

# DỰ BÁO MỨC ĐỘ Ô NHIỄM KHÔNG KHÍ TẠI BA TỈNH MIỀN BẮC VIỆT NAM: PHƯƠNG PHÁP HỌC MÁY VÀ HỌC SÂU

1<sup>st</sup> Cao Hoai Sang  
Information System  
name of organization (of Aff.)  
Ho Chi Minh City, Viet Nam  
21522541@gm.uit.edu.vn

2<sup>nd</sup> Nguyen Tran Gia Kiet  
dept. name of organization (of Aff.)  
name of organization (of Aff.)  
City, Country  
email address or ORCID

3<sup>rd</sup> Thi Thành Công  
dept. name of organization (of Aff.)  
name of organization (of Aff.)  
City, Country  
email address or ORCID

4<sup>th</sup> Nguyễn Hoàng Đăng Khoa  
dept. name of organization (of Aff.)  
name of organization (of Aff.)  
City, Country  
email address or ORCID

5<sup>th</sup> Given Name Surname  
dept. name of organization (of Aff.)  
name of organization (of Aff.)  
City, Country  
email address or ORCID

6<sup>th</sup> Given Name Surname  
dept. name of organization (of Aff.)  
name of organization (of Aff.)  
City, Country  
email address or ORCID

**Tóm tắt nội dung**—Mục tiêu chính của nghiên cứu này là dự đoán chất lượng không khí của ba tỉnh được chỉ định (Bắc Ninh, Hà Nội và Quảng Ninh) ở khu vực phía Bắc Việt Nam bằng cách kết hợp các thuật toán học máy và học sâu. Các mô hình bao gồm Gauss Newton Method Non-Linear, Residual Convolutional Neural Networks (ResCNN), Neural Hierarchical Interpolation for Time Series Forecasting (N-HITS), Dynamic Linear Model (DLM), Simple Exponential Smoothing (SES), Linear Regression (LR), Autoregressive Integrated Moving Average (ARIMA), Recurrent Neural Network (RNN), Gated Recurrent Unit (GRU), Long Short Term Memory (LSTM). Hiệu quả của tất cả các mô hình được đề cập trên được đo lường bằng Mean Absolute Percentage Error (MAPE), Root Mean Squared Error, nhằm đạt được độ chính xác tối đa trong dự báo chuỗi thời gian chất lượng không khí chính xác.

**Index Terms**—Nonlinear regression, Gauss-Newton, generalized least squares, iteratively reweighted least squares

## I. GIỚI THIỆU

Với sự gia tăng dân số nhanh chóng ở Việt Nam cùng với sự công nghiệp hóa ngày càng mạnh mẽ ở các khu vực trọng yếu, vấn đề chất lượng không khí nhanh chóng trở thành mối quan tâm chính, đặc biệt là ở các tỉnh đông dân cư ở khu vực phía Bắc Việt Nam. Sự suy giảm chất lượng không khí không chỉ gây ra những rủi ro lớn đối với sức khỏe của cư dân mà còn đe dọa sự cân bằng sinh thái mong manh của khu vực. Trong bối cảnh này, việc dự đoán chính xác và kịp thời các mức độ chất lượng không khí là rất cần thiết cho các chiến lược giảm thiểu hiệu quả và can thiệp y tế công cộng. Nghiên cứu của nhóm chúng tôi sử dụng sự kết hợp của các mô hình học máy và học sâu để dự đoán chất lượng không khí, bằng cách tận dụng sức mạnh của các thuật toán phức tạp như Gauss Newton Method Non-Linear, Resilient Convolutional Neural Networks

(ResCNN), Neural Hierarchical Interpolation for Time Series Forecasting (N-HITS), Dynamic Linear Model (DLM), Simple Exponential Smoothing (SES), Linear Regression (LR), Autoregressive Integrated Moving Average (ARIMA), Recurrent Neural Network (RNN), Gated Recurrent Unit (GRU), và Long Short Term Memory (LSTM).

Việc bao gồm một tập hợp đa dạng các mô hình cho phép chúng tôi khám phá các khía cạnh khác nhau của dữ liệu và so sánh hiệu suất của các phương pháp khác nhau. Chẳng hạn, các mô hình thống kê truyền thống như SES, LR, và ARIMA cung cấp tính giải thích và sự đơn giản, làm cho chúng trở thành các mô hình cơ sở có giá trị để so sánh. Mặt khác, các kiến trúc học sâu như RNN, GRU, và LSTM xuất sắc trong việc nắm bắt các phụ thuộc phi tuyến tính và phụ thuộc thời gian dài hạn, những đặc điểm thường có trong dữ liệu chuỗi thời gian chất lượng không khí. Ngoài ra, các phương pháp sáng tạo như ResCNN và N-HITS mang lại những lợi thế độc đáo trong việc xử lý các cấu trúc không gian và thứ bậc trong dữ liệu, nâng cao độ chính xác của các dự đoán.

## II. NGHIÊN CỨU LIÊN QUAN

### A. Gauss-Newton nonlinear method

Vào năm 2015, ứng dụng dự báo phục hồi sau xuất viện của Phạm Thị Hương được sử dụng trong luận văn thạc sĩ khoa học [1]. Sử dụng phương pháp Gauss-Newton phi tuyến để ước lượng giá trị nhỏ nhất của bình phương sai số. [1]

## III. TÀI NGUYÊN

### A. Bộ dữ liệu

Dự báo chất lượng không khí là rất quan trọng để giảm thiểu các tác động tiêu cực của ô nhiễm lên sức khỏe con người và môi trường. Phân tích chuỗi thời gian nổi lên như một công cụ

Identify applicable funding agency here. If none, delete this.

mạnh mẽ trong lĩnh vực này, cho phép dự đoán các mức độ chất lượng không khí trong tương lai dựa trên dữ liệu lịch sử.

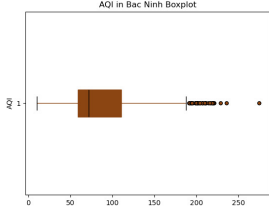
Do các vấn đề phổ biến liên quan đến bụi mịn ở khu vực phía Bắc, nhóm đã chọn một bộ dữ liệu chi tiết về chất lượng không khí của ba thành phố lớn ở miền Bắc Việt Nam: Bắc Ninh, Hà Nội và Quảng Ninh.

Bộ dữ liệu chủ yếu kéo dài từ năm 2021 đến năm hiện tại, 2024, bao gồm 6 cột tương ứng với các thành phần khác nhau trong không khí và chất lượng không khí được đánh giá thông qua Chỉ số Chất lượng Không khí (AQI): nồng độ PM2.5, nồng độ PM10, nồng độ O3, nồng độ NO2, nồng độ SO2 và nồng độ CO.

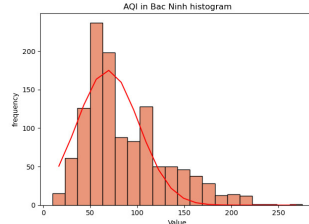
## B. Thống kê mô tả

Bảng I  
HA NOI, BAC NINH, QUANG NINH'S DESCRIPTIVE STATISTICS

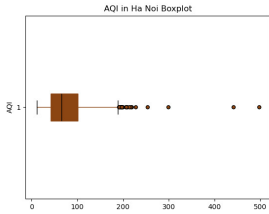
	HaNoi	BacNinh	QuangNinh
Count	2779	1190	929
Mean	75.283	86.87	32.117
Std	42.635	43.484	25.576
Min	11	10.0	5.0
25%	42.0	59.0	19.0
50%	66.0	72.0	25.0
75%	101.0	111.0	38.0
Max	498	275.0	500.0



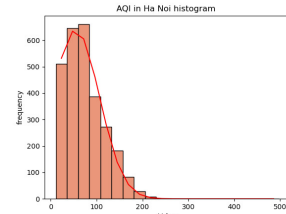
Hình 1. BacNinh AQI's boxplot



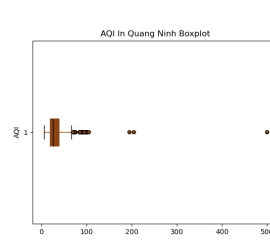
Hình 2. BacNinh AQI's histogram



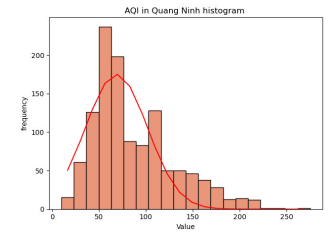
Hình 3. HaNoi AQI's boxplot



Hình 4. HaNoi AQI's histogram



Hình 5. QuangNinh AQI's boxplot



Hình 6. QuangNinh AQI's histogram

## IV. PHƯƠNG PHÁP LUẬN

### A. Gauss newton method nonlinear

1) *Least Squares*: Khoảng cách giữa một đường cong được khớp và một quan sát được gọi là sai số dư, hoặc lỗi.

$$Residuals = y_i - \hat{y}_i$$

Tổng của các sai số bình phương được tính bằng phương trình sau:

$$SSE = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Trong đó:

- $y_i$  là các giá trị quan sát được
- $\hat{y}_i$  là các giá trị khớp

2) *Phương pháp Newton*: Với hàm  $y = y_0 e^{-kt}$ , chúng ta tìm giá trị nhỏ nhất của SSE. Chúng ta tìm giá trị  $k$  bằng phương pháp Newton.

$$SSE = \sum_i^n (y_i - y_0 e^{-kt_i})^2$$

$$k_{\text{new}} = k_{\text{old}} - \frac{f'(k_{\text{old}})}{f''(k_{\text{old}})}$$

Hoặc chúng ta có thể giải thích bằng ma trận Hessian như sau:

$$\begin{pmatrix} k_{\text{new}} \\ y_{0,\text{new}} \end{pmatrix} = \begin{pmatrix} k_{\text{old}} \\ y_{0,\text{old}} \end{pmatrix} - H^{-1}G$$

Trong đó:

$$\bullet H = \begin{bmatrix} \frac{\partial^2 f}{\partial k^2} & \frac{\partial^2 f}{\partial k \partial y_0} \\ \frac{\partial^2 f}{\partial y_0 \partial k} & \frac{\partial^2 f}{\partial y_0^2} \end{bmatrix}$$

$$\bullet G = \begin{bmatrix} \frac{\partial f}{\partial k} \\ \frac{\partial f}{\partial y_0} \end{bmatrix}$$

Vấn đề với phương pháp Newton trong hồi quy phi tuyến là việc tính toán ma trận Hessian và nghịch đảo của nó gặp khó khăn. Để giải quyết vấn đề này, phương pháp Gauss-Newton thay thế bằng cách xấp xỉ ma trận Hessian.

Chúng ta có thể viết lại SSE như sau:

$$SSE = \sum_i^n r_i^2 = r^T r$$

Trong đó:

- $r$  là một vector chứa các sai số dư

3) *Phương pháp Gauss-Newton*: Chúng ta lấy đạo hàm của SSE theo các tham số trong mô hình bằng quy tắc chuỗi, chúng ta có được phương trình sau:

$$\frac{\partial SSE}{\partial \beta_j} = 2 \sum_i^n r_i \frac{\partial r_i}{\partial \beta_j}$$

Sau đó, bỏ số hai vì nó sẽ không ảnh hưởng đến việc ước tính các tham số. Tương ứng với ma trận Jacobian.

$$J_r = \begin{bmatrix} \frac{\partial r_1}{\partial \beta_1} & \frac{\partial r_1}{\partial \beta_2} \\ \frac{\partial r_2}{\partial \beta_1} & \frac{\partial r_2}{\partial \beta_2} \\ \vdots & \vdots \\ \frac{\partial r_n}{\partial \beta_1} & \frac{\partial r_n}{\partial \beta_2} \end{bmatrix}$$

Với phương trình sau cho tổng các sai số bình phương (SSE):

$$SSE = \sum_{i=1}^n (y_i - y_0 e^{-kt_i})^2$$

Chúng ta có:

$$\frac{\partial^2 SSE}{\partial \beta_j \partial \beta_k} = \sum_i^n \left( \frac{\partial r_i}{\partial \beta_j} \frac{\partial r_i}{\partial \beta_k} + r_i \frac{\partial^2 r_i}{\partial \beta_j \partial \beta_k} \right)$$

Sự khác biệt chính giữa phương pháp Newton và Gauss-Newton là phương pháp Gauss-Newton bỏ qua  $r_i \frac{\partial^2 r_i}{\partial \beta_j \partial \beta_k}$ . Do đó, đạo hàm bậc hai được xấp xỉ bằng hàm sau:

$$\frac{\partial^2 SSE}{\partial \beta_j \partial \beta_k} \approx \sum_i^n \left( \frac{\partial r_i}{\partial \beta_j} \frac{\partial r_i}{\partial \beta_k} \right) = J_r^T J_r$$

Sử dụng quy tắc cập nhật sau trong phương pháp Newton. Đối với Gauss-Newton, đơn giản chỉ cần cắm vào xấp xỉ cho ma trận Hessian và gradient.

$$\begin{pmatrix} k_{\text{new}} \\ y_{0,\text{new}} \end{pmatrix} = \begin{pmatrix} k_{\text{old}} \\ y_{0,\text{old}} \end{pmatrix} - (J_r^T J_r)^{-1} J_r^T r$$

Với  $\beta$  là một vector cột với các tham số được ước tính. Đối với ví dụ đơn giản chỉ ước tính hai tham số, phương trình trông như sau:

$$\beta_{\text{new}} = \beta_{\text{old}} - (J_r^T J_r)^{-1} J_r^T r(\beta_{\text{old}})$$

$$\begin{pmatrix} k_{\text{new}} \\ y_{0,\text{new}} \end{pmatrix} = \begin{pmatrix} k_{\text{old}} \\ y_{0,\text{old}} \end{pmatrix} - (J_r^T J_r)^{-1} J_r^T r \begin{pmatrix} k_{\text{old}} \\ y_{0,\text{old}} \end{pmatrix}$$

## B. Gated Recurrent Unit (GRU)

1) *Đơn vị tuần hoàn trong mạng nơ-ron*: Trước khi chúng ta đi vào chi tiết về GRU, hãy xem xét một đơn vị tuần hoàn thông thường trong mạng nơ-ron. Đơn vị này được sử dụng để xử lý dữ liệu tuần tự, chẳng hạn như chuỗi văn bản, âm thanh hoặc dữ liệu thời gian.

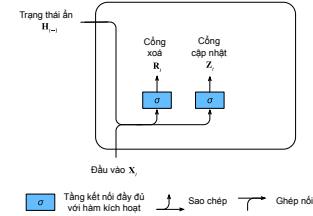
Mỗi đơn vị tuần hoàn có một trạng thái ẩn (hidden state) và nhận đầu vào từ bước thời gian trước đó. Nó sử dụng thông tin từ trạng thái ẩn và đầu vào hiện tại để tính toán trạng thái ẩn mới.

2) *GRU là gì?*: GRU là một biến thể của đơn vị tuần hoàn, được thiết kế để giải quyết một số vấn đề của kiến trúc LSTM (Long Short-Term Memory).

Sự khác biệt chính giữa GRU và LSTM nằm ở cách xử lý trạng thái ẩn và ô nhớ:

- **LSTM**: Sử dụng ba cổng (input gate, output gate và forget gate) để duy trì trạng thái của ô nhớ riêng biệt.
- **GRU**: Không có ô nhớ riêng biệt, thay vào đó, nó sử dụng một “vector kích hoạt ứng viên” và hai cổng (reset gate và update gate).
- **Cổng đặt lại (reset gate)**: Quyết định bao nhiêu trạng thái ẩn trước đó cần quên.
- **Cổng cập nhật (update gate)**: Quyết định bao nhiêu của vector kích hoạt ứng viên cần tích hợp vào trạng thái ẩn mới.

3) *Cổng xóa và cổng cập nhật*: Đầu tiên ta giới thiệu cổng xóa và cổng cập nhật. Ta thiết kế chúng thành các vector có các phần tử trong khoảng (0,1) để có thể biểu diễn các tổ hợp lỗi. Chẳng hạn, một biến xóa cho phép kiểm soát bao nhiêu phần của trạng thái trước đây được giữ lại. Tương tự, một biến cập nhật cho phép kiểm soát bao nhiêu phần của trạng thái mới sẽ giống trạng thái cũ.



Hình 7. Cổng xóa và cổng cập nhật trong GRU

Ta bắt đầu bằng việc thiết kế các cổng tạo ra các biến này. Hình 7 minh họa các đầu vào cho cả cổng xóa và cổng cập nhật trong GRU, với đầu vào ở bước thời gian hiện tại  $\mathbf{X}_t$  và trạng thái ẩn ở bước thời gian trước đó  $\mathbf{H}_{t-1}$ . Đầu ra được tạo bởi một tầng kết nối đầy đủ với hàm kích hoạt sigmoid.

Tại bước thời gian  $t$ , với đầu vào minibatch là  $\mathbf{X}_t \in \mathbb{R}^{n \times d}$  (số lượng mẫu:  $n$ , số lượng đầu vào:  $d$ ) và trạng thái ẩn ở bước thời gian gần nhất là  $\mathbf{H}_{t-1} \in \mathbb{R}^{n \times h}$  (số lượng trạng thái ẩn:  $h$ ), cổng xóa  $\mathbf{R}_t \in \mathbb{R}^{n \times h}$  và cổng cập nhật  $\mathbf{Z}_t \in \mathbb{R}^{n \times h}$  được tính như sau:

$$\mathbf{R}_t = \sigma(\mathbf{X}_t \mathbf{W}_{xr} + \mathbf{H}_{t-1} \mathbf{W}_{hr} + \mathbf{b}_r)$$

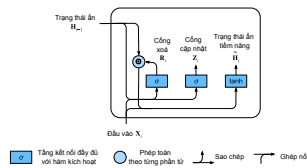
$$\mathbf{Z}_t = \sigma(\mathbf{X}_t \mathbf{W}_{xz} + \mathbf{H}_{t-1} \mathbf{W}_{hz} + \mathbf{b}_z).$$

Ở đây,  $\mathbf{W}_{xr}, \mathbf{W}_{xz} \in \mathbb{R}^{d \times h}$  và  $\mathbf{W}_{hr}, \mathbf{W}_{hz} \in \mathbb{R}^{h \times h}$  là các tham số trọng số và  $\mathbf{b}_r, \mathbf{b}_z \in \mathbb{R}^{1 \times h}$  là các hệ số điều chỉnh. Ta sẽ sử dụng hàm sigmoid để biến đổi các giá trị đầu vào nằm trong khoảng (0, 1).

4) *Hoạt động của cổng xóa*: Trong RNN thông thường, ta cập nhật trạng thái cổng xóa theo công thức

$$\mathbf{H}_t = \tanh(\mathbf{X}_t \mathbf{W}_{xh} + \mathbf{H}_{t-1} \mathbf{W}_{hh} + \mathbf{b}_h).$$

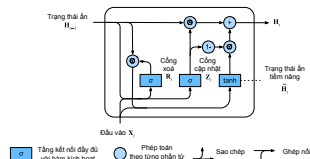
Hình 8 minh họa luồng tính toán sau khi áp dụng cổng xóa. Ký hiệu  $\odot$  biểu thị phép nhân theo từng phần tử giữa các tensor.



Hình 8. Tính toán trạng thái tiềm năng trong một GRU. Phép nhân được thực hiện theo phần tử

5) *Hoạt động của cổng cập nhật*: Cổng này xác định mức độ giống nhau giữa trạng thái mới  $\mathbf{H}_t$  và trạng thái cũ  $\mathbf{H}_{t-1}$  cũng như mức độ trạng thái tiềm năng  $\tilde{\mathbf{H}}_t$  được sử dụng. Biến cổng (gating variable)  $\mathbf{Z}_t$  được sử dụng cho mục đích này bằng cách áp dụng tổ hợp lỗi giữa trạng thái cũ và trạng thái tiềm năng. Ta có phương trình cập nhật cuối cho GRU.

$$\mathbf{H}_t = \mathbf{Z}_t \circ \mathbf{H}_{t-1} + (1 - \mathbf{Z}_t) \circ \tilde{\mathbf{H}}_t.$$



Hình 9. Tính toán trạng thái ẩn trong một GRU. Phép nhân được thực hiện cho từng phần tử

Nếu các giá trị trong cổng cập nhật  $\mathbf{Z}_t$  bằng 1, chúng ta chỉ đơn giản giữ lại trạng thái cũ. Trong trường hợp này, thông tin từ  $\mathbf{X}_t$  về cơ bản được bỏ qua, tương đương với việc bỏ qua bước thời gian  $t$  trong chuỗi phụ thuộc. Ngược lại, nếu  $\mathbf{Z}_t$  gần giá trị 0, trạng thái ẩn  $\mathbf{H}_t$  sẽ gần với trạng thái ẩn tiềm năng  $\mathbf{H}_t$ . Những thiết kế trên có thể giúp chúng ta giải quyết vấn đề tiêu biến gradient trong các mạng RNN và nắm bắt tốt hơn sự phụ thuộc xa trong chuỗi thời gian. Tóm lại, các mạng GRU có hai tính chất nổi bật sau:

- Cổng xóa giúp nắm bắt các phụ thuộc ngắn hạn trong chuỗi thời gian.
- Cổng cập nhật giúp nắm bắt các phụ thuộc dài hạn trong chuỗi thời gian.

## TÀI LIỆU

- [1] P. T. Huong, "Linear regression, polynomial regression, and applications, master's thesis in science,"2015.