Đại Học Bách Khoa Hà Nội

Trường Công nghệ thông tin và truyền thông

======o0o======

BÁO CÁO

MÔN: Nhập môn học máy và khai phá dữ liệu

Đề tài: dự đoán số ca mắc sốt xuất huyết ở Việt Nam sử dụng LSTM

Giảng viên hướng dẫn: PGS.TS Nguyễn Thị Kim Anh

Nhóm thực hiện 7:

Nguyễn Khắc Thái Bình	20204944
Nguyễn Sỹ Anh Khoa	20205088
Nguyễn Quốc Huy	20204987
Trần Văn Hiếu	20200231

Hà Nội, ngày 1 tháng 1 năm 2024

Mục lục

1. Tóm tắt

Từ trước đến nay, dịch bệnh sốt xuất huyết, với tác nhân gây bệnh chủ yếu là do muỗi Aedes aegypti thường chủ yếu xuất hiện vào thời điểm mùa mưa hàng năm, thời điểm mà muỗi sinh sản và phát triển mạnh nhất. Tuy nhiên, trong những năm gần đây tình hình dịch bệnh này đã có những diễn biến khó lường, không chỉ xảy ra ở quanh mùa mưa mà còn diễn biến quanh năm, nhất là vào những tháng cuối năm. Theo thống kê số lượng ca bệnh ở riêng Việt Nam trong năm nay thì cả nước hiện đã ghi nhận hơn 93.800 ca mắc sốt xuất huyết, 26 trường hợp tử vong. Tại Hà Nội, số ca mắc sốt xuất huyết vẫn tiếp tục tăng, toàn thành phố đã ghi nhận trên 15.300 ca. Chính vì vậy, nhóm muốn thông qua bài tập lớn môn học để phân tích những yếu tố về thời tiết có ảnh hưởng như thế nào đến sự phát triển, hoành hành của dịch bệnh để từ kết quả thu thập được có thể đưa ra các phân tích về các yếu tố môi trường với dịch bệnh, dự đoán được với những tình hình thời tiết của các thành phố thì số lượng ca bệnh sẽ diễn biến như thế nào.

2. Giới thiệu chung

Trong môi trường địa lý và khí hậu đa dạng của Việt Nam, sốt xuất huyết là một vấn đề sức khỏe cộng đồng quan trọng. Việc hiểu rõ mối liên quan giữa số ca mắc và các yếu tố thời tiết có thể giúp nâng cao khả năng dự đoán và quản lý dịch bệnh. Đề tài "Phân tích và dự đoán số ca mắc sốt xuất huyết ở Việt Nam dựa trên thời tiết" nhấn mạnh sự kết hợp giữa hai lĩnh vực quan trọng là y tế và khí tượng học để tạo ra một hệ thống thông tin động và hiệu quả.

Trong vài thập kỷ gần đây, sốt xuất huyết đã trở thành một thách thức lớn đối với hệ thống y tế Việt Nam. Điều này yêu cầu sự đổi mới trong việc thu thập và phân tích dữ liệu để có cái nhìn toàn diện về các yếu tố ảnh hưởng đến sự bùng phát của dịch bệnh. Đề tài này đặt ra mục tiêu nghiên cứu và xây dựng một mô hình phân tích sự biến động của sốt xuất huyết dựa trên các yếu tố thời tiết, từ nhiệt độ, độ ẩm đến mức lượng mưa.

Thông qua việc tích hợp dữ liệu y tế và dữ liệu thời tiết, chúng tôi mong muốn xây dựng một hệ thống có khả năng dự đoán xu hướng sốt xuất huyết và đánh giá rủi ro theo thời gian. Điều này có thể giúp các cơ quan y tế và chính phủ có cái nhìn sâu sắc về tình hình sức khỏe cộng đồng và áp đặt các biện pháp phòng chống hiệu quả.

Đồng thời, sự hiểu biết về ảnh hưởng của thời tiết sẽ mở ra cơ hội để phát triển các chiến lược ngăn chặn và ứng phó với sốt xuất huyết dựa trên dự đoán thời tiết. Hy vọng rằng, kết quả của nghiên cứu này sẽ góp phần quan trọng vào việc nâng cao khả năng dự báo và quản lý sức khỏe cộng đồng trong bối cảnh thách thức của các dịch bệnh truyền nhiễm.

3. Thu thập dữ liệu

3.1 Tổng quan

Với yêu cầu được đặt ra, bài toán cần có những dữ liệu về số ca mắc sốt xuất huyết, các dữ liệu về yếu tố thời tiết có thể ảnh hưởng đến tình hình dịch bệnh ở đây được nhóm đặt ra là thống kê về nhiệt độ và lượng mưa, độ ẩm. Từ việc đặt ra các mục tiêu dữ liệu cần phân tích, nhóm đã triển khai tìm kiếm những thông tin, thống kê về những nội dung trên.

3.2 Thông tin về số ca bệnh

Khi đặt vấn đề về thông tin số ca mắc bệnh sốt xuất huyết, nhóm đã tìm kiếm thông tin về chúng trên các trang web thống kê của Việt Nam, tuy nhiên các thông tin này ở trên các trang web thống kê vẫn còn thiếu chi tiết, chưa đủ để nhóm thu thập nên nhóm đã tìm kiếm thông tin này ở trên các trang web thống kê quốc tế về sốt xuất huyết.

Nhóm đã tìm kiếm được dữ liệu về số ca bệnh ở nguồn sau:

https://opendengue.org/data.html?fbclid=IwAR3OOmoFxiX5ZuISTlwdScDA3qz7_qWqoPo8F1RKM_6fHwQK7QvC3V9b0wA

Cụ thể dữ liệu đã thu thập được như sau:

	adm_0_na	adm_1_na	adm_2_na	full_name	ISO_A0	FAO_GAUL	RNE_iso_c	IBGE_code	calendar_st	calendar_e	Year	dengue_tot	case_defini	S_res	T_res	UUID
2	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	3/1/2021	9/1/2021	2021	101	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
3	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	10/1/2021	16/1/2021	2021	151	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
4	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	17/1/2021	23/1/2021	2021	201	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
5	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	24/1/2021	30/1/2021	2021	202	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
6	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	31/1/2021	6/2/2021	2021	100	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
7	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	7/2/2021	13/2/2021	2021	251	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
8	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	14/2/2021	20/2/2021	2021	101	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
9	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	21/2/2021	27/2/2021	2021	61	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
10	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	28/2/2021	6/3/2021	2021	99	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
11	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	7/3/2021	13/3/2021	2021	112	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
12	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	14/3/2021	20/3/2021	2021	70	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
13	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	21/3/2021	27/3/2021	2021	70	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
14	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	28/3/2021	3/4/2021	2021	99	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
15	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	4/4/2021	10/4/2021	2021	99	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
16	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	11/4/2021	17/4/2021	2021	75	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
17	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	18/4/2021	24/4/2021	2021	99	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
18	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	25/4/2021	1/5/2021	2021	94	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
19	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	2/5/2021	8/5/2021	2021	77	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
20	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	9/5/2021	15/5/2021	2021	150	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
21	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	16/5/2021	22/5/2021	2021	101	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
22	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	23/5/2021	29/5/2021	2021	99	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
23	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	30/5/2021	5/6/2021	2021	110	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
24	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	6/6/2021	12/6/2021	2021	97	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
25	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	13/6/2021	19/6/2021	2021	91	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
26	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	20/6/2021	26/6/2021	2021	53	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
27	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	27/6/2021	3/7/2021	2021	51	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05
28	AFGHANIST	NA	NA	AFGHANIST	AFG	1011446	AFG	NA	4/7/2021	10/7/2021	2021	67	Suspected	Admin0	Week	WHOEMRO-ALL-2021-Y01-05

Trong đó các cột adm_0, adm_1,adm_2 là các trường về địa chỉ (quốc gia, thành phố), các cột calendar time là cột lịch và cột dengue total là số ca mắc trong tuần đó

3.3 Thông tin về thời tiết

Với yêu cầu tìm kiếm thông tin thời tiết theo ngày của Việt Nam, cụ thể nhóm hướng tới dữ liệu thống kê của Hồ Chí Minh, nhóm đã tìm đến những trang web có thể crawl được thông tin lịch sử về thời tiết của thành phố và tìm được trang https://www.visualcrossing.com/ có dữ liệu lịch sử thông tin thời tiết của các thành phố. Ở trang web có chứa nhiều thông tin về thời tiết như nhiệt độ, lượng mưa, độ ẩm,.... phù hợp với yêu cầu về dữ liệu của nhóm.

Nhóm đã gọi api dữ liệu thời tiết từ trang web, sau đó dữ liệu nhận về dưới dạng json và nhóm có viết thêm chương trình chuyển đổi từ json sang csv. Tuy nhiên vì việc web chỉ giới hạn việc lấy data ở mức 1000 dòng/ngày nên nhóm phải gọi api nhiều lần trong tuần để thu thập đủ dữ liệu cần thiết.

Dưới đây là mã nguồn convert file.json sang file.csv:

```python
import os
import csv
import json
from collections import OrderedDict

# OPEN A JSON FILE
try:
    filename = input("Filename: ")
    extension = filename.split(".")[-1].lower()

    f = open(filename)

    if extension == "csv":
        # load csv file
        data = list(csv.reader(f))
        print("CSV file loaded")
    elif extension == "json":
        # load json file
        data = json.load(f,object_pairs_hook=OrderedDict)
        print("JSON file loaded")
    else:
        print("unsupported file type ... exiting")
        exit()
except Exception as e:
    # error loading file
    print("Error loading file ... exiting:",e)
    exit()
else:
    # CONVERT CSV TO JSON
    if extension == "csv":
        keys = data[0]
        converted = []

        for i in range(1, len(data)):
            obj = OrderedDict()
            for j in range(0,len(keys)):
```

```
            if len(data[i][j]) > 0:
                obj[keys[j]] = data[i][j]
            else:
                obj[keys[j]] = None
        converted.append(obj)

# CONVERT JSON TO CSV
if extension == "json":

    # get all keys in json objects
    keys = []
    for i in range(0,len(data)):
        for j in data[i]:
            if j not in keys:
                keys.append(j)

    # map data in each row to key index
    converted = []
    converted.append(keys)

    for i in range(0,len(data)):
        row = []
        for j in range(0,len(keys)):
            if keys[j] in data[i]:
                row.append(data[i][keys[j]])
            else:
                row.append(None)
        converted.append(row)

# CREATE OUTPUT FILE
converted_file_basename = os.path.basename(filename).split(".")[0]
converted_file_extension = ".json" if extension == "csv" else ".csv"

if(os.path.isfile(converted_file_basename + converted_file_extension)):
    counter = 1
    while os.path.isfile(converted_file_basename + " (" + str(counter) + ")" + converted_file_extension):
        counter += 1
    converted_file_basename = converted_file_basename + " (" + str(counter) + ")"

try:
    if converted_file_extension == ".json":
        with open(converted_file_basename + converted_file_extension, 'w') as outfile:
            json.dump(converted, outfile)
    elif converted_file_extension == ".csv":
        with open(converted_file_basename + converted_file_extension, 'w') as outfile:
            writer = csv.writer(outfile)
            writer.writerows(converted)
except:
    print("Error creating file ... exiting")
else:
    print("File created:",converted_file_basename + converted_file_extension)
```

4. Tiền xử lý dữ liệu

Nhóm đã thu thập dữ liệu số ca mắc sốt xuất huyết từ năm 1990-2019.

Dữ liệu số ca mắc sốt xuất huyết sau khi đã tiền xử lý bao gồm các trường Country, City, start, end, dengue. Tuy nhiên, dữ liệu đã bị thiếu mất năm 2011 và nhóm không thể thu thập dữ liệu năm này từ các nguồn khác nhau. Bên cạnh đó, từ năm 1994-2010, dữ liệu có đầy đủ thông tin số ca mắc theo 63 tỉnh, thành phố và theo tháng, từ năm 2012-2015 dữ liệu số ca mắc chỉ theo tháng và

2016-2019 dữ liệu được thống kê theo tuần. Do đó, rất từ dữ liệu rất khó để có thể phân tích được toàn bộ.

Về dữ liệu nhiệt độ, lượng mưa, thông tin thu thập được từ năm 1900-2010 theo từng ngày. Nhóm đã thay đổi từ dữ liệu ngày sang dữ liệu tháng bằng cách lấy trung bình cộng các số liệu khí hậu các ngày vào tháng. Dữ liệu này đến từ nguồn thông tin khác nên cần phải ghép dữ liệu vào số ca mắc để có thể nhận xét được sự ảnh hưởng của khí hậu tới số ca mắc sốt xuất huyết.

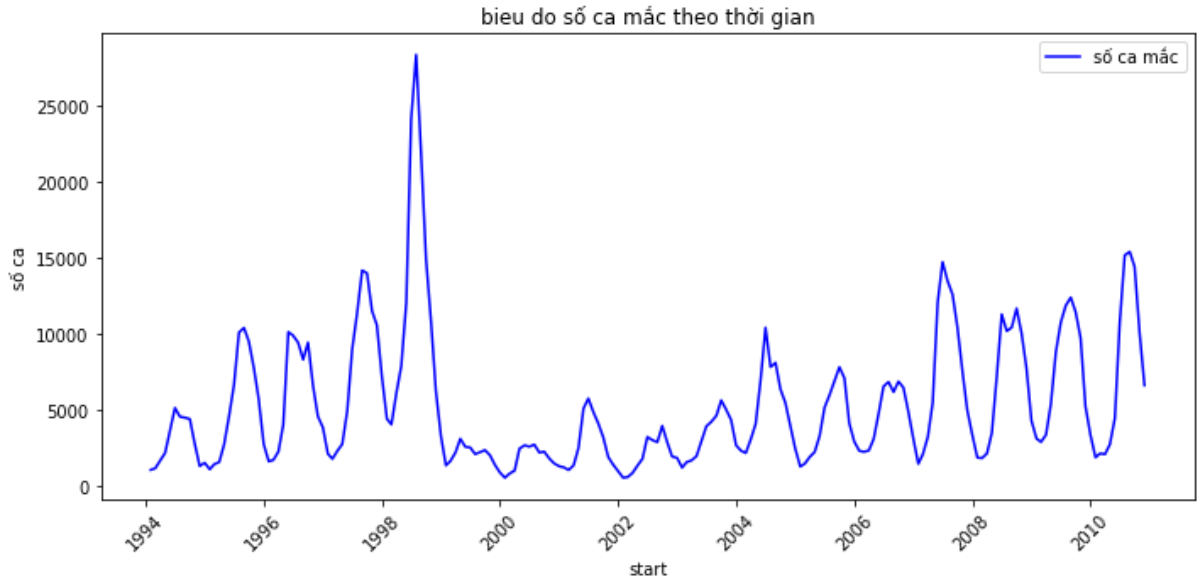
Từ những điều trên, sau khi đã tiền xử lý loại bỏ các trường dư thừa, bổ sung các trường dữ liệu còn thiếu, nhóm quyết định sử dụng dữ liệu sau cho mô hình học máy: **năm 1994-2010** với các trường thông tin đất nước, năm, tháng, ngày bắt đầu, ngày kết thúc, số ca mắc, lượng mưa, nhiệt độ

	country	year	Month	start	end	dengue	rainfall	temperature
0	VIET NAM	1994	2	1994-02-01	1994-02-28	1095	32.9093	22.3773
1	VIET NAM	1994	3	1994-03-01	1994-03-31	1199	62.2816	21.1490
2	VIET NAM	1994	4	1994-04-01	1994-04-30	1719	35.6912	26.6498
3	VIET NAM	1994	5	1994-05-01	1994-05-31	2208	235.4870	27.5385
4	VIET NAM	1994	6	1994-06-01	1994-06-30	3686	248.9170	26.9911
...	...	...	...	...	...	...	...	...
198	VIET NAM	2010	8	2010-08-01	2010-08-31	15169	258.2670	27.1810
199	VIET NAM	2010	9	2010-09-01	2010-09-30	15395	208.8840	26.9750
200	VIET NAM	2010	10	2010-10-01	2010-10-31	14411	210.1980	24.6371
201	VIET NAM	2010	11	2010-11-01	2010-11-30	10070	105.9540	22.7269
202	VIET NAM	2010	12	2010-12-01	2010-12-31	6639	70.3632	21.5358

203 rows × 8 columns

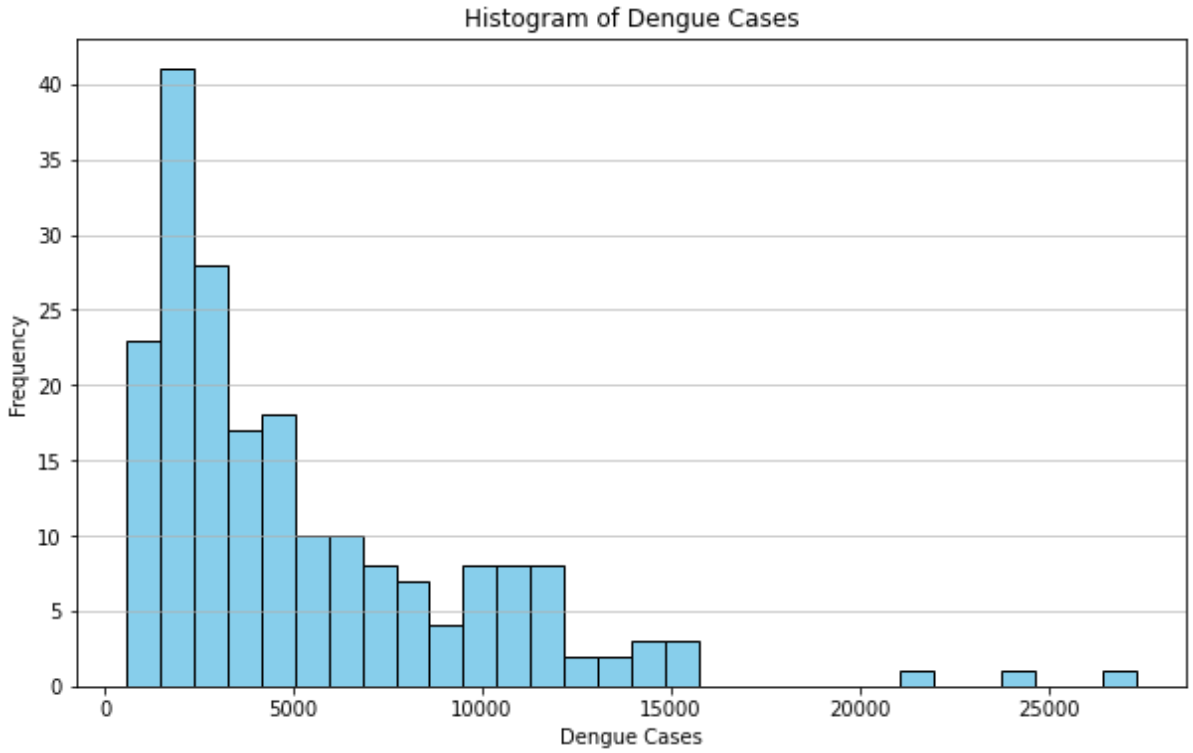
5. Phân tích dữ liệu

Biểu đồ đường cho số ca mắc theo thời gian:



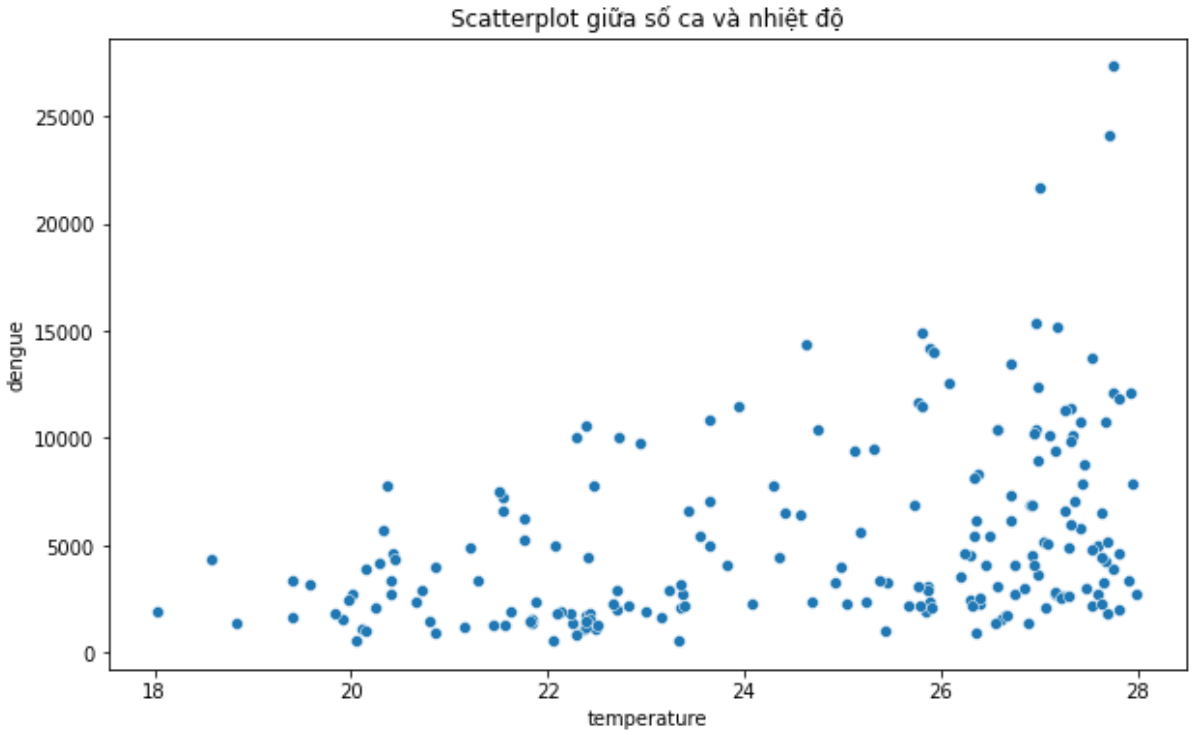
Nhận xét: Số ca mắc nhiều nhất là đạt đỉnh năm 1998 và có xu hướng lặp lại theo chu kỳ trong giai đoạn từ năm 2007 đến 2010

Biểu đồ các ca sốt xuất huyết:



Nhận xét: Ta thấy mật độ xuất hiện của số lượng ca mắc bệnh tương đối thường xuyên, mỗi lần bùng phát sẽ vào khoảng trên dưới 5000 ca mắc được ghi nhận.

Biểu đồ Scatterplot của mối quan hệ giữa ca mắc và nhiệt độ:

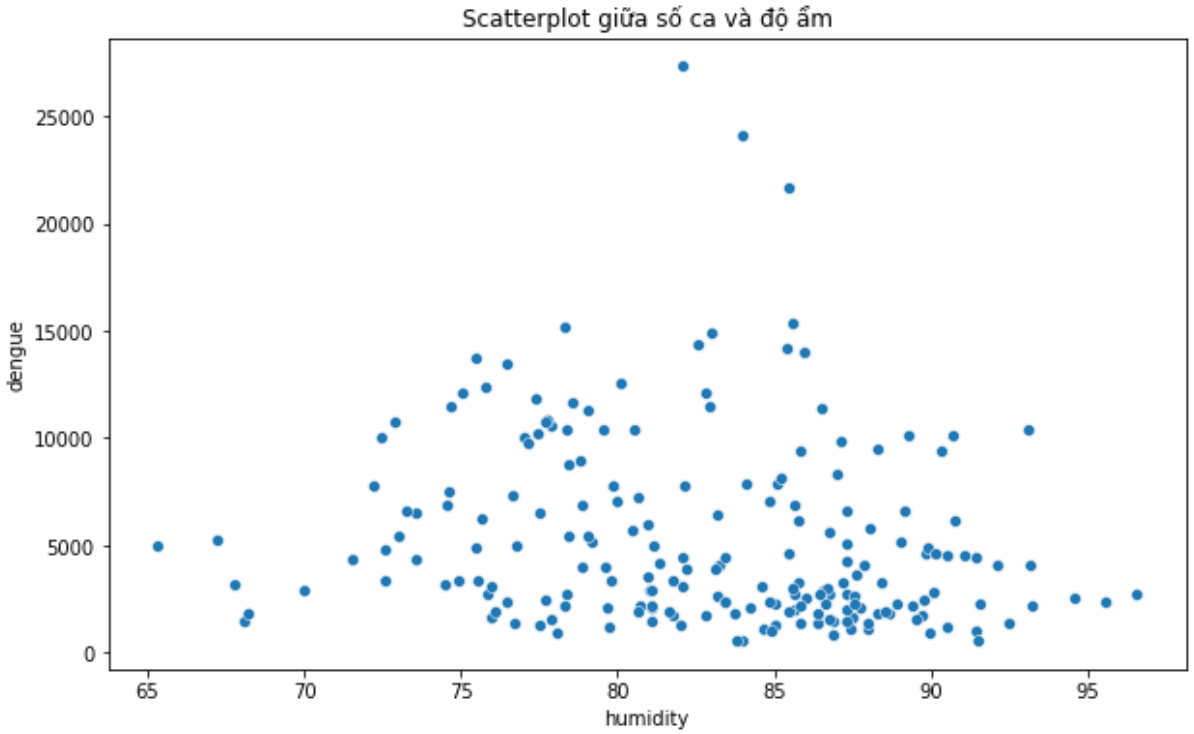


Nhận xét: Nhiệt độ càng cao khiến cho dịch bệnh lây lan nhanh chóng, đỉnh điểm số ca mắc khi nhiệt độ đạt 28 độ.

Biểu đồ Scatterplot của mối quan hệ giữa ca mắc và lượng mưa:

Nhận xét: Số ca mắc đạt đỉnh điểm khi lượng mưa trong khoảng từ 200mm đến 300mm.

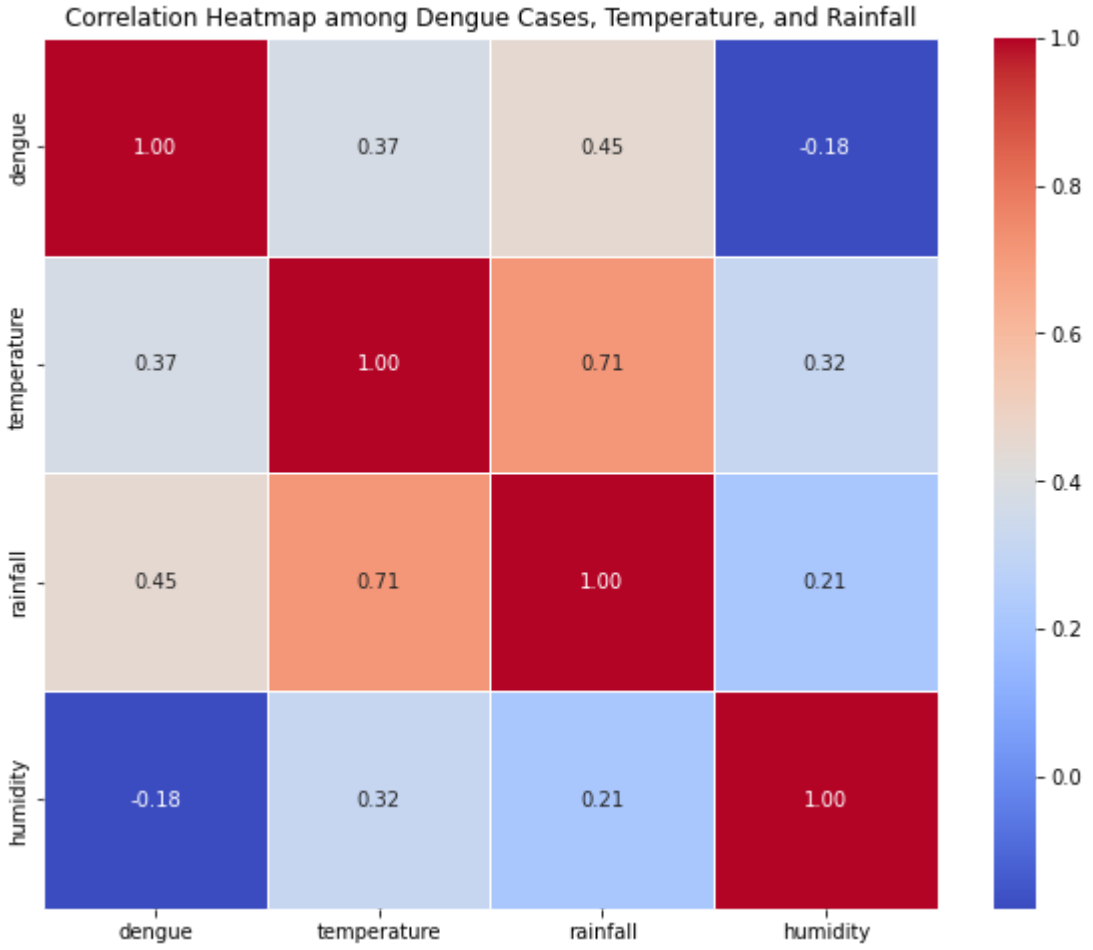
Biểu đồ Scatterplot của mối quan hệ giữa ca mắc và độ ẩm:



Nhận xét: Độ ẩm từ 80 - 90% là độ ẩm thích hợp cho dịch bệnh bùng phát.

Qua các biểu đồ, ta có thể nhận xét: Dịch bệnh bùng phát mạnh vào những năm 1997-1998, phần lớn là vào những tháng mưa như tháng 7, 8, 9. Nhiệt độ dễ bùng phát dịch nhất là khoảng 28 độ, lượng mưa khoảng từ 200 đến 300 mm, độ ẩm giao động từ 80%.

Biểu đồ Heatmap cho mối tương quan giữa ca mắc, nhiệt độ, lượng mưa và độ ẩm:



Ta có thể thấy, mức độ tương quan của các yếu tố khí hậu đến số ca mắc khá thấp, điều này nguyên nhân có thể do số liệu được kết hợp từ 2 trang web khác nhau. Do đó, dự đoán số ca mắc sốt xuất huyết dựa vào khí hậu sẽ đưa ra kết quả không chính xác. Vì vậy, chúng em quyết định dự đoán số ca mắc theo thời gian thực time-series.

6. Huấn luyện mô hình mạng LSTM

6.1. Mạng neuron nhân tạo

6.1.1. RNN

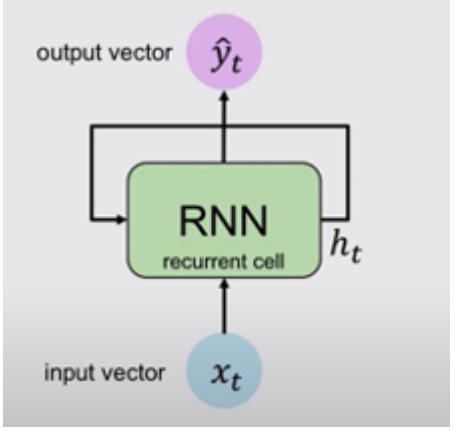
6.1.1.1. Sơ lược về mạng RNN

Mạng nơ-ron hồi quy (Recurrent Neural Network) là một dạng mạng nơ-ron nhân tạo chứa các vòng lặp bên trong nó, trong đó đầu ra của bước trước đó được đưa làm đầu vào của bước tiếp theo. Đối với những mạng nơ ron truyền thống khác, trong đó đầu vào và đầu ra của mạng là độc lập với nhau, kiến trúc này không phù hợp với một số bài toán đòi hỏi xử lý chuỗi các thông tin và cần ghi nhớ các thông tin được

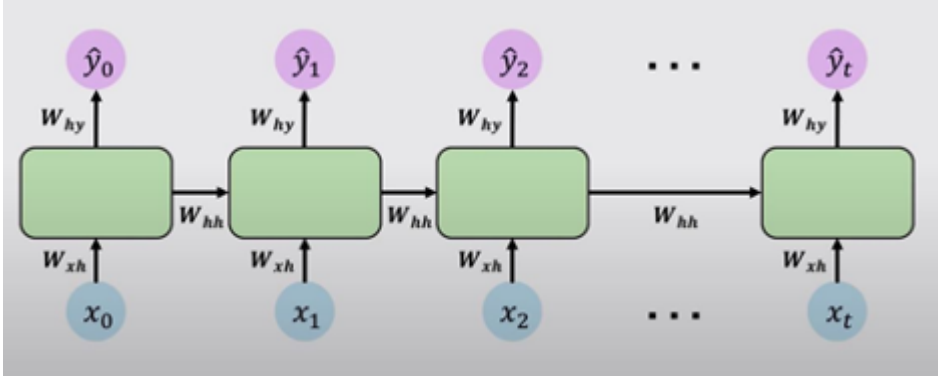
tính toán trước đó, ví dụ như bài toán dự đoán từ tiếp theo của một câu đòi hỏi phải ghi nhớ những từ đã được xuất hiện trước đó. Mạng nơ-ron hồi quy đã giải quyết được vấn đề này bằng cách sử dụng hidden state (nơi lưu trữ thông tin của chuỗi dữ liệu) được cập nhật trong mỗi vòng lặp.

6.1.1.2. Cấu trúc mạng RNN

Một mạng RNN tại mỗi bước lặp có cấu trúc như sau:



Phân giải theo thời gian:



Trong đó, tại mỗi thời điểm t, mạng RNN sẽ nhận dữ liệu đầu vào x(t), thực hiện cập nhật hidden state ht tính toán output tại thời điểm t: yt và truyền ht đến bước lặp tiếp theo.

Quy tắc chung cập nhật hidden state:

$$h_t = f(h_{t-1}, x_t)$$

Hàm thường được lựa chọn:

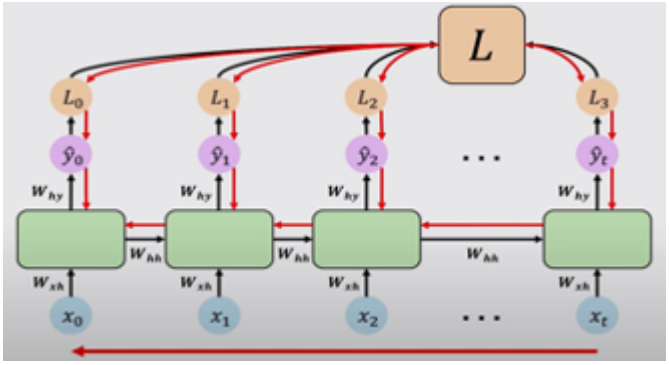
$$h_t = \tanh(W_{hh}h_{t-1}+W_{xh}x_t)$$

Tính toán output đầu ra:

$$y_t = W_{hy}h_t$$

Trong đó: W_{hh}, W_{xh}, W_{hy} là các vecto trọng số được sử dụng chung cho tất cả các vòng lặp.

6.1.1.3. Huấn luyện mạng RNN



Huấn luyện mạng RNN cũng tương tự như các mạng nơ-ron truyền thống, tuy nhiên giải thuật lan truyền ngược (backpropagation) phải thay đổi một chút. Đạo hàm tại mỗi đầu ra phụ thuộc không chỉ vào các tính toán tại bước đó, mà còn phụ thuộc vào các bước trước đó nữa, vì các tham số trong mạng RNN được sử dụng chung cho tất cả các bước trong mạng. Ví dụ, để tính đạo hàm tại t = 4 ta phải lan truyền ngược cả 3 bước phía trước rồi cộng tổng đạo hàm của chúng lại với nhau. Việc tính đạo hàm kiểu này được gọi là lan truyền ngược liên hồi (BPTT - Backpropagation Through Time).

6.1.1.4. Nhược điểm của RNN

Vấn đề về phụ thuộc xa (Long-Term Dependencies):

Khi thực hiện cập nhật trọng số đối với mạng rnn có nhiều vòng lặp, khi đó xuất hiện vấn đề về vanishing gradient khi cập nhật trong số với các vòng lặp tại thời điểm xa trước đó, dẫn đến việc mạng rnn hầu như không cập nhật tại các vòng lặp xa và sẽ có xu hướng ghi nhớ các thông tin ở thời điểm gần với hiện tại, do đó mạng sẽ không làm việc hiệu quả với những bài toán đòi hỏi phải xử lý các phụ thuộc xa (long-term dependencies)

Một số cách giải quyết:

- Sử dụng là ReLU thay vì hàm tanh
- Khởi tạo vector trọng số dạng đơn vị để giảm tốc độ suy biến của gradient
- Sử dụng Gate Cell (được sử dụng trong mạng LSTM, GRU, v..v..)

6.1.2. LSTM

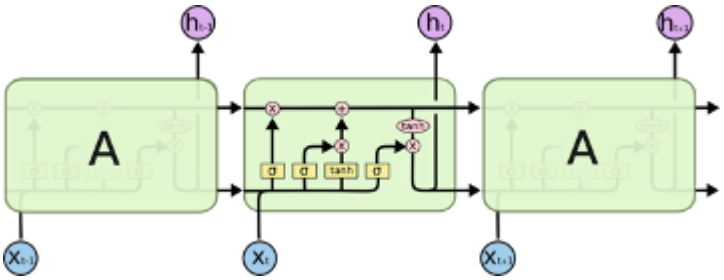
6.1.2.1. Sơ lược về mạng LSTM

Mạng bộ nhớ dài-ngắn (Long Short Term Memory networks), thường được gọi là LSTM - là một dạng đặc biệt của RNN, nó có khả năng học được các phụ thuộc xa. LSTM được giới thiệu bởi Hochreiter & Schmidhuber (1997), và sau đó đã được cải tiến và phổ biến bởi rất nhiều người trong ngành. Chúng hoạt động cực kì hiệu quả trên nhiều bài toán khác nhau nên dần đã trở nên phổ biến như hiện nay.

LSTM được thiết kế để tránh được vấn đề phụ thuộc xa (long-term dependency). Việc nhớ thông tin trong suốt thời gian dài là đặc tính mặc định của chúng, chứ ta không cần phải huấn luyện nó để có thể nhớ được. Tức là ngay nội tại của nó đã có thể ghi nhớ được mà không cần bất kì can thiệp nào.

6.1.2.2. Cấu trúc mạng LSTM

LSTM cũng có kiến trúc dạng chuỗi giống như RNN, nhưng các mô-đun trong nó có cấu trúc khác với mạng RNN chuẩn. Thay vì chỉ có một tầng mạng nơ-ron, chúng có tới 4 tầng tương tác với nhau một cách rất đặc biệt.

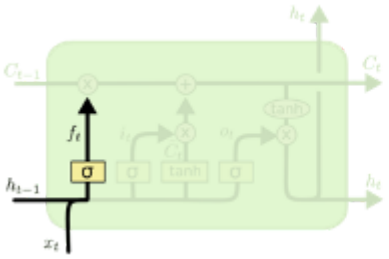


Chìa khóa của LSTM là trạng thái tế bào (cell state) - chạy xuyên suốt tất cả các mắt xích (các nút mạng) và chỉ tương tác tuyến tính đôi chút. Vì vậy mà các thông tin có thể dễ dàng truyền đi thông suốt mà không sợ bị thay đổi.

LSTM có khả năng bỏ đi hoặc thêm vào các thông tin cần thiết cho trạng thái tế bào thông qua một cơ chế được gọi là cổng (gate) bao gồm một mạng sigmoid và một phép nhân.

Trong mỗi bước lặp của LSTM thông tin được xử lý như sau:

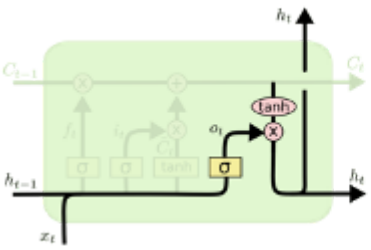
1. Quyết định xem thông tin nào cần bỏ đi từ trạng thái tế bào (cell state) thông qua forget gate layer qua đó, đầu ta của tâng sigmoid là một giá trị thuộc đoạn [0,1] quy định lượng thông tin được giữ lại của trạng thái tế bào ct-1 trong đó 1 có nghĩa là giữ lại toàn bộ, và 0 nghĩa là toàn bộ thông tin sẽ bị bỏ đi.



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

2. Lựa chọn thông tin mới sẽ được lưu và trong trạng thái tế bào (cell state), phần này bao gồm 3 bước: lựa chọn thông tin để thêm vào trạng thái tế bào, từ đó tính toán ra được thông tin sẽ được thêm và cập nhật thông tin đó vào trạng thái tế bào.

3. Tính toán output để gửi đến bước lặp tiếp theo, Giá trị đầu ra sẽ dựa vào trạng thái tế bào, nhưng sẽ được tiếp tục sàng lọc. Đầu tiên, ta chạy một tầng sigmoid để quyết định phần nào của trạng thái tế bào ta muốn xuất ra. Sau đó, ta đưa nó trạng thái tế bảo qua một hàm tanhtanh để co giá trị nó về khoảng [−1,1], và nhân nó với đầu ra của cổng sigmoid để được giá trị đầu ra ta mong muốn.



$$o_t = \sigma\left(W_o \; [h_{t-1}, x_t] \; + \; b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

6.1.3: Đánh giá mô hình

Để đánh giá mô hình học máy này, chúng em sử dụng 3 độ đo sau:

- phù hợp (R^2 score): R^2 score đo lường tỷ lệ biến thiên của biến phụ thuộc được giải thích bởi biến độc lập trong mô hình. Giá trị R^2 càng gần 1 thì mô hình càng tốt.

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \overline{y})^2}$$

- Mean Absolute Error (MAE): Đây là độ đo đo lường sự khác biệt trung bình giữa dự đoán và giá trị thực tế. MAE cho biết độ lệch trung bình của dự đoán so với giá trị thực tế.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n} |x_i - x|$$

- Mean Squared Error (MSE): Đây là độ đo phổ biến khác để đánh giá hiệu suất của mô hình. MSE tính toán giá trị trung bình của bình phương sự khác biệt giữa dự đoán và giá trị thực tế.

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n} (y_i - \tilde{y}_i)^2$$

6.2: Xây dựng mô hình LSTM

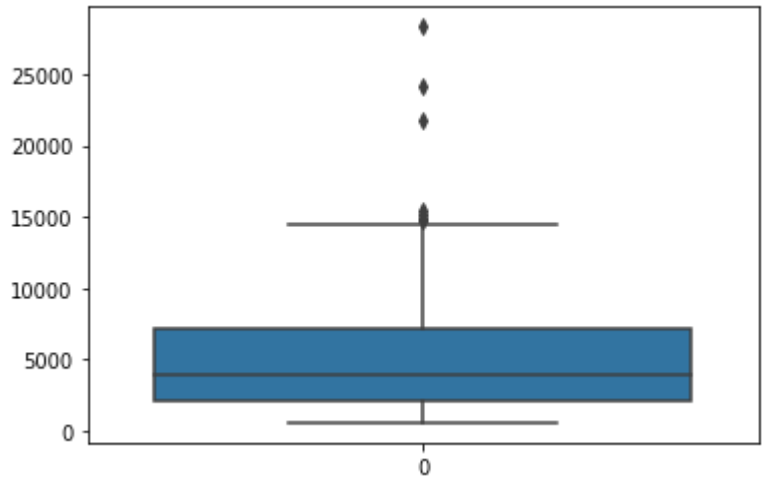
6.2.1: Tiền xử lý

Để điều chỉnh dữ liệu thực tế sao cho nó thích nghi với những phân tích chuỗi thời gian thì những dữ liệu tiếp tục qua một bước tiền xử lý dữ liệu. Việc tiền xử lý dữ liệu có thể giúp xử lý các vấn đề như tính toán các giá trị còn thiếu, loại bỏ các giá trị ngoại lai và tính toán các giá trị biến động.

Xác định giá trị ngoại lai trong phân tích chuỗi thời gian (Identifying Outliers in Time Series Analysis): Các giá trị ngoại lai là những quan sát cực đoan so với phần còn lại của dữ liệu. Các giá trị ngoại lai có thể làm hỏng ước tính của mô hình và do đó dẫn đến các dự đoán kém chính xác hơn. Dưới đây là xem xét dữ liệu một lần nữa, lần này phân tích dữ liệu đó để tìm các giá trị ngoại lai bằng

cách sử dụng hàm IQR, thu được kết quả sau:

	start	dengue
53	1998-07-01	24087
54	1998-08-01	28315
55	1998-09-01	21713
56	1998-10-01	14931
161	2007-07-01	14723
198	2010-08-01	15169
199	2010-09-01	15395

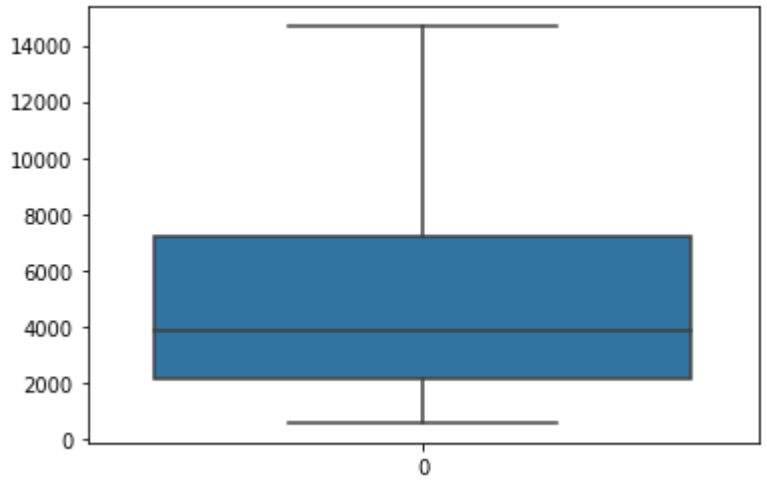


Để xử lý ngoại lai mà không làm mất mát thông tin của các điểm này, có 3 phương pháp chính:

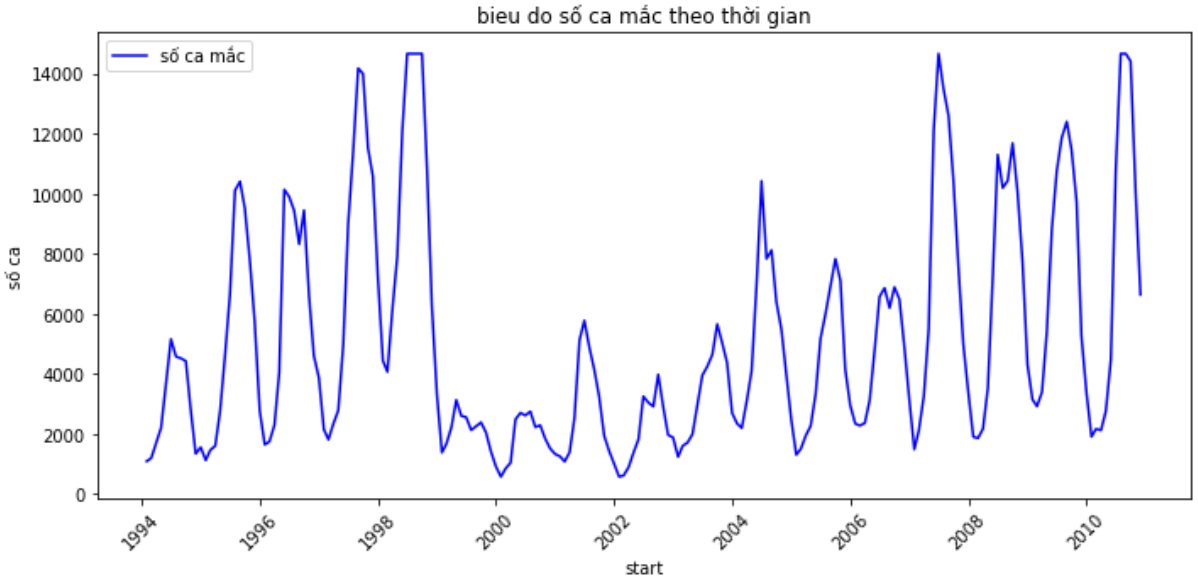
- thay thế các điểm ngoại lai bằng giá trị trung bình hoặc giá trị trung vị của dữ liệu.
- phương pháp xử lý ngoại lai bằng cách đưa các điểm ngoại lai về giới hạn dưới hoặc giới hạn trên
- phương pháp nội suy (interpolation) để ước lượng giá trị của các điểm ngoại lai dựa trên các điểm lân cận

Dựa vào số lượng điểm ngoại lai không quá lớn, tuy nhiên sự biến đổi lớn và dữ liệu theo thời gian thực, em sẽ dụng 2 phương pháp Tukey và nội suy để thay thế ngoại lai.

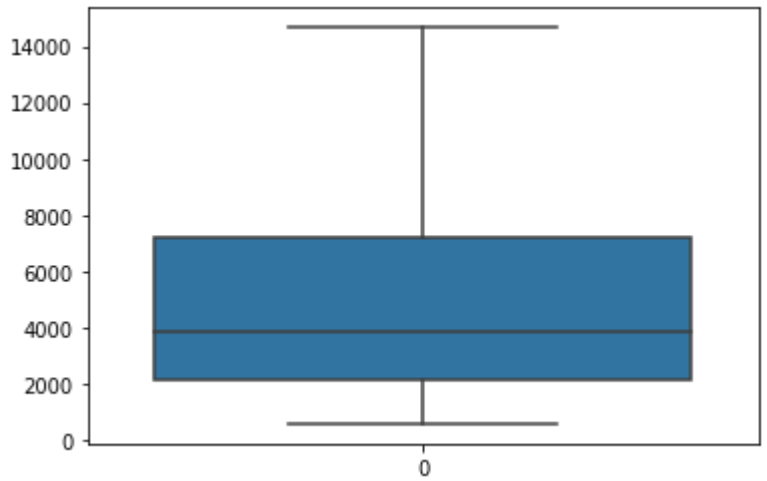
Xử lý ngoại lai đưa các điểm ngoại lai về giới hạn dưới hoặc giới hạn trên:



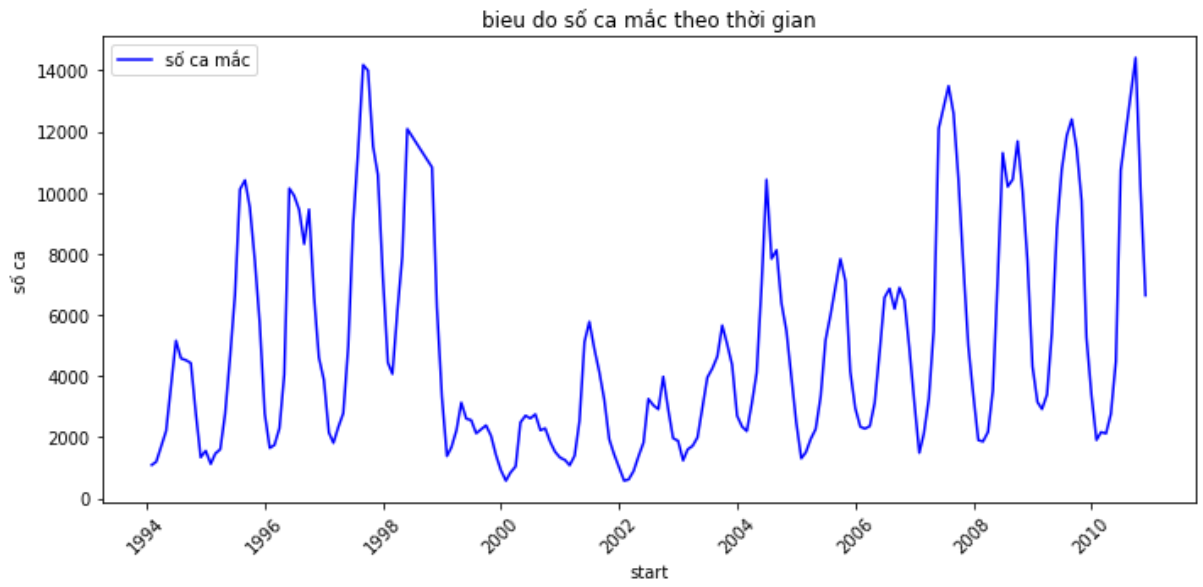
Biểu đồ theo thời gian



Xử lý ngoại lai bằng nội suy



Biểu đồ theo thời gian



Ta thấy từ 2 biểu đồ, phương pháp thay thế các điểm ngoại lai về giới hạn dưới hoặc giới hạn trên có điểm tương đồng lớn hơn và không làm mất mát nhiều thông tin, giá trị của điểm ngoại lai. Do đó, em sẽ thay thế điểm ngoại lai bằng cách này.

6.2.2: Huấn luyện mô hình

Đầu tiên, khai báo các thư viện tensorflow, keras để huấn luyện mô hình, chia dữ liệu thành 2 phần train và test với **160 ngày huấn luyện** và **47 ngày thử nghiệm**.

Tiếp theo, chuẩn hoá dữ liệu bằng phương pháp MinMax Scaler, biến đổi giá bán về một phạm vi 0 và 1, giữ nguyên khoảng biến thiên của dữ liệu gốc. Phương pháp này giúp đưa dữ liệu về cùng 1 tỉ lệ, giúp mô hình học tốt hơn. Công thức tính toán khi chuẩn hoá:

x' = a + (x - min) * (b - a) / (max - min)
trong đó x' là giá trị đã được chuẩn hóa của x, a và b là giá trị giới hạn của phạm vi chuẩn hóa (ví dụ: a=0, b=1 cho khoảng [0, 1]), min và max là giá trị tương ứng nhỏ nhất và lớn nhất trong dữ liệu ban đầu.

```
sc = MinMaxScaler(feature_range=(0,1))
sc_train = sc.fit_transform(data)
```

Sau khi chuẩn hoá dữ liệu, em sẽ tạo dữ liệu huấn luyện x_train là dữ liệu đầu vào và y_train là dữ liệu đầu ra, lấy 20 ngày đầu để huấn luyện cho ngày tiếp

theo. Xếp dữ liệu x_train và y_train thành mảng 1 chiều để đưa dữ liệu vào mô hình.

Tiếp theo, xây dựng mô hình. Mô hình huấn luyện bao gồm 5 lớp như đã trình bày :nhận dữ liệu đầu vào thông qua lớp input, sau đó sử dụng hai lớp LSTM để hiểu thông tin trong chuỗi dữ liệu theo thời gian. Lớp Dropout được sử dụng để tránh overfitting, và cuối cùng, mô hình dự đoán một giá trị duy nhất thông qua lớp đầu ra.

```
model = Sequential() // đầu vào
 model.add(LSTM(units=128,
input_shape=(x_train.shape[1],1),return_sequences=True)) // kích thước của mảng đầu vào
 model.add(LSTM(units=64)) // có 64 đơn vị LSTM
 model.add(Dropout(0.5)) //mỗi đơn vị đầu vào có xác suất 50% bị loại bỏ ngẫu nhiên
 model.add(Dense(1)) // dự đoán giá trị 1 số duy nhất
model.compile(loss='mean_absolute_error',optimizer='adam')
```

Tiếp theo, huấn luyện mô hình. Mô hình huấn luyện sau sẽ được lưu lại nếu sai số tuyệt đối trung bình của mô hình đó nhỏ hơn so với các lần huấn luyện trước, nếu lớn hơn mô hình sẽ tự động bỏ qua không lưu bằng hàm sai số và hàm tối ưu Adam.

Các tham số epochs=100, batch_size=50

```
best_model =
ModelCheckpoint(save_model,monitor='loss',verbose=2,save_best_only=True,
mode = 'auto')
 model.fit(x_train,y_train,epochs=100, batch_size=50, verbose=2,
callbacks=[best_model]) // huấn luyện 100 lần lặp
```

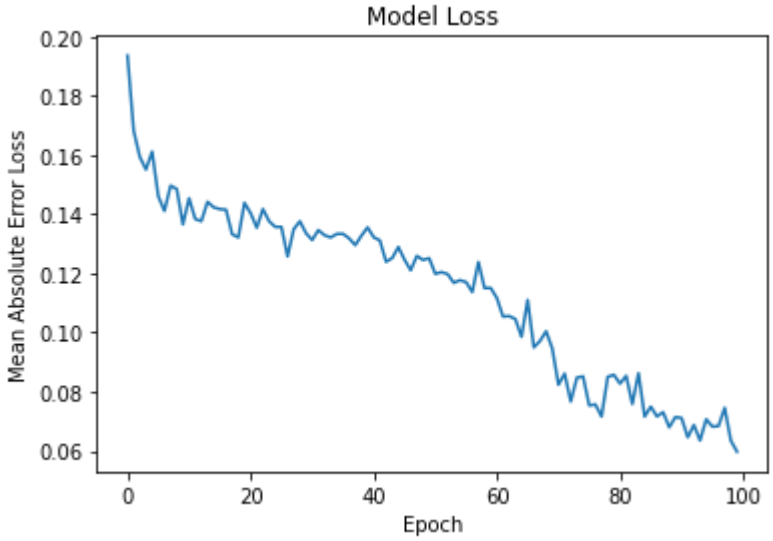
Tiến hành dự đoán cho dữ liệu x_train. Nhóm sẽ tải lại mô hình dự đoán vừa lưu và áp dụng vào dự đoán cho x_train với y_train là giá thực và y_predict là giá dự đoán. Tái sử dụng mô hình với tập test, với y_test là giá dự đoán và y_test_predict là giá dự đoán. Điều chỉnh lại dữ liệu về giá trị gốc do dữ liệu đang được chuẩn hoá bằng inverse_transform.

```
y_train = sc.inverse_transform(y_train)
final_model=load_model('Notebooks/data/save_model.hdf5') y_train_predict =
```

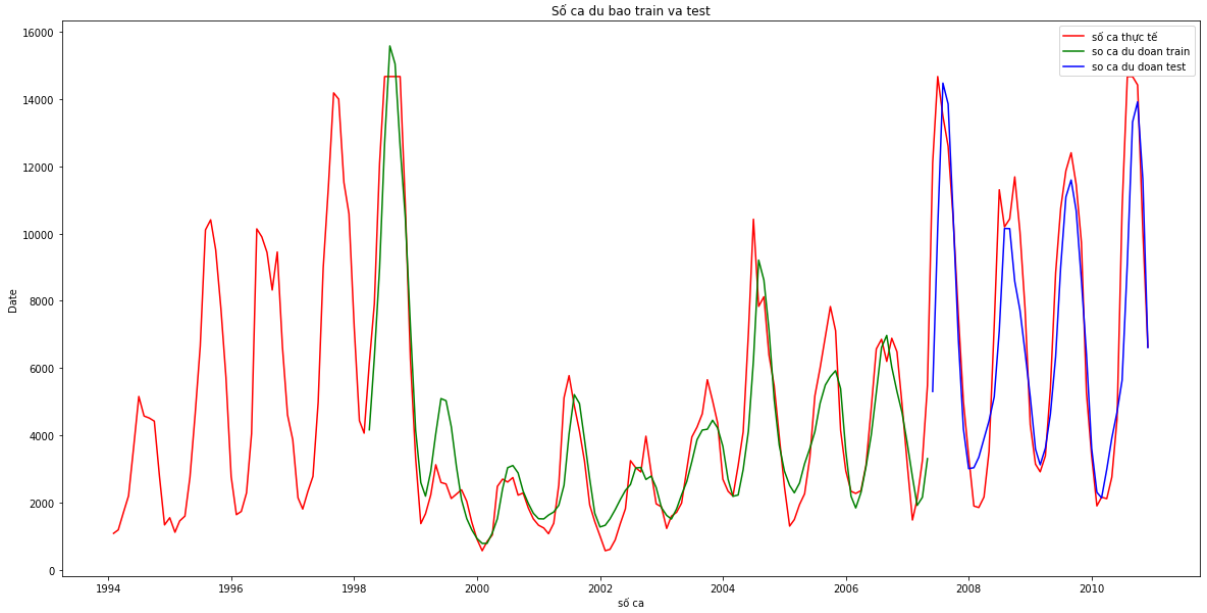
```
final_model.predict(x_train) y_train_predict=
sc.inverse_transform(y_train_predict)
```

7. Kết quả thực nghiệm

Biểu đồ của mô hình huấn luyện:



Đánh giá độ chính xác của mô hình. Em sẽ vẽ biểu đồ so sánh ở trên cả 2 tập train và test.



Biểu đồ dự đoán số ca mắc

Trong đó:

- đường màu đỏ là số ca mắc thực tế
- đường màu xanh lá cây là dữ liệu train trong 160 ngày, lấy 50 ngày đầu huấn luyện cho ngày tiếp theo
- đường màu xanh dương là dữ liệu dự đoán trong 48 ngày

Độ phù hợp tập train R2-score là 87.45%

Sai số tuyệt đối trung bình tập train: 808

Sai số bình phương trung bình tập train: 1205691

Độ phù hợp tập test là 75.03%

Sai số tuyệt đối trung bình tập train: 1452

Sai số bình phương trung bình tập train: 4536367

Bảng 1 dự đoán ở tập huấn luyện

Thời gian	Số ca mắc thực tế	Số ca mắc dự đoán
1998-04-01	6156.00	4171.454590
1998-05-01	7913.00	6322.066406
1998-06-01	12082.00	8965.229492
1998-07-01	14660.75	12685.840820
1998-08-01	14660.75	15569.900391
...	...	...
2007-04-01	3259.00	2166.000488
2007-05-01	5485.00	3313.917236

Bảng 2 dự đoán ở tập thử nghiệm

Thời gian	Số ca mắc thực tế	Số ca mắc dự đoán
2007-06-01	12099.00	5308.944824
2007-07-01	14660.75	10172.213867
2007-08-01	13487.00	14465.436523
2007-09-01	12588.00	13845.981445
…	…	…
2010-08-01	14660.75	9118.626953
2010-09-01	14660.75	13309.515625
2010-10-01	14411.00	13908.627930
2010-11-01	10070.00	11611.282227
2010-12-01	6639.00	6611.289062

8. Đánh giá mô hình

Mô hình dự đoán do có 208 dữ liệu nên việc dự đoán vẫn còn hạn chế về mức độ phù hợp ở cả tập huấn luyện và tập thử nghiệm. Tuy nhiên độ phù hợp khá ổn và các sai số không quá lớn. Việc thay thế ngoại lai bằng các điểm về giới hạn dưới hoặc giới hạn trên có độ chính xác cao hơn. Ngoài ra giá cả còn ảnh hưởng bởi nhiều yếu tố khác bên ngoài. Trong tương lai nhóm sẽ cải thiện mô hình để có độ phù hợp cao hơn và kết hợp các yếu tố ảnh hưởng để dự đoán chính xác hơn số ca mắc.

9. Kết luận

Khó khăn

Các dữ liệu được thu thập từ nhiều nguồn, số liệu thống kê số ca mắc không được công khai rõ ràng, chỉ thu thập được từ quá khứ rất xa, dữ liệu thu thập thiếu các trường dữ liệu cần thiết. Thêm vào đó dữ liệu thời tiết cần thu thập riêng biệt, sau đó phải kết nối các dữ liệu với nhau, có một số dữ liệu không ăn khớp (ví dụ độ ẩm và số ca mắc).

Tập dữ liệu huấn luyện mô hình dự đoán còn khá ít, do đó quá trình huấn luyện không được cao.

Kết quả thu được

Dự đoán được những năm số ca sốt xuất huyết tăng cao, và số ca mắc có chu kì để chúng ta có thể kiểm soát dịch bệnh dễ dàng hơn. Mô hình dự đoán với những số liệu dự đoán có thể chấp nhận được.

Trong tương lai, nhóm sẽ tìm kiếm thêm các trang web có thông tin về số ca mắc để bổ sung vào tập dữ liệu, xây dựng mô hình dự đoán chính xác hơn bằng việc sử dụng grid search, tối ưu các siêu tham số.

10. Phân công công việc

Nguyễn Khắc Thái Bình	Tiền xử lý dữ liệu, phân tích dữ liệu, huấn luyện mô hình, triển khai mô hình, hoàn thành báo cáo.
Nguyễn Sỹ Anh Khoa	Thu thập dữ liệu, hoàn thiện báo cáo.
Nguyễn Quốc Huy	Thu thập dữ liệu, hoàn thiện báo cáo, làm slide thuyết trình.
Trần Văn Hiếu	Thu thập dữ liệu,Tiền xử lý dữ liệu, hoàn thiện báo cáo.