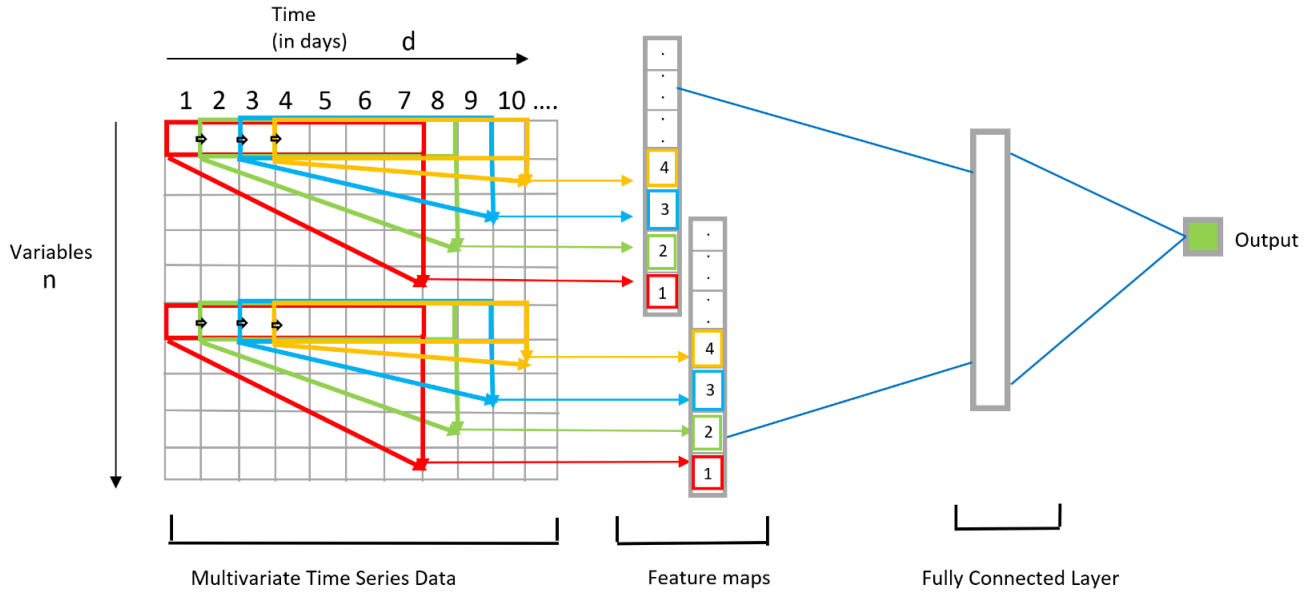


Figure 2.10: A convolutional kernel shown as a temporal window.

producing an output. The kernel moves along the time series data performing a convolution operation for a certain number of steps before generating an output. The output from kernel in each convolution position creates a feature map that acts as input to the next layer.

Pooling Layer

A pooling layer, illustrated in Figure 2.11 is function which is used to reduce the dimensionality of a feature map representation. Pooling involves selecting a subsampling operation, to be applied to feature maps. The size of the pooling operation or filter is smaller than the size of the feature map; specifically, it is typically 2 x 2 pixels applied with a stride of two pixels, it moves two pixels to the right as it scans the feature map.

There are two common functions used in the pooling operation. They are: Average Pooling and Max Pooling. Average pooling calculates the average value for each patch on the feature map, whereas Max pooling calculates the maximum value for each patch of the feature map. The result of using a pooling layer and creating down sampled or pooled feature maps is a summarized version of the features detected in the input.

The pooling layers take the feature map projections and distill them into the essential

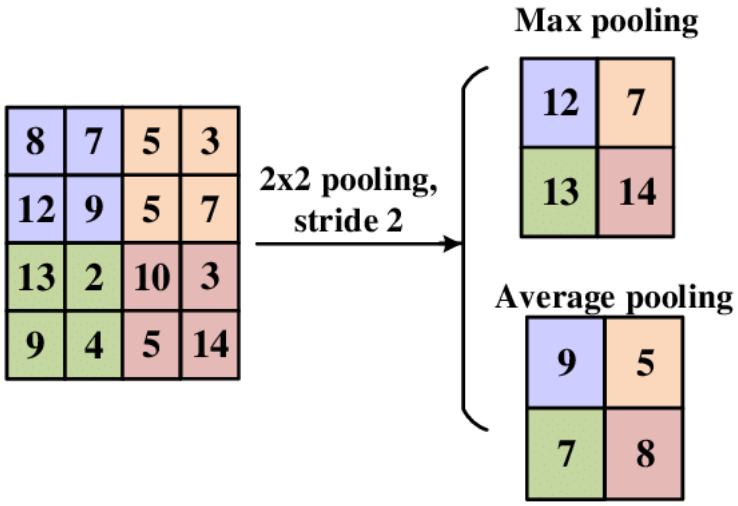


Figure 2.11: Pooling Layer in CNN [23].

elements. The convolutional and pooling layers can be repeated at depth, providing multiple layers of abstraction of the input signals. The output of these networks is often one or more fully-connected layers.

Dropout Layer

Dropout can be defined as a technique which is used to prevent the model from overfitting. Overfitting occurs in a model when it learns patterns from the training data which contain noise to an extent that it negatively impacts the model performance on unseen data. The term dropout refers to dropping out units both hidden and visible in a neural network [24]. By dropping a unit, it is removed from the network temporarily and the choice of which units to drop is random. Figure 2.12 shows the use of a dropout layer in a neural network.

Fully connected layer

A fully connected layer are standard neuron layers that perform the operation of classification or regression based on the given input from the convolutional layers. As shown in the Figure 2.9, the CNN models will be used to learn a function that will map a sequence of past observations as input to the output observations. The time-series data can be taken as a

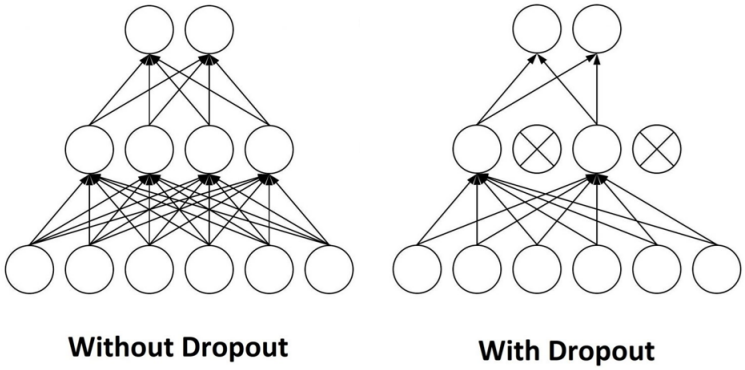


Figure 2.12: Dropout Layer [25].

one-dimensional (1D) grid that takes the samples at regular intervals.

2.4 Evaluation methods

2.4.1 Training Using a Validation Set

The preprocessed data is generally divided into three data sets: training, validation, and test data sets. The training set is the sample of data used to learn and fit the model's parameters. The validation set is used to find and optimize the best model to solve a given problem. The test set is a sample of data used to provide an unbiased evaluation of a final model fit to the training dataset. As the current dataset used in the research is time series, the data should be split in chronological order.

2.4.2 K-Fold Chronological Cross Validation

The k-fold cross-validation approach is a technique that ensures that every example is used in a test set. The cross-validation approach can reduce the probability of overestimating the model's effectiveness. In a standard k-fold cross-validation, the dataset is divided into n subsets of approximately equal size. The process of model generation and testing is then repeated for n times. Each time a different subset is selected as a test set and the remaining subsets are selected for training. The main intention of using a cross-validation approach is

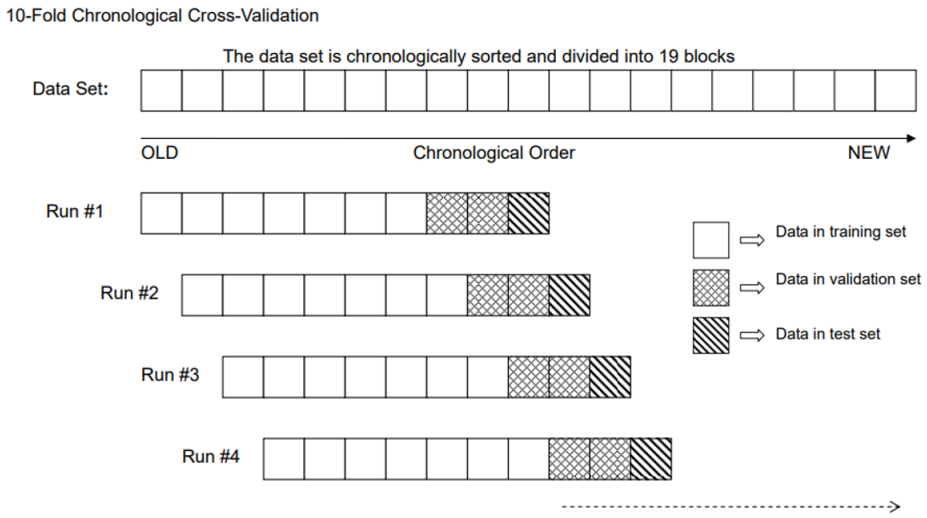


Figure 2.13: A chronological k-fold cross validation approach [26].

to better estimate the accuracy of the models, by calculating the mean accuracy over the n evaluations.

The standard cross-validation method cannot be used on time series data as the temporal order of the observations must be preserved during training and testing. A k-fold chronological cross-validation is a more realistic data evaluation method for any temporally sensitive model that selects the training and test sets [26]. A 10-fold chronological cross-validation is shown in Figure 2.13. The 10-fold starts by including seven blocks of training data, two blocks of validation data, and one block of test data in each run. The evaluation process keeps moving forward along the chronological order until the last data block in the data set is used as the test set.

2.4.3 Evaluation metrics

To measure the quality of a machine learning model, it is important to use an evaluation metric which will define how well the algorithm will work when tested with real life unseen datasets. The mean absolute error (MAE), mean absolute percentage error (MAPE), confidence intervals and hypothesis tests are the most commonly used evaluation metrics and methods.

Mean Absolute Error (MAE)

Mean Absolute Error (MAE) is the average of the absolute difference between actual values and predicted values.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{2.21}$$

Where, x_i is the actual value, y_i is the predicted value, and n is the number of data points.

Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) is the mean or average of the absolute percentage errors of forecasts. Error is defined as the difference between the actual value and the predicted value.

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} |\frac{y_i - x_i}{y_i}| \tag{2.22}$$

Where, x_i is the actual value, y_i is the predicted value, and n is the number of data points.

Confidence Interval

A confidence interval is a bound on the estimate of a population variable. It is an interval statistic used to quantify the uncertainty on an estimate. In deep learning, the confidence intervals can be used to present the skill of the predictive model. We use a confidence interval of 95% to measure the level of predictivity of the deep learning models developed.

Hypothesis Test

The model selection is an important part of applied deep learning. In the case of the regression problem, the model with minimum error (MAE or MAPE) can be estimated as the best model. With the help of statistical hypothesis testing, we determine which model is statistically more accurate than the other.

Chapter 3

Theory and Approach

This chapter provides the theory and approach to our research. Section 3.1 refines the problem statement and defines the hypothesis. Section 3.2 explains the approaches used in data collection, data preparation, and data cleaning for the models. Section 3.3 describes the model development and testing approach for the daily cases and the 7-day average cases prediction. Lastly, Section 3.4 and Section 3.5 discuss the evaluation methods and the software and the hardware configurations used in this research. The theory and the approaches presented in this chapter will be tested by the empirical studies discussed in Chapter 5.

3.1 Problem Refinement

Epidemiological forecasting is an important tool for health care systems during a pandemic such as COVID-19 for decision making and drafting health care policies. Predictive models can estimate the number of future COVID-19 cases in advance and offer insights into causal factors. With this information, health care systems can calculate the demand for hospital services and pharmaceuticals and determine the length of lockdowns or other public health measures. Given the degree of uncertainty with the pandemic, we set a success criterion of 15% MAPE (mean absolute percentage error), which is an accuracy of 85% or higher for predicting the number of COVID-19 cases per day.

3.2 Approach

To build a prototype machine learning model and understand the importance of the independent variables, we propose collecting the COVID-19 data from Ontario along with associated weather and mobility data. We use data from March 1, 2020, to December 31, 2020, prior to the initial vaccination rollout during the first week of January 2021. We explored several methods to model the relationship between the daily number of COVID-19 infections over space and time and the social mobility data and high-resolution weather and air quality data. The chief among these include:

1. Inductive decision trees (IDT) that are good for determining linear and non-linear relationships between one or more independent variables and the dependent variables.

2. Deep feed-forward artificial neural networks (ANN) could predict future days of case counts given several days' worth of independent variable values as a single input vector.

3. Deeper recurrent ANNs (RNN) could give several days worth of independent variable values as a sequence of daily vectors.

The first method is one of the oldest machine learning approaches that employ information theory and statistics to build a tree graph from a set of training data. Once constructed, the model can accurately predict the dependent variable given a set of input variables [27]. A recursive algorithm is used to build the IDT. The most important variable for deciding the value of the dependent variable is placed sequentially nearest the root of the tree. This approach has been used for time series prediction for problems such as weather and sales prediction [28]. The IDTs would be key to determining the most important variables affecting COVID-19 case counts and, therefore, would be used to eliminate irrelevant variables, thereby reducing the dimensionality of the input.

The second method is an older approach that uses deep feedforward artificial neural networks and a window that moves across the spatial-temporal data capturing what is considered important independent variables over space and time to predict the dependent variable. For example, the prior week's worth of daily social mobility, weather, air quality data, and the autoregressive terms might be used as a window to predict the next day's number of

infections. This approach to time-series modeling has been used effectively for predicting the next frame in a video [29], counting people in a moving crowd [30], in medicine [31], and traffic analysis [32]. The advantage of this approach is the relevance speed and which models can be trained and tested. Recently convolutional neural networks (CNN) have been used effectively for data prepared and used in this windowing manner. The disadvantage is that the size of the window and how far it reaches back in time and spatially to capture prior variables (features) must be manually selected.

The third method is a relatively new approach using deep recurrent neural networks (RNN). RNNs do not require the data to be prepared as a series of spatial-temporal windows into the past and nearby regions. The network uses recurrent connections and additional internal representation to capture the current context of the model and uses this as other input for making predictions. Most recently, Long Short-Term Memory RNNs or LSTMs have been used to learn long-term dependencies. They were introduced by Hochreiter and Schmidhuber (1997) and have been refined and popularized by many researchers. They have been used effectively to model complex sequences, including video frames [33], spoken and written language [34], sale prediction [35], traffic flows [51], and ship movement [37]. LSTM RNNs have the advantage of learning to select the most important variables (features) from the past depending upon the most recent network context and predictions. Unfortunately, LSTMs require a lot of computing power with long training times (hours and in some cases, days). We intend to solve this problem by using GPU-enabled devices to decrease the LSTM training time.

3.2.1 Data Collection

The first step in the process is identifying and collecting necessary data. Initial data was provided by the SMU team, which consisted of COVID case count data from Ontario. For our forecast modeling efforts, we selected just four of the 32 counties making up 35% of the data from Ontario; specifically Toronto, Peel, Durham, and York [38](see Figure 3.1). The total population of these counties is 5,797,924 (5.9 million), which accounts for 40% of the total population of Ontario (14,789,778).

Working with the SMU and Dalhousie researchers, we came to understand that there

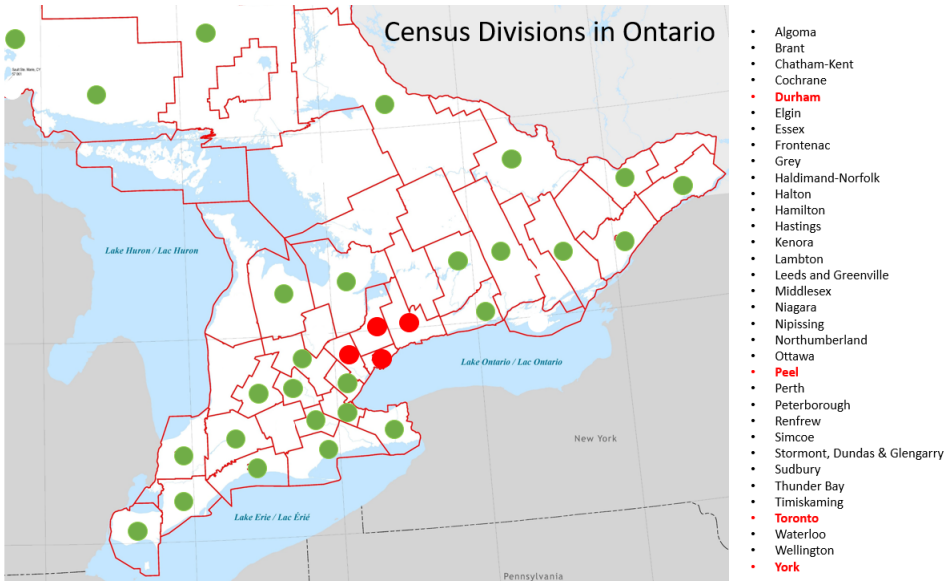


Figure 3.1: Geographical regions examined in Ontario(marked in red).

were at least five categories of variables that need to be considered by the predictive models:

- Demographics of people (age, gender)
- Time (day of week, day of year)
- Indoor and outdoor environmental factors (temperature, relative humidity, precipitation)
- Movement of people (public health restrictions, mobility)
- Recent and current status of disease (case counts)

It would be necessary to understand the fundamental linear relationships between variables in each of the above categories concerning COVID-19 case counts, taking into consideration various delay factors such as:

- The time for the disease to manifest symptoms in an infected person (3-14 days).
- The time before a COVID-19 sample could be taken from that person (1-7 days).
- Time required to complete the test on the sample and register it as positive (1-7).

- These steps could be completed in just three days or as many as 28 days, with an estimated average of 14 days [39].

We understood that the variables regarding the movement of people (and therefore viral spread opportunities) would be based on averages both in terms of volume of people and measurement time interval.

We observed that the number of case counts would rise in the fall of 2020 and beyond until such time as a vaccine was discovered, which meant that we would be looking at models that would need to not just interpolate (generalize) between training data points but project to numbers of cases not previously seen.

3.2.2 Data Preparation

This section describes the preprocessing steps used to prepare the data for the modeling; They are data aggregation, data cleaning, and data wrangling.

3.2.3 Data Aggregation

The above-mentioned data have been merged using Approximate Onset Date as the primary index. At the end of this step, the data consisted of 306 examples from March 1, 2020, till December 31, 2020.

3.2.4 Data Cleaning

Most of the data used in this research are human input data, which may contain some errors. The missing, redundant, and outlier values are replaced or removed during the data cleaning process. Of the original data consisting of case counts, calendar, weather, mobility, and indoor data, only the weather data had missing values. These had to be corrected before modeling could be undertaken. All variables which have 50% missing values were removed. This eliminated the following variables: solar_radiation, cloud_cover, snow_on_ground, max_humidex, and min_windchill.

3.3 Model Development

3.3.1 Inductive Decision Trees

Inductive Decision Trees are one of the oldest machine learning approaches that employ information theory and statistics to build a tree graph from a set of training data. Once constructed, the model can accurately predict the number of case counts given a set of input variables. A recursive algorithm is used to build the IDT. The most important variable for deciding the value of the dependent variable is placed sequentially nearest to the root of the tree. This approach has been successfully used for time series prediction in the past. IDTs can also be used to determine the most important variables affecting COVID-19 case counts and, therefore, can be used to eliminate irrelevant variables, thereby reducing the dimensionality of the input.

3.3.2 LSTMs

LSTMs networks are a type of recurrent neural network (RNN) used in time-series forecasting problems that effectively handle long data sequences. In real-world problems, Long Short-Term Memory Networks (LSTM) have the advantage of understanding non-linear relationships over time in the data. The LSTMs can learn complex relationships between the input and output variables to predict the number of cases effectively.

A six-layer LSTM is developed to forecast the daily case counts. The LSTM architecture consists of an input layer, two hidden LSTM layers, followed by two dense layers, and an output layer. We have considered two types of LSTMs in this research, an STL LSTM used to predict the next day (D+1) and an MTL LSTM which predicts all the following seven days into the future (D+1 to D+7).

3.3.3 CNNs

Although convolutional neural networks (CNNs) are generally used for image processing and computer vision, they can be used in time series forecasting [40]. In case of image recognition, the CNN detects the patterns in an input image which is a group of pixels using the convolutional layers. Similarly, the patterns in a time series data can be used to extract

the temporal correlations among different sequences, by taking the timesteps into account [50]. Unlike LSTMs in which the timesteps can be explicitly applied, the input data in a CNN is presented as a matrix. The variables across multiple timesteps are presented as an array of n x d, where n is the number of variables and d is the number of timesteps.

A CNN usually consists of convolutional layers, pooling layers, and fully connected layers. We use convolutional neural network consisting of two blocks of layers; 1 convolutional block and 1 fully connected block. The convolutional block contains two convolutional layers, followed by a max-pooling layer. Max-pooling is performed after each convolutional layer with a filter size of 14 and a stride of 2. We have considered two types of CNNs similar to LSTMs, using an STL CNN to predict the next day (D+1) and an MTL CNN which predicts all the following seven days into the future (D+1 to D+7).

3.4 Evaluation metrics

3.4.1 Mean Absolute Error (MAE) & Mean Absolute Percentage Error (MAPE)

To evaluate the model's performance, the actual number of case counts and the predicted number of case counts are compared using the Mean Absolute Error and the Mean Absolute Percentage Error shown in equations 3.1 and 3.2 respectively.

The MAE is the most intuitive metric used to understand the deviation of the predicted case counts from the actual values. However, MAPE is the main measure of model performance used in the experiments of Chapter 5. It takes into consideration the MAE normalized by the average number of case counts. The MAE is calculated in the number of case counts, as shown in equation 3.1.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{3.1}$$

Where, x_i is the actual value, y_i is the predicted value, and n is the number of data points.

The MAPE value is calculated as shown in equation 3.2, as a derivate of MAE value divided by the average number of case counts in the independent test set.

$$MAPE = \frac{MAE\ in\ case\ counts}{Average\ number\ of\ case\ counts\ in\ test\ set} \tag{3.2}$$

3.5 Implementation

3.5.1 Hardware and Software

Python was used as the primary programming language throughout the research. The deep neural networks are built using the Keras API to perform multiple experiments. Keras is a high-level library that works on top of TensorFlow. NVIDIA's CUDA architecture has been used wherever applicable to benefit from parallel processing on the local computer. The hardware and the software configurations used in this research are listed in Table 3.1 and 3.2 respectively. All the experiments discussed in Chapter 5 use this configuration.

Table 3.1: Hardware configuration of local computer.

Item	Type
CPU	Intel Core i7-9750H @2.6 GHz
GPU	GeForce GTX 1660 Ti
RAM	16 GB
VRAM	6 GB

Table 3.2: Software configuration.

Library	Version
Python	3.8.0
TensorFlow	2.4.0
Keras	2.4.3
Numpy	1.20.2
Pandas	1.3.0
Matplotlib	3.4.2
Scikit-learn	0.24.2
CUDA	11.5
cuDNN	8.1
Jupyter	6.4.5
WEKA	3.8.4

CuDNNLSTM is a fast LSTM implementation backed by cuDNN; a GPU accelerated deep learning library that runs on the CUDA architecture, with TensorFlow as the backend. Running the LSTM models on a CPU versus a CUDA accelerated GPU has decreased the model training time by 70%. In the case of sequence prediction problems where the input and the output to the model are both huge vectors, the CUDA architecture provides an interface that will allow the vector operations to take advantage of running the processes parallely on a GPU. Compared to a CPU, the processes are run sequentially, which is often time and resource-consuming.

WEKA, an open-source tool, has been used to perform exploratory data analysis and build IDTs to determine the variable importance [41]. IDTs help us understand which variables are of greatest importance for predicting case counts. This is often unachievable using neural networks because of the complexity of the representation in the models.

Chapter 4

Data Analysis and Variable Selection

This section identfies the relevant variables considered and summarizes an analysis of the relationships between each of the independent variables and the number of COVID-19 case counts. It also describes the variables divided into five categories such as Daily case count variables, Demographic variables, Calendar variables, Outdoor and Indoor environmental variables, and lastly Movement of People variables.

4.1 Data Sources

Specifically, the data used in this research are as follows :

- Confirmed positive cases of COVID-19 in Ontario including date of approximate onset of the symptoms. The initial data contained individual case outcome which was further grouped using the date of approximate onset of the symptoms. Factors such as age group and gender are further calculated as average over all the cases on a given day [38].

- Mobility data was acquired from the Meta (previously Facebook) website under the Facebook Data for Good initiative [42]. This is an open-source set of data, based on location history from smartphones. The dataset contains two metrics: Movement_rel_to_baseline (Mobility); and Proportion_users_staying_put. These two metrics are further defined in Section 4.3.7.

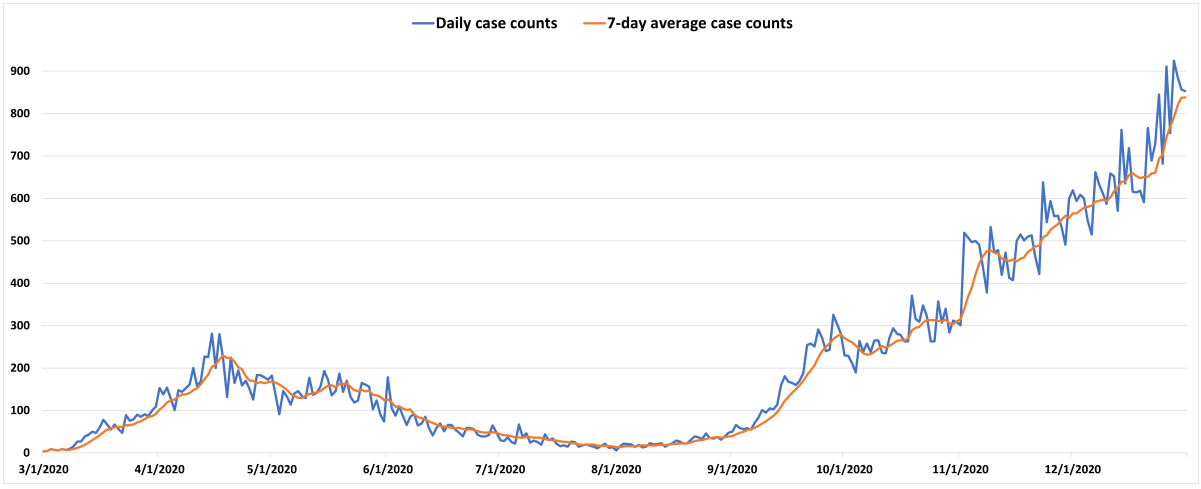


Figure 4.1: Daily case counts and 7-day average case counts

- Weather data was extracted from weatherstats.ca [43]. They provide high-dimensional weather data with approximately 60 different variables in readily available formats such as CSV.
- Indoor Relative Humidity (IRH) data was acquired from the Mississauga area of Toronto [36].
- Changes in the Public Health Restrictions were compiled from the Government of Ontario website which consists of all the information on lockdowns, opening and re-opening, and restrictions imposed in several Ontario regions from February 2020 till present [44].

4.2 Identification of Relevant Variables

As mentioned in Section 3.2.1, there are at least five categories of variables that need to be considered for predicting changes in COVID-19 case counts. The following list provides details on each of the variables in each category that has been considered during this project. A complete Meta-Data Report can be found in Appendix B.

Case counts - Daily case count (current day = D0, yesterday = D-1, tomorrow = D+1):

- Estimated number of cases occurring on the Accurate_Episode_Date. It is intended to estimate the number of new cases originating on that date, representing the number of transmitted cases.

7-Day average case counts:

- Average daily case count over the last 7 days, including today (D0). Figure 4.1 graphs these two important target variables over the 2020 study time frame.

Demographics:

- Age
  - Represents the average delay in the days between the dates when symptoms were observed and the date the person officially called the health authorities.
- Male percentage
  - Ratio of number of male cases over the Daily case count.
- Delay Mean
  - Average delay in the days between the dates when symptoms were observed and the date the person officially called the health authorities.

Transmission Type:

- Trans-Close Contact
  - Percentage of transmissions by close contacts such as family members or friends.
- Trans-CommSpread
  - Percentage of transmission by community spread.
- Trans-Missing
  - Percentage of cases in which the information about the type of the transmission is missing from the record.

- Trans-Unknown
  - Percentage of cases in which the approximate information about the type of the transmission is unknown by the patient.
- Trans-Outbreak
  - Percentage of transmission by an outbreak at a certain place. E.g.: At a sports stadium or a theatre.
- Trans-Travel
  - Percentage of transmission by air travel between the borders.

Calendar:

- Day of Year
  - Consecutive number of the day in a year(1 to 365).
- Day of week
  - Consecutive number of the day in a week starting from Sunday.

Indoor and outdoor environmental factors:
Maximum, Minimum, average and average hourly of the following :

- Air temperature
- Outdoor relative humidity
  - Amount of the water vapour that is present in the air in %.
- Wind speed
  - Average speed at which the air moves from high to low pressure due to changes in the temperature in km/h.

- Air pressure
  - Average pressure that is observed at a specific elevation and the true barometric pressure of that location.
- Visibility
  - Average visibility is the maximum horizontal distance through the atmosphere that objects can be seen by unaided eye.
- Health index
  - Average air quality index value for a particular day.
- Dew point
  - Value of the temperature to which air must be cooled to become saturated with the water vapour.
- Precipitation
  - Amount of rain/snow received.
- Snow
  - Amount of snowfall measured in *cm*.
- Snow on ground
  - Amount of the snow fall which is accumulated on the ground. Measured in *mm*.
- Wind gust
  - Brief increase in the speed of the wind usually in less than 20 seconds.
- Sea Pressure
  - Average pressure that is observed at a sea level and the true barometric pressure of that location.

- Indoor relative humidity
- Maximum humidex
- Minimum windchill
  - Effective lowering of the air temperature caused by the wind.
- Heat degree days
  - Given for each degree Celsius that the daily mean temperature departs below or above the baseline of 18 degree Celsius.
- Cool degree days
  - Given for each degree Celsius that the daily mean temperature departs below or above the baseline of 18 degree Celsius.
- Grow degree days with 5C, 7C and 10C as base
- Sunrise and Sunset time
- Daylight (in *hrs*)
- Sunrise and Sunset forecasted
- Minimum and maximum UV forecast
- Temperature forecasts
- Solar Radiation
- Cloud cover

Movement of People:

- Movement_relative_to_baseline (Mobility): Movement of people compared to a baseline period which predates most social distancing measures, based on cellular phone movement.
- Proportion_users_staying_put: The fraction of population that stays in place for longer time periods.
- Public health restrictions (personal bubbles, limited mobility, lockdown)

4.3 Analysis of Variables

4.3.1 Linear Correlation

Figure 4.2 shows the correlation of each independent variable with the number of new case counts for the current day (D0). We use this base information along with common knowledge of the relationships between variables to reduce the dimensionality of the input space. Most values with a correlation less than ± 0.1 were removed. And the min and max daily values for all weather variables, except max_wind_gust, were removed in favor of the average daily values.

Figure 4.3 through 4.6 show the Pearson linear correlation between each variable and the number of new COVID-19 case counts as a function of lag days. Variables that show their graphs slowly increasing or decreasing for all lag values such as DOY, Restrictions, Mobility, Male percentage, and Age indicate that a shift in their calendar date into the future generally means the best correlation is with a lag of zero days. The more interesting variables in terms of lag are avg_temperature, avg_health_index, IRH, avg_visibility, DOW, precipitation, and avg_relative_humidity (outdoor humidity). We see from the graphs of the remaining variables that data from the prior 14 days has the highest correlation with the current day's case counts. Outdoor air temperature has the highest negative correlation with case counts (-0.468) when there is lag of 10 days; that is current air temperature has its highest impact on case count 10 days into the future. Similarly, IRH has the highest negative correlation (-0.301) when there is lag of 6 days, DOW (-0.114) every seventh day (Sunday), avg_visibility (-0.248) with a lag of 14 days, precipitation (-0.066) with a lag of 12 days, and avg_health_index (-0.300) with a lag greater than 16 days.

4.3.2 Analysis of Daily Case Count Variables

Daily case count - Represents the number of new positive cases each day taking the following into consideration:

- The time required to complete the test on the sample and registered it as positive (1-7 days). The date that the test is recorded is referred to as the Test_Reported_Date.

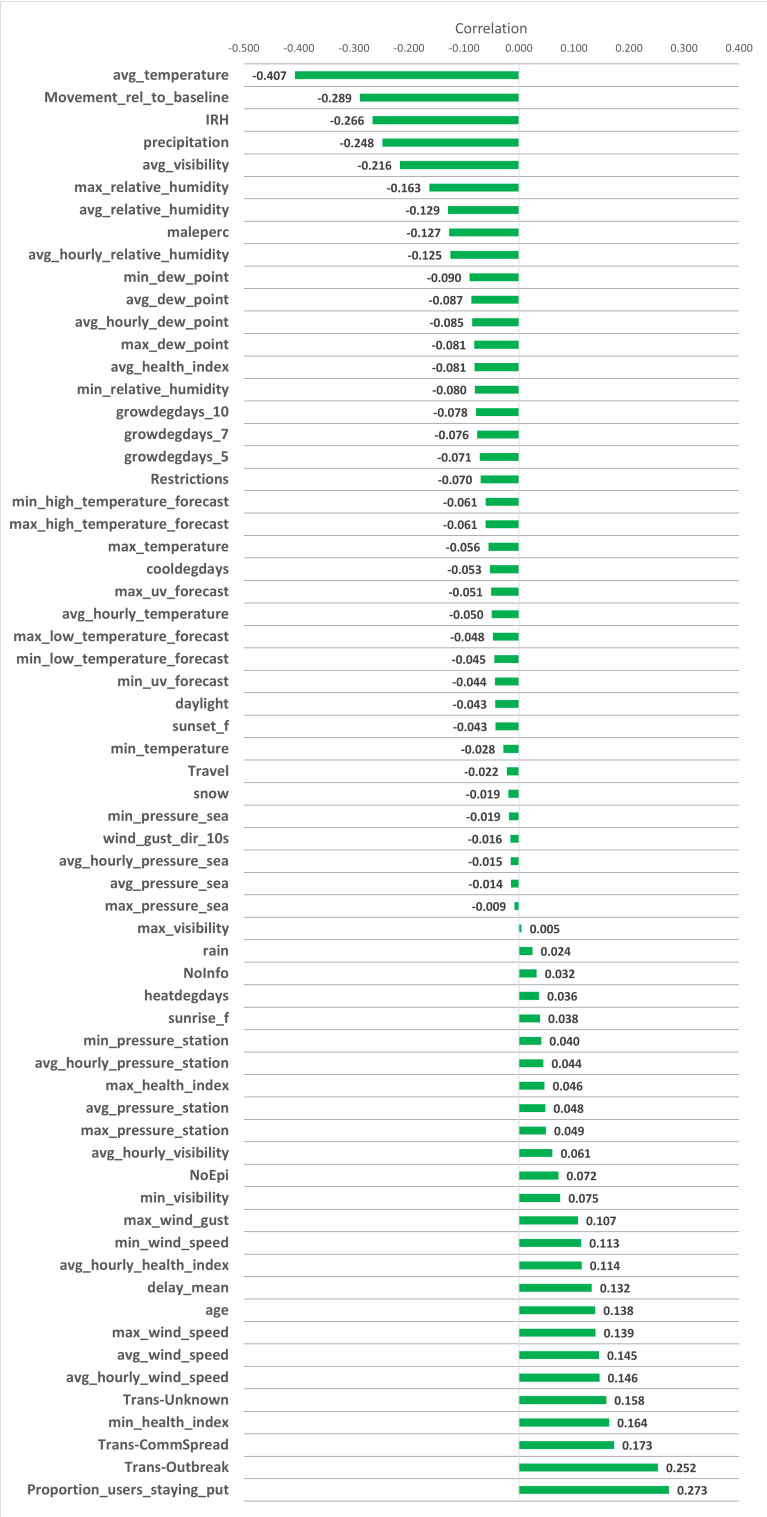


Figure 4.2: Correlation between independent variables and current Daily case counts (D0).

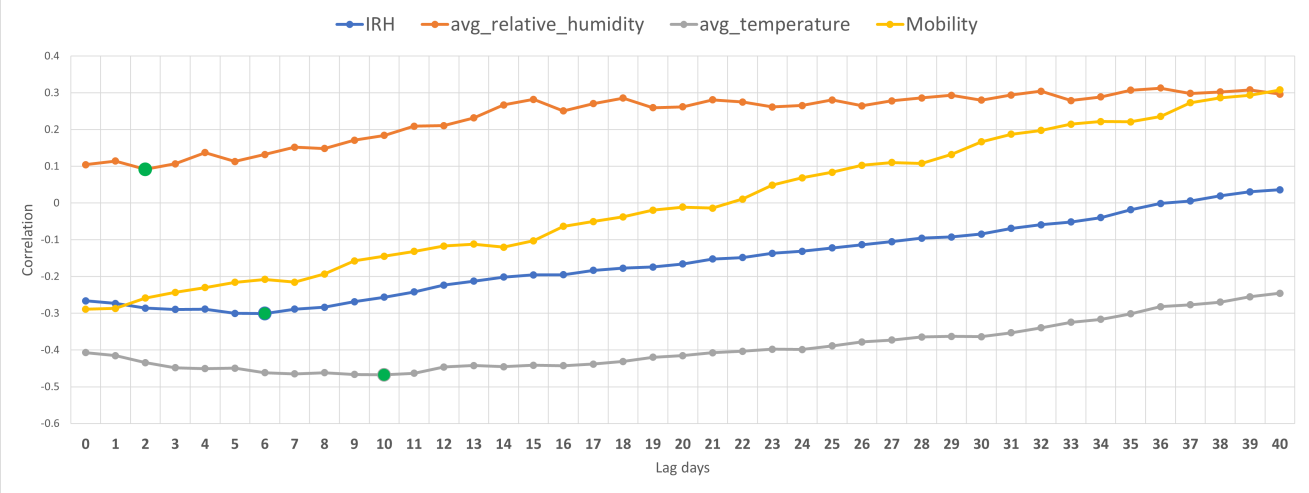


Figure 4.3: The graph shows the correlation between the independent variables: IRH, avg_relative_humidity, avg_temperature, and Mobility, with case counts as a function of lag days. The green dot depicts the highest correlation value.

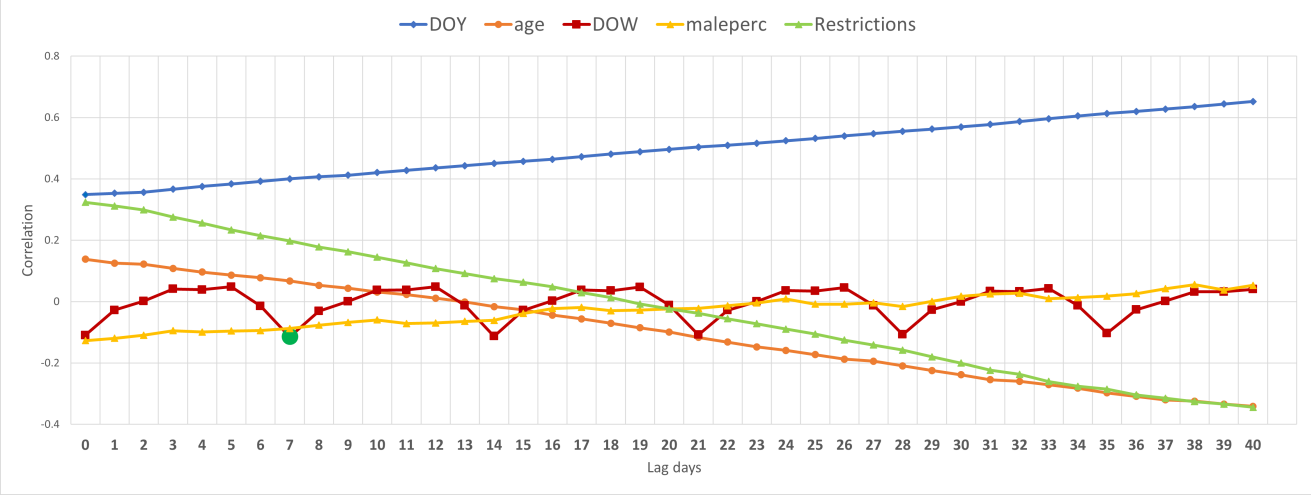


Figure 4.4: The graph shows the correlation between the variables: DOY, age, DOW, maleperc, and Restrictions, with case counts as a function of lag days. The green dot depicts the highest correlation value.

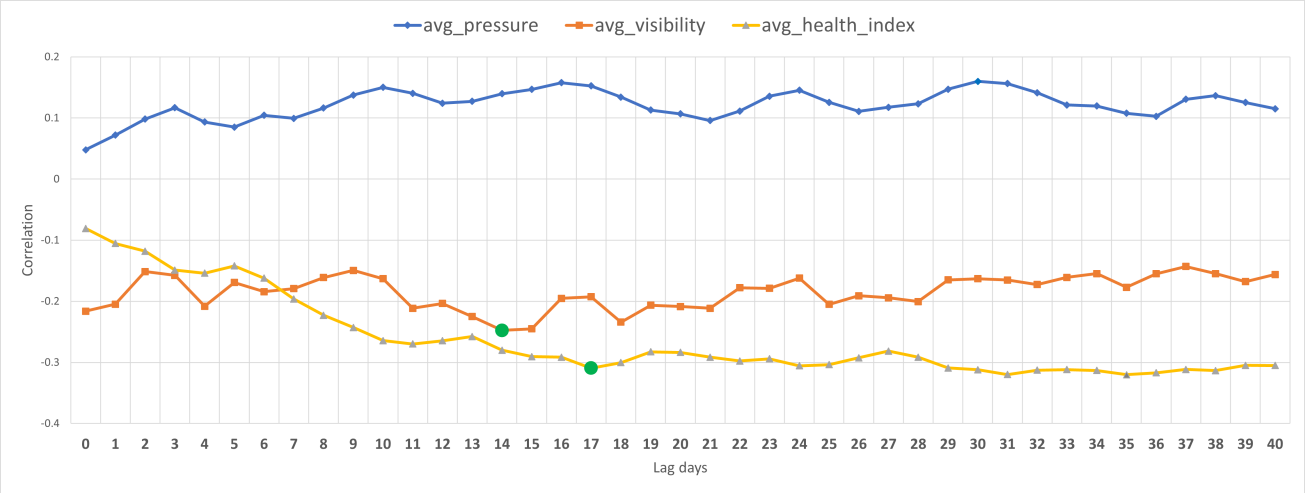


Figure 4.5: The graph shows the correlation between the variables: avg_pressure, avg_visibility, avg_health_index, with case counts as a function of lag days. The green dot depicts the highest correlation value.

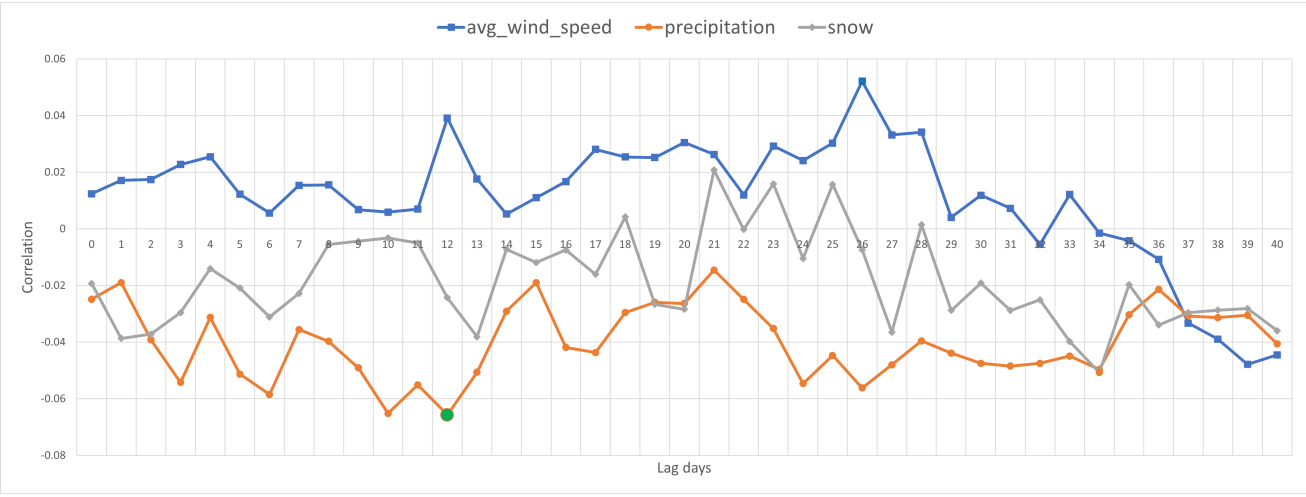


Figure 4.6: The graph shows the correlation between the variables: avg_wind_speed, precipitation and snow, with case counts as a function of lag days. The green dot depicts the highest correlation value.

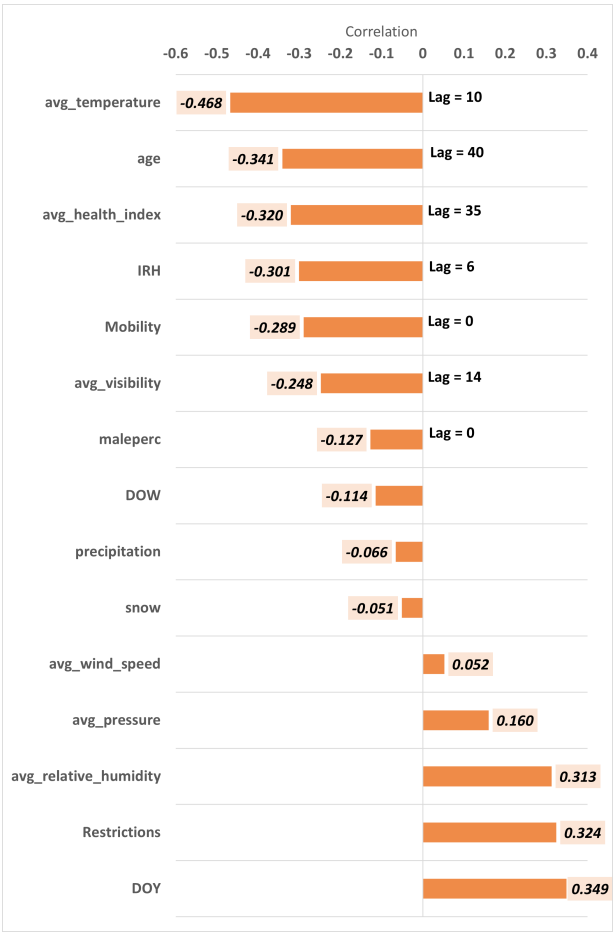


Figure 4.7: Bar graph of maximum correlation between independent variables and case counts.

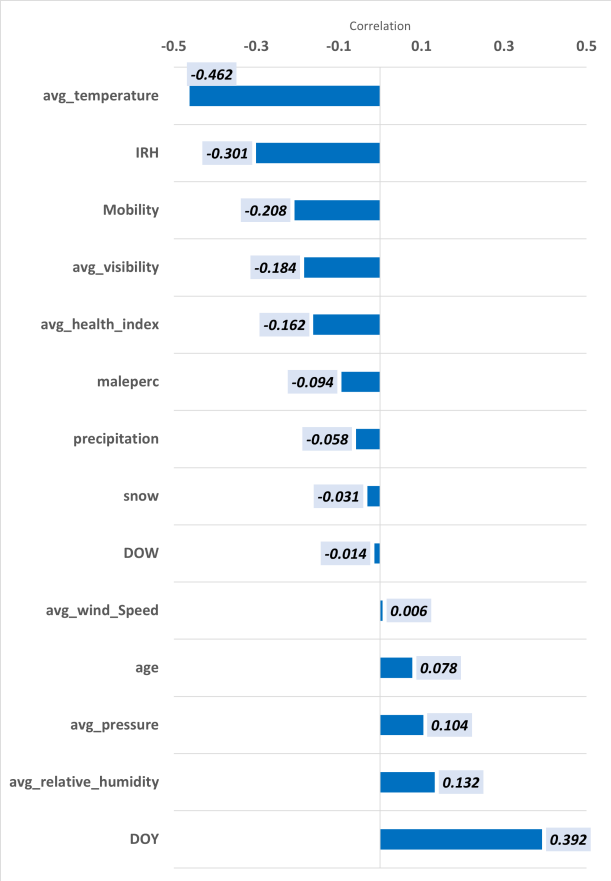


Figure 4.8: Bar graph of the correlation between each of major independent variables and the Daily case counts when there is a lag of 6 days.

- The time before a COVID-19 specimen is taken from a person (1-7 days). The date of taking sample from the person is referred to as the Specimen_Date.
- The time between the detection of symptoms and the report of a case (5-14 days). The date of reporting the case is referred to as the Case_Reported_Date.
- The time for the disease to manifest symptoms in an infected person following transmission.
- The estimated date of disease transmission based on all the prior dates is referred to as the Accurate_Episode_Date.
- The delay_mean is a function of the difference between the Accurate_Episode_Date and the Case_Reported_Date.
- Case count could include cases recorded as few as 7 days prior to or as many as 28 days prior to the Case_Reported_Date, with an estimated average of 14 days prior to the Case_Reported_Date [39].

7-day average case count - The rolling average of Daily case counts over the period D-6 (6 days past) through D0 (current day) (See Figure 4.1).

4.3.3 Analysis of Demographic Variables

- Age: Represents the average age of all the COVID cases on a given day. It has a significant correlation with daily COVID cases. Figure 4.9 shows the Daily case count by DOY broken out by Age range. This shows that in the winter and spring of 2020, those persons over 70 years contributed largely to the case counts, whereas during the fall of 2020 it was 20 to 39 year-olds who contributed significantly. The 40-69 group contributed strongly during both periods.
- Male percentage : The percentage of male patients on a particular day. It doesn't have a significant correlation with COVID-19.

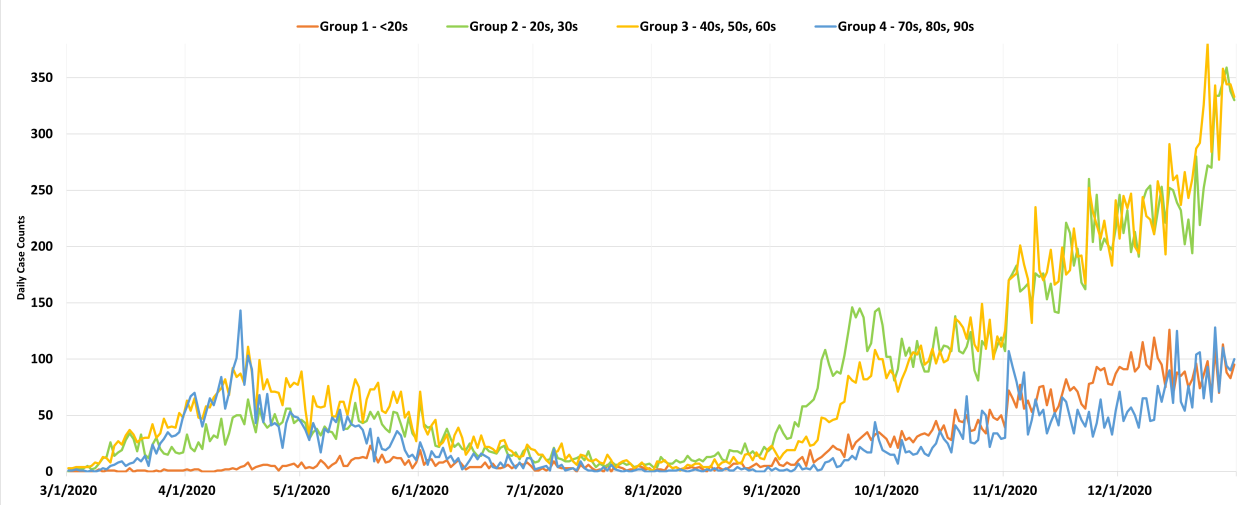


Figure 4.9: Daily case counts broken out by Age.

- During the period between March and June, the older age group were more affected by COVID, while the younger population had a greater number of COVID transmissions during the second half of the year.

- Delay_mean: The average delay in the days between when the symptoms were observed and the date when the person officially called the health authorities. It was removed because it varies widely over the study period and was thought to not be of consequence.

- The Transmission type is the suspected method of exposure to COVID-19, reported either by a patient or determined by a Public Health Unit after assessing the patient. Although the variables Trans-CloseContact, Trans-CommSpread, Trans-Missing, Trans-Unknown, Trans-Outbreak, Trans-Travel have a significant correlation with the Daily case counts, we believe that type of transmission is not a important factor in the prediction of case counts.

- Subsequently, Trans-Close Contact, CommSpread, Missing, Unknown, Outbreak, Travel, delay_mean were not used for model development.

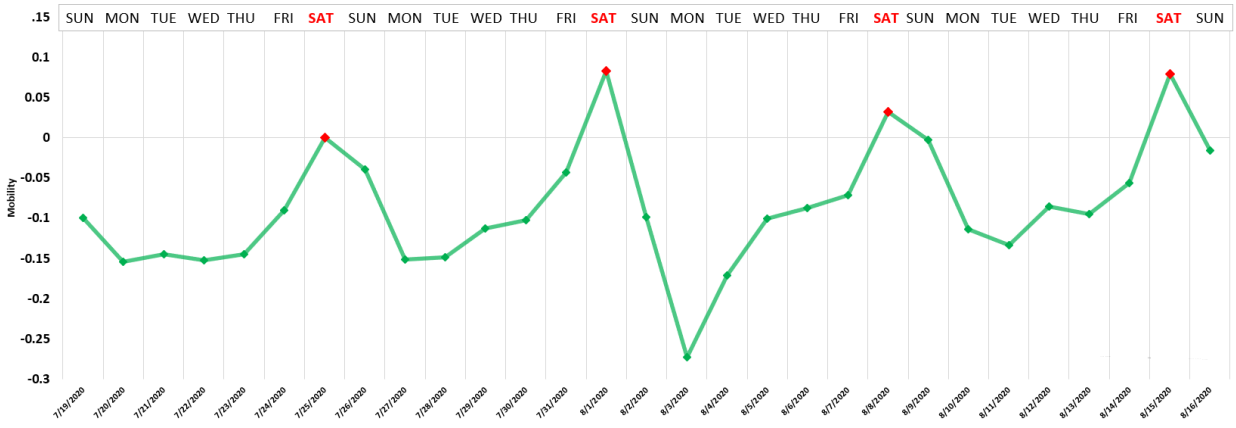


Figure 4.10: The graph shows daily case counts versus DOW. Peak days are normally on Saturdays (DOW = 5).

4.3.4 Analysis of Calendar Variables

- Day of the year: Represents the numeric day of the year 1 to 365. There is a strong linear correlation to Daily case counts just because transmission of the disease increases through the year 2020.

- Day of Week: An important factor because transmission of diseases varies significantly from the weekday to weekend, and from day to day over the week. The highest case counts occurred on Sundays (DoW 6) and Mondays (DoW 0) because there were more interactions between mixes of people on the weekends than during the week and the recording of positive tests was higher after a weekend. As shown in the Figure 4.10, the Mobility data shows that Saturday and Sunday (DoW 5 and 6) are the highest days of active movement in the Toronto area and Monday and Tuesday (DoW 0 and 1) tend to be the least active days.

4.3.5 Analysis of Outdoor Environmental Variables

- Outdoor air temperature: Fluctuates significantly in the Toronto area; from -4.2C in March to 28.4C in August. We see that air temperature has one of the strongest correlations with the transmission of COVID-19. The highest negative correlation of -0.468 occurs with a lag of 10 days between the temperature and the case counts.

- Outdoor relative humidity: There appears to be only a small relationship between the spread of COVID-19 and outdoor humidity in the Toronto region of Canada. Figure 4.11 contrasts the difference in its impact to that of indoor humidity.
- Wind speed: Shows a small correlation with Daily case counts depending upon the lag.
- Wind gust: No significant correlation with Daily case counts.
- Air pressure: No significant correlation with Daily case counts.
- Visibility: Directly related to temperature and relative humidity; we see the highest negative correlation (-0.248) occur at a lag of 14 days.
- Health index: Based on outdoor air quality; seems to have only a minor relationship to case counts with a correlation of -0.164 at six days of lag.
- Dew point: Calculated from outdoor air temperature and relative humidity levels, which are already available to our models.
- Windchill and Humidex: Calculated from variables already available to our models, and the original records had at least 60% missing values.
- Precipitation: Plays a minor role, with a high negative of -0.0657 at a lag of 12 days.
- Snow on ground: No significant correlation with case counts beyond that already reflected in the precipitation variable; however, it was left in as we felt that heavy snow levels could affect the ability to meet and therefore transmit the virus.
- Based on their relatively low correlation with the dependent Case Counts variable, the following, environment variables: heatdegdays, cooldegdays, growdegdays, Sunrise and Sunset forecast, Minimum and Maximum UV forecasts were not used for model development.

We found that all independent variables with significant correlation (avg_temperature, IRH, avg_visibility, DOW, precipitation, avg_health_index) have a lag within 14 days and an average of 10.5 days.

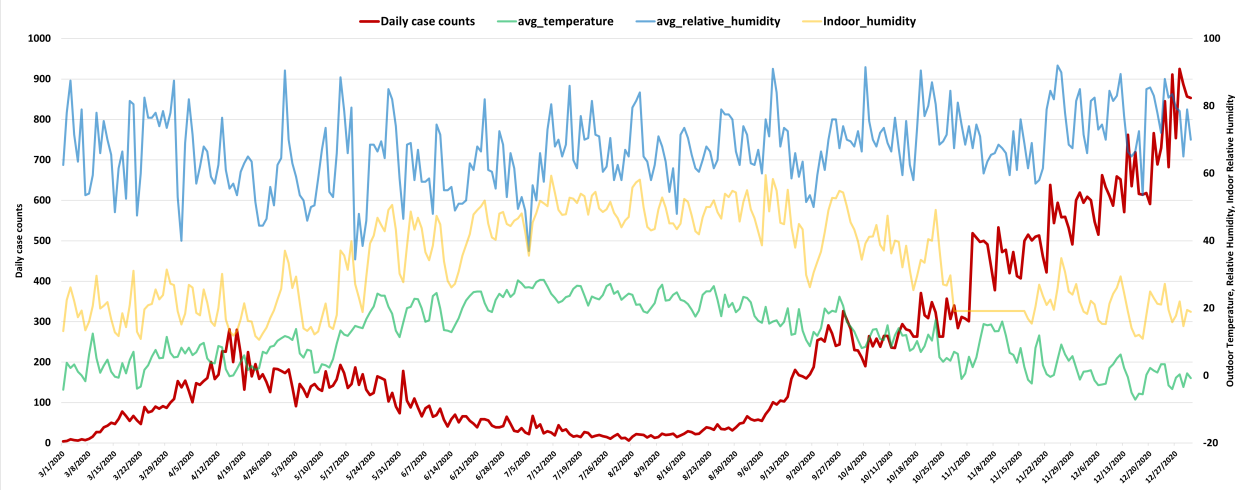


Figure 4.11: Comparision between Daily case counts, outdoor temperature and relative humidity and indoor relative humidity.

4.3.6 Analysis of Indoor Environmental Variables

During lower outdoor air temperatures and cold winter winds more people are working and living indoors or in vehicles for greater periods of time. This brings people into closer contact with each other and reduces their exposure to sunlight (which lowers vitamin D and melatonin levels as well as exercise and sweating). This provides a greater opportunity for viruses to spread. Building environmental conditions can contribute significantly to transmission; most significantly, indoor air temperature (IAT) and indoor relative humidity (IRH). The primary indoor environmental variables are:

- Density of people: Controlled by Public Health Restrictions.
- Quantity of fresh air: Typically 10% or less in office buildings, but as much as 30% in hospitals; there are engineering standards and building manager guidelines for, but fewer laws.
- Quality of air filtration between rooms: Air re-circulation in buildings causes the same air to be inhaled by multiple individuals on the same floor or multiple floors; various levels of filtration in the HVAC system of the building are meant to reduce the spread of disease, toxic vapors, and offensive smells. Less expensive filtration methods are

used in office buildings and malls, whereas more advanced and expensive HEPA filters are used in health care settings and hospitals.

- Indoor Air Temperature (IAT): Typically, kept at about 20-23.5C in summer and 23-25.5C in winter, because of variations in clothing and humidity.
- Indoor Relative Humidity (IRH): Known for some time to have a significant impact upon the well-being of the inhabitants.

This data showed a significant negative correlation between Daily case count and IRH, particularly when a lag of 6 days is considered (see Figures 4.7 and 4.8). One can see in Figure 4.11 that there is little correlation between Outdoor relative humidity and Daily case counts in the Toronto area, however there is significant correlation between Outdoor temperature and Indoor relative humidity and Daily case counts. The lowest IRH is 10.64% in March and the highest is 59.48% in August.

4.3.7 Analysis of Movement of People Variables

Mobility

Two variables were extracted from the Facebook website under an initiative known as Facebook for Good, which have released a set of datasets to help researchers around the world to respond to the COVID-19 crisis.

- Movement_rel_to_baseline (Mobility): Shows the movement of people compared to a baseline period which predates most social distancing measures. We use this variable as the major signal of people's movement.
- Proportion_users_staying_put: Shows the fraction of the population that stay indoors for longer time periods. It correlates strongly with Movement_relative_to_baseline so it is redundant.

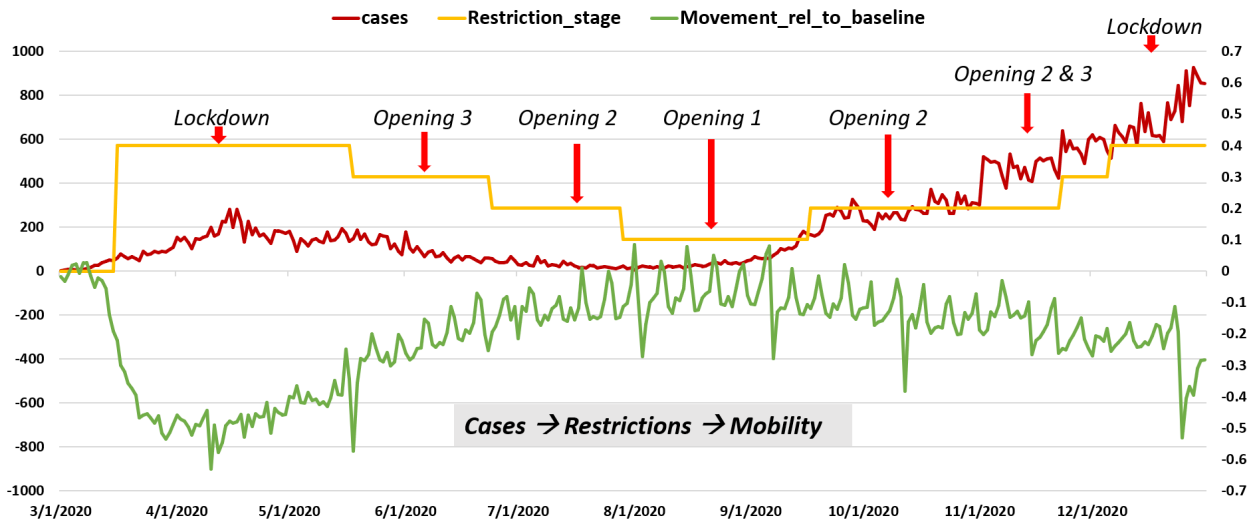


Figure 4.12: Daily case counts versus Restrictions and Mobility.

Public Health Restrictions

Data on changes in Public Health Restriction was compiled from the Government of Ontario website. We divided the stages of restrictions into 5 categories namely Normal, Opening 3, Opening 2, Opening 1 and Lockdown based on the dates available from the website when the restrictions have been initiated. We assigned each of these stages of Public Health Restrictions one of five levels 0-4 as shown in Table 4.1.

Table 4.1: Stages of Public Health Restrictions in Ontario.

Restriction	Stage Name	Indoor (max persons)	Outdoor (max persons)
0	Normal	-	-
1	Opening 3	25	10
2	Opening 2	10	25
3	Opening 1	5	10
4	Lockdown	5	0

Figure 4.12 shows a graph of Daily case counts, Mobility, and level of Public Health Restrictions. In our opinion, the graph suggests that the cause-and-effect sequence is from Case count to changes in restrictions and finally to changes in mobility. In fact, the linear correlation between restrictions and mobility is 0.9 with no lag and decreases with lag thereafter. One could conclude that the rise or fall in case counts results in an increase or

decrease in restrictions, followed by a decrease or increase in mobility, respectfully. So, one can see how the restrictions worked well to reduce mobility which naturally, over time, had an impact on the transmission of the disease.

4.4 Variable Selection

The data suggests that in the early part of the year during the colder months of March and April, transmission of COVID was the highest amongst the most vulnerable demographic (older population) despite restrictions and subsequent reductions in mobility. In contrast, a reduction in restrictions along with increased mobility to normal levels in the warmer months of June through August corresponds to a reduction in Daily case count numbers to their lowest in 2020. This suggests that variables beyond human interaction are at work in COVID transmission. The fall of 2020 confirms this because as mobility slowly decreases, the number of Daily case counts soars to their highest levels.

Based on the above analysis we reduced the original set of independent variables to the list of 17 including the case counts shown in Figure 4.13. These variables either had high linear correlation with Daily case counts considering a lag of up to 14 days, or they played an important combinatorial role in inductive decision tree models, as will be discussed in Chapter 5. We feel these variables capture the most important aspects of the demographics and movement of the population, the environments in which people were interacting, and the temporal aspects of the week or year. A complete Meta-Data Report can be found in Appendix C.

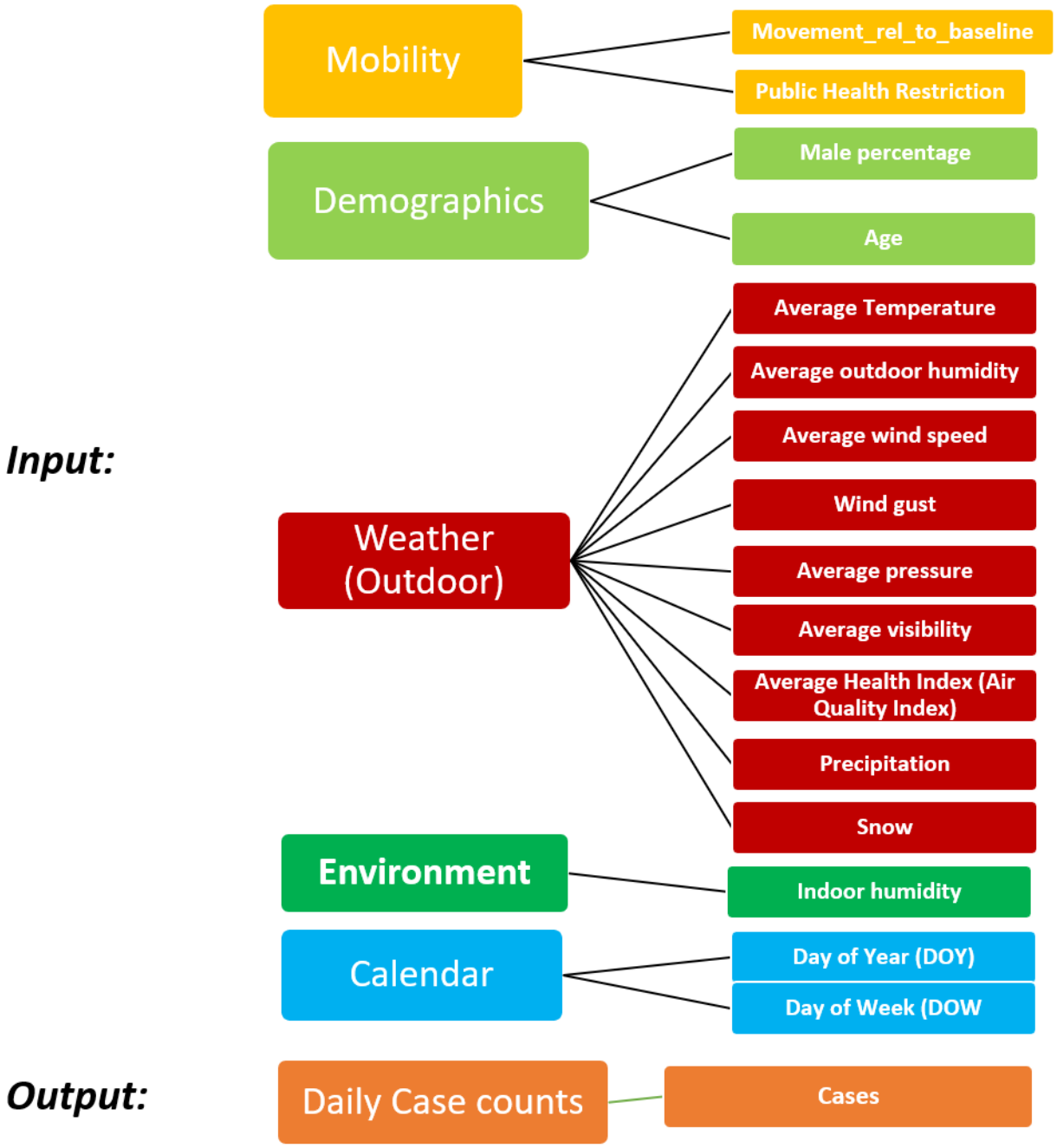


Figure 4.13: List of the selected variables

Chapter 5

Empiricial Studies

This chapter presents the results of empirical studies used to predict case counts and the 7-day average values for the next day (D+1), referred as STL models and for the next 7 days (D+1 to D+7), referred as MTL models. Sections 5.1 to 5.4 describe all the experiments performed.

5.1 Predicting COVID-19 using IDT

Inductive Decision Trees (IDTs) are popular machine learning algorithms because of their interpretable nature [45]. This experiment uses decision trees to understand the variable importance for both 7-day and 14-day prior data to predict the next day (D+1) and demonstrate the ability of decision trees to predict the unconventional case counts.

5.1.1 Experiment 1: Predicting tomorrows (D+1) case counts using 7 days of prior data

Objective

The experiment aims to develop IDT models that accurately predict COVID-19 case counts one day in advance (D+1) using seven days of prior data (D-6 to D-0), and then analyze these models to determine the most important attributes.

Data and Methods

For this experiment, the model used 306 examples from March 1, 2020, until December 31, 2020. The training set was composed of 268 examples from March 1, 2020, to November 23, 2020, with 38 test examples from November 24, 2020, to December 31, 2020.

The 7-day and the 14-day data was manually processed using excel, as decision trees do not have the power to process the past data automatically like LSTMs. We used the WEKA Machine Learning software and developed multiple decision tree models and tested on the independent test set until the lowest MAPE was determined.

Results and Discussion

Various experiments of hyperparameters such as C and M were done to find the optimal parameter setting. After several preliminary trials of the WEKA M5P IDT software to develop models, the best learning parameter settings were found to be M=4 with Linear Regression Models at leaves of trees.

Using all input variables with the autoregressive variables, the best IDT model produced an MAE = 72.66 and a MAPE = 10.96% on the independent test set. This indicates that better models can be developed using additional variables. A simple persistence model (that uses todays case count as tomorrows prediction) performs much worse with an MAE of 88.09 cases and a MAPE of 13.28% on the same test set (based on a daily mean case count of 662 over the test set). The best model using all input variables without autoregressive variables produced an MAE = 109.95 and a MAPE = 16.58% on the independent test set (based on a daily mean case count of 662 over the test set).

Figure 5.1 shows the graph of predicted versus actual case counts generated by this model. This is a more illuminating model because it focuses on variables other than prior case count values. Figure 5.2 shows the resulting decision tree and, therefore, the most important variables for forecasting Daily case counts. The most significant variables were DOY, Mobility (Movement_rel_to_baseline), age, IRH, and avg_temperature (outdoor). DOY plays an important role in determining time of the year, such as before the end of Winter 2020 (March 23), spring and summer (March 24 Sept 24), or colder days after Sept 24.

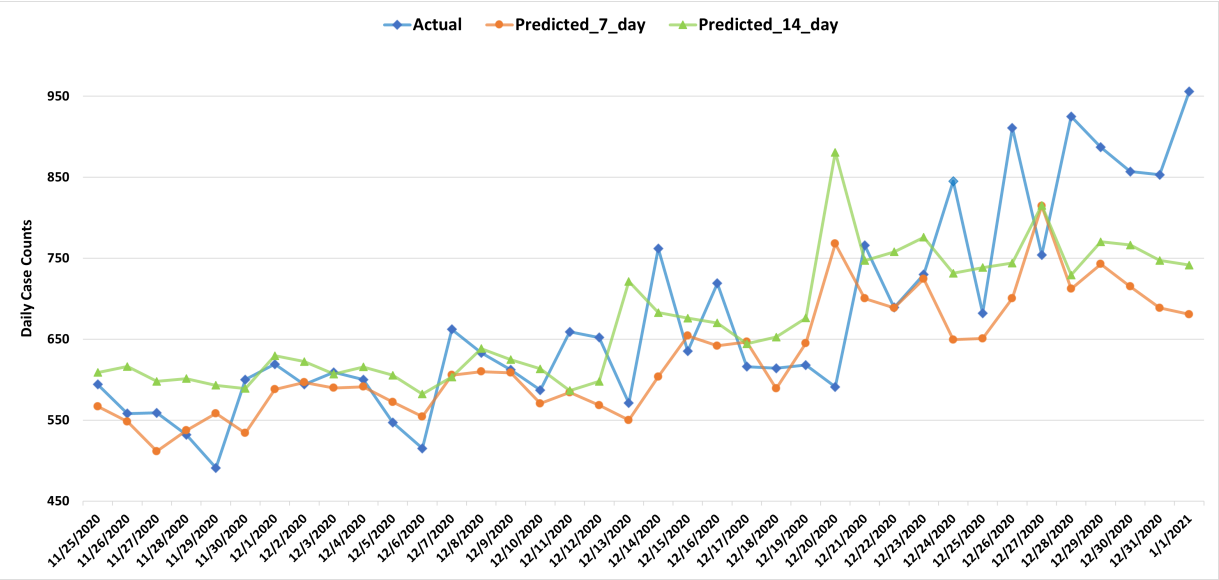


Figure 5.1: Experiment 1, 2: IDT 7-day and 14-day predicted versus actual daily case counts.

Within the warmer Spring and Summer period, the outdoor air temperature and relative humidity play key roles in determining case counts.

A model developed without DOY but with Mobility produces an MAE = 178.29. This suggests that Mobility confounds efforts to create a general model from the available data if DOY is not present. A model developed with DOY but without mobility confirmed this with an MAE = 107.29. Furthermore, a model developed with neither Mobility nor DOY had an MAE = 260.85 on the test set. These results agree with the findings of several prior studies, such as [46], that found the relationship of mobility to viral transmission to be very dependent on the time of the year, with two major phases, one before and one after June 2020.

5.1.2 Experiment 2: Predicting tomorrows (D+1) case counts using 14 days of prior data

Objective

This experiment aims to develop IDT models that accurately predict COVID case counts one day in advance (D+1) using 14 days of prior data (D-14 to D-0) and then analyze

```
DOY|D-6 <= 257.5 :
|   Mobility|D-6 <= -0.276 :
|   |   age|D-5 <= 55.123 :
|   |   |   DOY|D-6 <= 92.5 : LM1 (11/12.265%)
|   |   |   DOY|D-6 >  92.5 : LM2 (40/12.915%)
|   |   age|D-5 >  55.123 :
|   |   |   DOY|D-6 <= 93.5 : LM3 (6/7.102%)
|   |   |   DOY|D-6 >  93.5 :
|   |   |   |   avg_wind_speed|D-3 <= 19.5 : LM4 (8/7.582%)
|   |   |   |   avg_wind_speed|D-3 >  19.5 :
|   |   |   |   |   Mobility|D-4 <= -0.562 : LM5 (2/0.337%)
|   |   |   |   |   Mobility|D-4 >  -0.562 : LM6 (4/1.26%)
|   Mobility|D-6 >  -0.276 :
|   |   avg_temperature|D-0 <= 20.6 :
|   |   |   DOY|D-6 <= 244.5 :
|   |   |   |   Mobility|D-6 <= -0.2 :
|   |   |   |   |   Mobility|D-4 <= -0.24 : LM7 (8/4.599%)
|   |   |   |   |   Mobility|D-4 >  -0.24 : LM8 (5/4.595%)
|   |   |   |   Mobility|D-6 >  -0.2 :
|   |   |   |   |   Mobility|D-4 <= -0.014 :
|   |   |   |   |   |   avg_temperature|D-5 <= 23.8 : LM9 (18/5.416%)
|   |   |   |   |   |   avg_temperature|D-5 >  23.8 :
|   |   |   |   |   |   |   age|D-2 <= 36.381 : LM10 (3/2.52%)
|   |   |   |   |   |   |   age|D-2 >  36.381 : LM11 (2/2.694%)
|   |   |   |   |   Mobility|D-4 >  -0.014 : LM12 (9/5.3%)
|   |   |   DOY|D-6 >  244.5 :
|   |   |   |   avg_relative_humidity|D-0 <= 67 :
|   |   |   |   |   Mobility|D-1 <= -0.096 : LM13 (4/2.536%)
|   |   |   |   |   Mobility|D-1 >  -0.096 : LM14 (2/2.357%)
|   |   |   |   avg_relative_humidity|D-0 >  67 : LM15 (6/3.367%)
|   |   avg_temperature|D-0 >  20.6 :
|   |   |   maleperc|D-5 <= 0.515 :
|   |   |   |   DOY|D-6 <= 184.5 : LM16 (18/6.821%)
|   |   |   |   DOY|D-6 >  184.5 :
|   |   |   |   |   DOY|D-6 <= 226.5 :
|   |   |   |   |   |   DOY|D-6 <= 191 : LM17 (5/3.996%)
|   |   |   |   |   |   DOY|D-6 >  191 : LM18 (13/3.018%)
|   |   |   |   |   DOY|D-6 >  226.5 :
|   |   |   |   |   |   avg_wind_speed|D-2 <= 15.75 : LM19 (4/4.841%)
|   |   |   |   |   |   avg_wind_speed|D-2 >  15.75 : LM20 (4/0.996%)
|   |   |   maleperc|D-5 >  0.515 : LM21 (25/4.615%)
DOY|D-6 >  257.5 :
|   DOY|D-6 <= 299.5 :
|   |   DOY|D-6 <= 285.5 :
|   |   |   IRH|D-3 <= 45.079 :
|   |   |   |   avg_relative_humidity|D-1 <= 68.5 : LM22 (11/4.863%)
|   |   |   |   avg_relative_humidity|D-1 >  68.5 :
|   |   |   |   |   avg_relative_humidity|D-4 <= 71.25 : LM23 (6/1.97%)
|   |   |   |   |   avg_relative_humidity|D-4 >  71.25 : LM24 (3/1.982%)
|   |   |   IRH|D-3 >  45.079 :
|   |   |   |   Mobility|D-2 <= -0.101 : LM25 (4/10.935%)
|   |   |   |   Mobility|D-2 >  -0.101 : LM26 (4/10.874%)
|   |   DOY|D-6 >  285.5 :
|   |   |   Mobility|D-6 <= -0.181 : LM27 (3/6.373%)
|   |   |   Mobility|D-6 >  -0.181 : LM28 (11/10.182%)
|   DOY|D-6 >  299.5 : LM29 (29/17.823%)
```

Figure 5.2: Experiment 1: 7-day input IDT model predicting Daily case counts for D+1.

these models to determine the most important input attributes. We have increased the prior number of days from 7 to 14 as the correlations show that the lag between input variables and the COVID-19 case counts can be as much as 14 days.

Data and Methods

For this experiment, the model used 306 examples from March 1, 2020, until December 31, 2020. The training set was composed of 268 examples from March 1, 2020, to November 23, 2020, with 38 test examples from November 24, 2020, to December 31, 2020. Models were developed with and without the autoregressive Daily case count variables and tested on the same independent test set until the lowest MAPE was determined.

Results and Discussion

The best WEKA M5P IDT models were developed with M=4 and Linear Regression Models at leaves of trees. The best model using all input variables, including the autoregressive variables, produced an MAE = 98.29 and a MAPE = 14.8% on the independent test set (based on a daily mean case count of 662 over the test set). The best IDT model using all input variables except the autoregressive variables produced an MAE = 103.95 and a MAPE = 15.67% on the independent test set (based on a daily mean case count of 662 over the test set).

Figure 5.1 shows the graph comparing actual versus predicted values for IDTs using 7 days and 14 days of prior data. Figure 5.3 shows a decision tree when using all but the autoregressive variables to forecast Daily case counts. The most important variables are similar to the 7-day model (DOY, Mobility (Movement_rel_to_baseline), Age, IRH, and avg_temperature (outdoor)). However, we see a heavy emphasis on values from days D-13 and D-12. For example, the most important variable is the DOY from 13 days earlier (D-13), with a major break based on it being September 7 (250th day); this is followed by focusing on the amount of mobility that occurred 13 days earlier (D-13), if the date is before September 7. This makes sense since a major increase in case counts occurs in early September, and the 13-day lead time would provide the opportunity for the virus to incubate and generate symptoms.

```
DOY|D-13 <= 250.5 :
|   Mobility|D-13 <= -0.312 :
|   |   age|D-12 <= 56.066 : LM1 (44/12.978%)
|   |   age|D-12 >  56.066 :
|   |   |   maleperc|D-1 <= 0.436 :
|   |   |   |   Mobility|D-4 <= -0.562 : LM2 (2/0.308%)
|   |   |   |   Mobility|D-4 >  -0.562 :
|   |   |   |   |   maleperc|D-2 <= 0.405 : LM3 (4/1.153%)
|   |   |   |   |   maleperc|D-2 >  0.405 : LM4 (3/4.398%)
|   |   |   maleperc|D-1 >  0.436 : LM5 (9/8.072%)
|   Mobility|D-13 >  -0.312 :
|   |   avg_temperature|D-4 <= 20.57 :
|   |   |   DOY|D-13 <= 237 :
|   |   |   |   Mobility|D-5 <= -0.24 :
|   |   |   |   |   maleperc|D-3 <= 0.461 :
|   |   |   |   |   |   Mobility|D-13 <= -0.218 : LM6 (3/2.773%)
|   |   |   |   |   |   Mobility|D-13 >  -0.218 : LM7 (3/3.711%)
|   |   |   |   |   maleperc|D-3 >  0.461 : LM8 (9/1.495%)
|   |   |   |   Mobility|D-5 >  -0.24 :
|   |   |   |   |   age|D-12 <= 40.72 :
|   |   |   |   |   |   avg_relative_humidity|D-7 <= 78.5 : LM9 (11/4.836%)
|   |   |   |   |   |   avg_relative_humidity|D-7 >  78.5 : LM10 (3/0.769%)
|   |   |   |   |   age|D-12 >  40.72 :
|   |   |   |   |   |   dayofweek|D-10 <= 4.5 :
|   |   |   |   |   |   |   max_wind_gust|D-2 <= 47.152 : LM11 (7/2.948%)
|   |   |   |   |   |   |   max_wind_gust|D-2 >  47.152 :
|   |   |   |   |   |   |   |   maleperc|D-12 <= 0.516 : LM12 (2/2.774%)
|   |   |   |   |   |   |   |   maleperc|D-12 >  0.516 : LM13 (3/1.906%)
|   |   |   |   |   |   dayofweek|D-10 >  4.5 : LM14 (4/0.267%)
|   |   |   DOY|D-13 >  237 :
|   |   |   |   DOY|D-13 <= 243.5 : LM15 (5/3.813%)
|   |   |   |   DOY|D-13 >  243.5 :
|   |   |   |   |   Mobility|D-12 <= -0.106 : LM16 (2/2.158%)
|   |   |   |   |   Mobility|D-12 >  -0.106 : LM17 (5/2.662%)
|   |   avg_temperature|D-4 >  20.57 :
|   |   |   avg_temperature|D-7 <= 20.4 :
|   |   |   |   age|D-7 <= 35.429 : LM18 (5/4.939%)
|   |   |   |   age|D-7 >  35.429 : LM19 (14/6.482%)
|   |   |   avg_temperature|D-7 >  20.4 :
|   |   |   |   IRH|D-10 <= 48.179 : LM20 (27/6.327%)
|   |   |   |   IRH|D-10 >  48.179 : LM21 (25/3.894%)
DOY|D-13 >  250.5 :
|   DOY|D-13 <= 292.5 :
|   |   DOY|D-13 <= 278.5 :
|   |   |   IRH|D-3 <= 45.079 : LM22 (20/8.277%)
|   |   |   IRH|D-3 >  45.079 :
|   |   |   |   Mobility|D-2 <= -0.101 : LM23 (4/10.012%)
|   |   |   |   Mobility|D-2 >  -0.101 : LM24 (4/5.276%)
|   |   DOY|D-13 >  278.5 :
|   |   |   Mobility|D-6 <= -0.181 : LM25 (3/5.835%)
|   |   |   Mobility|D-6 >  -0.181 : LM26 (11/9.323%)
|   DOY|D-13 >  292.5 : LM27 (36/15.23%)
```

Figure 5.3: Experiment 2: 14-day input IDT model predicting Daily case counts for D+1.

In comparison, a model developed without DOY but with Mobility produces a high MAE = 210.29. This is significantly worse than the persistence model, with an MAE of 88.09 cases and a MAPE of 13.28%. At the same time, a model developed with DOY but without Mobility performed with an MAE = 139.22, which is also worse than the persistence model. Furthermore, a model developed with neither Mobility nor DOY had an MAE = 200.9 on the test set. These results agree with the models developed using only seven daysâ worth of prior data; it shows that without DOY, Mobility confounds efforts to create a general model from the available data.

5.2 Predicting COVID-19 using a Standard ANN

5.2.1 Experiment 3: Predicting tomorrows (D+1) case counts using a STL ANN

This section records the approach and results of using deep feedforward artificial neural networks and an input window that moves across the spatial-temporal data to capture crucial independent variables over space and time to predict future COVID case counts. A window of fourteen (14) days' worth of daily social mobility, weather, and air quality data, along with the autoregressive terms, is used to predict the number of case counts for the next day (D+1).

Objective

This experiment aims to develop a Multi-Layer Perceptron (MLP) model that accurately predicts Daily COVID case counts for tomorrow (D+1) using a sequence of 14 days' worth of prior data. The main intention of using MLP for time series forecasting is to test how well the standard neural network works against the widely used time series forecasting algorithms such as the LSTMs and CNNs.

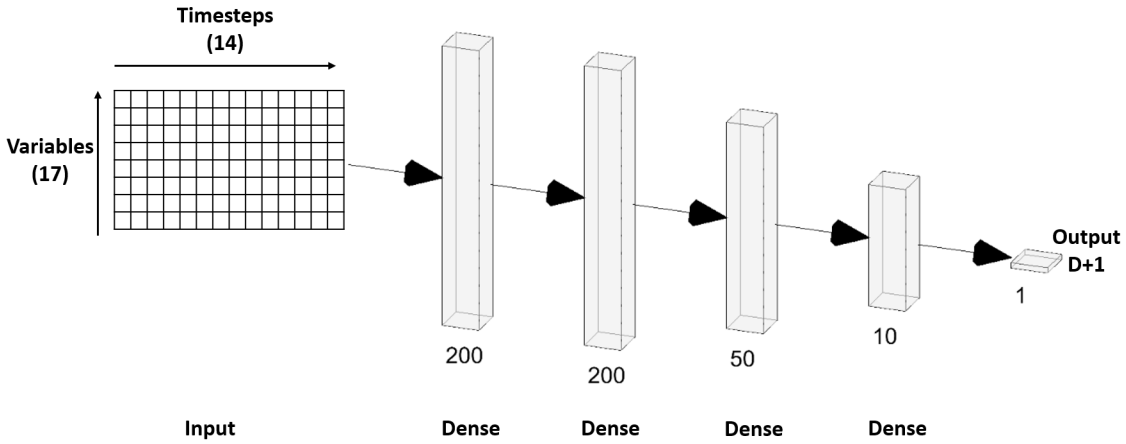


Figure 5.4: Standard STL ANN architecture.

Data and Methods

This experiment used 306 examples from March 1, 2020, until December 31, 2020, wherein all the variables listed in Section 4.4 were used. The training set was composed of 245 examples, from March 1, 2020, until October 31, 2020, validation set with 23 examples from November 1, 2020, to November 23, 2020, and 38 test examples from November 24, 2020, till December 31, 2020.

After numerous trials of network configurations and hyperparameter settings, the best architecture, as shown in the Figure 5.4, was a deep neural network consisting of 4 Dense layers followed by an output layer. The first two dense layer contain 200 nodes each followed by 50 and 10 nodes and a output layer. Rectified Linear Unit (ReLU) is used as the activation function in each of the dense layers. The linear activation function is used for the dense output layer to perform regression. This linear activation function of the output layer produces the COVID case count, which is an integer value. Adam optimizer was used with a variable learning rate of 0.0001, and the cost function was the mean absolute error (MAE). The networks were trained for 200 epochs, with a batch size of 1.

Results and Discussion

The ANN models have an MAE of 98.62 cases and a MAPE of 14.87%. Figure 5.5 depicts the graph of actual vs predicted case counts which shows that the majority of the test data

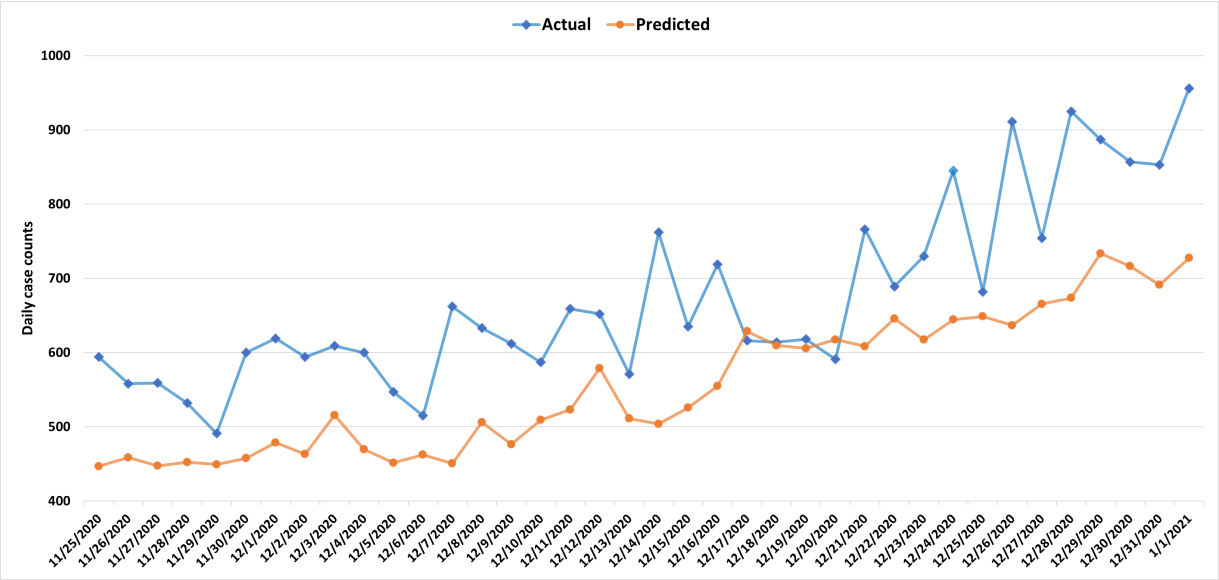


Figure 5.5: Experiment 3: Standard STL ANN model actual versus predicted case counts for D+1.

is underpredicted by the ANN model.

5.3 Predicting COVID-19 using CNNs

5.3.1 Experiment 4: Predicting tomorrows (D+1) case counts using a STL CNN

This section records the approach and results of using deep feedforward artificial neural networks and an input window that moves across the spatial-temporal data capturing what is considered important independent variables over space and time to predict future COVID case counts. Fourteen (14) days' worth of daily social mobility, weather, and air quality data along with the autoregressive terms are used as a window to predict the next day (D+1). Recently convolutional neural networks (CNN) have been used effectively for data prepared and used in this windowing manner. The disadvantage of CNNs, as compared to LSTMs, is that the size of the window and how far it reaches back in time to capture salient variables (features) must be manually selected. CNNs do come with the advantage of being faster to

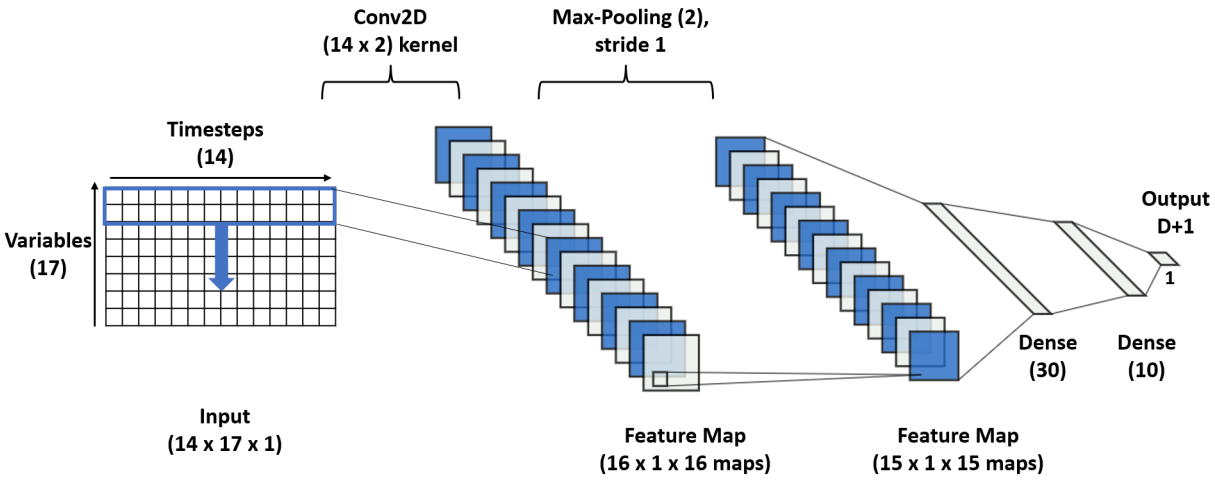


Figure 5.6: STL CNN architecture.

train and test than the LSTMs.

Objective

This experiment aims to develop STL CNN models that accurately predict Daily COVID case counts for tomorrow (D+1) using a sequence of 14 days' worth of prior data.

Data and Methods

These models used 306 examples from March 1, 2020, until December 31, 2020. The training set was composed of 245 examples, from March 1, 2020, until October 31, 2020, validation set with 23 examples from November 1, 2020, to November 23, 2020, and 38 test examples from November 24, 2020, till December 31, 2020. These models used all variables listed in Section 4.4.

After numerous trials of network configurations and hyperparameter settings, the best architecture, as shown in Figure 5.6, was a deep convolutional neural network consisting of two blocks of layers; one convolutional block, and one fully connected block. The convolutional block contains a convolutional layer with a kernel of size 14 x 2, followed by a max-pooling layer. Max-pooling is performed after each convolutional layer with a filter size of 2 and a stride of 1. Both convolutional and max-pooling layers create feature maps of size

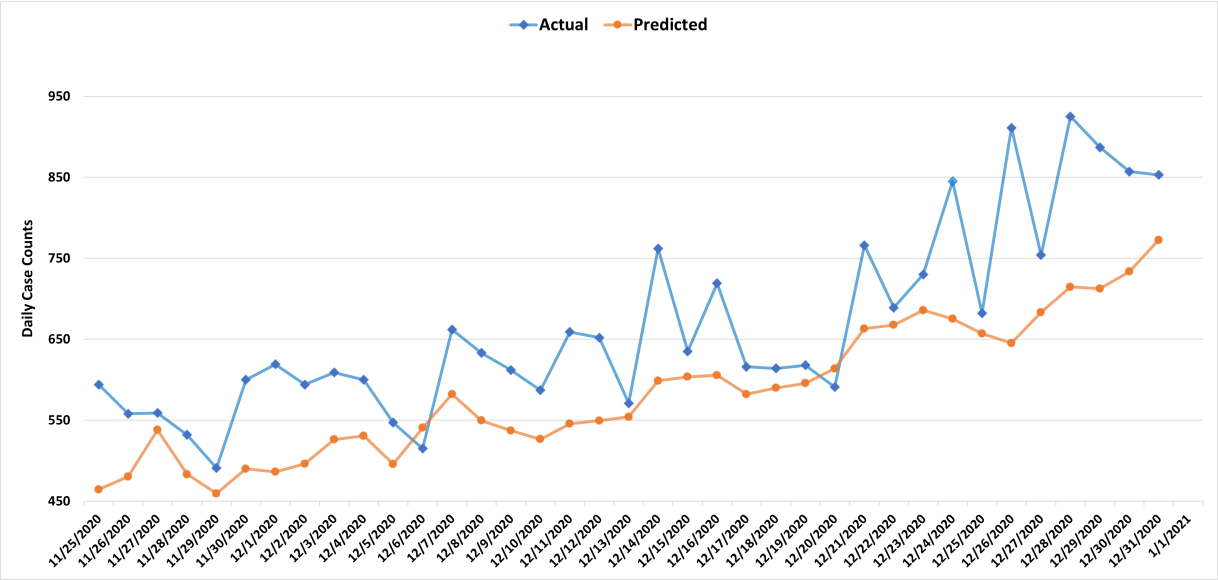


Figure 5.7: Experiment 4: STL CNN model actual versus predicted case counts for D+1.

16 and 15 respectively, which are connected to a fully connected layer in which two dense layers of 30 and 10 nodes each are present. Rectified Linear Unit (ReLU) is used as the activation function in each convolutional block and in the two dense layers after max-pooling. A linear activation function is used for the dense output layer to perform regression. This linear activation function of the output layer produces the COVID case count, which is an integer value. Adam optimizer was used with a variable learning rate of 0.0001, and the cost function was the mean absolute error (MAE). The networks were trained for 1200 epochs, with a batch size of 1.

Results and Discussion

The CNN models have an MAE of 84.6 cases and a MAPE of 12.08%. Figure 5.7 depicts the graph of predicted versus actual case counts. We notice from the graph that the STL CNN model often underpredicts the case counts. Compared to the best IDT model in Experiment 2 (see Section 5.1.2), which has a MAE of 98.29 and MAPE of 14.8%, the CNN models do not perform better. Based on a p-value of 6.89, there is no significant statistical difference between the two models.

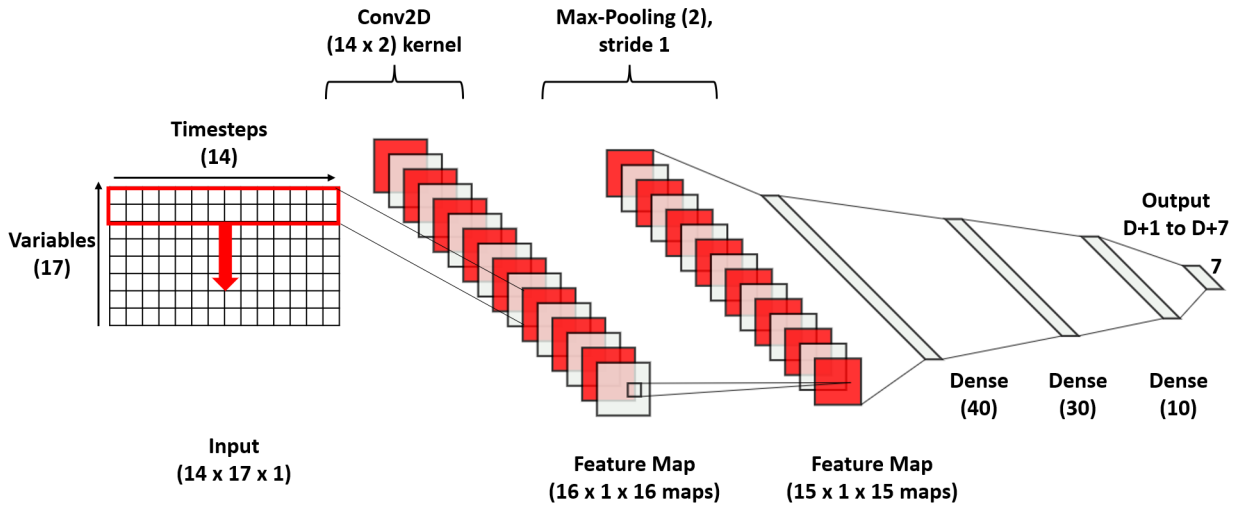


Figure 5.8: MTL CNN architecture.

5.3.2 Experiment 5: Predicting next 7 days (D+1 to D+7) case counts using a MTL CNN

Objective

This experiment aims to develop MTL CNN models that accurately predict the Daily case counts for the next 7 days (D+1 through D+7) using a sequence of 14 days' worth of prior data. This is an example of a Multiple Task Learning (MTL) model.

Data and Methods

The data used in this experiment is the same data used in the above experiment. After several trials of network configurations and hyperparameter settings, the best architecture, as shown in Figure 5.8, was a deep network consisting of two blocks of layers; one convolutional block, and one fully connected block. The convolutional block contains one convolutional layer with a kernel of size 14 x 2, followed by a max-pooling layer. Max-pooling is performed after each convolutional layer with a filter size of 2 and a stride of 1. Both convolutional and max-pooling layers create feature maps of size 16 and 15 respectively, which are connected to a fully connected layer in which three dense layers of 40, 30 and 10 nodes each are present. The output layer contains seven nodes for the 7 days (D+1 to D+7). A Rectified Linear Unit

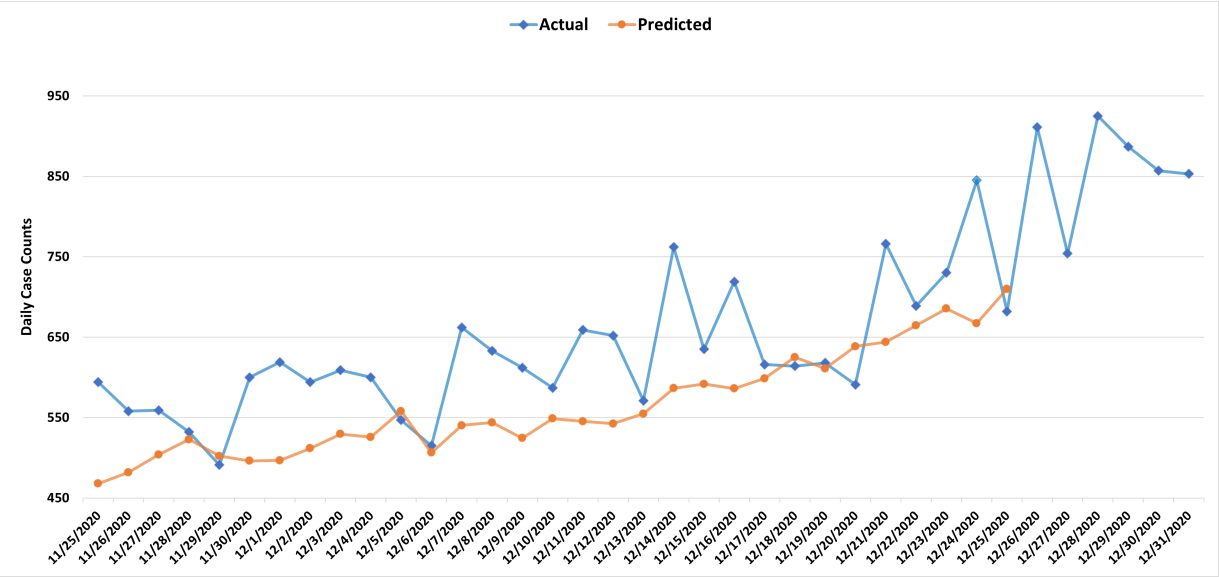


Figure 5.9: Experiment 5: CNN model actual versus predicted case counts for D+1.

(ReLU) is used as the activation function in each dense hidden layer. The output nodes use a linear activation function to produce the COVID case count, which we round to an integer value. Model development was repeated five times using different random initial weights, and the average MAE and MAPE were calculated.

Results and Discussion

Figure 5.9 depicts the graph of predicted versus actual case counts for D+1. The model has an average MAE of 93.75 cases and MAPE of 13.94% overall 7 days (D+1 through D+7) and an MAE of 80.08 and MAPE of 11.9% for the D+1 prediction. This is an excellent result that demonstrates the value of using MTL models where multiple related outputs are predicted by the same neural network.

Figure 5.10 shows the MAE values for all the days D+1 to D+7. We can see that the first 3 days D+1 to D+3 perform well compared to the STL CNN model, which has a MAE of 84.6, with MAPE of 12.08% on D+1, and a 95% CI of 5.52 based on five repeated model runs.

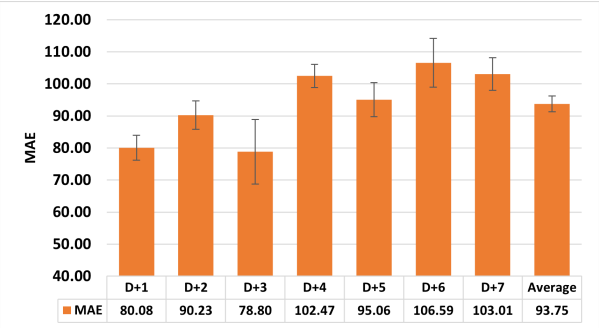


Figure 5.10: Experiment 5: MAE for all 7 days D+1 to D+7, error bars showing 95% CI.

5.3.3 Experiment 6: Predicting tomorrows (D+1) 7-day average case counts using a STL CNN

Objective

This experiment aims to develop STL CNN models that accurately predict 7 day average COVID case counts for tomorrow (D+1) using a sequence of 14 days' worth of prior data.

Data and Methods

These models used 306 examples from March 1, 2020, until December 31, 2020. The training set was composed of 245 examples, from March 1, 2020, until October 31, 2020, validation set with 23 examples from November 1, 2020, to November 23, 2020, and 38 test examples from November 24, 2020, till December 31, 2020. These models used all variables listed in Section 4.4.

The architecture used in this model is the same as Experiment 4 (see Section 5.3.1). The model development was repeated five times using different random initial weights, and the average MAE and MAPE were calculated.

Results and Discussion

The CNN models predicting the 7-day average values have an MAE of 41.8 cases and a MAPE of 6.52%, with a 95% CI of 4.14. These values are the average over five consecutive runs set with different random initial weights. Comparing these results with Experiment 4

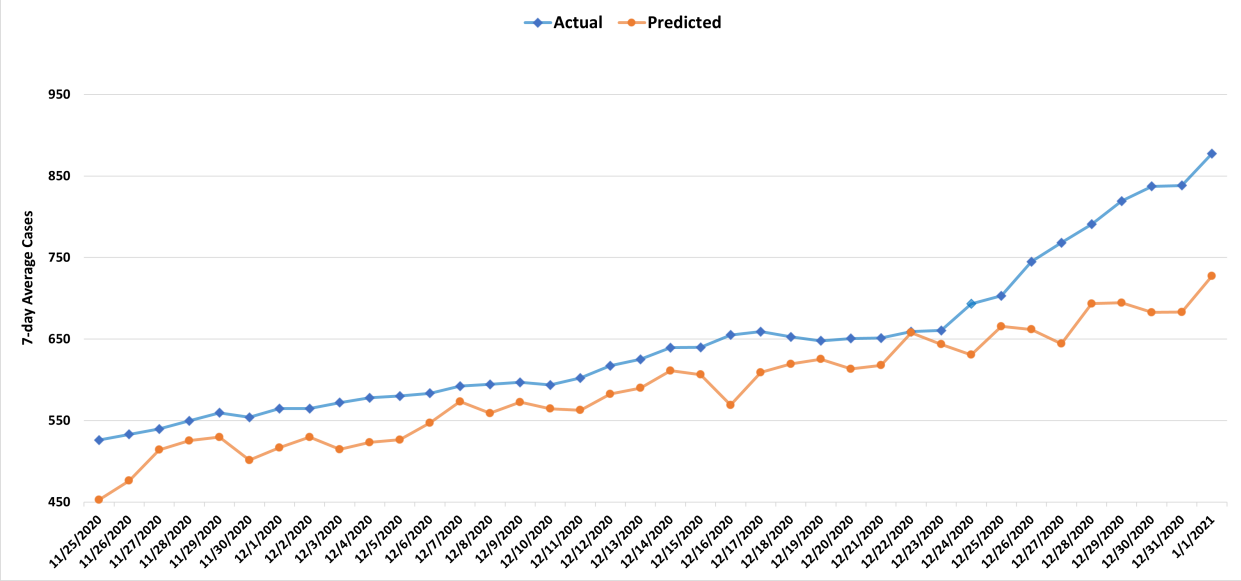


Figure 5.11: Experiment 6: CNN model actual versus predicted 7-day average case counts for D+1.

(see Section 5.3.1), where we predict the Daily case counts of D+1 using CNNs, the 7-day average models perform better because as shown in Figures 4.1 and 5.11, the sequence of 7-day average values are smooth compared to the sequence of case counts.

5.3.4 Experiment 7: Predicting next 7 days (D+1 to D+7) 7-day average case counts using a MTL CNN

Objective

This experiment aims to develop MTL CNN models that accurately predict 7-day average case counts for the next 7 days (D+1 through D+7) using a sequence of 14 days' worth of prior data.

Data and Methods

The data and the architecture used in this experiment are the same which are used in Experiment 5 (see Section 5.3.2). Model development was repeated five times using different random initial weights, and the average MAE and MAPE were calculated.

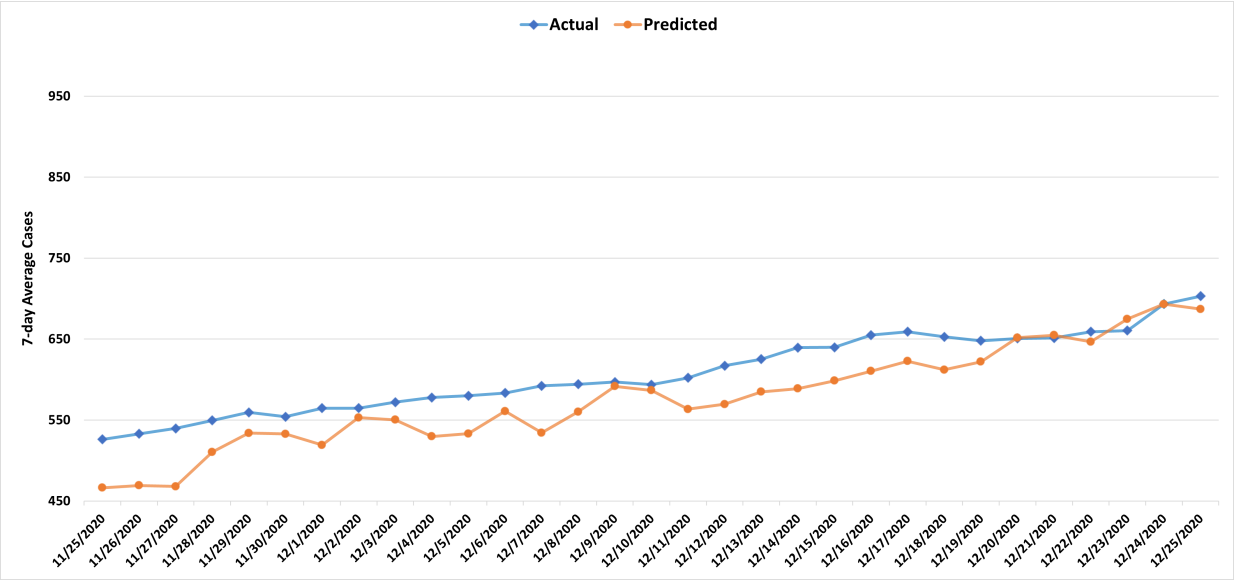


Figure 5.12: Experiment 7: CNN model actual versus predicted case counts for D+1 on 7 -day average cases.

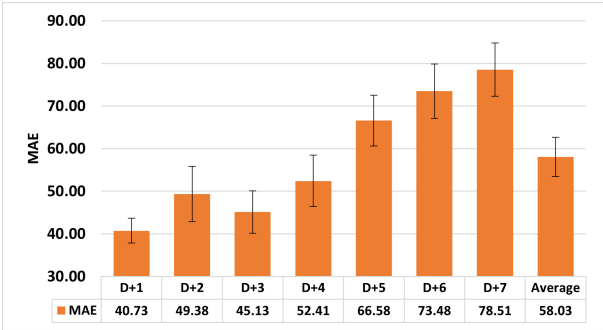


Figure 5.13: Experiment 7: MAE for all 7 days D+1 to D+7, error bars showing 95% CI.

Results and Discussion

Figure 5.12 depicts the graph of predicted versus actual case counts for D+1. This is an excellent result that demonstrates the value of using MTL models where multiple related outputs are predicted by the same neural network. The model has an average MAE of 58.03 cases and MAPE of 9.35% overall for 7 days (D+1 through D+7) and an MAE of 40.73 and MAPE of 6.89%, and a 95% CI of 4.41 for the D+1 prediction. Figure 5.13 shows the MAE values over all the 7 days.

5.4 Predicting COVID-19 using LSTM Networks

5.4.1 Experiment 8: Predicting tomorrows (D+1) case counts using a STL LSTM

The following experiments use a Long Short-Term Memory (LSTM) network. The LSTMs do not require the data to be prepared as a series of spatial-temporal windows into the past and nearby regions. These have the advantage of learning to select the most important variables from the past depending upon the most recent network context and predictions. We compare the LSTM model performance to a baseline persistence model and CNN models, developed and tested on the same data.

Objective

This experiment aims to develop a STL LSTM model that accurately predicts Daily COVID case counts for tomorrow (D+1) using a sequence of 14 days' worth of data.

Data and Methods

These models used 306 examples from March 1, 2020, until December 31, 2020. The list of input variables used is shown in Section 4.4. The training set was composed of 245 examples from March 1 to October 31, 2020, a validation set with 23 examples from November 1 to November 23, 2020, and 38 test examples from November 24 to December 31, 2020.

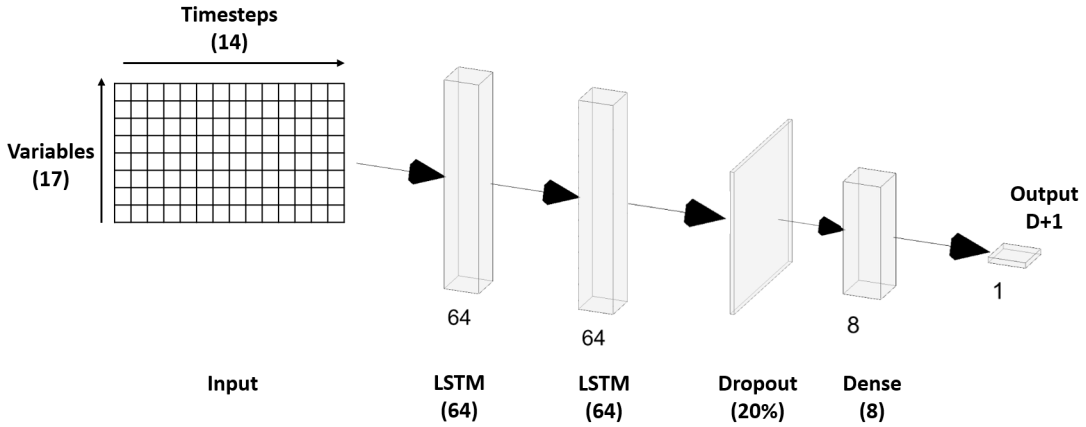


Figure 5.14: STL LSTM architecture.

After numerous trials of network configurations and hyperparameter settings, the best architecture, as shown in Figure 5.14, was a deep network consisting of 2 LSTM layers followed by one dense hidden layer of 8 nodes, along with a dropout of 20% in between LSTM and dense layer and followed by an output layer. Each LSTM layer contained 64 LSTM nodes. A Rectified Linear Unit (ReLU) is used as the activation function for each of the dense hidden layer nodes. The output node uses a linear activation function to produce the COVID case count, rounded to an integer value. Model development was repeated five times using different random initial weights, and the average MAE and MAPE were calculated.

Results and Discussion

LSTM models have an MAE of 64.9 cases and a MAPE of 9.22%. Compared to Experiment 4 (see Section 5.3.1), which uses a deep STL CNN on the same training and testing data to predict D+1 Daily case counts, the STL LSTMs perform better. The CNN had a higher MAE of 84.6, and MAPE of 12.08%, and a 95% CI of 5.52. Figure 5.15 shows the graph comparing actual versus predicted values for LSTM and CNN predicting D+1. As we can see from the graph, the LSTM follows the actual values more precisely than the CNN model. We believe that the ability of LSTMs to retain memory from its prior timesteps has helped it to perform better than the CNN.

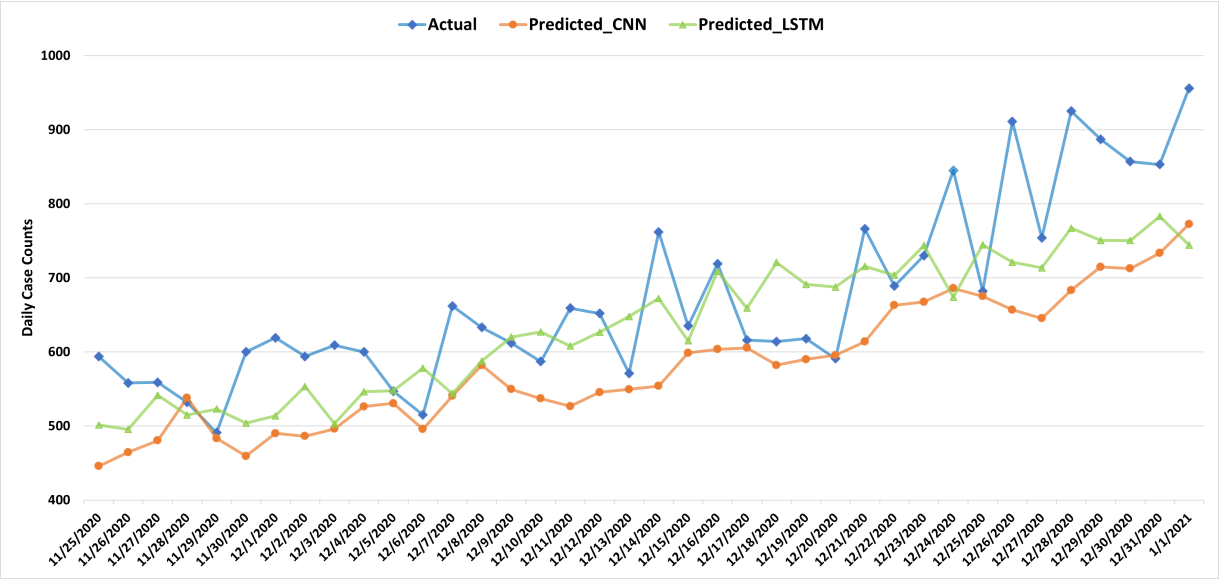


Figure 5.15: Experiment 8: LSTM versus CNN model, actual versus predicted case counts for D+1.

5.4.2 Experiment 9: Predicting next 7 days (D+1 to D+7) case counts using a MTL LSTM

Objective

This experiment aims to develop MTL LSTM models that accurately predict the Daily case counts for the next 7 days (D+1 through D+7) using a sequence of 14 days' worth of prior data.

Data and Methods

This model uses 306 examples from March 1, 2020, until December 31, 2020, as in Experiment 8. The training set comprises 245 examples from March 1 to October 31, 2020, the validation set has 23 examples from November 1 to November 23, 2020, and the test set has 38 test examples from November 24 to December 31, 2020.

After several trials of network configurations and hyperparameter settings, the best architecture, as shown in Figure 5.16, was a deep network consisting of 2 LSTM layers followed by two dense hidden layers of 128 and 64 nodes and then an output layer containing 7 nodes

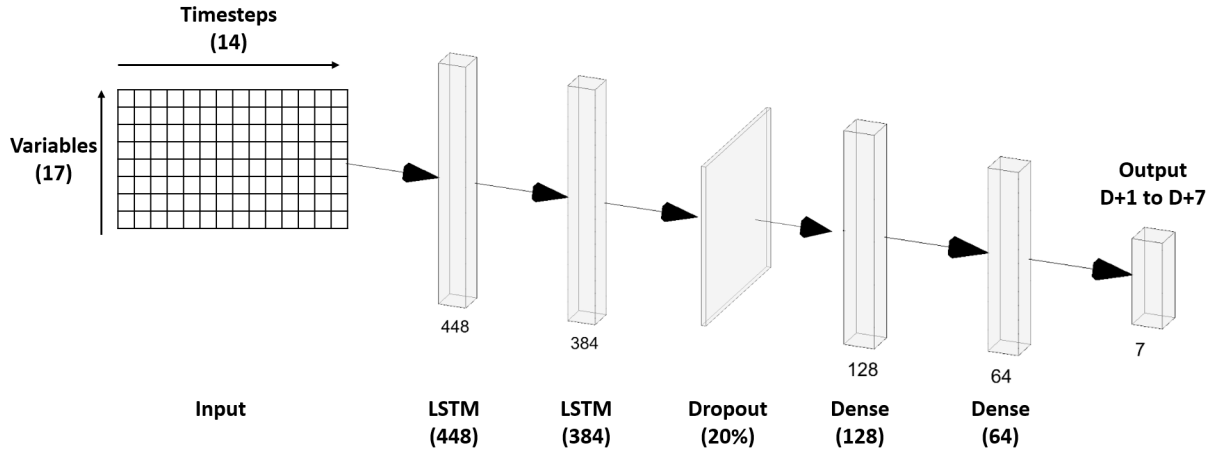


Figure 5.16: MTL LSTM architecture.

for the 7 days (D+1 through D+7). The first LSTM layer contains 448 LSTM nodes, and the second LSTM layer has 384. A dropout of 20% has been introduced between the LSTM and the dense layers to prevent the model from overfitting. A Rectified Linear Unit (ReLU) is used as the activation function in each of the dense hidden layers, and the output nodes use a linear activation function to produce the COVID case count, which we round to an integer value. Model development was repeated five times using different random initial weights, and the average MAE and MAPE were calculated.

Results and Discussion

The model has an average MAE of 62.56 cases and MAPE of 9.44%, with a 95% CI of 6.68 over all 7 days (D+1 through D+7) and an MAE of 52.51 and MAPE of 8.6%, with a 95% CI of 1.07 for the D+1 prediction. Figure 5.17 depicts the graph of predicted versus actual case counts for D+1. This is an excellent result that demonstrates the value of using MTL models where multiple related outputs are predicted by the same neural network.

Figure 5.18 shows the MAE values over all days. However, we note that the test set is slightly smaller (ending on Dec 25) because the last D+1 date is 6 days prior to the end of the test period. If we test the prior STL LSTM model on the same data, it has an MAE of 64.9 and MAPE of 9.22% which is still less than the MTL model. Based on the p-value of 0.06, there is no statistical difference between these two models at 95% CI. Compared with

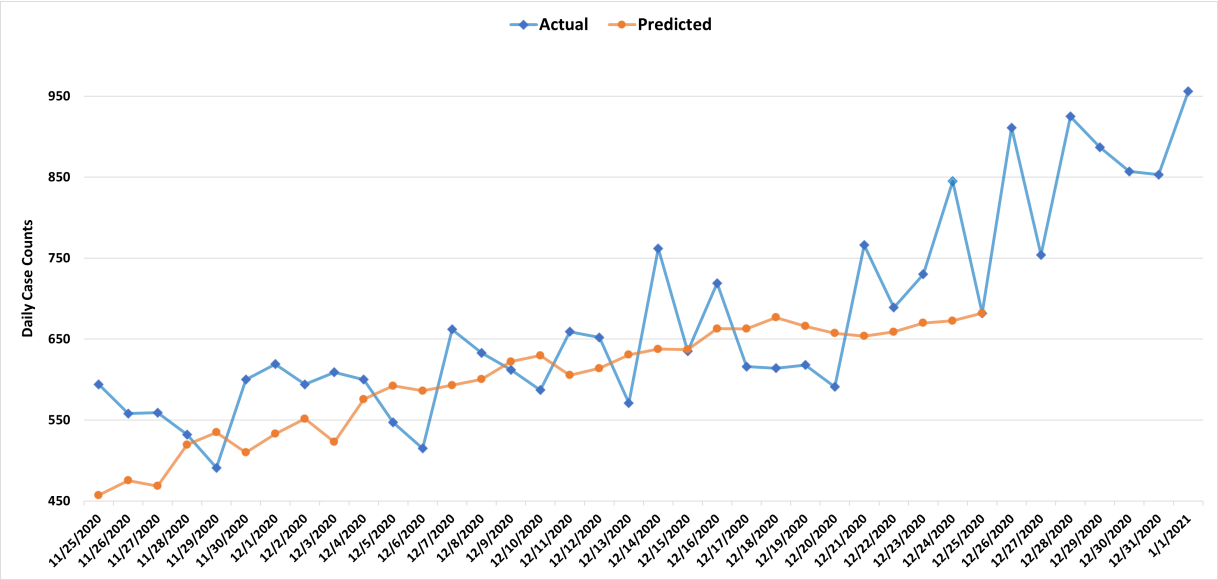


Figure 5.17: Experiment 9: LSTM model actual versus predicted case counts for D+1.

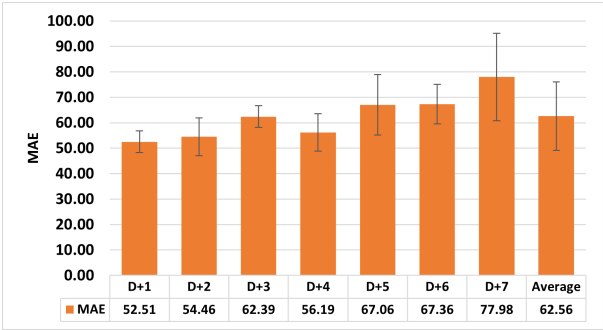
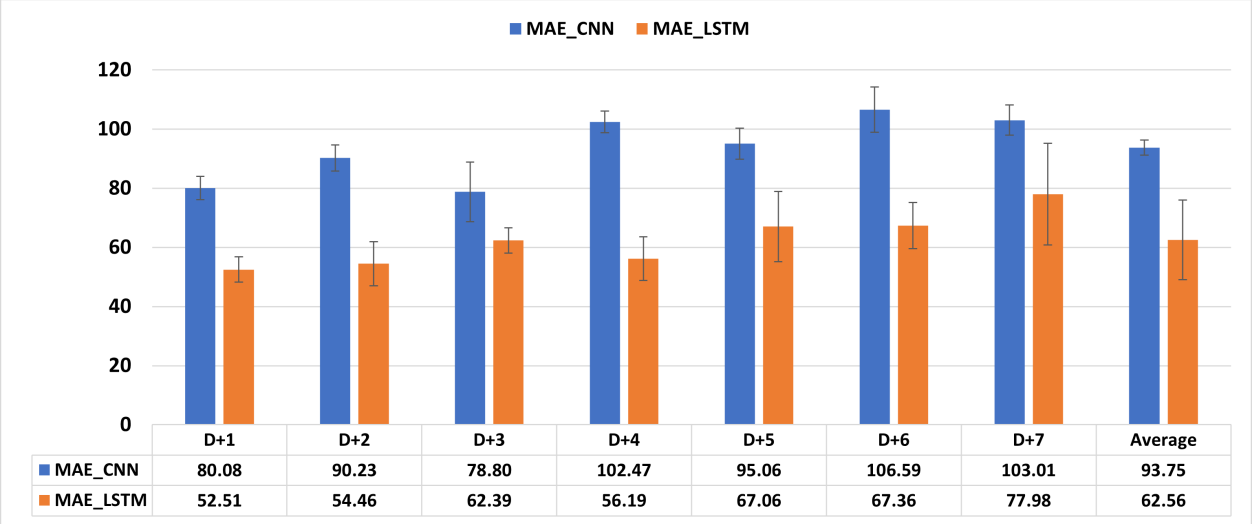


Figure 5.18: Experiment 9: MAE for all 7 days D+1 to D+7, error bars showing 95% CI.

the results of Experiment 5 (see Section 5.3.2) which uses MTL CNN for predicting 7 days of daily case counts, the current MTL LSTM models also perform better statistically based on a p-value of 0.001. Although the LSTM models have a higher variance based on the error bars in Figure 5.19, when it comes to the predicted values the MTL CNN models underpredict the case counts (see Figure 5.9), while the MTL LSTM model predict the actual case counts well. Thereby, the LSTM models are preferred.



	D+1	D+2	D+3	D+4	D+5	D+6	D+7	Average
MAE_CNN	80.08	90.23	78.80	102.47	95.06	106.59	103.01	93.75
MAE_LSTM	52.51	54.46	62.39	56.19	67.06	67.36	77.98	62.56

Figure 5.19: Experiment 9: MTL-LSTM versus CNN MAE values for all days, error bars showing 95% CI.

5.4.3 Experiment 10: Predicting tomorrows (D+1) 7-day average case counts using a STL LSTM

Objective

This experiment aims to develop STL LSTM models that accurately predict 7-day average COVID case counts for tomorrow (D+1) using a sequence of 14 days' worth of prior data.

Data and Methods

This experiment uses the same data that is described in Experiment 8 (see Section 5.4). The only difference is that the output variable daily case counts has been replaced with 7-day average case counts. The models' architecture is the same, as explained in Experiment 8 (see Section 5.4). The network is trained using an Adam optimizer with a learning rate of 0.0001. The training was performed for 300 epochs, with a batch size of 1.

Results and Discussion

The LSTM models predicting the 7-day average values have an MAE of 21.94 cases and a MAPE of 3.34%. Compared to Experiment 6 (see Section 5.3.3), a similar experiment

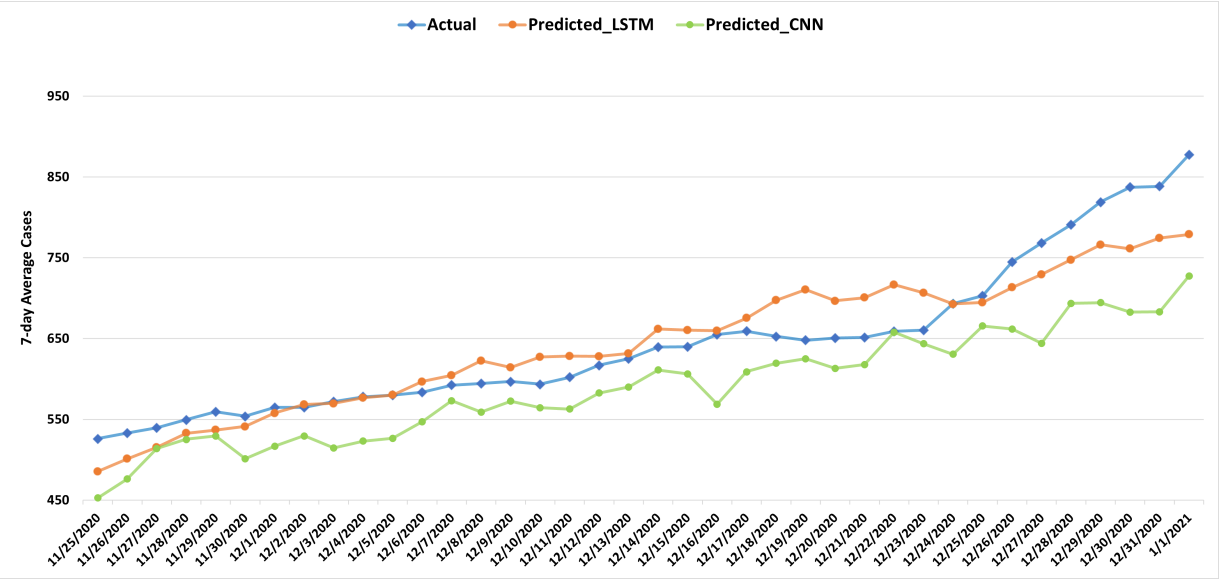


Figure 5.20: Experiment 10: STL LSTM versus CNN actual vs predicted 7-day average case counts for D+1.

performed using CNNs, the current LSTM model performs better. Figure 5.20 shows the actual and the predicted graphs for both LSTM and CNN models predicting D+1 7-day average case counts. As we can see in the graph, the CNN model tends to underpredict as compared to the LSTM model.

5.4.4 Experiment 11: Predicting next 7 days (D+1 to D+7) 7-day average case counts using a MTL LSTM

Objective

This experiment aims to develop LSTM models that accurately predict the 7-day average case counts for the next 7 days (D+1 through D+7) using a sequence of 14 days' worth of prior data.

Data and Methods

This experiment uses the same data that is described in Experiment 8 (see Section 5.4). The only difference is that the output variable daily case counts has been replaced with the 7-day

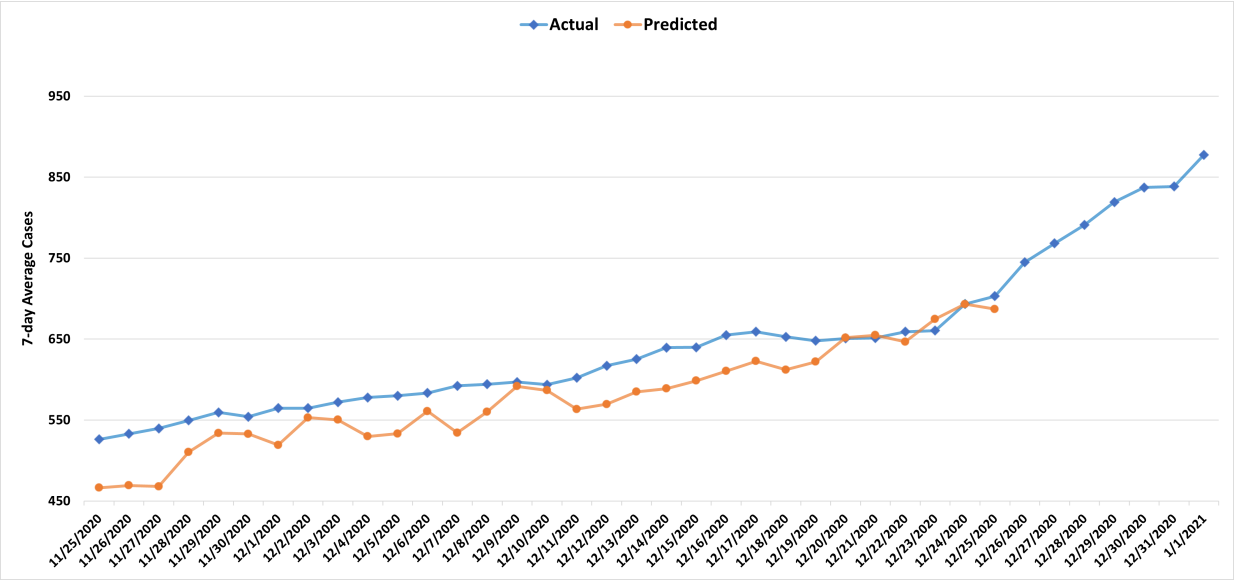


Figure 5.21: Experiment 11: LSTM model actual vs predicted 7-day average case counts for D+1.

average case counts. The model's architecture is the same, as explained in Experiment 8 (see Section 5.4). The network is trained using an Adam optimizer with a learning rate of 0.0001. The training was performed for 600 epochs, with a batch size of 1.

Results and Discussion

The model has an average MAE of 44.96 cases and MAPE of 7.45% over all 7 days (D+1 to D+7) and an MAE of 36.15 and MAPE of 5.97% for the D+1 prediction. Figure 5.21 depicts the graph of predicted versus actual case counts for D+1. However, we note that the test set is slightly smaller (ending on Dec 25) because the last D+1 date is 6 days prior to the end of the test period. Figure 5.22 shows the MAE values over all the predicted 7 days.

Compared with the results of Experiment 7 (see Section 5.3.4) which uses CNN for predicting 7 days of daily case counts, the LSTM models perform better. Based on a p-value of 0.049, we can say that these two models are statistically different, thereby the LSTM model is preferred. Figure 5.23 shows the comparision between MTL LSTM and MTL CNN (see Section 5.3.4) predicting 7-day average case counts over all the 7 days.

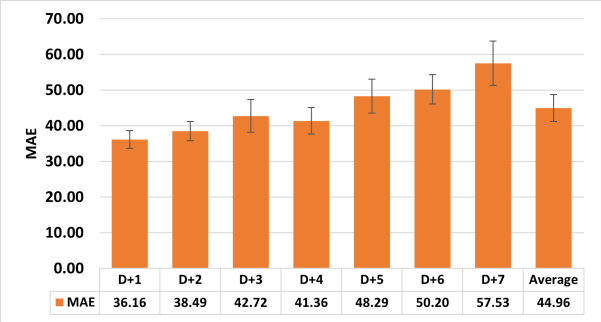


Figure 5.22: Experiment 11: MAE for all 7 days D+1 to D+7, error bars showing 95% CI.

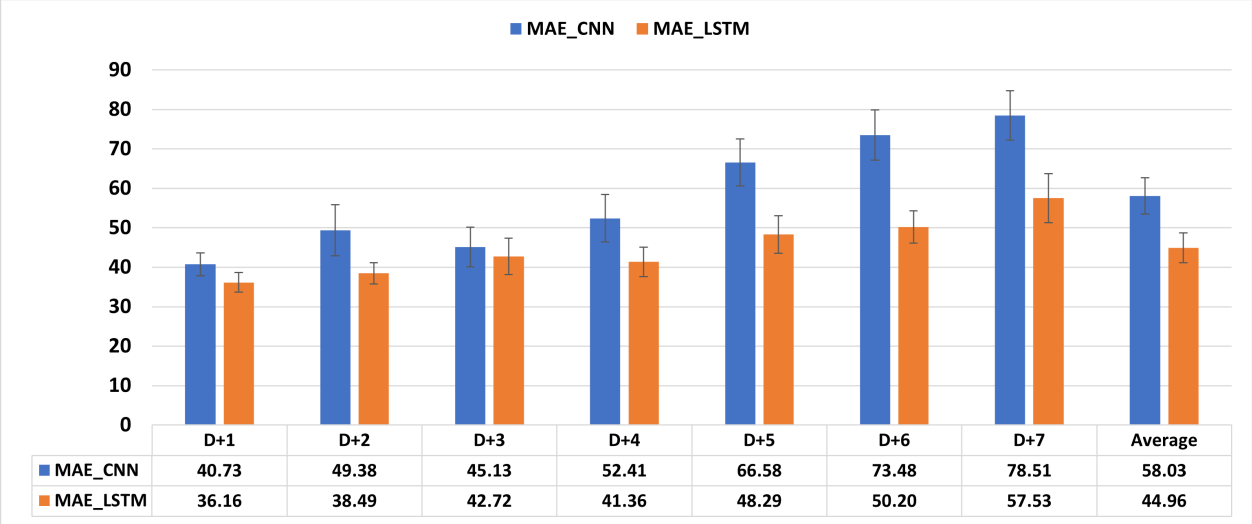


Figure 5.23: Experiment 11: LSTM vs CNN MAE values for all days, error bars showing 95% CI.

5.4.5 Experiment 12: Predicting tomorrows (D+1) Daily case counts using k-fold chronological cross validation

Objective

This experiment aims to emulate the iterative development and testing of COVID forecast models over a period of several windows of data. This is what would occur in a real-world scenario. We develop STL LSTM models to predict the Daily case count for tomorrow (D+1) as accurately as possible over two weeks using a sequence of 14 days' worth of prior data. We then move the window forward by two weeks.

Data and Methods

These models use the same data set of 306 examples from March 1, 2020, until December 31, 2020. A k-fold chronological cross-validation technique is used to develop robust models, where for this experiment k=5. This is meant to emulate the periodic development and use of predictive models in a real-world scenario as more data continues to arrive. The complete dataset is used to create 5 folds of training, validation, and test examples as one moves forward in time, beginning from March 1, 2020.

As shown in Table 5.1, the training and validation for the first fold data is from March 1 to Oct 14, and the test set is from Oct 15 to Oct 28. For the second fold, the training and validation data is from March 1 to October 28, and the test set is from Oct 29 to Nov 11. And so, on up until the last test set ending on Dec 24. The loss on the validation set is monitored using an early stopping technique to prevent overfitting. The independent test set is used to measure each model's performance, and the average MAE and MAPE are reported for each output. Table 5.1 and Figure 5.24 shows the division of data into 5-folds.

Table 5.1: Dataset divided for chronological 5-fold cross validation.

	Training	**Validation**	**Testing**
Fold 1	3/1 to 9/26	9/27 to 10/14	10/15 to 10/28
Fold 2	3/1 to 10/9	10/10 to 10/28	10/29 to 11/11
Fold 3	3/1 to 10/22	10/23 to 11/11	11/12 to 11/25
Fold 4	3/1 to 11/4	11/5 to 11/25	11/26 to 12/9
Fold 5	3/1 to 11/18	11/19 to 12/9	12/10 to 12/24

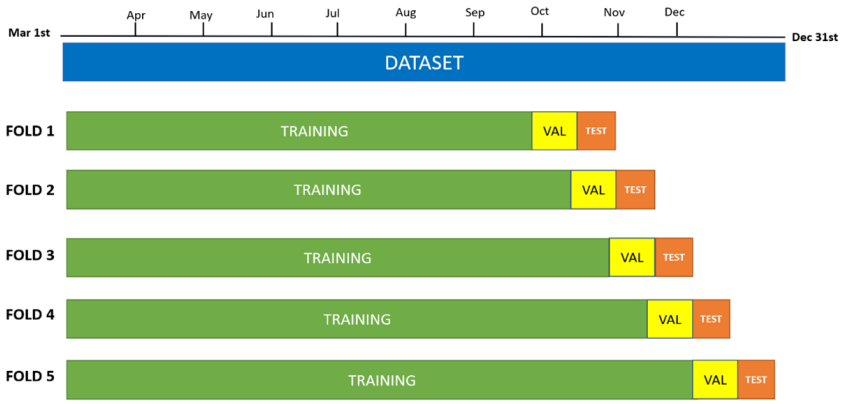


Figure 5.24: Experiment 12: The chronological 5-fold cross validation approach used.

The same network configuration and hyperparameter settings used in Experiment 8 (Section 5.4) were used for this experiment: 2 LSTM layers each of 64 nodes, followed by one dense hidden layer of 8 ReLU nodes, with a dropout layer in between and then the output layer. The output node uses a linear activation function to produce the COVID case count, rounded to an integer value. Model development was repeated five times using different random initial weights, and the average MAE and MAPE were calculated.

Results and Discussion

Table 5.2 shows the performance results of the 5-fold chronological cross-validation runs for the STL models that predict the Daily case counts for D+1. Figure 5.25 shows the graphs of predicted versus the actual number of cases for D+1 using these models. On average, the models were 90% accurate in predicting Daily Case counts over the Oct 15 to Dec 24 period. The model has acheived a average MAE of 45.4 and a MAPE of 9.2%. One can see from Figure 5.25 that the model generally follows the rise and fall of COVID cases over this period.

Table 5.2: Experiment 12: Performance of LSTM STL models over 5-folds predicting daily case counts.

	MAE	MAPE
Fold 1	28.90	8.99
Fold 2	55.37	12.96
Fold 3	43.85	8.78
Fold 4	37.13	6.30
Fold 5	61.77	9.15
Average	45.40	9.24
Stdev	13.33	2.39
95%CI	11.68	2.09

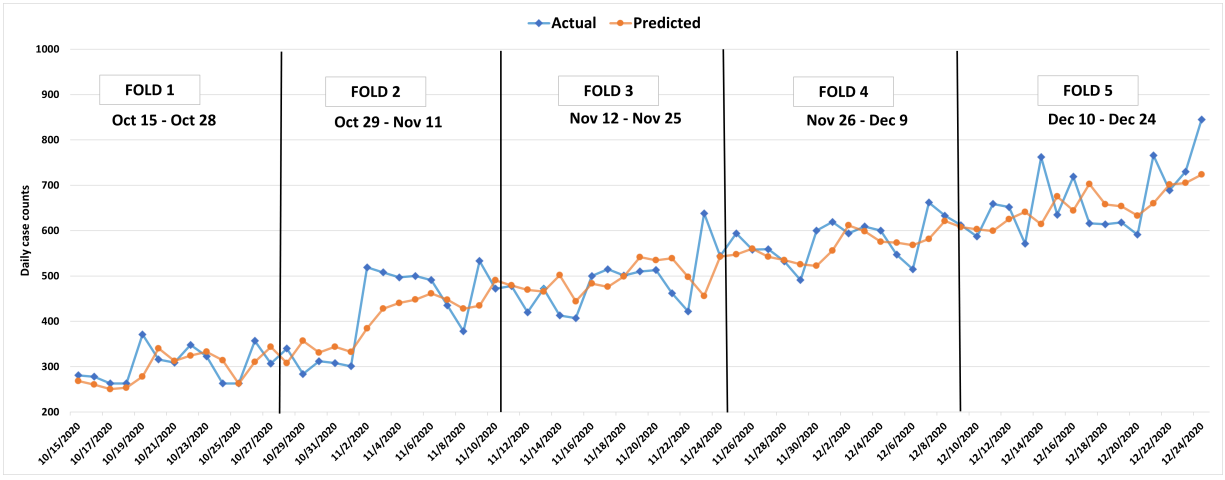


Figure 5.25: Experiment 12: LSTM models predicted versus actual case counts for D+1 over 5-folds predicting daily case counts.

5.4.6 Experiment 13: Predicting tomorrows (D+1) 7-day average case counts using k-fold chronological cross validation

Objective

The goal of this experiment is the same as the last but for the 7-Day average case count. That is to develop STL LSTM models to predict the 7-Day average case count for tomorrow (D+1) as accurately as possible over a period of two weeks using a sequence of 14 days' worth of prior data. The window is then moved forward by two weeks.

Data and Methods

These models use the same data set of 306 examples from March 1, 2020, until December 31, 2020. The same 5-fold chronological cross-validation technique is used; the complete dataset is used to create 5 folds of training, validation, and test examples as one move forward in time beginning from March 1, 2020. For each of the five runs, the loss on the validation set is monitored using an early stopping technique to prevent overfitting. The independent test set is used to measure each modelâs performance and the average MAE and MAPE is reported for each output. The same network configuration and hyperparameter settings used in Experiment 12 (see Section 5.4.5) was used for this experiment. Model development was repeated 5 times using different random initial weights and the average MAE and MAPE calculated.

Results and Discussion

Table 5.3 shows the performance results of the 5-fold chronological cross-validation runs for the STL models that predict the 7-Day average case counts for D+1. Figure 5.26 shows the graphs of predicted versus actual number of cases for D+1 using these models. This method produces models that have an accuracy of between 96.21 and 99.14% over the period of Oct 15 to Dec 24, 2020. In comparison, the baseline Persistence model over the same test sets had an MAE of 7.94 cases and a MAPE of 1.74%. This surprising result is driven by the smooth-flowing change in the 7-Day average as compared to the more stochastic nature of the Daily case count.

Table 5.3: Experiment 13: Performance of LSTM STL models over 5-folds predicting 7-day case counts.

	MAE	MAPE
Fold 1	4.08	1.40
Fold 2	14.48	3.78
Fold 3	9.17	1.68
Fold 4	9.09	1.60
Fold 5	5.52	0.86
Average	8.47	1.87
Stdev	4.03	1.12
95% CI	3.53	0.98

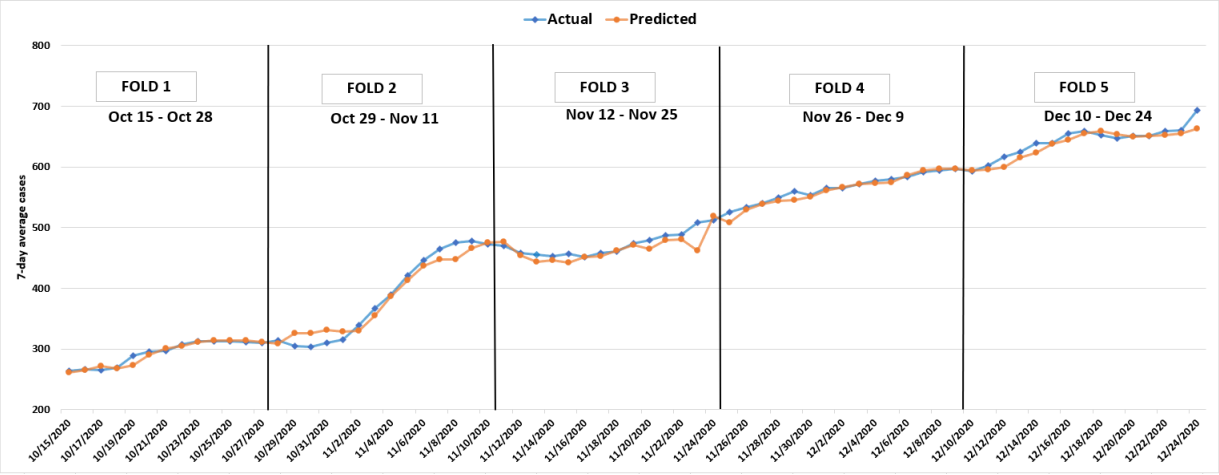


Figure 5.26: Experiment 13: STL LSTM models predicted versus actual case counts for D+1 over 5-folds predicting 7-day case counts.

5.4.7 Experiment 14: Predicting next 7 days (D+1 to D+7) case counts using k-fold chronological cross validation

Objective

This experiment aims to emulate the iterative development and testing of COVID forecast models over a period of several windows of data. We develop MTL LSTM models that accurately predict the Daily case counts for the next 7 days (D+1 through D+7) using a sequence of 14 days' worth of prior data. We then move the window forward by two weeks.

Data and Methods

This experiment uses the same data that is described in Experiment 12 (see Section 5.4.5). The complete dataset is used to create 5 folds of training, validation, and test examples as one moves forward in time, beginning from March 1, 2020. The independent test set is used to measure each models' performance, and the average MAE and MAPE are reported for each output.

Results and Discussion

Table 5.4 shows the performance results of the 5-fold chronological cross-validation runs for the MTL models that predict the Daily case counts for D+1 to D+7. Figure 5.27 shows the graphs of actual versus predicted values of D+1 in the MTL model.

Table 5.4: Experiment 14: Performance of LSTM MTL models over 5-folds predicting daily case counts.

	MAE	**MAPE**
Fold 1	33.82	10.59
Fold 2	79.65	18.53
Fold 3	68.36	13.80
Fold 4	66.91	11.51
Fold 5	65.78	10.10
Average	62.90	12.91
Stdev	15.36	3.09
95% CI	13.46	2.71

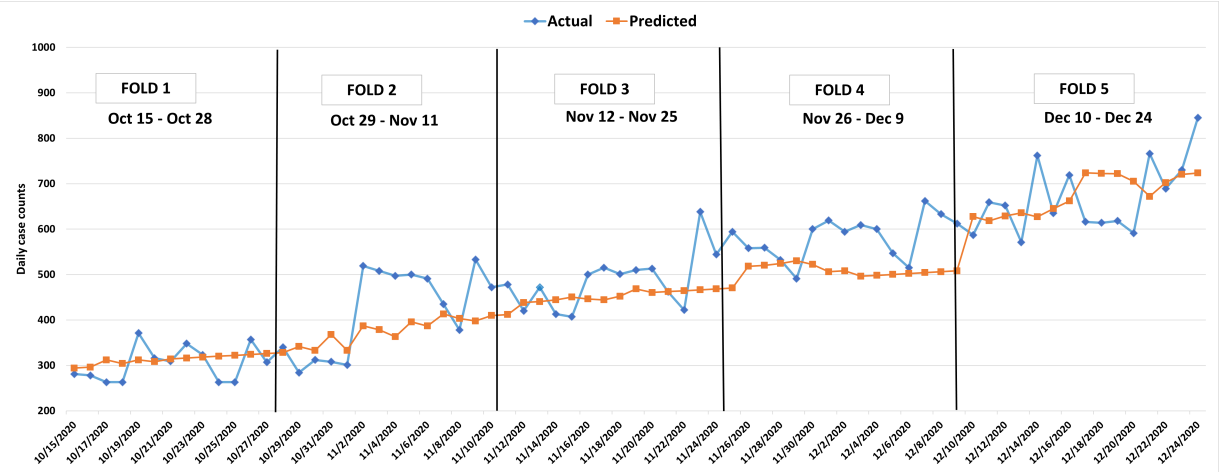


Figure 5.27: Experiment 14: MTL LSTM models predicted versus actual case counts for D+1 over 5-folds predicting daily case counts.

5.4.8 Experiment 15: Predicting next 7 days (D+1 to D+7) 7-day average case counts using k-fold chronological cross validation

Objective

This experiment aims to emulate the iterative development and testing of COVID forecast models over a period of several windows of data. We develop MTL LSTM models that accurately predict the 7-day average case counts for the next 7 days (D+1 through D+7) using a sequence of 14 days' worth of prior data. We then move the window forward by two weeks.

Data and Methods

This experiment uses the same data that is described in Experiment 13 (see Section 5.4.6). The complete dataset is used to create 5 folds of training, validation, and test examples as one moves forward in time, beginning from March 1, 2020. The independent test set is used to measure each models' performance, and the average MAE and MAPE are reported for each output.

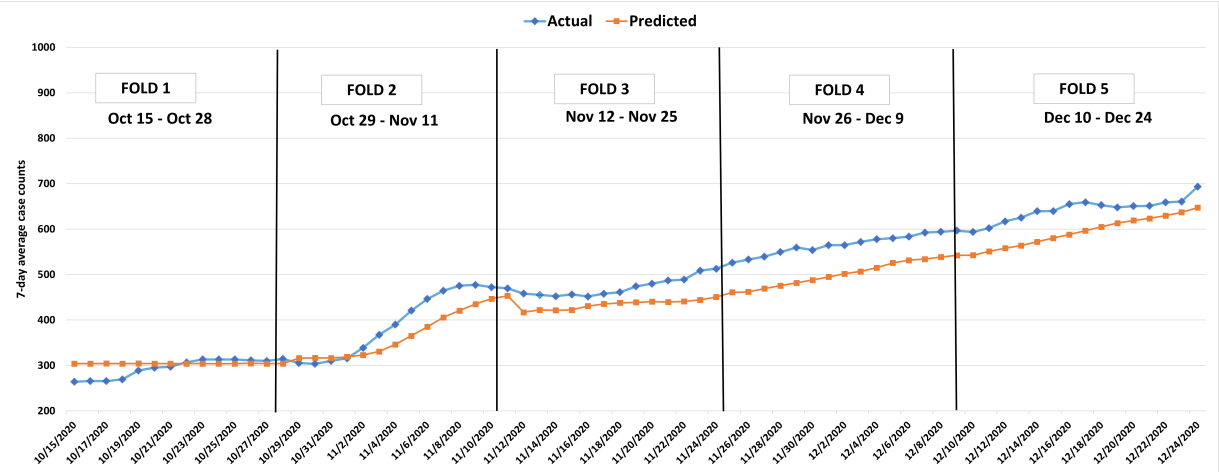


Figure 5.28: Experiment 15: MTL LSTM models predicted versus actual case counts for D+1 over 5-folds predicting 7-day case counts.

Results and Discussion

Table 5.5 shows the performance results of the 5-fold chronological cross-validation runs for the MTL models that predict the 7-day average case counts for D+1 to D+7. Figure 5.28 shows the graphs of actual versus predicted values of D+1 in the MTL model

Table 5.5: Experiment 15: Performance of LSTM STL models over 5-folds predicting 7-day case counts.

	MAE	**MAPE**
Fold 1	16.89	5.73
Fold 2	31.66	7.98
Fold 3	40.56	8.51
Fold 4	63.89	11.23
Fold 5	47.98	7.46
Average	40.20	8.18
Stdev	17.59	2.00
95% CI	15.42	1.75

Single-Task Learning (STL) D+1 models	**Multi-Task Learning (MTL)** D+1 to D+7 models
Exp 1: IDT 7-day ⇒ D+1 Daily	-
Exp 2: IDT 14-day ⇒ D+1 Daily	-
Exp 3: ANN ⇒ D+1 Daily	-
Exp 4: CNN ⇒ D+1 Daily	**Exp 5:** CNN ⇒ D+1 to D+7 Daily
Exp 6: CNN ⇒ D+1 Averages	**Exp 7:** CNN ⇒ D+1 to D+7 Averages
Exp 8: LSTM ⇒ D+1 Daily	**Exp 9:** LSTM ⇒ D+1 to D+7 Daily
Exp 10: LSTM ⇒ D+1 Averages	**Exp 11:** LSTM ⇒ D+1 to D+7 Averages
Exp 12: k-fold LSTM ⇒ D+1 Daily	**Exp 14:** k-fold LSTM ⇒ D+1 to D+7 Daily
Exp 13: k-fold LSTM ⇒ D+1 Averages	**Exp 15:** k-fold LSTM ⇒ D+1 to D+7 Averages

Figure 5.29: List of STL and MTL Models.

5.5 Comparision of Models

Figure 5.29 shows the list of all Single Task Learning (STL) and Multi Task Learning (MTL) models built using both daily case counts and 7-day average values to predict next day (D+1) and the next week (D+1 to D+7 respectively).

Comparing the STL models shown in Figure 5.30, we see that of all the models predicting Daily case counts, the K-fold LSTM model (Experiment 12) is the best model which has the lowest MAE and MAPE values. Similarly, in the models predicting 7-day average case counts, the K-fold LSTM model (Experiment 13) is the best model which has the lowest MAE and MAPE values.

Comparing the MTL models shown in Figure 5.31, we see that of all the models predicting Daily case counts, the MTL LSTM model (Experiment 9) is the best model has the lowest MAE and MAPE values. Similarly, in the models predicting 7-day average case counts, the MTL LSTM model (Experiment 11) is the best model having the lowest MAE and MAPE values.

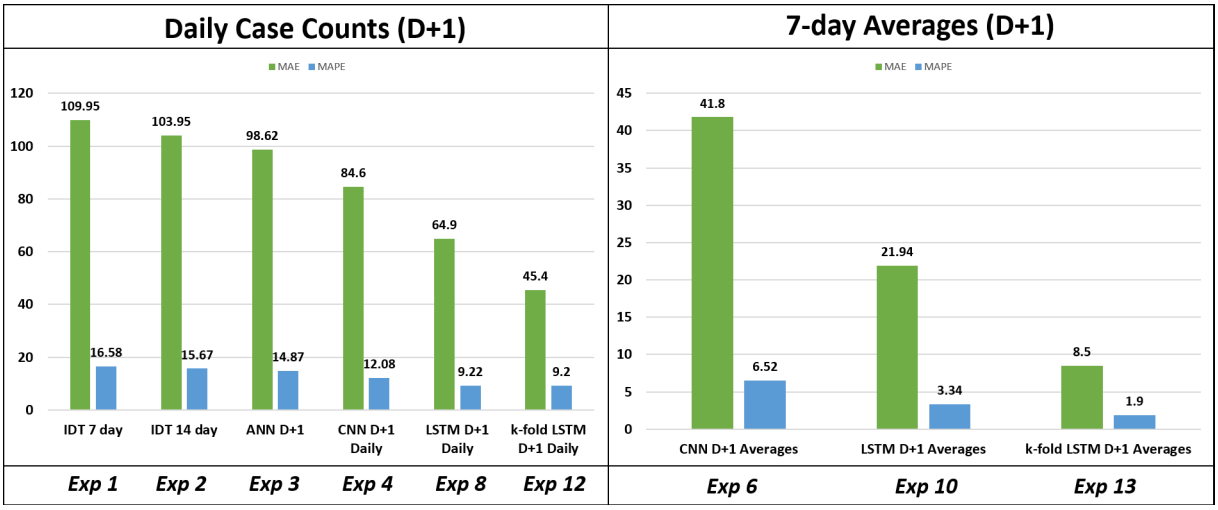


Figure 5.30: Comparision of STL Models.

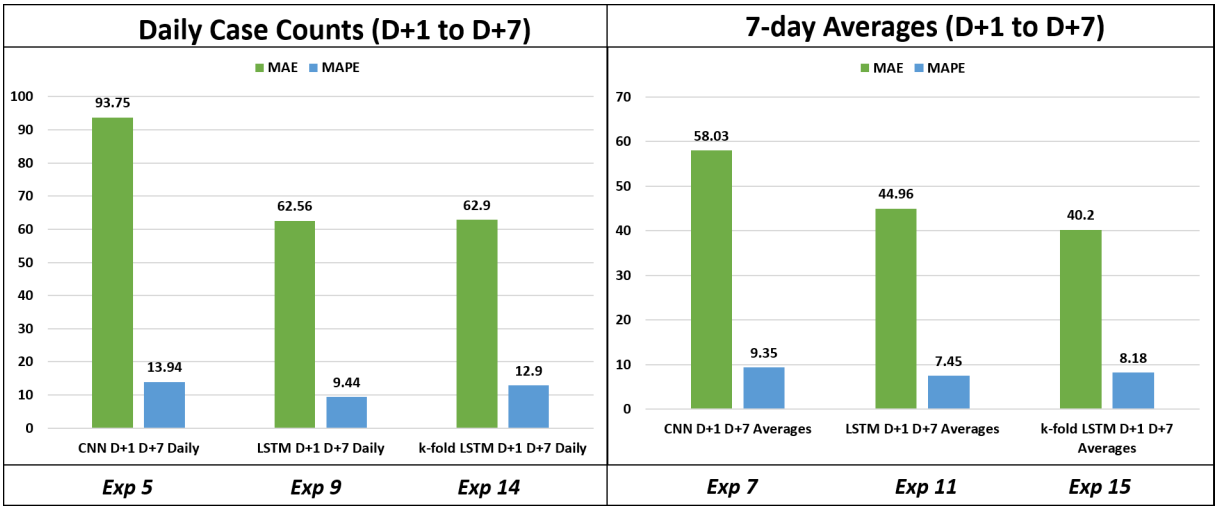


Figure 5.31: Comparision of MTL Models.

Chapter 6

Conclusion and Future work

6.1 Summary

Epidemiological forecasting models are important tools for provincial health care systems to combat viral infections such as COVID-19. These models can be used to understand the changes in a pandemic and to estimate the future demand for health services. Using forecasting models, the policymakers and health care management can estimate the changes in the number of COVID-19 cases in advance [13].

In this research, several predictive models are built, tested, and compared for one day (D+1) and one week ahead (D+1 to D+7) predictions. The models were developed using two deep neural network approaches: Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) neural networks. A 5-fold chronological cross-validation approach employed these methods to develop predictive models using data from March 1 to October 14, 2020, and test them on data covering October 15 to December 24, 2020. Each model used 14 days' worth of prior data (including the Daily case count autoregressive values) as input and predicted tomorrow's case count, day D+1. The prior 14 days of data as input is used to train and test the models.

The best models for predicting tomorrow's (D+1) Daily case count and 7-Day average case count were based on the LSTM deep neural network architecture. The best STL LSTM model predicting daily case counts had an average MAE of 45.4 counts and an average MAPE of 9.23% with a 95% CI of 2.09% over all the 5-folds (see Section 5.4.5). The best

STL LSTM model predicting 7-Day average case counts had an average MAE of 8.47 counts and an average MAPE of 1.87% with a 95% CI of 0.98% over all the 5-folds (see Section 5.4.6).

6.2 Findings and Contributions

The following are the important findings from this research:

1. An analysis of the 2020 Ontario data suggests that in the early part of the year, during the colder time frame of March and April, the transmission of COVID was higher amongst the most vulnerable demographic (older population) despite restrictions and subsequent reductions in mobility (see Figure 4.9). In contrast, a reduction in restrictions and increased mobility to normal levels in the warmer months of June through August corresponds to a decrease in Daily case count numbers to their lowest in 2020 (see Figure 4.12). This suggests that variables beyond human interaction are at work in COVID transmission. The fall of 2020 confirms this because as mobility slowly decreases, the number of Daily case counts soars to their highest levels, mainly in the 20-39-year-olds (see Figure 4.9).

2. Guided by the above analysis, we reduced the original set of independent variables to the list of 17 shown in Figure 4.13. These variables capture the most important aspects of demographics and population movement, environments in which they were interacting, and the temporal aspects of the week or year. The variables had a high linear correlation with Daily case counts considering a lag of up to 14 days or played an important combinatorial role within early inductive decision trees (IDT)

3. The predictive models show that the Day of Year (DOY), Mobility (Movement_rel_to_baseline), and Age naturally played important roles in the models but in different ways depending upon if it is before or after June 2020. Figure 4.12 shows a graph of Daily case counts, Mobility, and level of Public Health Restrictions. The graphs suggest that the cause-and-effect sequence is from case count to changes in restrictions and finally to changes in mobility. In fact, the linear correlation between restrictions and mobility

is 0.9 with no lag and decreases with a lag after that. One could conclude that the rise or fall in case counts results in an increase or decrease in restrictions, followed by a decrease, or increase in mobility, respectively. One can see from Figure 4.12 how restrictions worked well to reduce mobility which naturally, over time, had an impact on the transmission of the disease.

4. Figure 4.9 shows that during the first wave of 2020, those over 70 years of age contributed largely to the case counts, whereas, during the second wave, it was the 20- to 39-year-olds who contributed significantly. The 40-69 group contributed strongly during both periods. Therefore, we believe that the Public Health Restrictions and environment driven by the seasons played a significant role in determining which age groups were affected the most. There are lessons to be learned and reflected in the recommendations below.

5. We found the most significant and interesting environmental variables affecting the models were Outdoor average air temperature (avg_temperature) and Indoor Relative Humidity (IRH), as shown in Figure 4.11. Related studies from all over the world have shown that outdoor air temperature plays a significant role in transmitting the COVID-19 virus. During the coldest portions of the year, when humans spend more time indoors or in vehicles, air quality drops within buildings, most significantly Indoor Relative Humidity (IRH) levels. Moderate to high indoor temperatures coupled with low IRH (below 20%) are prime conditions for viral transmission because:

   (a) Water vapor ejected from an infected person's mouth can linger longer and travel further in the air because of evaporation, and

   (b) Dry skin conditions, particularly in a recipient's airway, can make for more optimal conditions for transmission.

6.3 Future Work and Recommendations

The following proposes some areas of future work and recommendations based on our analysis of the data and the predictive models developed from the Ontario data.

1. It needs to be better communicated that indoor environments play an important role in transmitting the disease during colder parts of the year. Within portions of buildings where persons from different regions or so-called 'bubbles' are expected to meet, such as airports, seaports, train stations, and bus stations, the IRH should be raised to the ASHRAE recommended level of 30-50% IRH. The cost of doing this is recognized, and difficult decisions need to be taken.

2. Early Public Health Restrictions should first focus on the most vulnerable portion of the population, typically seniors. If COVID-19-like diseases, restrictions, and changes in operating procedures specific to senior facilities and nursing homes should be planned for, tested, and redesigned as needed to reduce the chances of the disease making its way to the most vulnerable.

3. Future modeling should focus on modeling the general trend in disease transmission provided by the 7-Day moving average case count versus the Daily case count. This is mainly because of the noise introduced by estimates of when a person contracted the disease driven by variations in testing and reporting sequence during the pandemic.

Appendix A

Research Ethics Board Approval

Research Ethics Board

Acadia University Box 181
Wolfville, Nova Scotia
Canada B4P 2R6
http://reb.acadiau.ca

Application for Ethical Review of Research Involving Humans

Complete this form electronically and submit it, along with your **Application Package**, by **email attachment** to smaitzen@acadiau.ca. Please attach it as a single **Microsoft Word** or **PDF** file. No digital signature is required on your documents.

The Research Ethics Board (REB) strongly encourages you to consult the *Tri-Council Policy Statement, Second Edition* (TCPS2), when preparing your application. TCPS2 can be found at this link. **Incomplete forms will be rejected**.

Name of Principal Investigator:	Daniel L. Silver
Faculty, Staff, Graduate student, Undergraduate student?	Faculty
Department, School, or Program:	Jodrey School of Computer Science
Telephone number:	cell: 902-679-9315 / office: (902) 585-1413
Email address:	danny.silver@acadiau.ca
Supervisor (if you are a student):	
Supervisor's email address:	

Title of your project: The role of environmental determinants and social mobility in viral infection transmission in Ontario and Nova Scotia.

Type of project (e.g., Honours or Master's thesis; externally funded project; part of a research program):

Masters thesis, funded by the Nova Scotia COVID-19 Health Research Coalition, part of an SMU, Dal, Acadia partnership effort.

Other investigators on this project:	Their email addresses:
Rinda Digamarthi, MSc student and RA, Acadia	157742d@acadiau.ca
Yigit Aydede, Professor, Saint Mary's University	yigit.aydede@smu.ca
Mutlu Yuksel, Professor, Dalhousie University	multu@dal.ca

Funders/sponsors of your project (if any): Nova Scotia COVID-19 Health Research Coalition

Proposed start date of your research: September 21, 2020 **(4-6 weeks are required for review.)**

Enter the date of your application below to certify that you will follow all TCPS2 regulations and REB requirements in conducting your research.

September 14, 2020

For student researchers, enter the date below on which your supervisor approved your submission of this application. You must also "cc" your supervisor on your email submission of this application.

September 14, 2020

Research Summary

1. **Purpose**: the objectives of the study; its hypotheses (if any); why the study is needed

   A collaborative partnership, called the Nova Scotia COVID-19 Health Research Coalition, to develop a COVID-19 research response strategy has formed among NSHA Research & Innovation (Nova Scotia Health Authority), Dalhousie University, Research Nova Scotia, the Dalhousie Medical Research Foundation, the QEII Health Sciences Centre Foundation, the IWK Foundation, and the Dartmouth General Hospital Foundation (DGHF). These partners have collectively committed a minimum $1.5 million to support the Nova Scotia research community. This research effort will inform the best COVID-19 practices and support healthcare decision making and planning that benefits the population of Nova Scotia.

   The proposal submitted by our research team (Yigit Aydede - SMU, Mutlu Yuksel - DAL, Daniel Silver - Acadia) was selected by the Coalition for funding of $36,900 (see the attached application and award notice). $12,000 from this award will come to Acadia for a Masters level research project in the School of Computer Science. Using Nova Scotia COVID-19 test data, the proposed study plans to analyze the transmission of viral pathogens and the number of positive COVID-19 cases in response to local climatic and air quality conditions as well as the level and the mode of mobility in Nova Scotia. We will analyze these data with respect to the speed of transmission measured by the demand for the COVID-19 tests and the number of positive tests by region by time. Advance statistical time series methods and deep recurrent neural networks will be use to develop models to predict the spread of the disease. As input variables, we will use local mobility data extracted from Apple, Google, and Facebook application programming interfaces (API), and high-dimensional high-resolution local weather and air quality data obtained from industry providers such as Breezometer and Climacell.

   While awaiting the data from the Nova Scotia Health Authority (NSHA), the SMU and Acadia team will work with a preliminary dataset from the Province of Ontario that is publicly available by county at Ontario.ca (specifically: https://data.ontario.ca/dataset/confirmed-positive-cases-of-covid-19-in-ontario). Mobility and high resolution weather data will be obtained for these county regions in the same manner as it will be for postal code regions in Nova Scotia. This will allow us to hone our knowledge of the problem and modeling skills prior to receiving the NS data.

2. **Methodology**: how the subjects will be chosen, how they will be contacted, and by whom; who will conduct the research and where; what the subjects will be asked to do; what data will be taken

   Our analysis combines information from multiple data sources on (i) the nonpharmaceutical interventions measured by social mobility indices of Apple, Google and Facebook, (ii) the daily number of infections abstracted from 811 Triage, and (iii) high-dimensional high-resolution weather and air quality data.
   The daily data on the number of infections (Covid19 or common flu) at the postal code level will be extracted from health authorities. Since COVID19 tests for the public are regulated by 811 Triage, each referral by 811 Triage represents a significant level of pre-determined symptoms. This provides us with information on the number of new infections each day by postal code. The observed variation

in the incidence of viral infections may be influenced in part by variation in reporting. In order to remove such random variation from the data, we may use a moving average of the data to better estimate days with zero or very low cases.

SMU will apply advanced time series statistical modeling methods. They will develop Autoregressive Distributed Lag (ARDL) models with fixed-effect dummies and ARDL with a random-coefficient structure. They have two explicit objectives: prediction and model selection so that sparsity can be achieved by different objective functions. First, SMU wish to determine if there is an underlying data generating a process for the infection transmission that can be captured by a model that represents the "true" sparsity with relevant variables (predictors). They will apply an Adaptive Lasso regression proven to be consistent when applied to time-series and panel data. Second, SMU will find a model that has the most predictive power in forecasting the transmission rate. To accomplish this, they will employ cross-validation methods to find the optimal penalization in a range of Lasso family applications from Elastic Net to Bootstrapped- Lasso. They will also experiment with multiple nonparametric machine learning models specifically suitable for a time series problems.

Acadia will have very similar objectives: (1) To develop the most predictive model possible using a time-series moving window cross-validation approach and (2) To determine the key predictor variables and their potential non-linear interaction to produce accurate estimates of viral disease spread. This will be accomplished by using state-of-the-art Long Short-Term Memory (LSTM) artificial neural networks, which is a kind of recurrent neural network (RNN) and one of the best predictive models given the chain-like nature of time series and panel data. Acadia's work will focus on predicting the number of new cases of COVID-19 each day, secondarily the Acadia team will look at models that classify the next day as having more or less COVID-19 cases than the day before. Sensitivity analysis will be done on the models to provide insight into the relationship between the input variables and response variable. Less accurate decision tree models may be used to provide some insight into these relationships, if it seems warranted.

3. **Consent**: how informed consent will be obtained (**Note: The REB requires parent/guardian consent for any research subject under 18 years of age who is not a registered student at Acadia University**.)

   All health care data has been collected by provincial health authorities.
   The preliminary data that we are using from the Province of Ontario has already been collected and is now in the public domain and freely available to all researchers. The data consists of the number of COVID cases that have tested positive per day per region. The NSHA has already collected the COVID19 test data for Nova Scotia. They are preparing the data for our use with privacy and integrity in mind such that the data cannot be traced back to any individuals. There is no identifying information beyond the region name (county for ON data, postal code for NS data). We will throw away any data that has less than 5 test cases per region.

   Aggregated smartphone mobility will be obtained for each region from publicly accessible web sources using APIs from Apple, Google and Facebook.

   The high-resolution weather data will come from industry web sources such as Breezometer or Climacell.

4. **Debriefing**: how the subjects will be debriefed following their participation

   Not applicable. Neither the PI nor RA will be collecting data from subjects.

5. **Risks**: any expected risks to the subjects and how such risks will be minimized

   Not applicable. Neither the PI nor RA will be collecting data from subjects.

   There is no risk because the data (COVID-19 Test numbers) will be aggregated by region and time at NSHA. The preliminary Ontario data being used is in the public domain and is similarly aggregated by region and time.

6. **Safety**: if applicable, how the safety of subjects will be monitored

   Not applicable. Neither the PI nor RA will be collecting data from subjects.

7. **Confidentiality**: how the confidentiality of the subjects and data will be assured

   The data consists of the number of COVID cases that have tested positive per day per region (county or postal code). The data aggregation has been completed by the respective health authority. We do not have access to individual records. The health authority has prepared the data for our us with privacy and integrity in mind such that the data cannot be traced back to any individuals. There is no identifying information beyond the region name (county for ON data, postal code for NS data). We will throw away any data that has less than 5 test cases per region.

   The data will be contained in the researchers' local computers. If requested by NSHA, the data will be transmitted in an encrypted format over the Internet, and the encryption key will be sent independently by text message. The Ontario data can be transmitted in open format as it is in the public domain.

8. **Compensation**: how, if at all, the subjects will be compensated

   Not applicable

9. **Deception**: if deception will be used, why it is necessary

   Not applicable

Consent forms that will be used

Not applicable.

Surveys, questionnaires, or interview questions that will be used

Not applicable.

Advertisements that will be used to alert or attract research subjects

Not applicable.

Research protocols, if any

Not applicable.

Contract agreements, if any

Please see the attached NSHA application and award and the Saint Mary's University letter of funds transfer to Acadia University.

Confidentiality agreements, if any, between the researcher and his/her source of funding

Not applicable.

From: Stephen Maitzen <stephen.maitzen@acadiau.ca>
Date: Monday, September 14, 2020 at 11:03 AM
To: Daniel Silver <danny.silver@acadiau.ca>
Subject: RE: REB Application

Dear Dr. Silver,

Thank you for contacting me. Because your research involves only the analysis of anonymized data that were originally collected for a different purpose and that cannot be linked to individuals, your project is exempt from ethics review under TCPS2 Article 2.4. You do not need to submit a research ethics application.

Best wishes for a successful project.

****
Stephen Maitzen, PhD
W. G. Clark Professor of Philosophy
Head, Department of Philosophy
Chair, Research Ethics Board
Acadia University

From: Danny Silver <danny.silver@acadiau.ca>
Sent: Monday, September 14, 2020 10:11 AM
To: Stephen Maitzen <stephen.maitzen@acadiau.ca>
Subject: REB Application

Hello Dr. Maitzen (Stephen) .. Please find attached an application for Ethical Review of Research.
This is a joint project with Saint Mary's and Dalhousie where we are using data concerning COVID-19 collected and aggregated by health authorities in Ontario and Nova Scotia. We will be using counts of persons who have approached 811 with symptoms by region by date, or have been found positive with COVID based on a test by region by date. No personal information is included in the data from the provinces which is publicly available in Ontario and will soon be in NS (we hope).

I have attached the review response from SMU for your information.
They did not require a full review. I can send you their REB app, if you wish.

==========
Daniel L. Silver
Professor, Jodrey School of Computer Science
Director, Acadia Institute for Data Analytics
Acadia University,
Office 314, Carnegie Hall,
Wolfville, Nova Scotia Canada B4P 2R6

t. (902) 585-1413
f. (902) 585-1067

Appendix B

Meta Data Report for Original Source Data

The Meta Data Report for all the variables considered during this project can be found at the following link.

Appendix C

Meta Data for Prepared Modeling Data

The Meta Data Report for just those variables used in the development of machine learning and deep learning models can be found at the following link.

Bibliography

[1] Soliman, M., Lyubchich, V., Gel, Y. R. (2019). Complementing the power of deep learning with statistical model fusion: Probabilistic forecasting of influenza in Dallas County, Texas, USA. Epidemics, 28, 100345. https://doi.org/10.1016/j.epidem.2019.05.004

[2] Xu, Q., Gel, Y. R., Ramirez Ramirez, L. L., Nezafati, K., Zhang, Q., Tsui, K. L. (2017). Forecasting influenza in Hong Kong with Google search queries and statistical model fusion. PLOS ONE, 12(5), e0176690. https://doi.org/10.1371/journal.pone.0176690

[3] Schultz, D. M. (2022). Questions about Tosepu et al. (2020) âCorrelation between weather and Covid-19 pandemic in Jakarta, Indonesia.â Science of The Total Environment, 154078. https://doi.org/10.1016/j.scitotenv.2022.154078

[4] Xu, R., Rahmandad, H., Gupta, M., DiGennaro, C., Ghaffarzadegan, N., Amini, H., Jalali, M. S. (2021). Weather, air pollution, and SARS-CoV-2 transmission: a global analysis. The Lancet Planetary Health, 5(10), e671 to e680. https://doi.org/10.1016/s2542-5196(21)00202-3

[5] Zhang, C., Liao, H., Strobl, E., Li, H., Li, R., Jensen, S. S., Zhang, Y. (2020). The Role of Weather Conditions in COVID-19 Transmission: A Study of a Global Panel of 1236 Regions. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.3652202

[6] Askitas, N., Tatsiramos, K., Verheyden, B. (2020). Lockdown Strategies, Mobility Patterns and COVID-19. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.3619687

[7] Lai, S., Ruktanonchai, N. W., Carioli, A., Ruktanonchai, C. W., Floyd, J. R., Prosper, O., Zhang, C., Du, X., Yang, W., Tatem, A. J. (2021). Assessing the Effect of Global

Travel and Contact Restrictions on Mitigating the COVID-19 Pandemic. Engineering, 7(7), 914 to 923. https://doi.org/10.1016/j.eng.2021.03.017

[8] Ahlawat, A., Wiedensohler, A., Mishra, S. K. (2020). An Overview on the Role of Relative Humidity in Airborne Transmission of SARS-CoV-2 in Indoor Environments. Aerosol and Air Quality Research, 20(9), 1856 to 1861. https://doi.org/10.4209/aaqr.2020.06.0302

[9] Sharif, A. (2020, November 5). The Relationship Between Humidity Air Quality Indoors. Edge. Retrieved May 1, 2021, from https://edgesustainability.com/the-relationship-between-humidity-air-quality-in-a-building/

[10] Education, I. C. (2021c, November 5). Machine Learning. Machine Learning. Retrieved October 31, 2020, from https://www.ibm.com/cloud/learn/machine-learning

[11] Fumo, D. (2018, June 21). Types of Machine Learning Algorithms You Should Know. Medium. Retrieved October 31, 2020, from https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861

[12] OsiÅski, B., Budek, K. (2022, February 15). Reinforcement Learning. Deepsense.Ai. Retrieved February 28, 2022, from https://deepsense.ai/what-is-reinforcement-learning-the-complete-guide/

[13] M, R. (2021, December 12). Decision tree — Non-parametric supervised learning algorithms — Clairvoyant Blog. Medium. Retrieved January 10, 2022, from https://blog.clairvoyantsoft.com/upside-down-trees-that-divide-and-conquer-e893c8f73ee8

[14] Education, I. C. (2021b, August 3). Neural Networks. Neural Networks. Retrieved October 31, 2020, from https://www.ibm.com/cloud/learn/neural-networks

[15] Sulaiman, K., Hakim Ismail, L., Adib Mohammad Razi, M., Shalahuddin Adnan, M., Ghazali, R. (2019). Water Quality Classification Using an Artificial Neural Network (ANN). IOP Conference Series: Materials Science and Engineering, 601(1), 012005. https://doi.org/10.1088/1757-899x/601/1/012005

[16] Abirami, S., Chitra, P., Multilayer Perceptron - an overview — ScienceDirect Topics. Multilayer Perceptron. Retrieved May 10, 2021, from https://www.sciencedirect.com/topics/computer-science/multilayer-perceptron

[17] Wood, T. (2020, September 2). Backpropagation. DeepAI. Retrieved March 1, 2021, from https://deepai.org/machine-learning-glossary-and-terms/backpropagation

[18] Adam, H. (2020, October 12). Autoregressive Integrated Moving Average (ARIMA). Investopedia. Retrieved January 14, 2021, from https://www.investopedia.com/terms/a/autoregressive-integrated-moving-average-arima.asp

[19] Education, I. C. (2021a, April 7). Recurrent Neural Networks. Recurrent Neural Networks. Retrieved June 10, 2021, from https://www.ibm.com/cloud/learn/recurrent-neural-networks

[20] Or, B. (2021, December 16). The Exploding and Vanishing Gradients Problem in Time Series. Medium. Retrieved January 10, 2022, from https://towardsdatascience.com/the-exploding-and-vanishing-gradients-problem-in-time-series-6b87d558d22

[21] Understanding LSTM Networks – colahâs blog. (2015, August 27). Colahâs Blog. Retrieved May 10, 2021, from https://colah.github.io/posts/2015-08-Understanding-LSTMs/

[22] LSTM Networks. LSTM. Retrieved May 15, 2021, from https://apmonitor.com/do/index.php/Main/LSTMNetwork

[23] Yingge, H., Ali, I., Lee, K. Y. (2020). Deep Neural Networks on Chip - A Survey. 2020 IEEE International Conference on Big Data and Smart Computing (BigComp). Retrieved January 20, 2021, from https://doi.org/10.1109/bigcomp48618.2020.00016

[24] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15, 1929 to 1958.

[25] Baeldung, B. (2020, August 13). How ReLU and Dropout Layers Work in CNNs. Baeldung on Computer Science. Retrieved March 1, 2022, from https://www.baeldung.com/cs/ml-relu-dropout-layers

[26] Fu, C., Silver, D.L., Blustein, J. (2006). Chronological Sampling for Email Filtering.

[27] Onyari, E.K., Ilunga, F.M. (2013). Application of MLP Neural Network and M5P Model Tree in Predicting Streamflow: A Case Study of Luvuvhu Catchment, South Africa.

[28] El Houssainy, A. R., Haitham, F., Amal Mohamed, A. F. (2021). Time Series Forecasting Using Tree Based Methods. Journal of Statistics Applications Probability, 10(1), 229â244. https://doi.org/10.18576/jsap/100121

[29] Taylor, G. W., Fergus, R., LeCun, Y., Bregler, C. (2010, September). Convolutional learning of spatio-temporal features. In European conference on computer vision (pp. 140-153). Springer, Berlin, Heidelberg.

[30] Miao, Y., Han, J., Gao, Y., Zhang, B. (2019). ST-CNN: Spatial-Temporal Convolutional Neural Network for crowd counting in videos. Pattern Recognition Letters, 125, 113 to 118. https://doi.org/10.1016/j.patrec.2019.04.012

[31] Tavakolian, M., Hadid, A. (2019). A Spatiotemporal Convolutional Neural Network for Automatic Pain Intensity Estimation from Facial Dynamics. International Journal of Computer Vision, 127(10), 1413 to 1425. https://doi.org/10.1007/s11263-019-01191-3

[32] Yao, H., Tang, X., Wei, H., Zheng, G., Li, Z. (2019, July). Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction. In Proceedings of the AAAI conference on artificial intelligence (Vol. 33, No. 01, pp. 5668 to 5675).

[33] Yunbo Wang, Mingsheng Long, Jianmin Wang, Zhifeng Gao, and Philip S. Yu. 2017. PredRNN: Recurrent Neural Networks for Predictive Learning using Spatiotemporal LSTMs. In NIPS. 879–888

[34] Yin, W. (2017, February 7). Comparative Study of CNN and RNN for Natural Language Processing. arXiv.Org. https://arxiv.org/abs/1702.01923

[35] Elmasdotter, A., Nystromer, C. (2018). A comparative study between LSTM and ARIMA for sales forecasting in retail.

[36] Pinchin Ltd. 2360 Meadowpine Blvd., Unit 2, Mississauga, ON L5N 6S2, Canada, contact: Stephen Booth, Director, GTA OHS Indoor Environmental Quality, T: 905.363.1301 â C:416.816.5071

[37] Jiang, X. (2017, May 7). TrajectoryNet: An Embedded GPS Trajectory Representation for arXiv.Org. Retrieved December 10, 2021, from https://arxiv.org/abs/1705.02636?msclkid=fe8f7c72b1a011eca8693cb2d16f7732

[38] Government of Ontario. (2020, February 10). Case numbers, spread and deaths. COVID-19 (Coronavirus) in Ontario. Retrieved March 10, 2022, from https://covid-19.ontario.ca/data/case-numbers-and-spread

[39] Lauer, S. A., Grantz, K. H., Bi, Q., Jones, F. K., Zheng, Q., Meredith, H. R., Azman, A. S., Reich, N. G., Lessler, J. (2020). The Incubation Period of Coronavirus Disease 2019 (COVID-19) From Publicly Reported Confirmed Cases: Estimation and Application. Annals of Internal Medicine, 172(9), 577 to 582. https://doi.org/10.7326/m20-0504

[40] Brownlee, J. (2020, August 28). How to Develop Convolutional Neural Network Models for Time Series Forecasting. Machine Learning Mastery. https://machinelearningmastery.com/how-to-develop-convolutional-neural-network-models-for-time-series-forecasting/

[41] The University of Waikato. Weka Wiki. WEKA. Retrieved March 10, 2022, from https://waikato.github.io/weka-wiki/

[42] Meta (Facebook) - Data for Good Initiative. (2020, March 1). Movement Range Maps - Humanitarian Data Exchange. Facebook Data for Good - Mobility Data. Retrieved March 10, 2022, from https://data.humdata.org/dataset/movement-range-maps?

[43] Weather Stats. Canada Weather Stats. Weatherstats Website. Retrieved March 10, 2022, from https://www.weatherstats.ca/

[44] Government of Ontario. COVID-19 public health measures and advice. COVID-19 (Coronavirus) in Ontario. Retrieved March 10, 2022, from https://covid-19.ontario.ca/public-health-measures

[45] Molnar, C. (2022, March 29). 5.4 Decision Tree — Interpretable Machine Learning. Decision Tree — Interpretable Machine Learning. Retrieved April 1, 2022, from https://christophm.github.io/interpretable-ml-book/tree.html

[46] Nouvellet, P., Bhatia, S., Cori, A., Ainslie, K., Baguelin, M., Bhatt, S., Boonyasiri, A., Brazeau, N. F., Cattarino, L., Cooper, L. V., Coupland, H., Cucunuba, Z. M., Cuomo-Dannenburg, G., Dighe, A., Djaafara, B. A., Dorigatti, I., Eales, O. D., van Elsland, S. L., Nascimento, F. F., FitzJohn, R. G.,Donnelly, C. A. (2021). Reduction in mobility and COVID-19 transmission. Nature communications, 12(1), 1090. https://doi.org/10.1038/s41467-021-21358-2

[47] R, A. (2021, December 16). Machine Learning Explanation : Supervised Learning Unsupervised Learning. Medium. Retrieved January 10, 2022, from https://arifromadhan19.medium.com/machine-learning-explanation-supervised-learning-unsupervised-learning-6d4c7f2bebb2

[48] Panneerselvam, L. (2021, April 14). Activation Functions - What are Activation Functions. Analytics Vidhya. Retrieved December 20, 2021, from https://www.analyticsvidhya.com/blog/2021/04/activation-functions-and-their-derivatives-a-quick-complete-guide/

[49] Zhang, G. (2003b). Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing, 50, 159â175. https://doi.org/10.1016/s0925-2312(01)00702-0

[50] Hoseinzade, E., Haratizadeh, S. (2019). CNNpred: CNN-based stock market prediction using a diverse set of variables. Expert Systems with Applications, 129. https://doi.org/10.1016/j.eswa.2019.03.029

[51] R. Fu, Z. Zhang and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," 2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), 2016, pp. 324-328, doi: 10.1109/YAC.2016.7804912.