Mathematical Sciences and Informatics Journal
Vol. 3, No. 1, May 2022, pp. 75-85
http://www.mijuitmjournal.com DOI : 10.24191/mij.v2i2.18278

Web Scraping and Regression Analysis based on Machine Learning for COVID-19 with Rapid Software Platform

Aizal Yusrina Idris
Department of Computer Science and Engineering, Yanbu Industrial College,Yanbu Al-Sinaiyah, Saudi Arabia
idrisa@rcyci.edu.sa

Razan Bamoallem
Department of Computer Science and Engineering, Yanbu Industrial College,Yanbu Al-Sinaiyah, Saudi Arabia
bamoallemr@rcyci.edu.sa

Mohamad Harith Azfar Mohamad Hatta
Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Shah Alam, Selangor
2021782731@student.uitm.edu.my

Article Info

Article history:

Received March 05, 2022
Revised March 26, 2022
Accepted May 08, 2022

Keywords:

Web scraping
Regression analysis
Machine learning
COVID-19
Python
RapidMiner

ABSTRACT

Since the recent incidence of global COVID-19 pandemic, expertise from different domains including scientists, clinicians, and healthcare experts keep on exploring for technologies to manage the COVID-19 data. Updated and accurate data collection is very critical for them to make a more effective and efficient decision on any aspects of the emergency consequences and events. Although some of them are inexpert data scientists, the important skills and knowledges to extract the recent data on COVID-19 is web data extraction and analysis. While tremendous of literature can be referred from the academic databases, it is difficult to find the report that presents the basis and fundamental methods for implementing web data analysis in a simple way with a rapid software platform. This paper demonstrates a simple framework for implementing web data extraction or web scraping to be analyzed in a rapid software platform. Python scripting language is the simple tool to conduct the web scraping method while RapidMiner is the rapid software for implementing the data visualization and analysis. Simple linear regression based on machine learning approach has been implemented with the RapidMiner to predict COVID-19 death based on the collected data. This paper will be useful for academicians and industry practitioners to conduct a more robust data analysis to accommodate a more challenge issue such as big data analytics in any domains.

Corresponding Author:
Aizal Yusrina Idris
Yanbu Industrial College,Yanbu Al-Sinaiyah, Saudi Arabia
email: idrisa@rcyci.edu.sa

1 Introduction

The world is on the rise of complexity and uncertainty mainly since the COVID-19 impact. The COVID-19 infectious disease caused by the SARS-CoV-2 virus, was declared as pandemic by the World Health Organization (WHO) in the early months of 2020. Between 2020-2022, a huge number of peer reviewed journal articles with COVID-19 search keywords have been published to be available in major databases such as 170,290 articles in the Web of Sciences and 204,187 in the SCOPUS. This is a great indicator to show a rapid progress of interest and attention on COVID-19 to be resolved through a vast amount of critical research. Analyzing the recent states and preparing the future actions for COVID-19 or other pandemics is highly critical.

The web is a reliable and major source of information for COVID-19 to be viewed by many peoples from different background and expertise. Recently, with the continuous information overload over the web, collecting data from the web to be saved to a file system or database for later retrieval and analysis is becoming more crucial compared to conventional data collection methods[1]. Due to

PENERBIT PRESS
UNIVERSITI TEKNOLOGI MARA

the fact that heterogeneous data will be massively generated on the web, web scraping is widely acknowledged as the powerful and remarkable technique for collecting data. The techniques must be grasp not just for scientist with computer expertise but also for the inexpert data scientist from many domains of research areas.

The contributions of this paper are two-folds. Firstly, it presents the framework of implementing web scraping with Python scripting language. Secondly, it demonstrates a simple way to conduct pre-processing data mining on the collected data in a rapid software tool to further been analyzed using a simple linear regression in predicting COVID-19 death. The fundamental methodology presented in this paper will be valuable for researchers, academicians, politicians and scientists mainly to those inexpert in programming.

2 Literature Review

Web scraping is the recent technique in Information and Communication Technology (ICT) and big data technology to allow web data extraction that are publicly available on the internet such as from websites and social media contents. Examples of social media that commonly used for web scraping are Facebook, Twitter and Instagram. The challenge of web scraping is the contents are constantly change and once the website's structure has changed, the scraper tool might not be able to navigate the sitemap correctly or find the relevant information. However, the good side is that the changes to websites are normally small and incremental, so only minimal adjustments are needed to the scraper tool. Constant maintenance on the scraper tool would need a rapid and easy platform and programming codes. Python is the most commonly used scripting language to support a variety of data science tasks including web scraping[2].

Web scraping on COVID-19 information has been the most interesting topic since the pandemic was declared in 2020. Researchers in [3], used Twitter and web news to conduct a predictive analytic of the outbreak. Similarly, Twitter was used in [4] for the researchers to perform semantic analysis and opinion mining from the global netizen on the COVID-19. Focused on Instagram social media, the researchers have extracted text and images from the website to be analyzed and presented in a form of interactive data visualization. Concerned with employees emotion due to remote work during COVID-19, researchers in [5] have conducted a sentiment analysis based on Twitter data
scraping. Sentiment analysis based on web scraping data collection have also been conducted in other domains related to COVID-19 impacts such is online learning[6], tourism[7] and Bitcoin[8].

Besides web scraping, machine learning is the promising technology in resolving COVID-19 issue[9]. As for examples, machine learning was used in COVID-19 diagnosis to estimate the patient risk of infection [10]. More interesting in [11], the researchers used a set of chest x-ray images to develop a machine learning classification model to detect the level of COVID-19 infections. Deep learning is an advanced technique in machine learning used by researchers in [12] to develop the COVID-19 forecasting model. Other than machine learning approach, predicting mortality risk has been conducted in [13] by using convention logistics regression method. Similarly, by using the conventional systematic review, meta-analysis and meta-regression, the relationships between diabetes mellitus and COVID-19 mortality risks can identified in [14].

In all the mentioned literatures, no detail implementation has been reported on the web scraping technique. In[15], the researchers presents the taxonomy of web extraction tools but rather focused on the theoretical than demonstrating the implementation codes. This paper filling the gap to demonstrates the use of Python programming to easily implement the process of web scraping. Furthermore, how to use the collected data to be analyzed in a rapid software is also presented in this paper.

3. Research Method

3.1 Data collection

Conventional method for data collection mostly relied on survey of questionnaires, interview and observation but web scraping is the latest appealing way to automate the process of data collection. Python programming language is simple to be used by inexpert programmer as it provides a huge set of libraries that were developed to performed specific application, including web scraping.

As a second higher programming language, all the libraries were developed from the complex programming language like C, C++ or JAVA by the Python developers. Thus, Python is categorized as a scripting language, which can be practiced easier compared to the complex programming language. There are many libraries have been developed for Python to support web scraping, including Scrapy, Request, Urllib, Beautiful Soup and Selenium. Beautiful Soup is the best choice to be used mainly if the web documents are not structured[16]. Figure 1 shows the first part of the Python codes to implement web scraping with Beautiful Soup library.

```
In [10]: #import
         import requests
         from bs4 import BeautifulSoup
         import pandas as pd

         dataset = pd.DataFrame()

In [11]: #grab html page
         from urllib.request import urlopen as uReq

In [12]: #parse html tags, call BeautifulSoup
         from bs4 import BeautifulSoup

In [13]: #define the html page's url
         covidUrl = 'https://www.worldometers.info/coronavirus/'

In [14]: #to check content of covidUrl variable
         covidUrl

Out[14]: 'https://www.worldometers.info/coronavirus/'

In [15]: page = requests.get(covidUrl)
         soup = BeautifulSoup(page.content, 'html.parser')

In [16]: #To parse HTML to the extract table
         table = soup.find(id='nav-tabContent')
         table = table.find(id = 'nav-today')
         table = table.find(id = '')
         table = table.find(id = 'main_table_countries_today')

In [17]: #finds all the tables with the tag tr in html
         table_rows = table.find_all('tr')
         l = []

         #To loop through all the table to get row data each of the table
         for tr in table_rows:
             td = tr.find_all('td') #finding all column for given tables
             row = [tr.text for tr in td]
             if len(row) == 0:
                 continue
             row = row[:9] #extraction of first 8 cloums
             l.append(row) #creating list of lists to represent the table data

         #To create dataframe from the table of html
         dataset = pd.DataFrame(l, columns=["No","Country","Total Cases","New Cases","T
         otal Deaths","New Deaths","Total Recoverd","Active Cases","Serious Cases"])
```

Figure 1. The Python codes to call Beautiful Soup library

The Python codes in Figure 1 were developed in Jupyter Notebook[17] platform that can be installed in a local computer or with online Jupyter Notebook such as JupyterLab, CoCalc and Colab. Jupyter Notebook is an interactive web-based editor for creating, sharing and editing documents that consists of program codes, rich media and text. Jupyter Notebook are embraced by the research community to allow easy and flexible platform for programmers mainly for the inexpert. For this research, the website address for the data collection is https://www.worldometers.info/coronavirus/ to be set in a variable named as covidURL (refer to In [13]). The next step is to inspect the website's document object model (DOM) by right-clicking on the page and selecting the *Inspect* option or using a keyboard shortcut (Ctrl+Shift+I). The developer tool as depicted in Figure 2 allows the users to interactively explore the websites' DOM.

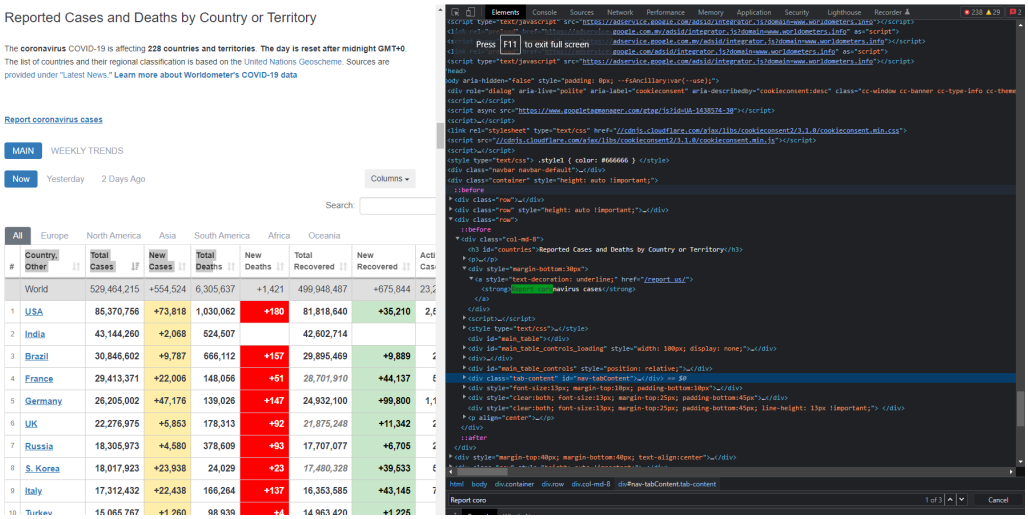


Figure 2. The HTML codes viewed in developer tool for the websites' DOM

The developer tool will display the websites DOM when the *Elements* tab is selected and users can see the structure of the websites with the clickable HTML elements. Selecting the element codes line by line will be linked to be highlighted in the website page. In Figure 1, line In [16] consists of the codes to set the websites contents to be extracted. Then, converting the dataset series into a Python data frame can be written with codes given in Figure 3. Figure 4 presents the Python codes to display to data frame contents.

```
In [9]: # Function to Clean the DataSet
        def dataframeCleaner(dataset):

            dataset = dataset.drop(dataset.head(8).index) # Deleting the last row
            dataset = dataset.drop(dataset.tail(7).index) # Deleting the last row
            dataset = dataset.replace(r'^\s*$', 0, regex=True)# converting empty string to 0
            return dataset
```

```
In [10]: #To print the dataset
         print(dataset)
```

```
    No            Country Total Cases New Cases Total Deaths New Deaths  \
0       \nNorth America\n   3,380,547    +7,873      174,169       +668
1       \nSouth America\n   2,422,021    +1,290       90,914        +49
2             \nEurope\n   2,454,446                193,079
3               \nAsia\n   2,473,515       +66       59,904         +1
4             \nAfrica\n     450,658                 10,920
..  ..              ...         ...         ...          ...        ...
226               Total:   2,473,515       +66       59,904         +1
227               Total:     450,658                 10,920
228               Total:       9,902        +5          126
229               Total:         721                    15
230               Total:  11,191,810    +9,234      529,127       +718

    Total Recovered Active Cases Serious Cases
0         1,504,826       +4,823     1,701,552
1         1,533,749      +33,108       797,358
2         1,411,604         +300       849,763
3         1,653,365          +62       760,246
4           217,705                    222,033
..              ...         ...           ...
226       1,653,365          +62       760,246
227         217,705                    222,033
228           8,916                        860
229             651                         55
230       6,330,816      +38,293     4,331,867

[231 rows x 9 columns]
```

Figure 3. The Python codes to convert the web extraction data in a data frame

```
In [11]: # To cleaning the dataset using user defined function
         dataset = dataframeCleaner(dataset)
         dataset
```

Out[11]:

	No	Country	Total Cases	New Cases	Total Deaths	New Deaths	Total Recoverd	Active Cases	Serious Cases
8	1	USA	2,890,588	0	132,101	0	1,235,488	0	1,522,999
9	2	Brazil	1,543,341	0	63,254	0	978,615	+32,700	501,472
10	3	Russia	667,883	0	9,859	0	437,893	0	220,131
11	4	India	649,889	0	18,669	0	394,319	0	236,901
12	5	Spain	297,625	0	28,385	0	N/A	N/A	N/A
...	...	...	...	...	...	...	...	...	...
219	212	St. Barth	6	0	0	0	6	0	0
220	213	Anguilla	3	0	0	0	3	0	0
221	214	Saint Pierre Miquelon	1	0	0	0	1	0	0
222	215	China	83,545	+3	4,634	0	78,509	+10	402
223	0	Total:	3,380,547	+7,873	174,169	+668	1,504,826	+4,823	1,701,552

216 rows × 9 columns

Figure 4. Display the data frame contents

By using Python codes *dataset.to_csv("dataset.csv")* the data frame contents were converted into a CSV file to be ready used in the RapidMiner for the next research activities started with data inspection.

3.2 Data inspection

The data from web scraping have to be inspected and most probably problem of the data is missing values, which can be done in RapidMiner. Figure 5 shows that the data in CSV consists of some missing values to be processed.

Row No.	No	Country	Total Cases	New Cases	Total Deaths	New Deaths	Total Recov...	Active Cases	Serious Cas...
1	1	USA	2032633	6140	113380	325	774877	1397	1144376
2	2	Brazil	719449	8562	37840	528	325602	0	356007
3	3	Russia	485253	8595	6142	171	242397	11709	236714
4	4	UK	289140	1741	40883	286	?	?	?
5	5	Spain	288797	0	27136	0	?	?	?
6	6	India	273860	7932	7696	223	133764	4669	132400
7	7	Italy	235561	283	34043	79	168646	2062	32872
8	8	Peru	199696	0	5571	0	89556	0	104569
9	9	Germany	186317	112	8802	19	170200	600	7315

Figure 5. The collected web data in CSV with missing value

By using *Replace Missing Values* operator as seen in Figure 7, the missing values of the dataset can be replaced by any replenishment value. RapidMiner provides alternatives of techniques to replace the missing value, which by default when user set *none,* the missing values are not be replaced. Other techniques are *minimum, maximum, average, zero* and *value*[18]. Figure 6 shows the process of replacing missing values with *average* technique and the results can be viewed in Figure 7.

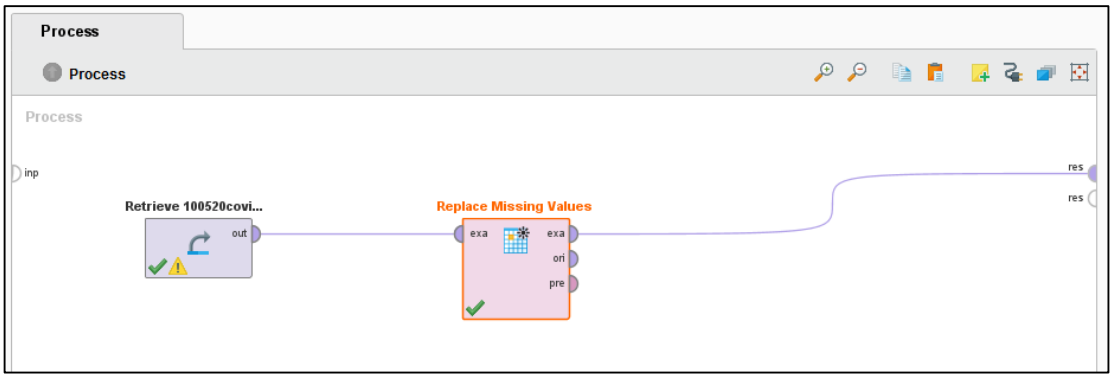


Figure 6. RapidMiner Proses for replacing the missing values

Row No.	No	Country	Total Cases	New Cases	Total Deaths	New Deaths	Total Recov...	Active Cases	Serious Cas...
1	1	USA	2032633	6140	113380	325	774877	1397	1144376
2	2	Brazil	719449	8562	37840	528	325602	0	356007
3	3	Russia	485253	8595	6142	171	242397	11709	236714
4	4	UK	289140	1741	40883	286	20350	210	19423
5	5	Spain	288797	0	27136	0	20350	210	19423
6	6	India	273860	7932	7696	223	133764	4669	132400
7	7	Italy	235561	283	34043	79	168646	2062	32872
8	8	Peru	199696	0	5571	0	89556	0	104569
9	9	Germany	186317	112	8802	19	170200	600	7315

Figure 7. The new dataset after replacing the missing values

3.3 Data visualization

Data visualization is to provide a better insight of the collected data in graphical forms such as maps or graphs to highlight the COVID-19 trends and patterns. Data visualization is important for the decision makers to absorb information quickly and make faster decisions. In this study, word cloud and bubble data visualization have been developed in the RapidMiner software tool.

3.4 Predictive analysis

In this research, simple linear regression based on machine learning approach was used to develop a death prediction model based on the total cases and Figure 8 is the process in RapidMiner.

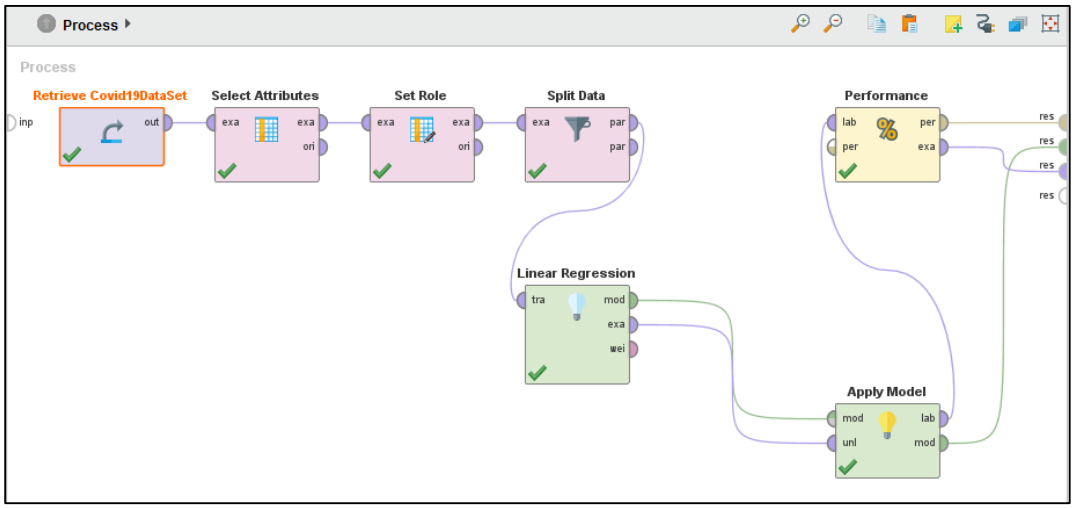


Figure 8. Implementing machine learning with simple linear regression algorithm

By using machine learning approach, the collected dataset needs to be split for training and testing datasets. In the *Split* Data operator, the split ratio was set to 70:30 from the 300 records. *Set Role* operator is to set the dependent variable in this case is the COVID-19 death.

4. Results and Discussion

4.1 Data visualization

Word cloud is a collection or cluster of words depicted in different sizes to be used in this research to present the pattern of the collected data as seen in Figure 9. The United States of America faced the highest cases of COVID-19 followed with Brazil based on the collected data.

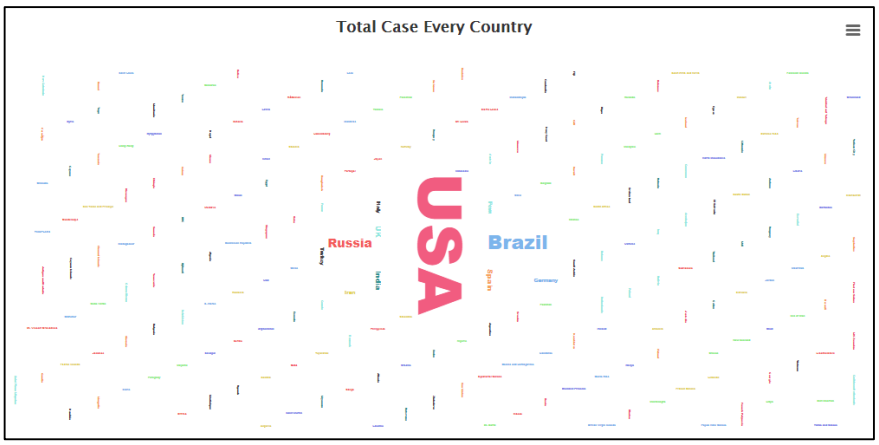


Figure 9. Word cloud

Furthermore, the function of the packed bubble chart to display data in a cluster of circles. Dimensions define the individual bubbles and measures represent the size and color of the different circles. Based on the packed bubble in Figure 10, the United States of America is the most total death in the world followed by the United Kingdom.

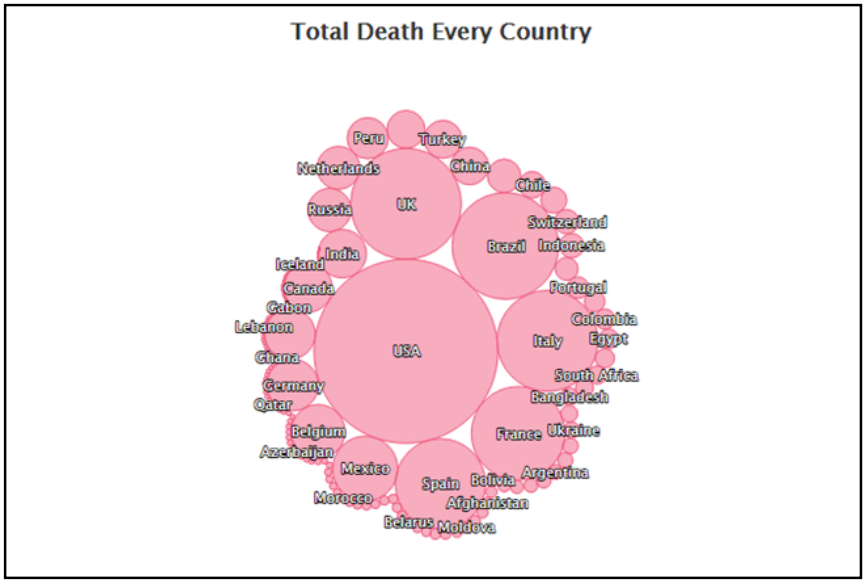


Figure 10. Bubble chart

4.2 Linear Regression

Figure 11 presents the total death based on the prediction and the real values. The prediction values have been produced closely matched with the real values to present the good accuracy of the machine learning linear regression model.

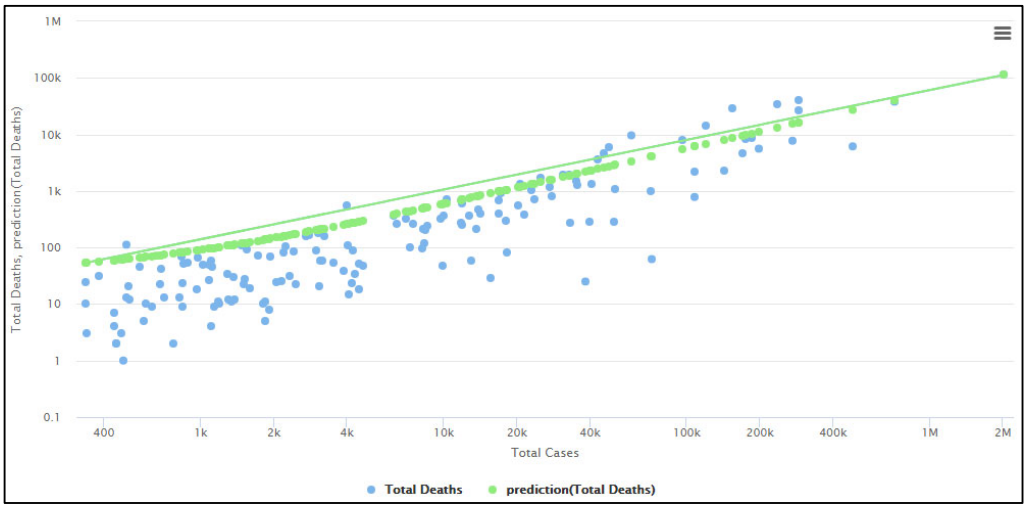


Figure 11. Prediction chart

The R squared of the regression model is 0.89, which considered as good fit enough to present the prediction accuracy as presented in Figure 12, from the RapidMiner.

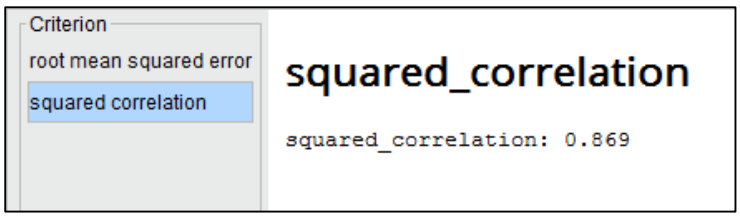


Figure 12. Result of R squared

The squared correlation coefficient or R squared shows the percentage variation in y (total death), which is explained by the independent variable (total cases). The higher the value means the better is the prediction model. Figure 13 is the correlation coefficient between the IV and the DV.

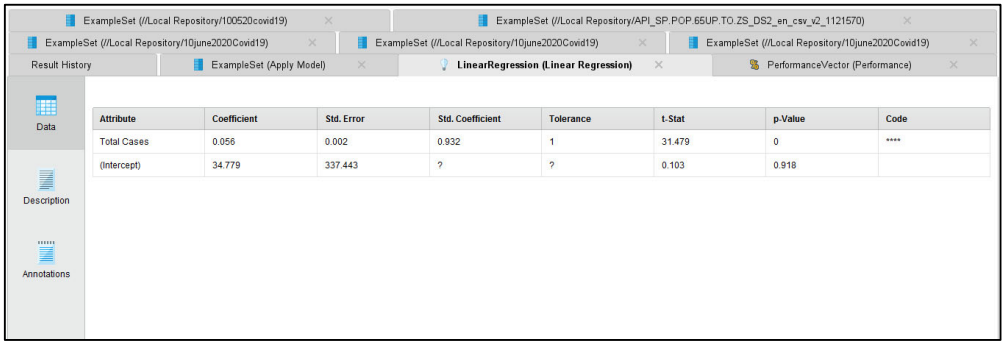


Figure 13. Result of correlation coefficient

The correlation coefficient represents that every total case of COVID-19 will increase the total death of the world population by 0.056. Based on the table, the p-value is less than 0.05, to reject the null hypothesis that total cases not influencing the total death.

5. Conclusion

This paper provides the fundamental implementation techniques on web mining with web scraping data collection and machine learning linear regression on COVID-19. The knowledges are useful for inexpert data scientists to conduct a more extensive web data extraction and rapid data analysis. This research can be expanded by using a more complex data from the website as well as to deploy another type of machine learning predictive analytic. Auto model machine learning and deep learning are the recent advancement of machine learning that would be useful for implementing predictive analytic on a complex dataset.

Acknowledgments

The authors would like to thanks Universiti Teknologi MARA and Yanbu Industrial College for the support.

Conflict of Interest

The authors declare no conflict of interest in the subject matter or materials discussed in this manuscript.

References

[1] A. Monelli and S. B. Sriramoju, “An overview of the challenges and applications towards web mining,” in *2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2018 2nd International Conference on*, 2018, pp. 127–131.

[2] S. Raschka, J. Patterson, and C. Nolet, “Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence,” *Information*, vol. 11, no. 4, p. 193, 2020.

[3] K. Jahanbin and V. Rahmanian, “Using Twitter and web news mining to predict COVID-19 outbreak,” *Asian Pac. J. Trop. Med.*, vol. 13, 2020.

[4] K. Jahanbin, V. Rahmanian, N. Sharifi, and F. Rahmanian, “Sentiment analysis and opinion mining about COVID-19 vaccines of twitter data,” *Pak J Med Heal. Sci*, vol. 15, no. 1, pp. 694–695, 2021.

[5] C. Zhang, M. C. Yu, and S. Marin, “Exploring public sentiment on enforced remote work during COVID-19.,” *J. Appl. Psychol.*, vol. 106, no. 6, p. 797, 2021.

[6] K. K. Bhagat, S. Mishra, A. Dixit, and C.-Y. Chang, “Public opinions about online learning during COVID-19: a sentiment analysis approach,” *Sustainability*, vol. 13, no. 6, p. 3346, 2021.

[7] Y. Lu and Q. Zheng, “Twitter public sentiment dynamics on cruise tourism during the COVID-19 pandemic,” *Curr. Issues Tour.*, vol. 24, no. 7, pp. 892–898, 2021.

[8] T. Pano and R. Kashef, “A complete VADER-based sentiment analysis of bitcoin (BTC) tweets during the era of COVID-19,” *Big Data Cogn. Comput.*, vol. 4, no. 4, p. 33, 2020.

[9] S. Kushwaha *et al.*, “Significant applications of machine learning for COVID-19 pandemic,” *J. Ind. Integr. Manag.*, vol. 5, no. 04, pp. 453–479, 2020.

[10] Y. Zoabi, S. Deri-Rozov, and N. Shomron, “Machine learning-based prediction of COVID-19 diagnosis based on symptoms,” *npj Digit. Med.*, vol. 4, no. 1, pp. 1–5, 2021.

[11] M. A. Elaziz, K. M. Hosny, A. Salah, M. M. Darwish, S. Lu, and A. T. Sahlol, “New machine learning method for image-based diagnosis of COVID-19,” *PLoS One*, vol. 15, no. 6, p. e0235187, 2020.

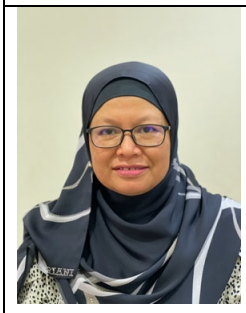
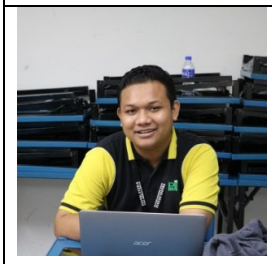
[12] I. Rahimi, F. Chen, and A. H. Gandomi, “A review on COVID-19 forecasting models,” *Neural Comput. Appl.*, pp. 1–11, 2021.

[13] S. Bhandari *et al.*, “Logistic regression analysis to predict mortality risk in COVID-19 patients from routine hematologic parameters,” *Ibnosina J. Med. Biomed. Sci.*, vol. 12, no. 2, p. 123, 2020.

[14] I. Huang, M. A. Lim, and R. Pranata, “Diabetes mellitus is associated with increased mortality

and severity of disease in COVID-19 pneumonia--a systematic review, meta-analysis, and meta-regression," *Diabetes & Metab. Syndr. Clin. Res. & Rev.*, vol. 14, no. 4, pp. 395–403, 2020.
[15] A. H. F. Laender, B. A. Ribeiro-Neto, A. S. Da Silva, and J. S. Teixeira, "A brief survey of web data extraction tools," *ACM Sigmod Rec.*, vol. 31, no. 2, pp. 84–93, 2002.
[16] K. Jarmul and R. Lawson, *Python Web Scraping*. Packt Publishing Ltd, 2017.
[17] N. Silaparasetty, "Introduction to Jupyter Notebook," in *Machine Learning Concepts with Python and the Jupyter Notebook Environment*, Springer, 2020, pp. 91–118.
[18] E. D. Madyatmadja, S. I. Jordan, and J. F. Andry, "Big Data Analysis Using Rapidminer Studio To Predict Suicide Rate In Several Countries," *ICIC Express Lett. Part B Appl.*, vol. 12, no. 8, 2021.

Biography of all authors

Picture	Biography	Authorship contribution
	Aizal Yusrina Idris is currently a lecturer in Yanbu Industrial College, Kingdom of Saudi Arabia. She is currently pursuing her PhD in Education Training, in University Malaya, Malaysia. She has a Master in Information Technology (Computer Science) from the University Kebangsaan Malaysia, Malaysia, and first degree also from the same university in Information Technology (Industrial Computing). Her research interest is the use of IT in teaching and learning, as well as in teacher's training, and system's evaluation methods in education. She can be contacted at idrisa@rcyci.edu.sa	Design the research work and completing the article
	Razan Bamoallem is currently a lecturer in Yanbu Industrial College, Kingdom of Saudi Arabia. She has completed her first degree in Computer Science from the same university. Her also has a Master in Computer Science, from Glasgow University, UK. Her research interest is in Human Computer Interaction and software interface design. She can be contacted at bamoallemr@rcyci.edu.sa	Data collection
	**Mohamad Harith Azfar Mohamad Hatta** is a master student in the Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA,Shah Alam,Selangor Malaysia. He has completed her first degree in Computer Science from the same university at the Perak branch, Tapah Campus. His research interest is	Regression analysis

	machine learning and information retrieval. He can be contacted at 2021782731@student.uitm.edu.my His LinkedIn profile can be accessed at linkedin.com/in/mohamad-harith-azfar-mohamad-hatta-329aa0141	