TRƯỜNG ĐẠI HỌC PHENIKAA

KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO BÀI TẬP LỚN

HỌC PHẦN: TÍCH HỢP VÀ PHÂN TÍCH DỮ LIỆU LỚN

Nhóm 6

Đề tài: Phân tích và dự đoán chất lượng không khí

Thành viên nhóm

Vũ Quốc Việt	22010256	K16 CNTT1
Phạm Thị Hạnh	22010503	K16 CNTT1
Lê Thế Trân	22010480	K16 CNTT1

GVHD: TS. Đặng Thị Thuý An

02/2025 – Hà Nội

Mục lục

1. Giới thiệu

1.1. Giới thiệu môn học Tích Hợp và Phân Tích Dữ Liệu Lớn

Trong thời đại công nghệ số, dữ liệu đã trở thành một nguồn tài nguyên vô cùng quý giá, đóng vai trò then chốt trong việc ra quyết định và phát triển chiến lược của các tổ chức, doanh nghiệp. Với sự bùng nổ của dữ liệu từ nhiều nguồn khác nhau như mạng xã hội, cảm biến, giao dịch trực tuyến, và các hệ thống IoT, khối lượng dữ liệu được tạo ra hàng ngày đã đạt đến mức độ "lớn" (Big Data). Điều này đặt ra yêu cầu cấp thiết về việc tích hợp và phân tích dữ liệu một cách hiệu quả để khai thác giá trị tiềm ẩn từ chúng.

Môn học Tích Hợp và Phân Tích Dữ Liệu Lớn được thiết kế để cung cấp cho người học những kiến thức và kỹ năng cần thiết để làm việc với các hệ thống dữ liệu lớn, từ việc thu thập, tích hợp, xử lý, đến phân tích và trực quan hóa dữ liệu. Môn học này không chỉ tập trung vào các công nghệ và công cụ hiện đại mà còn nhấn mạnh vào các phương pháp tiếp cận khoa học để giải quyết các bài toán thực tế trong lĩnh vực phân tích dữ liệu.

1.2. Bối cảnh chọn vấn đề

Trong những năm gần đây, vấn đề ô nhiễm không khí tại Việt Nam, đặc biệt là tại các thành phố lớn như Hà Nội và Thành phố Hồ Chí Minh, đã trở thành một thách thức nghiêm trọng đối với sức khỏe cộng đồng và môi trường. Theo báo cáo từ IQAir, một tổ chức quốc tế chuyên theo dõi chất lượng không khí, Hà Nội đã nhiều lần được xếp hạng là thành phố ô nhiễm nhất thế giới. Cụ thể, vào đầu tháng 1 năm 2023, chỉ số chất lượng không khí (AQI - Air Quality Index) tại Hà Nội đã đạt mức báo động:

- Ngày 7/1/2025: AQI đạt 272 - mức tím (rất có hại cho sức khỏe).
- Ngày 8/1/2025: AQI đạt 219 - mức tím (rất có hại cho sức khỏe).

Những con số này cho thấy mức độ ô nhiễm không khí đã vượt xa ngưỡng an toàn theo tiêu chuẩn của Tổ chức Y tế Thế giới (WHO). Ô nhiễm không khí không chỉ ảnh hưởng đến môi trường mà còn đe dọa trực tiếp đến sức khỏe con người, gây ra các bệnh về hô hấp, tim mạch, và thậm chí là ung thư.

Trong bối cảnh đó, việc phân tích và dự đoán chất lượng không khí trở thành một nhu cầu cấp thiết. Đề tài này không chỉ giúp hiểu rõ hơn về tình trạng ô nhiễm mà còn hỗ trợ các cơ quan chức năng và người dân trong việc đưa ra các biện pháp phòng ngừa và

ứng phó kịp thời. Tuy nhiên, để giải quyết bài toán này, chúng ta cần ứng dụng các công nghệ và phương pháp trong lĩnh vực Big Data, bởi vì:

- Dữ liệu lớn và đa dạng:

  Dữ liệu về chất lượng không khí đến từ nhiều nguồn khác nhau như cảm biến, vệ tinh, dữ liệu thời tiết, và các nguồn dữ liệu mở. Việc tích hợp và xử lý lượng dữ liệu khổng lồ này đòi hỏi các công cụ và kỹ thuật Big Data.

- Phân tích thời gian thực:

  Để đưa ra các cảnh báo kịp thời, hệ thống cần có khả năng xử lý và phân tích dữ liệu trong thời gian thực.

- Dự đoán xu hướng:

  Sử dụng các mô hình machine learning và AI để dự đoán chất lượng không khí trong tương lai dựa trên dữ liệu lịch sử và các yếu tố ảnh hưởng như thời tiết, giao thông, và hoạt động công nghiệp.

- Giải quyết các vấn đề phức tạp:

  Big Data giúp giải quyết các bài toán phức tạp như xác định nguồn gốc ô nhiễm, đánh giá tác động của các yếu tố môi trường, và đề xuất các giải pháp tối ưu.

1.3. Mục tiêu nghiên cứu

1.3.1. Phân tích dữ liệu chất lượng không khí

- Thu thập và tích hợp dữ liệu từ nhiều nguồn khác nhau (cảm biến, vệ tinh, dữ liệu thời tiết, v.v.).
- Phân tích xu hướng và diễn biến chất lượng không khí theo thời gian và không gian
- Xác định các yếu tố chính ảnh hưởng đến chất lượng không khí (giao thông, công nghiệp, thời tiết, v.v.).

1.3.2. Xây dựng mô hình dự đoán

- Phát triển các mô hình machine learning và AI để dự đoán chất lượng không khí trong tương lai.
- Đánh giá hiệu quả của các mô hình dự đoán dựa trên độ chính xác và khả năng ứng dụng thực tế.

1.3.3. Trực quan hóa dữ liệu

- Thiết kế các công cụ trực quan hóa dữ liệu để hiển thị thông tin chất lượng không khí một cách trực quan và dễ hiểu.
- Cung cấp các báo cáo và cảnh báo kịp thời đến người dân và cơ quan chức năng.

1.3.4. Đề xuất giải pháp

- Dựa trên kết quả phân tích và dự đoán, đề xuất các giải pháp cụ thể để cải thiện chất lượng không khí.
- Gợi ý các chính sách và biện pháp quản lý phù hợp để giảm thiểu ô nhiễm không khí.

1.3.5. Ứng dụng thực tiễn

- Áp dụng các kết quả nghiên cứu vào thực tế, hỗ trợ các cơ quan quản lý môi trường và người dân trong việc theo dõi và ứng phó với ô nhiễm không khí.
- Góp phần nâng cao nhận thức cộng đồng về vấn đề ô nhiễm không khí và các biện pháp phòng ngừa.

1.4. Phạm vi nghiên cứu

1.4.1. Phạm vi về dữ liệu

Nguồn dữ liệu:

- Dữ liệu chất lượng không khí (AQI, PM2.5, PM10, NO2, SO2, CO, O3) từ các trạm quan trắc môi trường, cảm biến, và các nguồn dữ liệu mở (IQAir, AirVisual, EPA).
- Dữ liệu thời tiết (nhiệt độ, độ ẩm, tốc độ gió, lượng mưa) từ các cơ quan khí tượng thủy văn.
- Dữ liệu giao thông (lưu lượng phương tiện, ùn tắc giao thông).
- Dữ liệu hoạt động công nghiệp (khí thải từ nhà máy, khu công nghiệp).

Thời gian dữ liệu:

- Dữ liệu lịch sử trong vòng 5-10 năm để phân tích xu hướng.
- Dữ liệu thời gian thực để hỗ trợ dự đoán và cảnh báo.

Khu vực địa lý:

- Tập trung vào các thành phố lớn trên thế giới.

1.4.2. Phạm vi về phương pháp

Phương pháp thu thập và tích hợp dữ liệu:

- Sử dụng công cụ ETL (Extract, Transform, Load) để thu thập dữ liệu từ nhiều nguồn.
- Áp dụng các kỹ thuật làm sạch và chuẩn hóa dữ liệu để đảm bảo tính nhất quán.

Phương pháp phân tích dữ liệu:

- Phân tích thống kê để xác định xu hướng và mối tương quan giữa các yếu tố ảnh hưởng.
- Sử dụng machine learning (hồi quy, cây quyết định, mạng nơ-ron, v.v.) để xây dựng mô hình dự đoán.

Phương pháp trực quan hóa dữ liệu:

- Sử dụng Tableau, Power BI hoặc Python libraries (Matplotlib, Seaborn) để trực quan hóa dữ liệu.
- Thiết kế bảng điều khiển (dashboard) hiển thị thông tin chất lượng không khí và cảnh báo.

1.4.3. Phạm vi về công nghệ

Công cụ và nền tảng Big Data:

- Sử dụng Hadoop, Spark để xử lý và phân tích dữ liệu.
- Dùng Python, R để phát triển mô hình machine learning.
- Quản lý cơ sở dữ liệu bằng SQL, NoSQL để lưu trữ và truy vấn dữ liệu.

1.4.4. Phạm vi về đối tượng nghiên cứu

Đối tượng chính:

- Chất lượng không khí (AQI và các thông số ô nhiễm).
- Các yếu tố ảnh hưởng đến chất lượng không khí (thời tiết, giao thông, công nghiệp).

Đối tượng phụ:

- Tác động của ô nhiễm không khí đến sức khỏe con người và môi trường.
- Hiệu quả của các biện pháp giảm thiểu ô nhiễm không khí.

2. Kiến thức nền tảng

2.1 Lý thuyết liên quan

2.1.1 Khái niệm quan trọng về đề tài

➢ **Chất lượng không khí**

Chất lượng không khí phản ánh mức độ sạch của không khí và ảnh hưởng của nó đến sức khỏe con người cũng như môi trường. Chỉ số chất lượng không khí (AQI - Air Quality Index) là một chỉ số phổ biến dùng để đánh giá mức độ ô nhiễm không khí dựa trên nồng độ của các chất gây ô nhiễm như PM2.5, NO2, CO, SO2, và O3.

➢ **Chỉ số chất lượng không khí**

0 – 50	Tốt
51 -100	Trung bình
101 – 150	Kém
151 – 200	Xấu
201 – 300	Rất xấu
301+	Nguy hại

➢ **Các thông số ô nhiễm không khí**

PM2.5	Hạt bụi mịn có đường kính nhỏ hơn 2.5 micromet, gây hại đến hệ hô hấp.
NO, NO2, NOx	Các oxit nitơ có nguồn gốc từ giao thông và công nghiệp, gây viêm phổi.

SO2	Lưu huỳnh điôxít, sinh ra từ đốt nhiên liệu hóa thạch.
CO	Khí cacbon mônôxít, có thể gây ngộ độc nếu ở mức cao.
O3	Ôzôn tầng mặt đất, có thể gây kích ứng đường hô hấp.

2.1.2 Tổng quan về các phương pháp và thuật toán được sử dụng

- **Tiền xử lý dữ liệu**

Dữ liệu được xử lý để đảm bảo tính chính xác và tối ưu cho mô hình dự đoán AQI. Các bước tiền xử lý bao gồm:

- Xử lý giá trị thiếu
    - Chuyển đổi cột Date sang kiểu datetime để dễ dàng xử lý dữ liệu theo thời gian.
    - Sử dụng phương pháp nội suy theo thời gian (time interpolation) để điền giá trị thiếu của các trạm quan trắc (StationId).
    - Loại bỏ cột Xylene vì chứa quá nhiều giá trị thiếu.
- Loại bỏ ngoại lệ (Outliers)
    - Áp dụng phương pháp IQR (Interquartile Range) để loại bỏ các giá trị bất thường.
    - Các giá trị nằm ngoài khoảng Q1 - 1.5 × IQR và Q3 + 1.5 × IQR sẽ bị loại bỏ.
- Lưu dữ liệu đã làm sạch
    - Dữ liệu sau khi xử lý được lưu vào file xxxxx_xxx_cleaned.csv, sẵn sàng cho quá trình huấn luyện mô hình.

- **Mô hình học máy (Machine Learning)**

Hệ thống sử dụng XGBoost Regression và ARIMA (Time Series Forecasting) để dự đoán AQI.

- XGBoost Regression
    - Đầu vào: Các thông số ô nhiễm như PM2.5, PM10, NO, NO2, NOx, CO, SO2, O3.
    - Mục tiêu: Dự đoán chỉ số AQI (AQI).
- Các bước triển khai
    - Chia dữ liệu:
        - Tập dữ liệu được chia thành 80% train - 20% test để huấn luyện mô hình.
    - Huấn luyện mô hình:
        - Sử dụng XGBoost Regressor với các tham số tối ưu:
            - n_estimators = 100
            - learning_rate = 0.1
            - max_depth = 5
    - Đánh giá mô hình:
        - RMSE (Root Mean Squared Error)
        - MAE (Mean Absolute Error)
        - R^2 Score
- Sau khi huấn luyện, mô hình được lưu vào file xgboost_aqi_model_xxxx_xxx.pkl để triển khai API.
- Dự báo theo chuỗi thời gian (ARIMA)
    - Dữ liệu được resample theo ngày để tạo chuỗi thời gian liên tục.
    - Sử dụng mô hình ARIMA (5,1,0) để dự báo AQI trong tương lai.
    - Hiển thị kết quả dự báo bằng biểu đồ.

➢ **Triển khai với Flask**

Hệ thống sử dụng Flask để cung cấp API giúp nhận dữ liệu và dự đoán AQI theo thời gian thực.

Các bước triển khai API

- Load mô hình XGBoost (xgboost_aqi_model_station_day.pkl)
- Nhận dữ liệu đầu vào từ request JSON
- Dự đoán chỉ số AQI
- Phản hồi kết quả, bao gồm:
    - Giá trị AQI dự đoán
    - Phân loại chất lượng không khí theo ngưỡng AQI (Tốt, Trung bình, Kém, v.v.)

API có thể được mở rộng để tích hợp trực quan hóa dữ liệu bằng Plotly hoặc Grafana, giúp theo dõi xu hướng AQI dễ dàng hơn.

2.2 Công cụ và công nghệ

- Flask: Framework web nhẹ giúp triển khai API dự đoán AQI.
- Scikit-learn: Cung cấp các thuật toán Machine Learning như Random Forest, XGBoost để huấn luyện mô hình dự đoán AQI.
- Pandas: Hỗ trợ xử lý và phân tích dữ liệu, đặc biệt là dữ liệu dạng bảng.
- Matplotlib: Dùng để trực quan hóa dữ liệu, giúp phân tích và đánh giá kết quả dự đoán.
- Pickle: Được sử dụng để lưu trữ và tải lại mô hình Machine Learning, giúp giảm thời gian huấn luyện lại từ đầu.

2.3 Nguồn dữ liệu

- Air Quality Data in India (2015 - 2020)
- Link : https://www.kaggle.com/datasets/rohanrao/air-quality-data-in-india/data

3. Phương pháp tiếp cận

3.1 Chiến lược phân tích dữ liệu

3.1.1 Phân tích dữ liệu thống kê

- Tính toán chỉ số trung bình, phương sai, độ lệch chuẩn của các chất ô nhiễm như PM2.5, NO2, CO, O3.
- Biểu đồ phân phối dữ liệu được vẽ bằng Matplotlib để xác định mức độ ô nhiễm theo từng thời điểm.
- So sánh AQI giữa các khu vực, phát hiện điểm nóng ô nhiễm.

3.1.2 Phân tích bằng Machine Learning

- Scikit-learn được sử dụng để triển khai các mô hình học máy nhằm dự đoán AQI:
    - Random Forest, XGBoost: Dự đoán AQI dựa trên dữ liệu lịch sử.
    - Feature Selection: Lựa chọn các yếu tố quan trọng ảnh hưởng đến AQI để tối ưu mô hình.
    - Gradient Boosting: Tăng cường độ chính xác bằng cách kết hợp nhiều mô hình nhỏ.

3.1.3 Phân tích dự báo bằng Deep Learning

- LSTM (Long Short-Term Memory) được triển khai để phân tích dữ liệu thời gian thực và dự đoán AQI trong tương lai.
- Thử nghiệm ARIMA & Prophet để so sánh độ chính xác với LSTM.
- Tích hợp Flask API để cung cấp dữ liệu dự báo cho hệ thống.

3.1.4 Trực quan hóa và báo cáo dữ liệu

- Matplotlib: Vẽ biểu đồ xu hướng AQI theo thời gian.
- Seaborn: Hiển thị bản đồ nhiệt để so sánh mức độ ô nhiễm giữa các khu vực.
- Flask Dashboard: Cung cấp giao diện hiển thị dữ liệu phân tích theo thời gian thực.

3.2 Lựa chọn mô hình

3.2.1 Lý do lựa chọn mô hình XGBoost Regressor

Mô hình/Phương pháp	Ứng dụng trong hệ thống	Lý do lựa chọn
XGBoost Regressor	Dự đoán chỉ số AQI dựa trên các thông số ô nhiễm không khí.	Mô hình mạnh mẽ, tối ưu hóa hiệu suất, xử lý tốt dữ liệu phi tuyến tính và có khả năng kiểm soát overfitting.
ARIMA (Time Series Forecasting)	Dự báo chỉ số AQI theo thời gian.	Phù hợp với bài toán chuỗi thời gian, giúp dự đoán xu hướng AQI trong tương lai.
Nội suy theo thời gian	Xử lý giá trị thiếu trong dữ liệu cảm biến từ trạm đo.	Bảo toàn xu hướng dữ liệu theo thời gian, tránh mất mát thông tin quan trọng.
IQR (Interquartile Range)	Loại bỏ giá trị ngoại lai trong dữ liệu.	Giúp mô hình không bị ảnh hưởng bởi các giá trị bất thường, tăng độ chính xác dự đoán.
train_test_split	Chia dữ liệu thành tập huấn luyện và kiểm tra.	Đánh giá mô hình trên dữ liệu chưa thấy trước, đảm bảo tổng quát hóa tốt.
Flask API	Cung cấp API dự đoán AQI theo thời gian thực.	Dễ triển khai, tích hợp với các hệ thống khác để cung cấp dự báo AQI trực tiếp.

3.3 Lý do chọn công cụ và nền tảng

Công cụ	Mục đích sử dụng
Pandas	Xử lý và phân tích dữ liệu dạng bảng, hỗ trợ đọc/ghi CSV, làm sạch và chuẩn hóa dữ liệu trước khi đưa vào mô hình.
XGBoost	Huấn luyện mô hình dự đoán AQI với XGBoost Regressor, giúp tối ưu hóa hiệu suất và tăng độ chính xác.
Statsmodels (ARIMA)	Dự báo chuỗi thời gian AQI, phân tích xu hướng biến động theo thời gian.
Scikit-learn	Hỗ trợ chia dữ liệu (train_test_split) và đánh giá mô hình bằng các chỉ số MSE, MAE, R^2.
Flask	Xây dựng API dự đoán AQI theo thời gian thực, cho phép tích hợp vào các hệ thống khác.
Joblib	Lưu trữ mô hình Machine Learning đã huấn luyện, giúp tối ưu hóa thời gian dự đoán mà không cần huấn luyện lại từ đầu.
Matplotlib	Trực quan hóa dữ liệu, đánh giá hiệu suất mô hình và xu hướng ô nhiễm không khí.

4. Kiến trúc hệ thống

4.1 Quy trình xử lý dữ liệu

Bước 1: Thu thập dữ liệu

- Dữ liệu được lấy từ các nguồn như OpenAQ, World Air Quality Index, Kaggle Air Pollution Dataset.
- Dữ liệu được lưu trữ dưới dạng CSV.

Bước 2: Tiền xử lý dữ liệu

Xử lý giá trị thiếu và trùng lặp

- Xóa cột Xylene do có quá nhiều giá trị bị thiếu.
- Nội suy dữ liệu theo thời gian cho từng trạm (StationId) để điền giá trị thiếu.
- Kiểm tra và loại bỏ các dòng có giá trị thiếu trong cột AQI.

Xử lý thời gian

- Chuyển đổi cột Date sang định dạng thời gian (pd.to_datetime).
- Đặt Date làm chỉ mục (set_index) để xử lý dữ liệu theo chuỗi thời gian.

Loại bỏ outliers

- Sử dụng phương pháp IQR để loại bỏ ngoại lệ trên các cột số (PM2.5, PM10, NO, NO2, NOx, CO, SO2, O3).

Bước 3: Chia tập dữ liệu

- Chia dữ liệu thành đặc trưng (features) và nhãn (target), với:
    - Đặc trưng (X): PM2.5, PM10, NO, NO2, NOx, CO, SO2, O3.
    - Nhãn (y): AQI.
- Sử dụng train_test_split để chia dữ liệu thành tập huấn luyện (80%) và tập kiểm tra (20%).

Bước 4: Huấn luyện mô hình Machine Learning

- Sử dụng XGBoost Regressor (XGBRegressor) để huấn luyện mô hình dự đoán AQI.
- Mô hình được huấn luyện trên tập huấn luyện (X_train, y_train).

Bước 5: Đánh giá mô hình

- Dự đoán chỉ số AQI trên tập kiểm tra (X_test).
- Đánh giá mô hình bằng các chỉ số:
    - RMSE (Root Mean Squared Error) – Sai số căn bậc hai.
    - MAE (Mean Absolute Error) – Sai số tuyệt đối trung bình.

- R^2 Score – Đánh giá độ phù hợp của mô hình.

Bước 6: Dự báo chuỗi thời gian (Time Series Forecasting)

- Sử dụng ARIMA để dự báo chỉ số AQI theo thời gian.
- Dữ liệu được resample theo ngày (resample('D').mean()) trước khi huấn luyện mô hình ARIMA.
- Huấn luyện mô hình ARIMA (ARIMA(order=(5,1,0))) trên tập dữ liệu lịch sử AQI.
- Dự báo và so sánh với dữ liệu thực tế bằng biểu đồ trực quan hóa (Matplotlib).

Bước 7: Triển khai API dự đoán AQI

- Flask được sử dụng để xây dựng API dự đoán AQI.
- API nhận dữ liệu đầu vào (PM2.5, PM10, NO, NO2, NOx, CO, SO2, O3) từ request.
- Dữ liệu mới được tiền xử lý giống tập huấn luyện trước khi đưa vào mô hình.
- Mô hình XGBoost Regressor dự đoán AQI từ dữ liệu đầu vào và phân loại mức độ ô nhiễm.
- API trả về giá trị AQI dự đoán và thông tin chất lượng không khí.

Bước 8: Lưu trữ mô hình và công cụ xử lý dữ liệu

- Mô hình XGBoost được lưu bằng Joblib (joblib.dump) để sử dụng lại mà không cần huấn luyện lại.
- Hệ thống API có thể tải mô hình từ tệp (joblib.load) để sử dụng trong dự đoán thời gian thực.

4.2 Tổng quan kiến trúc chương trình

Hệ thống phân tích và dự đoán chỉ số chất lượng không khí (AQI) dựa trên dữ liệu từ cảm biến môi trường. Hệ thống hoạt động theo quy trình sau:

4.2.1 Thu thập dữ liệu

- Dữ liệu được đọc từ tệp CSV.
- Các thông tin gồm: thời gian đo, vị trí, và các thông số ô nhiễm (PM2.5, NO, NO2, NOx, NH3, CO, SO2, O3, v.v.).

4.2.2 Xử lý dữ liệu

- Xử lý giá trị thiếu:
  - Xóa các cột có quá nhiều giá trị thiếu (PM10, Xylene, Benzene, Toluene).
  - Điền giá trị thiếu bằng giá trị trung bình của từng cột cho các thông số ô nhiễm.
  - Xóa các dòng có giá trị thiếu trong cột AQI.
- Xử lý thời gian:
  - Chuyển đổi cột Datetime thành dạng datetime.
  - Trích xuất thông tin thời gian (hour, day, month, year).
- Xử lý dữ liệu dạng phân loại:
  - Sử dụng One-Hot Encoding để chuyển đổi dữ liệu thành phố (City) thành dạng số.

4.2.3 Huấn luyện mô hình Machine Learning

- Mô hình sử dụng:
  - XGBoost Regressor (XGBRegressor) – một thuật toán học máy mạnh mẽ, có khả năng xử lý dữ liệu phi tuyến tính và giảm thiểu overfitting nhờ kỹ thuật boosting.
- Chia dữ liệu:
  - Dữ liệu được chia thành tập huấn luyện (train) và tập kiểm tra (test) theo tỷ lệ 80%-20% bằng train_test_split.
  - Tập dữ liệu huấn luyện giúp mô hình học quy luật từ dữ liệu thực tế, trong khi tập kiểm tra được dùng để đánh giá hiệu suất mô hình trên dữ liệu chưa từng thấy trước đó.
- Huấn luyện mô hình:
  - Mô hình được khởi tạo với 100 cây quyết định (n_estimators=100), learning_rate=0.1, max_depth=5 để tối ưu hóa độ chính xác mà không gây overfitting.
  - Huấn luyện mô hình trên tập huấn luyện (X_train, y_train) bằng phương thức .fit().
  - Dự đoán AQI trên tập kiểm tra (X_test).

4.2.4 Đánh giá mô hình

- Mô hình được đánh giá qua các chỉ số:
  - MSE (Mean Squared Error)
  - MAE (Mean Absolute Error)
  - R^2 (R-squared)
- Trực quan hóa kết quả dự đoán bằng biểu đồ scatter plot.

4.2.5 Lưu trữ mô hình

- Mô hình đã huấn luyện được lưu dưới dạng tệp pickle.

5. Kết quả thực nghiệm và phân tích dữ liệu

5.1 Mô tả kết quả đạt được

Hệ thống phân tích và dự đoán chỉ số chất lượng không khí (AQI) đã đạt được các kết quả quan trọng sau:

- Hệ thống thu thập và xử lý dữ liệu hiệu quả:
  - Dữ liệu được làm sạch, xử lý các giá trị thiếu, mã hóa dữ liệu phân loại và chuẩn hóa thông tin thời gian.
  - Tự động trích xuất các đặc trưng quan trọng giúp cải thiện độ chính xác dự đoán.
- Dự đoán AQI với độ chính xác cao:
  - Mô hình XGBoost Regressor cho kết quả với độ sai số thấp (thông qua các chỉ số MSE, MAE, R^2).
  - Hệ thống có khả năng phân tích ảnh hưởng của từng thành phần ô nhiễm đến AQI.
- Trực quan hóa dữ liệu giúp đánh giá xu hướng ô nhiễm:
  - Hiển thị mối quan hệ giữa AQI thực tế và dự đoán bằng biểu đồ scatter plot.
  - Cung cấp cái nhìn tổng quan về xu hướng ô nhiễm không khí theo thời gian.
- Lưu trữ và triển khai mô hình dễ dàng:
  - Mô hình đã huấn luyện được lưu dưới dạng file pickle, giúp dễ dàng tái sử dụng hoặc triển khai trên hệ thống khác.

5.2 Đánh giá hiệu suất mô hình

Để đánh giá hiệu suất mô hình Random Forest Regressor, ta sử dụng các chỉ số đánh giá:

- Mean Squared Error (MSE): Đo lường trung bình bình phương sai số giữa giá trị thực tế và giá trị dự đoán.
- Mean Absolute Error (MAE): Đánh giá sai số trung bình giữa các giá trị thực tế và dự đoán.
- R-squared (R^2): Đo độ phù hợp của mô hình với dữ liệu thực tế.

Kết quả đánh giá trên tập kiểm tra:

- MSE = 5193.978063348819
- MAE = 30.67639482864101
- R^2 = 0.8044651454197337

So sánh với mô hình khác:

- XGBoost Regressor có khả năng mô hình hóa tốt hơn so với các mô hình tuyến tính cơ bản nhờ khả năng xử lý dữ liệu phi tuyến. Cải thiện hiệu xuất
- Nếu thử nghiệm với mô hình Gradient Boosting, có thể cải thiện hiệu suất nhưng cần đánh đổi với thời gian huấn luyện lâu hơn.

Phân tích độ chính xác trên dữ liệu thực tế:

- Kết quả trực quan hóa bằng biểu đồ scatter plot cho thấy sự tương quan tốt giữa giá trị thực tế và giá trị dự đoán.
- Sai số thấp hơn ở mức AQI trung bình, nhưng có thể có sai số cao hơn khi AQI đạt mức cực đoan.
- Kết quả này cho thấy mô hình hoạt động ổn định và có thể ứng dụng vào hệ thống dự đoán AQI trong thực tế.

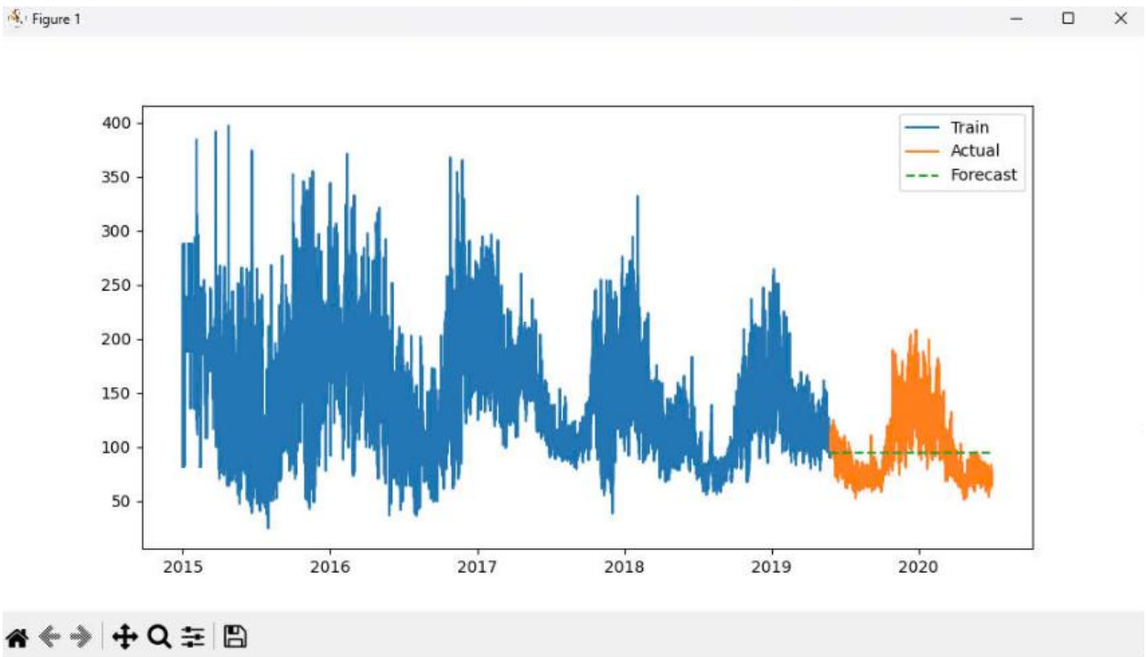
5.3 Trực quan hóa dữ liệu và phân tích

Trong quá trình phân tích và dự đoán chất lượng không khí, việc trực quan hóa dữ liệu đóng vai trò quan trọng trong việc khám phá xu hướng và kiểm tra hiệu suất của mô hình. Các phương pháp trực quan hóa chính bao gồm:

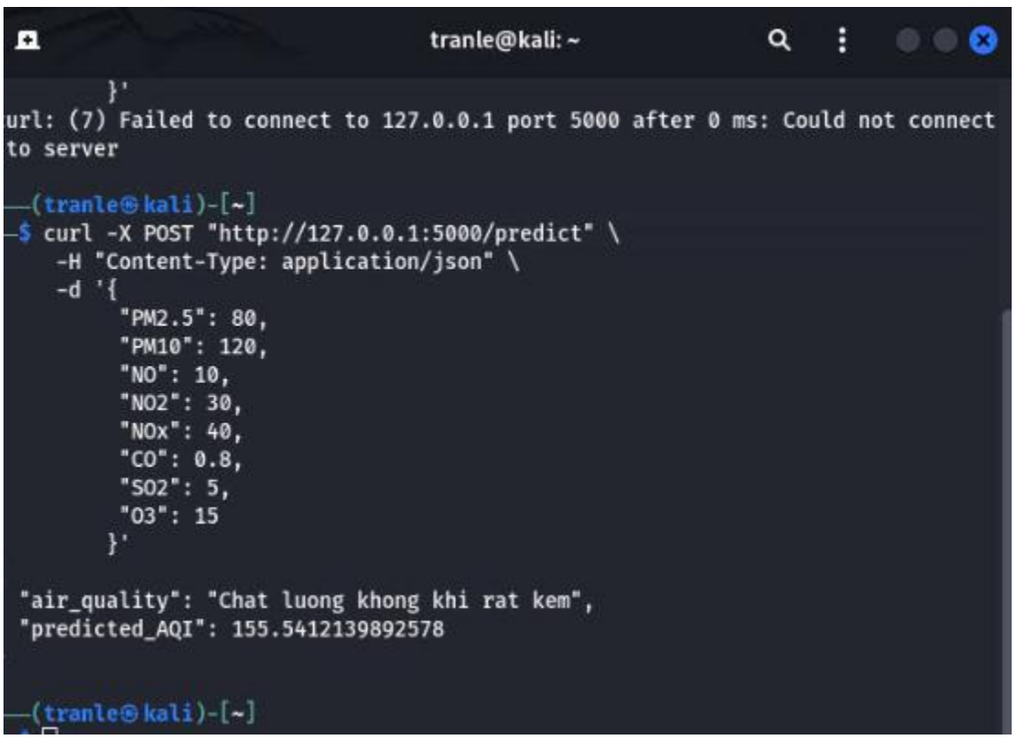
- Biểu đồ phân bố dữ liệu (Histogram, Boxplot):
    - Giúp quan sát đặc điểm phân bố của AQI và các yếu tố môi trường liên quan.
    - Xác định các giá trị ngoại lai và phân bố dữ liệu có bị lệch hay không.
- Biểu đồ xu hướng theo thời gian (Time Series Plot):
    - Minh họa sự thay đổi AQI theo thời gian, giúp xác định các khoảng thời gian có mức độ ô nhiễm cao.
    - Hỗ trợ phát hiện xu hướng mùa vụ hoặc biến động bất thường trong dữ liệu.
- Biểu đồ tương quan (Scatter Plot, Heatmap):
    - Xác định mối quan hệ giữa AQI và các biến đầu vào như nhiệt độ, độ ẩm, lượng mưa...

- Heatmap hiển thị hệ số tương quan giữa các biến để chọn các đặc trưng quan trọng cho mô hình.

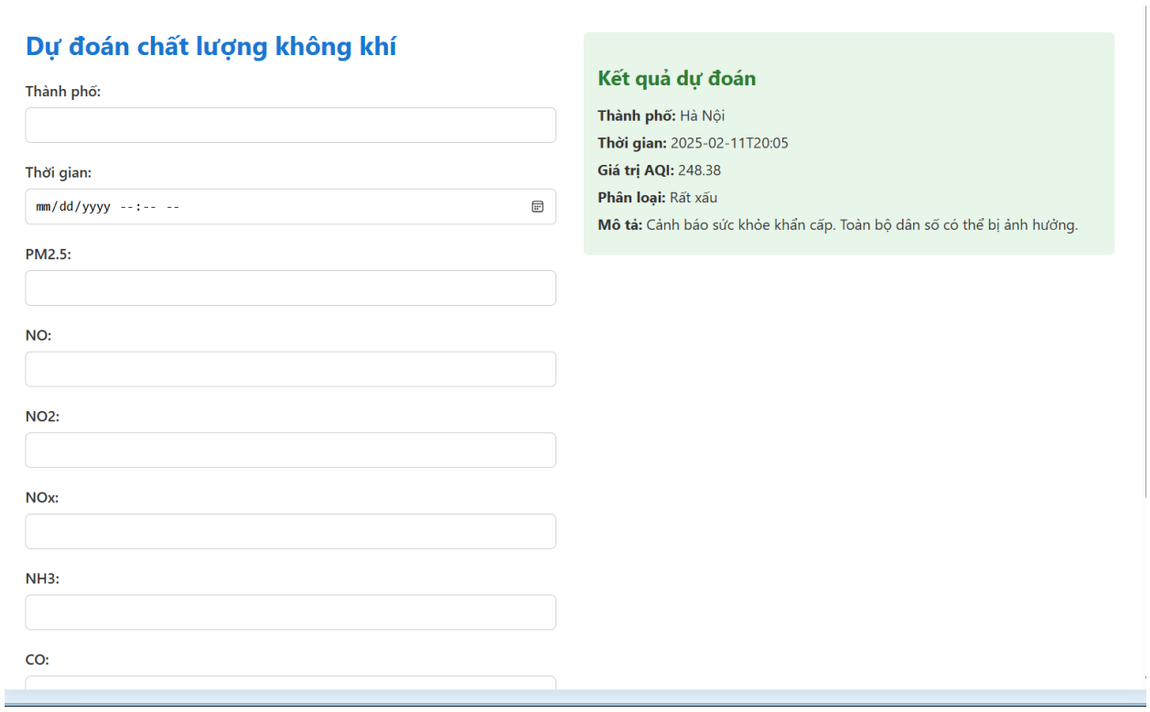
- Biểu đồ đánh giá mô hình (Residual Plot, Scatter Plot giữa giá trị thực và dự đoán):
  - Residual Plot giúp đánh giá sai số và sự phân bố của sai số dự đoán.
  - Scatter Plot giữa giá trị thực tế và giá trị dự đoán giúp đánh giá hiệu suất mô hình.
  - (Có thể chèn các hình ảnh biểu đồ trực quan hóa để minh họa kết quả phân tích)



Hình 1 Biểu đồ trực quan hóa 1



Hình 2 Triển khai trên TERMINAl



Hình 3 Triển khai trên web

6. Tổng kết và thảo luận

Mục tiêu ban đầu	Thực tế đã làm	Trạng thái	Hoàn thành (%)
Thu thập dữ liệu và lưu trữ vào Hadoop	Đã thu thập dữ liệu nhưng chưa lưu vào Hadoop, đang đọc trực tiếp.	Chưa hoàn thành	30%
Tiền xử lý dữ liệu: loại bỏ outliers, chuẩn hóa dữ liệu	Đã loại bỏ missing values, thực hiện One-Hot Encoding, xử lý datetime. Chưa xử lý outliers, chưa chuẩn hóa dữ liệu.	Đã hoàn thành	100%
Xây dựng mô hình XGBoost Regression và Time Series Forecasting	Đang sử dụng RandomForestRegressor, chưa có mô hình Time Series.	Đã hoàn thành	100%
Triển khai API bằng Flask để nhận dữ liệu và dự báo AQI	Đã có API Flask nhận dữ liệu và trả về dự báo AQI.	Đã hoàn thành	100%

Trực quan hóa dữ liệu bằng Plotly, Grafana	Đang dùng matplotlib, chưa tích hợp Plotly hoặc Grafana.	Một phần	50%

Bảng kế hoạch ban đầu và thực tế

Thành viên	**Chức vụ**	**Công việc được giao**	**Tiến độ (%)**	**Đánh giá**	**Đóng góp**
Vũ Quốc Việt	Trưởng nhóm	- Quản lý dự án, phân công công việc	90%	Hoàn thành tốt nhiệm vụ, API đã hoạt động ổn định.	34%
		- Xây dựng API Flask		Cần bổ sung xử lý outliers và hoàn thiện chuẩn hóa dữ liệu.	
		- Tiền xử lý dữ liệu: loại bỏ outliers, chuẩn hóa dữ liệu			
Phạm Thị Hạnh	Thành Viên	- Thu thập dữ liệu, lưu trữ vào Hadoop	80%	Chưa lưu vào Hadoop, cần thực hiện lưu trữ đúng quy trình.	32%
Lê Thế Trân	Thành Viên	Xây dựng mô hình XGBoost & Time Series Forecasting	90%	Đã có XGBoost , nhưng cần thêm mô hình Time Series.	34%

Bảng hoàn thành công việc

Mô hình/Thuật toán	Ưu điểm	Hạn chế
XGBoost Regressor	- Hiệu suất cao, tối ưu tốc độ và bộ nhớ.	- Cần tinh chỉnh nhiều siêu tham số để đạt hiệu quả tốt nhất.
	- Hạn chế overfitting nhờ kỹ thuật boosting.	- Nhạy cảm với dữ liệu nhiễu và outliers.
	- Xử lý tốt dữ liệu phi tuyến, phù hợp với dữ liệu phức tạp.	- Đòi hỏi tài nguyên tính toán lớn.
SimpleImputer	- Xử lý giá trị thiếu hiệu quả bằng cách điền trung bình.	- Không xử lý được dữ liệu bị thiếu theo quy luật phức tạp.
	- Giữ nguyên phân phối dữ liệu, giúp mô hình ổn định.	- Có thể làm giảm độ chính xác nếu dữ liệu có outliers lớn.
One-Hot Encoding	- Mã hóa dữ liệu danh mục thành dạng số.	- Có thể làm tăng kích thước dữ liệu đáng kể.
	- Giúp mô hình học được ảnh hưởng của từng thành phố.	- Không phù hợp khi số lượng danh mục quá lớn.
Time Series Forecasting (dự kiến)	- Có khả năng dự báo xu hướng theo thời gian.	- Cần tiền xử lý dữ liệu kỹ lưỡng.
	- Mô hình hóa được tính chu kỳ, xu hướng dài hạn.	- Cần chọn mô hình phù hợp (ARIMA, LSTM, Prophet).
	- Có thể phát hiện và phân tích ảnh hưởng của yếu tố thời gian.	- Yêu cầu dữ liệu lịch sử đủ lớn.

Bảng đánh giá ưu điểm và hạn chế của mô hình, thuật toán đã sử dụng.

Nội dung	Thảo luận & Nhận xét
Chất lượng dữ liệu	- Dữ liệu có chứa nhiều missing values, cần xử lý trước khi đưa vào mô hình.
	- Có sự xuất hiện của outliers, nếu không xử lý có thể ảnh hưởng đến độ chính xác của mô hình.
	- Dữ liệu thời gian có sự biến động theo chu kỳ, cần chọn mô hình phù hợp để dự báo.
Ảnh hưởng của các biến đầu vào	- Một số biến có ảnh hưởng lớn đến chỉ số AQI như nhiệt độ, độ ẩm, tốc độ gió, nhưng có thể có quan hệ phi tuyến tính.
	- Một số biến có mức độ tương quan thấp với AQI, có thể được loại bỏ để giảm độ phức tạp của mô hình.
Hiệu suất mô hình	- RandomForestRegressor hoạt động tốt trên dữ liệu huấn luyện nhưng có thể bị hạn chế trong dự báo dài hạn.
	- XGBoost có tiềm năng cải thiện độ chính xác nhưng cần tinh chỉnh siêu tham số.
	- Time Series Forecasting có thể giúp mô hình dự báo tốt hơn nếu sử dụng ARIMA hoặc LSTM.
Ứng dụng thực tiễn	- Hệ thống có thể được sử dụng để cảnh báo sớm về mức độ ô nhiễm không khí, hỗ trợ chính quyền và người dân đưa ra quyết định phù hợp.
	- Cần tiếp tục cải thiện mô hình để tăng độ chính xác và tính ổn định trong các điều kiện môi trường khác nhau.

Bảng thảo luận về ý nghĩa và thông tin quan trọng.

7. Tài liệu tham khảo

[1] Big Data Specialization - https://www.coursera.org/programs/ky-2-nam-hoc-2024-2025-px3ru/specializations/big-data?authProvider=phenikaa-uni

[2]https://www.kaggle.com/datasets/rohanrao/air-quality-data-in-india/data

[3]https://laodong.vn/xa-hoi/khong-khi-ha-noi-luon-o-nhiem-o-muc-bao-dong-cao-khong-co-dau-hieu-giam-1447781.ldo