ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

ĐỒ ÁN PHÂN TÍCH DỮ LIỆU KINH DOANH

ĐỀ TÀI: ÁP DỤNG PHÂN TÍCH DỮ LIỆU KINH DOANH

Dự đoán giá xăng RON-95 tại VIỆT NAM

Giảng viên hướng dẫn: ThS. Dương Phi Long

Thực hiện: Nhóm 2 – Khoa Hệ Thống Thông Tin

Thành viên:

Trương Thanh Quang	MSSV: 23521295
Đào Bảo Phúc	MSSV: 23521192
Ngô Nhật Quân	MSSV: 23521258
Huỳnh Trần Anh Thư	MSSV: 23521535

Học kỳ: HK1 – Năm học 2025–2026

TP. Hồ Chí Minh – 2025

ÁP DỤNG PHÂN TÍCH DỮ LIỆU KINH DOANH
Dự đoán giá xăng RON-95 tại VIỆT NAM

Abstract

Trong bối cảnh nền kinh tế thị trường đầy biến động, việc dự báo chính xác giá xăng dầu, đặc biệt là xăng RON 95, đóng vai trò then chốt trong việc hoạch định chính sách năng lượng và quản lý rủi ro tài chính tại Việt Nam. Bài báo này trình bày một nghiên cứu thực nghiệm về việc ứng dụng và so sánh hiệu suất của các mô hình Học máy và Học sâu trong bài toán dự báo giá xăng RON 95 dựa trên dữ liệu theo tuần. Nghiên cứu đề xuất một khung phân tích chuỗi thời gian đa biến, tích hợp các yếu tố ngoại sinh có tác động mạnh đến giá xăng thành phẩm bao gồm: giá dầu thô thế giới, Chỉ số giá tiêu dùng và tỷ giá hối đoái . Các mô hình được triển khai và đánh giá bao gồm [Linear, SVR, ARIMAX, XGBOOST, RNN, LSTM và GRU]. Kết quả thực nghiệm dựa trên các chỉ số đánh giá sai số như RMSE, MAE và MAPE cho thấy mô hình Linear Regrestion đạt hiệu suất có thể chấp nhận được trong việc nắm bắt các xu hướng phi tuyến tính và biến động ngắn hạn của thị trường. Nghiên cứu này cung cấp cơ sở khoa học quan trọng cho việc lựa chọn giải pháp dự báo tối ưu hỗ trợ ra quyết định trong lĩnh vực năng lượng tại Việt Nam.

I. Giới thiệu

Năng lượng hóa thạch, đặc biệt là xăng RON 95, đóng vai trò huyết mạch trong nền kinh tế Việt Nam, chiếm tỷ trọng lớn trong chi phí vận tải và tác động trực tiếp đến chỉ số giá tiêu dùng (CPI). Thống kê từ Bộ Công Thương cho thấy giá xăng thành phẩm có biên độ dao động trung bình năm từ 15% đến 20%, gây áp lực đáng kể lên lạm phát và kế hoạch sản xuất của doanh nghiệp [1].

Bối cảnh thị trường năng lượng trong nước đã có những thay đổi căn bản kể từ khi Nghị định 80/2023/NĐ-CP có hiệu lực, rút ngắn chu kỳ điều hành giá xuống còn 07 ngày. Cơ chế này tuy tăng cường tính thị trường nhưng đồng thời làm gia tăng độ biến động và nhiễu của chuỗi dữ liệu giá, đặt ra yêu cầu cấp thiết về các công cụ dự báo định lượng chính xác nhằm hỗ trợ quản trị rủi ro [2].

Tuy nhiên, việc xây dựng mô hình dự báo giá xăng RON 95 tại Việt Nam đối mặt với hai rào cản kỹ thuật đặc thù, tạo nên khoảng trống nghiên cứu cần được giải quyết.

Thứ nhất là thách thức về sự bất đồng bộ tần suất dữ liệu trong mô hình đa biến. Bài toán yêu cầu tích hợp chuỗi giá xăng tuần với các biến vĩ mô quan trọng như CPI, tỷ giá VND/USD và giá dầu thô chuẩn (BRENT, WTI). Vấn đề nảy sinh khi các chỉ số vĩ mô thường được công bố theo tháng, trong khi biến mục tiêu lại biến thiên theo tuần. Sự chênh lệch này đòi hỏi các kỹ thuật tiền xử lý và nội suy phù hợp để đồng bộ hóa dữ liệu mà không làm sai lệch cấu trúc xu hướng gốc [3].

Thứ hai là hạn chế về quy mô dữ liệu mẫu. Do cơ chế điều hành giá mới chỉ được áp dụng trong thời gian gần đây, dữ liệu lịch sử khả dụng chỉ dừng lại ở mức hơn 300 điểm dữ liệu tuần. Đây là trở ngại lớn đối với các kiến trúc mạng nơ-ron sâu như Long Short-Term Memory hay Gated Recurrent Unit, vốn dễ rơi vào trạng thái quá khớp khi thiếu hụt dữ liệu huấn luyện. Vì vậy, việc tinh chỉnh siêu tham số và áp dụng các kỹ thuật regularization trở thành yêu cầu bắt buộc để đảm bảo khả năng tổng quát hóa của mô hình.

Nghiên cứu này giải quyết các thách thức trên thông qua việc thiết lập khung phân tích so sánh giữa các thuật toán Máy học và Học sâu trên tập dữ liệu tuần giới hạn. Trọng tâm nghiên cứu là đánh giá hiệu quả của việc tích hợp các biến ngoại sinh đa khung thời gian nhằm tìm ra kiến trúc tối ưu cho bài toán dự báo ngắn hạn tại thị trường Việt Nam.

Cấu trúc bài báo được tổ chức như sau:

- Phần II lược khảo các công trình nghiên cứu liên quan.
- Phần III trình bày phương pháp luận và thiết kế mô hình.
- Phần IV thảo luận kết quả thực nghiệm.
- Phần V là kết luận cùng hướng phát triển.

II. Các nghiên cứu gần đây

Sagheer và Kotb (2019) đề xuất mô hình LSTM sâu để dự báo sản lượng dầu thô và chứng minh rằng LSTM vượt trội so với các mô hình truyền thống như ARIMA và RNN. Kết quả của nghiên cứu cho thấy mô hình học sâu có khả năng nắm bắt quan hệ phi tuyến và phụ thuộc dài hạn tốt hơn trong các chuỗi thời gian năng lượng [4].

He (2023) áp dụng các mô hình chuỗi thời gian cổ điển như ARIMA, SVR... để dự báo giá bán lẻ xăng tại Mỹ theo tuần. Nghiên cứu cho thấy các mô hình tuyến tính truyền thống vẫn đạt sai số thấp và ổn định trong bối cảnh dữ liệu điều chỉnh theo chu kỳ, cho thấy không phải lúc nào mô hình phức tạp cũng cho kết quả tốt hơn [5].

Ở góc độ truyền dẫn giá, Nguyen (2013) phân tích mối quan hệ giữa giá dầu thô và giá xăng bán lẻ. Nghiên cứu chỉ ra rằng biến động của giá dầu thô quốc tế có ảnh hưởng trực tiếp và đáng kể đến giá xăng, một kết quả phù hợp với bối cảnh điều hành giá tại Việt Nam, nơi giá xăng phụ thuộc mạnh vào giá dầu thô nhập khẩu [6].

III. Phương pháp nghiên cứu

3.1 Bộ dữ liệu

Bộ dữ liệu được xây dựng dựa trên giá bán lẻ xăng RON95 tại Việt Nam, thu thập từ cổng thông tin chính thức của PVOIL/Petrolimex. Trước năm 2019, giá xăng được công bố theo chu kỳ 15 ngày, sau đó giảm còn 10 ngày và hiện nay được điều chỉnh mỗi tuần (thường vào thứ Năm). Biến mục tiêu price biểu diễn giá bán (đồng/lít).

Ngoài biến mục tiêu, nghiên cứu sử dụng các biến ngoại sinh liên quan đến kinh tế vĩ mô và thị trường năng lượng như Chỉ số Giá Tiêu dùng (CPI), tỷ giá USD/VND, các chỉ số dầu thô quốc tế (Brent, WTI). Dữ liệu CPI được công bố hàng tháng bởi Tổng cục Thống kê (GSO) – Bộ Tài chính. Các biến ngoại sinh khác đa số có tần suất tuần nhưng thường cập nhật vào Chủ nhật.

Toàn bộ dữ liệu được chuẩn hóa theo tần suất tuần để bảo đảm tính đồng bộ thời gian. Tập dữ liệu cuối cùng bao gồm 378 tuần, trong giai đoạn từ cuối năm 2018 đến tháng 11 năm 2025.

Tiền xử lý dữ liệu: Do chu kỳ cập nhật giá xăng không đều, giá tại thời điểm điều chỉnh được giữ nguyên cho hai tuần liên tiếp nhằm phản ánh đúng cơ chế hiệu lực giá. Cách tiếp cận này tương đồng với phương pháp resampling theo step-wise hold trong các nghiên cứu dự báo giá năng lượng ngắn hạn [7].

Dữ liệu CPI (tần suất tháng) được giữ nguyên cho toàn bộ các tuần thuộc tháng đó , phù hợp với cách xử lý dữ liệu vĩ mô tần suất thấp được khuyến nghị trong [8], [9].

Xảy ra hiện tượng lệch mốc thời gian cập nhật giữa các biến: biến price thường cập nhật vào thứ Năm, trong khi các biến ngoại sinh khác ghi nhận giá trị vào Chủ nhật. Do đó, toàn bộ biến đầu vào được căn chỉnh về cùng mốc tham chiếu vào thứ Năm, thông qua kỹ thuật đồng bộ hóa dữ liệu bất đồng bộ [10].

- Dữ liệu có tần suất đồng nhất;
- Tất cả biến sử dụng chung mốc thời gian;
- Chuỗi thời gian liên tục, không tồn tại giá trị rỗng sau chuẩn hóa;

Sau khi làm sạch, dữ liệu được chia theo thứ tự thời gian với tỉ lệ 70% cho tập huấn luyện, 15% cho đánh giá và 15% cho tập kiểm thử, đảm bảo tính toàn vẹn của chuỗi thời gian và khả năng đánh giá hiệu quả mô hình một cách khách quan. Toàn bộ các mô hình được triển khai và huấn luyện trên cùng một tập dữ liệu xử lý để đảm bảo tính nhất quán trong quá trình so sánh và đánh giá.

Các biến chính trong tập dữ liệu:

- **price:** Giá bán lẻ xăng RON95 tại Việt Nam (VND/lít).
- **brent:** Giá dầu Brent quốc tế (USD/thùng).
- **WTI:** Giá dầu thô WTI quốc tế (USD/thùng).
- **usd_vnd:** Tỷ giá hối đoái giữa VND và USD (VND/USD).
- **cpi:** Chỉ số giá tiêu dùng (CPI) của Việt Nam (dữ liệu tháng được chuẩn hóa theo tuần).

Feature engineering cho chuỗi thời gian

Từ dữ liệu thô gồm các biến gốc như `price`, `Brent`, `WTI`, `usd_vnd`, chúng em thực hiện các bước feature-engineering để tạo các biến bổ sung nhằm giúp mô hình học được các mối quan hệ thời gian, biến động, xu hướng và tác động gián tiếp. Cụ thể:

- **Lag_1:** giá xăng của tuần trước ($price_{t-1}$). Việc dùng lag giúp mô hình nắm được tính *tự tương quan* (autocorrelation) trong chuỗi.

- **Price_Pct, Brent_Pct, WTI_Pct, USDVND_Pct:** biến phần trăm thay đổi giữa tuần hiện tại và tuần trước, phản ánh biến động tương đối thay vì chỉ giá tuyệt đối.
- **Brent_WTI_Spread, Brent_WTI_Ratio:** chênh lệch và tỉ lệ giữa giá Brent và WTI — nhằm khai thác ảnh hưởng từ cấu trúc dầu thô quốc tế tới giá xăng nội địa.
- **USD_Brent_Interact:** biến tương tác giữa tỷ giá và biến động giá dầu ($usd\_vnd \times Brent\_Pct$) — nhằm bắt tác động gián tiếp khi quy đổi dầu nhập khẩu bằng VND.
- **Moving average (MA):** các biến trung bình (`Price_MA3`, `Brent_MA3`, `USDVND_MA3`) để làm mượt nhiễu ngắn hạn, giúp mô hình học xu hướng rõ hơn.

Dữ liệu sau khi tạo features được loại bỏ missing value để đảm bảo tính đồng nhất trước khi đưa vào mô hình.
Việc tạo các feature như lag, rolling mean, biến thay đổi phần trăm, spread, interaction... đặt trong khung chung của *feature engineering for time series*, vốn được ghi nhận giúp cải thiện hiệu suất dự báo so với dùng dữ liệu thô [11].

3.2 Các mô hình dự đoán

3.2.1 Linear Regression

Hồi quy tuyến tính là mô hình nền tảng trong thống kê và học máy, được dùng để mô tả mối quan hệ tuyến tính giữa biến phụ thuộc y (giá xăng Việt Nam) và một hoặc nhiều biến độc lập x_i (các yếu tố vĩ mô).
Mô hình hồi quy tuyến tính đơn (1 biến độc lập) được biểu diễn như:

$$y = \beta_0 + \beta_1 x + \epsilon \tag{1}$$

Trong đó:

- y: biến phụ thuộc (giá xăng trong nước);
- x: biến độc lập (ví dụ: usd_index);
- β_0: hệ số chặn (intercept);
- β_1: hệ số hồi quy biểu thị mức thay đổi trung bình của y khi x tăng 1 đơn vị;
- ϵ: thành phần sai số ngẫu nhiên.

Khi có nhiều biến độc lập cùng ảnh hưởng đến giá xăng, mô hình hồi quy tuyến tính bội được sử dụng:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_i x_i + \epsilon \tag{2}$$

với β_i là hệ số hồi quy của từng biến độc lập x_i (ví dụ: CPI, USD/VND, giá dầu,...). Các hệ số β_i được ước lượng bằng phương pháp *Bình phương tối thiểu* (Ordinary Least Squares – OLS):

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{3}$$

Trong đó:

- $\mathbf{X}$: ma trận các biến độc lập (ma trận thiết kế) có kích thước $n \times p$.
- $\mathbf{y}$: vector biến phụ thuộc kích thước $n \times 1$.
- $\hat{\boldsymbol{\beta}}$: vector hệ số ước lượng $p \times 1$.
- $(\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\mathbf{X}^\mathsf{T}$: được gọi là nghịch đảo giả MoorePenrose của ma trận $\mathbf{X}$.

Mục tiêu của phương pháp hồi quy là tối thiểu hóa sai số giữa giá trị quan sát thực tế y_i và giá trị dự đoán $\hat{y}_i$. Kết quả ước lượng thu được từ mô hình cho phép xác định mức độ và chiều hướng ảnh hưởng của các biến độc lập đến biến phụ thuộc — trong trường hợp này là giá xăng.

Dấu của các hệ số β_i thể hiện tác động cùng chiều (+) hoặc ngược chiều (−) của từng biến giải thích đối với biến mục tiêu, qua đó cung cấp cơ sở định lượng để đánh giá mối quan hệ giữa các yếu tố kinh tế vĩ mô và biến động giá xăng.

3.2.2 Support Vector Regression (SVR)

Support Vector Regression (SVR) là mở rộng của thuật toán *Support Vector Machine (SVM)* cho bài toán hồi quy, có khả năng xử lý quan hệ phi tuyến giữa các biến. Thay vì giảm thiểu tổng sai số như Linear Regression, SVR tìm một hàm $f(x)$ sao cho phần lớn điểm dữ liệu nằm trong khoảng sai số ε cho phép, đồng thời mô hình có độ phức tạp nhỏ nhất.
Hàm hồi quy tuyến tính của SVR được viết:

$$f(x) = \mathbf{w}^T\mathbf{x} + b \tag{4}$$

Trong đó:

- $\mathbf{w}$: vector trọng số (*weight vector*), thể hiện mức độ ảnh hưởng của các biến đầu vào đến giá trị dự đoán.
- b: hệ số chặn (*bias term*), giúp điều chỉnh mô hình để tối ưu hóa độ khớp với dữ liệu.
- $\mathbf{x}$: vector đặc trưng đầu vào (*feature vector*).

Bài toán tối ưu của SVR là:

$$\min_{\mathbf{w},b,\xi,\xi^*} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{5}$$

$$\text{Với các ràng buộc: } \begin{cases} y_i - (\mathbf{w}^T\mathbf{x}_i + b) \le \varepsilon + \xi_i \\ (\mathbf{w}^T\mathbf{x}_i + b) - y_i \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases}$$

Trong đó

- ε: ngưỡng sai số chấp nhận được (*epsilon-insensitive margin*).
- ξ_i, ξ_i^*: các biến nới lỏng (*slack variables*) cho phép vi phạm ràng buộc ε, đảm bảo mô hình vẫn có thể học được khi dữ liệu có nhiễu.
- C: hệ số phạt (*regularization parameter*), cân bằng giữa độ phẳng của siêu phẳng (mức độ đơn giản của mô hình) và sai số cho phép.

Với dữ liệu phi tuyến, SVR sử dụng *Kernel Trick* để ánh xạ dữ liệu sang không gian đặc trưng cao hơn, phổ biến nhất là hàm RBF (Radial Basis Function):

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|^2) \tag{6}$$

Trong đó:

- $\mathbf{x}_i, \mathbf{x}_j$: hai điểm dữ liệu bất kỳ trong không gian đầu vào.
- γ (*gamma*): tham số điều chỉnh độ cong (*spread*) của hàm RBF.

Khi $\mathbf{x}_i$ và $\mathbf{x}_j$ **gần nhau**, khoảng cách $\|\mathbf{x}_i - \mathbf{x}_j\|^2$ nhỏ $\Rightarrow$ giá trị $K(\mathbf{x}_i, \mathbf{x}_j)$ gần 1, nghĩa là hai điểm có mức độ tương đồng cao.

Ngược lại, khi $\mathbf{x}_i$ và $\mathbf{x}_j$ **xa nhau**, giá trị $K(\mathbf{x}_i, \mathbf{x}_j)$ gần 0, nghĩa là mức độ tương đồng giữa hai điểm thấp.

Nhờ vậy, SVR có thể mô hình hóa quan hệ phi tuyến giữa giá xăng và các yếu tố kinh tế như CPI, tỷ giá, lãi suất, v.v.

3.2.3 Autegressive Integrated Moving Average with Exogenous Variables (ARIMAX)

ARIMAX là phiên bản mở rộng của mô hình ARIMA, cho phép tích hợp thêm các biến ngoại sinh (*exogenous variables*) để cải thiện độ chính xác của dự báo. Nếu như ARIMA chỉ dựa vào lịch sử của chính chuỗi thời gian đó, thì ARIMAX mô hình hóa mối quan hệ giữa biến phụ thuộc y_t với các biến độc lập bên ngoài X_t.
Mô hình này đặc biệt hiệu quả trong các bài toán kinh tế lượng, ví dụ như dự báo giá xăng dầu (biến phụ thuộc) dựa trên sự biến động của giá dầu thô thế giới và tỷ giá hối đoái (biến ngoại sinh).
Tên gọi ARIMAX bao gồm 4 thành phần chính:

AR (AutoRegressive): Phần tự hồi quy, mô tả sự phụ thuộc của giá trị hiện tại y_t vào các giá trị quá khứ của chính nó $(y_{t-1}, y_{t-2}, \ldots)$.

I (Integrated): Phần tích hợp, biểu thị số lần sai phân d để đưa chuỗi dữ liệu (cả biến phụ thuộc và biến ngoại sinh) về trạng thái dừng (*stationary*).

MA (Moving Average): Phần trung bình động, mô tả sự phụ thuộc của giá trị hiện tại vào các sai số ngẫu nhiên trong quá khứ.

X (Exogenous): Các biến ngoại sinh. Đây là các yếu tố bên ngoài hệ thống nhưng có tác động đồng thời hoặc trễ đến biến mục tiêu. Trong mô hình ARIMAX, các biến này được thêm vào phương trình hồi quy tuyến tính.

Công thức tổng quát của mô hình ARIMAX(p, d, q) với k biến ngoại sinh được biểu diễn như sau:

$$y'_t = c + \sum_{i=1}^{p} \phi_i y'_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \sum_{m=1}^{k} \beta_m x_{m,t} + \varepsilon_t \tag{7}$$

Trong đó:

- y'_t: giá trị của chuỗi thời gian tại thời điểm t (đã qua sai phân bậc d nếu cần).
- c: hằng số (intercept).
- ϕ_i: hệ số tự hồi quy bậc i.
- θ_j: hệ số trung bình động bậc j.
- $x_{m,t}$: giá trị của biến ngoại sinh thứ m tại thời điểm t (hoặc $t - b$ nếu có độ trễ).
- β_m: hệ số hồi quy tương ứng với biến ngoại sinh thứ m, phản ánh mức độ tác động của biến ngoài lai lên y_t.
- ε_t: sai số ngẫu nhiên (white noise) với giả định $E(\varepsilon_t) = 0$ và phương sai σ^2 không đổi.

Về mặt toán tử, mô hình ARIMAX cũng có thể được viết gọn lại dưới dạng:

$$\phi_p(B)(1 - B)^d y_t = \theta_q(B)\varepsilon_t + \sum_{m=1}^{k} \beta_m x_{m,t} \tag{8}$$

Trong đó B là toán tử trễ ($By_t = y_{t-1}$). Phương trình này cho thấy giá trị dự báo là sự kết hợp tuyến tính của lịch sử chuỗi, sai số quá khứ và giá trị hiện tại của các biến giải thích bên ngoài.

Quy trình xây dựng mô hình ARIMAX: Quy trình tương tự như ARIMA nhưng có thêm bước xử lý biến ngoại sinh:

Quy trình xây dựng mô hình ARIMAX:

1. **Xác định bộ biến ngoại sinh:** Sử dụng phương pháp so sánh chỉ số AIC / BIC. Tiến hành thử nghiệm các tập biến ngoại sinh khác nhau; mô hình nào có chỉ số AIC và BIC thấp hơn thì được xem là tốt hơn, từ đó xác định được các biến phù hợp nhất để đưa vào mô hình.
2. **Xác định tham số p, d, q bằng AUTO-ARIMA:** Chọn bộ tham số tối ưu bằng cách dò tìm tự động trên không gian tham số và đánh giá các mô hình ứng viên dựa trên các tiêu chí thông dụng như AIC và BIC.
3. **Ước lượng mô hình:** Các tham số thành phần AR, MA và hệ số hồi quy của biến ngoại sinh được ước lượng bằng phương pháp Ước lượng hợp lý cực đại (Maximum Likelihood Estimation - MLE) thông qua hàm `fit()` của mô hình ARIMAX.
4. **Kiểm tra độ phù hợp của mô hình:** Độ phù hợp được đánh giá toàn diện thông qua các chỉ số thông tin (AIC, BIC) và các chỉ số đo lường sai số dự báo (MAE, RMSE, MAPE).
5. **Dự báo và đánh giá mô hình:** Mô hình sau khi hoàn thiện được sử dụng để thực hiện dự báo. Hiệu quả của mô hình được đánh giá dựa trên sai số giữa giá trị dự báo và dữ liệu thực tế (tập kiểm tra).

Tóm lại: ARIMAX là một công cụ mạnh mẽ khắc phục hạn chế của ARIMA trong việc bỏ qua các yếu tố tác động bên ngoài. Đối với bài toán dự báo giá xăng, việc áp dụng ARIMAX cho phép mô hình "học" được sự nhạy cảm của giá xăng trong nước đối với các cú sốc từ giá dầu thế giới hoặc biến động tỷ giá, từ đó mang lại kết quả dự báo sát thực tế hơn.

3.2.4 Extreme Gradient Boosting (XGBoost)

XGBoost (eXtreme Gradient Boosting) [**?**] là một thuật toán học máy mạnh mẽ dựa trên cây quyết định, thuộc họ mô hình *Gradient Boosting*. Khác với các phương pháp *bagging* như Random Forest xây dựng các cây độc lập, XGBoost sử dụng kỹ thuật **boosting tuần tự**. Cụ thể, các cây mới được sinh ra để sửa lỗi của các cây trước đó bằng cách tối ưu hoá hàm mất mát thông qua đạo hàm bậc nhất và bậc hai.
Mô hình dự đoán tại bước lặp t đối với mẫu dữ liệu x_i được biểu diễn như sau:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i), \quad \text{với } f_t \in \mathcal{F} \tag{9}$$

trong đó $\hat{y}_i^{(t-1)}$ là kết quả dự đoán tại bước trước, $f_t(x_i)$ là giá trị trọng số của cây quyết định thứ t, và $\mathcal{F}$ là không gian các cây quyết định (regression trees).

Để tối ưu hóa mô hình, hàm mục tiêu (Objective Function) tại bước t được thiết lập bao gồm hàm mất mát và thành phần điều chuẩn (regularization) nhằm kiểm soát độ phức tạp:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t)}\right) + \Omega(f_t) \tag{10}$$

Thành phần regularization $\Omega(f)$ đóng vai trò quan trọng trong việc ngăn chặn hiện tượng quá khớp (overfitting), được định nghĩa là:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{11}$$

Trong đó:

- $l(\cdot)$: hàm mất mát đo lường sai số giữa thực tế và dự báo (ví dụ: MSE cho bài toán hồi quy);
- T: số lượng lá (leaf nodes) của cây;
- w_j: trọng số (weight) của lá thứ j;
- γ, λ: các siêu tham số kiểm soát mức độ phạt lên cấu trúc cây.

Điểm đột phá của XGBoost nằm ở việc sử dụng khai triển Taylor bậc hai để xấp xỉ hàm mất mát, giúp quá trình tối ưu diễn ra nhanh và chính xác hơn. Hàm mục tiêu (sau khi lược bỏ các hằng số) trở thành:

$$\mathcal{L}^{(t)} \approx \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2 \right] + \Omega(f_t) \tag{12}$$

với $g_i = \partial_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$ và $h_i = \partial^2_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$ lần lượt là đạo hàm bậc nhất và bậc hai của hàm mất mát.
Nhờ khả năng xử lý dữ liệu khuyết thiếu tự động, hỗ trợ tính toán song song và cơ chế regularization mạnh, XGBoost đặc biệt phù hợp và thường đạt hiệu suất cao đối với dữ liệu dạng bảng (tabular data) như bài toán dự báo biến động giá xăng dầu trong nghiên cứu này.

3.2.5 Recurrent Neural Network (RNN)

Recurrent Neural Network (RNN) là kiến trúc mạng nơ-ron chuyên xử lý dữ liệu chuỗi thời gian, cho phép mô hình ghi nhớ thông tin từ các bước trước đó thông qua cơ chế hồi tiếp (recurrent connections). Khác với mạng truyền thẳng, RNN duy trì một trạng thái ẩn h_t đại diện cho ngữ cảnh quá khứ, giúp mô hình nắm bắt được phụ thuộc theo thời gian.
Quá trình lan truyền tiến (forward pass) của RNN được mô tả như:

$$h_t = f(W_{xh} x_t + W_{hh} h_{t-1} + b_h) \tag{13}$$

$$\hat{y}_t = W_{hy} h_t + b_y \tag{14}$$

Trong đó:

- x_t: đầu vào tại thời điểm t (giá xăng, chỉ số USD, CPI...);
- h_t: trạng thái ẩn hiện tại lưu trữ thông tin chuỗi;
- $\hat{y}_t$: giá trị dự đoán;
- W_{xh}, W_{hh}, W_{hy}: ma trận trọng số học được trong quá trình huấn luyện;
- b_h, b_y: vector bias;
- $f(\cdot)$: hàm kích hoạt phi tuyến như tanh hoặc ReLU.

Hàm mất mát (Loss Function) được sử dụng phổ biến trong RNN là:

$$\mathcal{L} = \frac{1}{n} \sum_{t=1}^{n} (y_t - \hat{y}_t)^2 \tag{15}$$

Các trọng số được cập nhật qua thuật toán *Backpropagation Through Time (BPTT)* để học quan hệ tạm thời. RNN có thể gặp hiện tượng *vanishing gradient* khi chuỗi quá dài, làm giảm khả năng học phụ thuộc dài hạn. Do đó, các biến thể như **LSTM** (Long Short-Term Memory) và **GRU** (Gated Recurrent Unit) được đề xuất nhằm khắc phục vấn đề này thông qua cơ chế cổng điều khiển (gates).

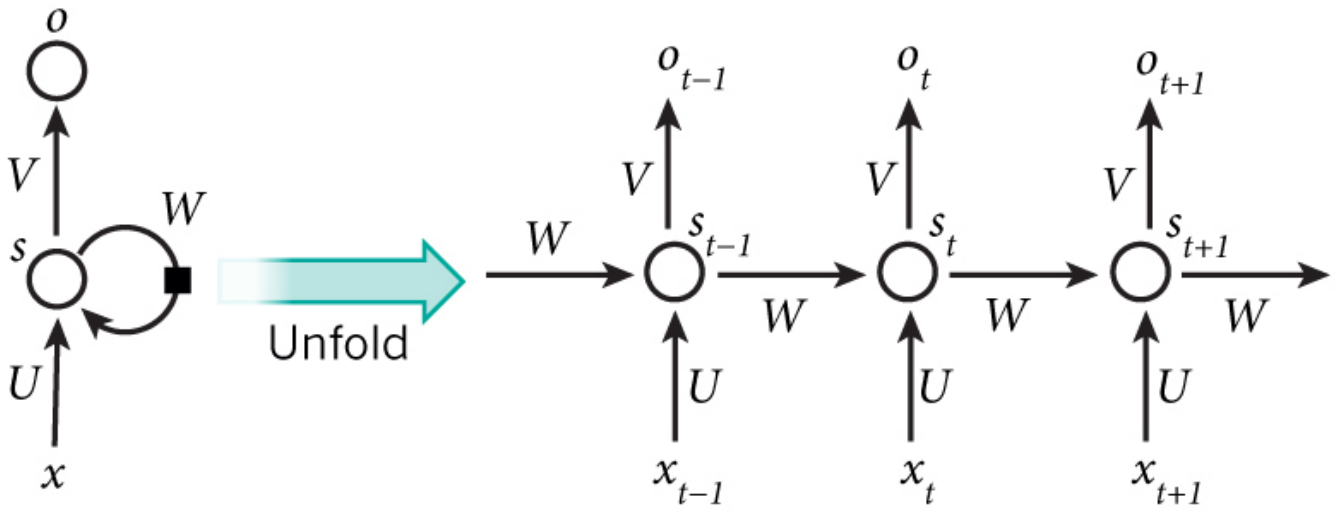

Figure 1: Cấu trúc tổng quát của mô hình RNN.

3.2.6 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) là một biến thể nâng cao của mạng nơ-ron hồi quy (RNN), được đề xuất nhằm giải quyết vấn đề *vanishing gradient* và *exploding gradient* trong quá trình huấn luyện. Kiến trúc LSTM được thiết kế với các "cổng" điều khiển (*gates*) giúp mô hình có khả năng chọn lọc thông tin cần ghi nhớ hoặc quên đi theo thời gian. Mỗi đơn vị LSTM (*LSTM cell*) bao gồm ba cổng chính: *Forget Gate*, *Input Gate*, và *Output Gate*, cùng hai trạng thái đặc trưng — *trạng thái ô nhớ* (C_t) và *trạng thái ẩn* (h_t).

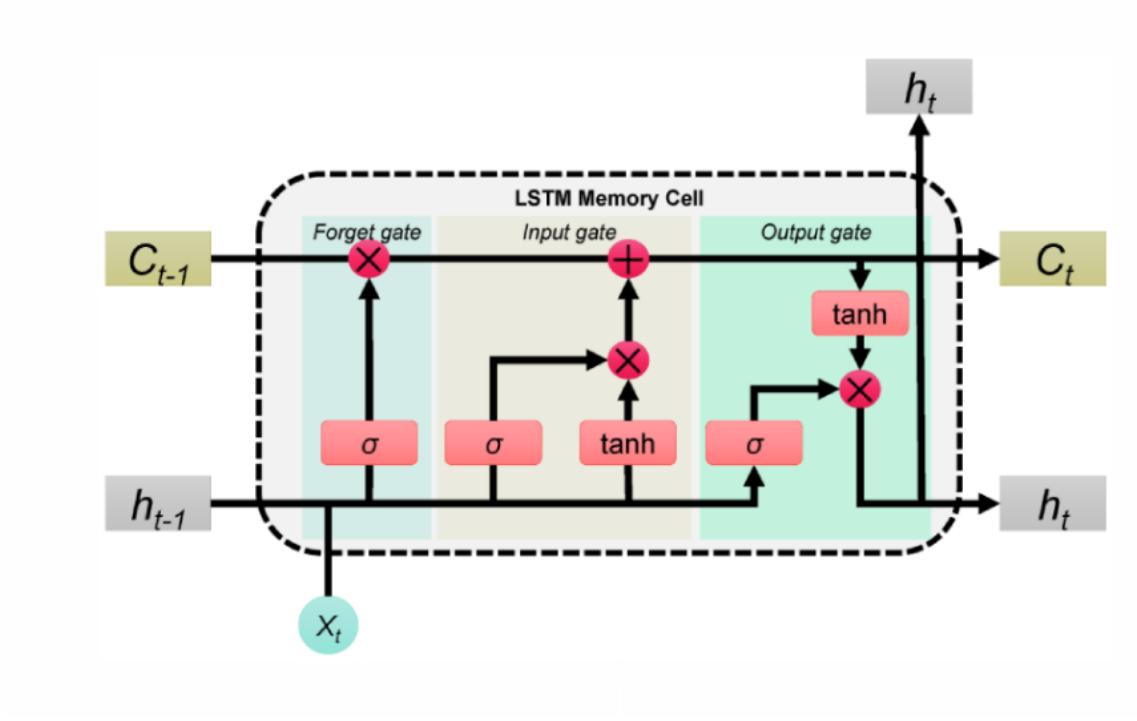

Figure 2: Cấu trúc tổng quát của một khối LSTM, gồm ba cổng điều khiển chính.

Forget Gate: Đây là bước đầu tiên trong mỗi khối LSTM. Cổng quên chịu trách nhiệm xác định phần thông tin nào từ bộ nhớ trước đó (C_{t-1}) nên được giữ lại hoặc loại bỏ. Cổng này nhận hai đầu vào: trạng thái ẩn trước đó (h_{t-1}) và đầu vào hiện tại (x_t), sau đó tính toán:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{16}$$

Trong đó:

- f_t: *hệ số vector* (*forget vector*), giá trị trong khoảng $[0, 1]$.
- h_{t-1}: *trạng thái ẩn* tại thời điểm $(t-1)$.
- x_t: *đầu vào hiện tại* tại thời điểm t.
- W_f, b_f: *trọng số* và *hệ số chệch* của Forget Gate.
- σ: *hàm kích hoạt sigmoid*, đảm bảo giá trị đầu ra nằm trong khoảng $[0, 1]$.

Nếu giá trị f_t gần 1, thông tin trong C_{t-1} sẽ được duy trì; ngược lại, nếu gần 0, thông tin tương ứng sẽ bị quên đi. Điều này giúp mô hình "lọc nhiễu" từ các bước thời gian quá xa, giữ lại những thông tin quan trọng cho dự báo hiện tại.

Input Gate: Cổng vào xác định lượng thông tin mới được thêm vào bộ nhớ tại thời điểm hiện tại. Nó gồm hai thành phần hoạt động song song:

(1) Một lớp sigmoid quyết định mức độ thông tin mới được phép thêm vào ô nhớ:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{17}$$

Trong đó:

- i_t: *vector hệ số đầu vào* (*input vector*), giá trị trong khoảng $[0, 1]$.
- h_{t-1}: *trạng thái ẩn* của bước thời gian trước $(t-1)$.
- x_t: *đầu vào hiện tại* tại thời điểm t.
- W_i, b_i: *trọng số* và *hệ số chệch* của cổng đầu vào.
- σ: *hàm kích hoạt sigmoid*, giúp xác định mức độ thông tin mới được phép ghi vào bộ nhớ.

(2) Một lớp tanh tạo ra các giá trị ứng viên $\tilde{C}_t$ — đây là thông tin mới tiềm năng có thể được lưu trữ:

$$\tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C) \tag{18}$$

Trong đó:

- $\tilde{C}_t$: *ứng viên trạng thái bộ nhớ* (*candidate cell state*), biểu diễn thông tin mới tiềm năng được thêm vào bộ nhớ hiện tại.
- tanh: *hàm kích hoạt phi tuyến* giúp giới hạn giá trị trong khoảng $[-1, 1]$, đảm bảo ổn định gradient trong quá trình huấn luyện.

Sau khi hai thành phần này được tính, LSTM cập nhật bộ nhớ thông qua kết hợp giữa phần cần giữ lại $(f_t * C_{t-1})$ và phần thông tin mới được chọn $(i_t * \tilde{C}_t)$:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{19}$$

Trong đó

- C_t: *trạng thái bộ nhớ hiện tại* (*cell state*).
- $f_t * C_{t-1}$: *phần thông tin được giữ lại* từ bước trước thông qua **Forget Gate**.
- $i_t * \tilde{C}_t$: *thông tin mới được chọn lọc thêm vào bộ nhớ* thông qua **Input Gate**.

Công thức trên thể hiện cách LSTM duy trì thông tin dài hạn trong C_t, đồng thời liên tục điều chỉnh nó dựa trên dữ liệu mới.

Output Gate: Cổng đầu ra điều chỉnh lượng thông tin từ bộ nhớ (C_t) được sử dụng để sinh ra trạng thái ẩn hiện tại (h_t) — chính là đầu ra của cell tại thời điểm t. Trước hết, cổng sigmoid xác định phần nào của ô nhớ sẽ được xuất ra:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{20}$$

Trong đó

- o_t: *vector cổng đầu ra* (*output gate vector*), xác định phần thông tin trong bộ nhớ được phép xuất ra.
- W_o, b_o: *trọng số* và *hệ số chệch* của cổng đầu ra.
- σ: *hàm kích hoạt sigmoid*, giúp giới hạn đầu ra trong khoảng $[0, 1]$.
- tanh: *hàm kích hoạt phi tuyến* giúp điều chỉnh mức độ kích hoạt của trạng thái bộ nhớ C_t.

Sau đó, giá trị trạng thái ẩn được tính bằng:

$$h_t = o_t * \tanh(C_t) \tag{21}$$

Trong đó

- $\tanh(C_t)$: giúp *chuẩn hóa giá trị bộ nhớ* về khoảng $[-1, 1]$, tránh hiện tượng *bão hòa giá trị* trong quá trình lan truyền ngược (backpropagation).
- o_t: *điều chỉnh mức độ "hiển thị" của thông tin* từ bộ nhớ C_t, ra đầu ra thực tế, quyết định phần nào của trạng thái ẩn được xuất ra.

3.2.7 Gated Recurrent Unit (GRU)

Gated Recurrent Unit (GRU) là một biến thể cải tiến của mạng LSTM được giới thiệu bởi Cho et al. (2014). GRU được thiết kế nhằm đơn giản hóa cấu trúc của LSTM trong khi vẫn duy trì khả năng học các phụ thuộc dài hạn trong chuỗi dữ liệu. Không giống như LSTM có ba cổng điều khiển và hai trạng thái (C_t, h_t), GRU chỉ sử dụng **hai cổng** — *Update Gate* và *Reset Gate* — và chỉ có một trạng thái duy nhất h_t để vừa lưu trữ vừa truyền thông tin.

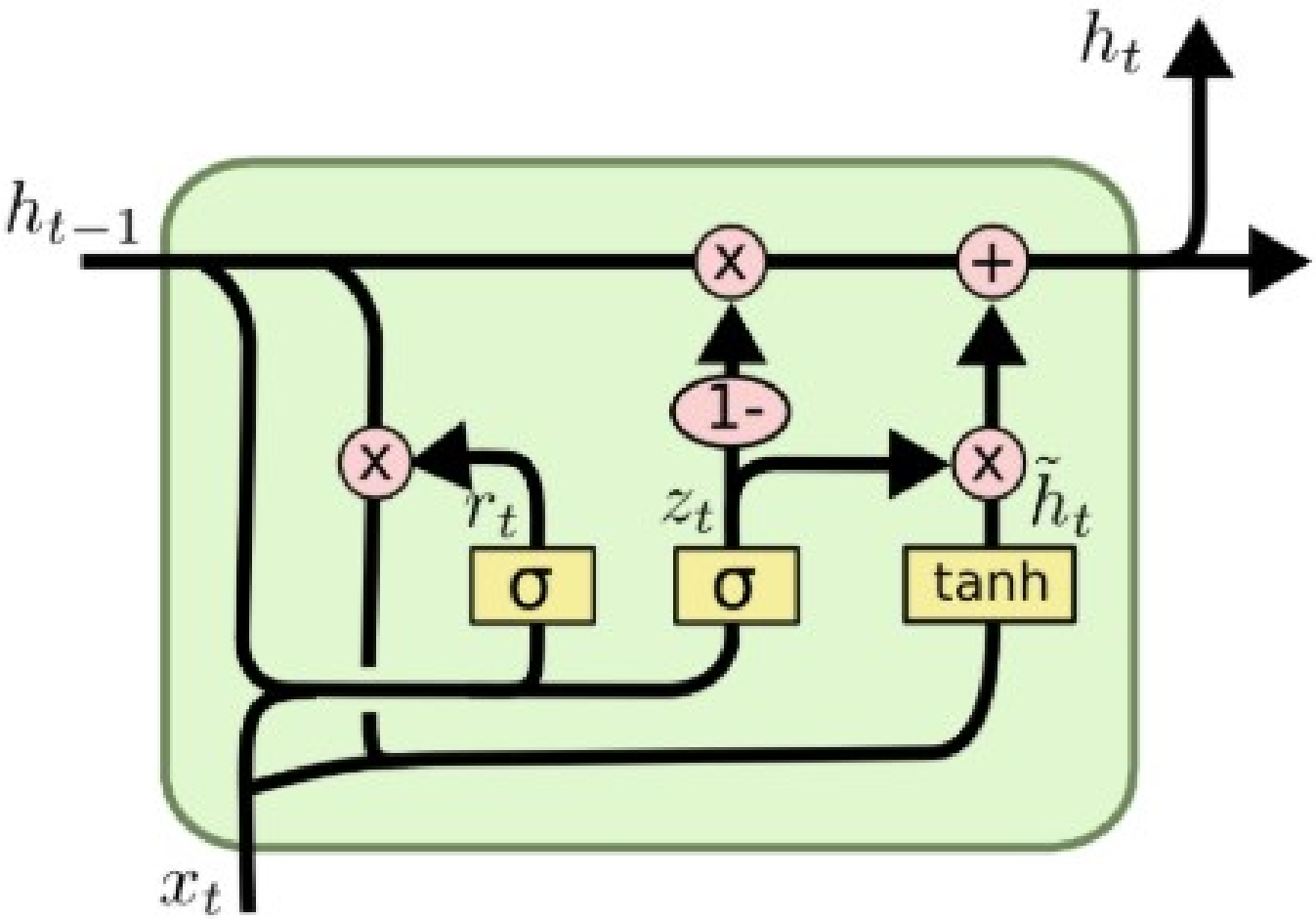

Figure 3: Cấu trúc mô hình GRU

Reset Gate: Cổng này điều khiển cách kết hợp thông tin mới từ đầu vào hiện tại với thông tin cũ từ bước trước. Nó giúp mô hình "quên có chọn lọc" một phần thông tin quá khứ khi cần xử lý các chuỗi dữ liệu có tính thay đổi nhanh. Công thức được biểu diễn như sau:

$$r_t = \sigma(W_r[h_{t-1}, x_t] + b_r) \tag{22}$$

Trong đó

- r_t: *hệ số đặt lại*, xác định lượng thông tin từ quá khứ cần được quên hoặc giữ lại khi tính toán trạng thái mới.

Giá trị r_t càng nhỏ thì thông tin từ trạng thái ẩn trước (h_{t-1}) bị "reset" càng nhiều, tức mô hình chú trọng hơn vào dữ liệu mới (x_t).
Update Gate: Cổng cập nhật quyết định lượng thông tin từ quá khứ được giữ lại để hình thành trạng thái hiện tại. Nếu giá trị của z_t gần 1, mô hình giữ lại phần lớn thông tin trước đó; ngược lại, nếu gần 0, mô hình sẽ cập nhật hoàn toàn dựa trên dữ liệu mới.

$$z_t = \sigma(W_z[h_{t-1}, x_t] + b_z) \tag{23}$$

Trong đó

- z_t: *hệ số cập nhật* (*update coefficient*), xác định mức độ ảnh hưởng của trạng thái quá khứ lên trạng thái hiện tại.

Candidate Activation: Dựa trên cổng reset, GRU tính toán trạng thái ứng viên $\tilde{h}_t$ — đại diện cho thông tin mới được đề xuất để cập nhật:

$$\tilde{h}_t = \tanh(W_h[r_t * h_{t-1}, x_t] + b_h) \tag{24}$$

Ở đây, $r_t * h_{t-1}$ cho phép mô hình quyết định phần nào của thông tin cũ sẽ được dùng để tạo ứng viên mới.
Cập nhật trạng thái ẩn: Cuối cùng, trạng thái ẩn hiện tại h_t được tính bằng cách kết hợp giữa trạng thái cũ và trạng thái ứng viên theo trọng số của cổng cập nhật:

$$h_t = (1 - z_t) * \tilde{h}_t + z_t * h_{t-1} \tag{25}$$

Công thức này cho phép GRU kiểm soát mượt mà giữa việc "ghi nhớ" và "cập nhật" thông tin, mà không cần tách riêng hai trạng thái như trong LSTM.

3.3 Phương pháp đánh giá mô hình

Để đánh giá hiệu suất dự báo của các mô hình, nghiên cứu sử dụng các chỉ số thống kê phổ biến trong phân tích hồi quy và dự báo chuỗi thời gian. Các chỉ số này phản ánh độ chính xác, khả năng tổng quát hoá và mức độ phù hợp của mô hình so với dữ liệu thực tế.
1) Hệ số xác định R^2 (Coefficient of Determination): Hệ số R^2 đo lường tỷ lệ biến thiên của giá trị thực (y_i) được giải thích bởi mô hình dự báo ($\hat{y}_i$). Giá trị R^2 nằm trong khoảng [0, 1]; càng gần 1, mô hình càng phù hợp với dữ liệu.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{26}$$

Trong đó:

- y_i: giá trị thực tế tại quan sát thứ i;
- $\hat{y}_i$: giá trị dự đoán của mô hình;
- $\bar{y}$: giá trị trung bình của biến phụ thuộc;
- n: số lượng quan sát.

2) Sai số tuyệt đối trung bình (Mean Absolute Error – MAE): MAE đo lường sai số trung bình tuyệt đối giữa giá trị thực và giá trị dự đoán:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{27}$$

Trong đó

- y_i: *giá trị thực tế.*
- $\hat{y}_i$: *giá trị dự đoán.*
- n: *số lượng quan sát.*

MAE phản ánh sai lệch trung bình tuyệt đối của mô hình; giá trị càng nhỏ, dự báo càng chính xác.
Chỉ số này ít nhạy cảm với ngoại lệ (outliers) hơn so với RMSE, giúp mô hình được đánh giá ổn định hơn.
3) Sai số phần trăm tuyệt đối trung bình (Mean Absolute Percentage Error – MAPE): MAPE cho biết mức sai lệch trung bình theo phần trăm giữa giá trị dự báo và giá trị thực tế:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{28}$$

Trong đó

- y_i: giá trị thực tế.
- $\hat{y}_i$: giá trị dự đoán.
- n: số lượng quan sát.

MAPE cho biết mức sai lệch trung bình tính theo phần trăm so với giá trị thực tế. Giá trị MAPE càng nhỏ, mô hình càng chính xác.
Thông thường, MAPE $< 10\%$ được xem là rất tốt trong các bài toán dự báo kinh tế. Chỉ số này đặc biệt hữu ích khi so sánh các mô hình trên các tập dữ liệu có quy mô khác nhau.

4) Căn bậc hai sai số bình phương trung bình (Root Mean Square Error – RMSE): RMSE là chỉ số đánh giá phổ biến trong dự báo chuỗi thời gian, đo lường mức độ lệch bình phương giữa giá trị thực và giá trị dự đoán:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{29}$$

Trong đó

- y_i: giá trị thực tế.
- $\hat{y}_i$: giá trị dự đoán.
- n: số lượng quan sát.

Khác với MAE, RMSE phạt nặng hơn cho các sai số lớn, do đó nhạy cảm hơn với giá trị ngoại lai. Giá trị RMSE càng nhỏ, mô hình dự báo càng chính xác và sai lệch trung bình giữa dự đoán và thực tế càng thấp.
Tóm tắt:

- R^2 và Adjusted R^2: đo độ phù hợp của mô hình (goodness-of-fit);
- MAE và RMSE: đo mức sai số tuyệt đối và bình phương;
- MAPE: đánh giá sai số tương đối theo phần trăm, dễ diễn giải trực quan.

5) Độ chính xác về xu hướng (Directional Accuracy - DA):
Chỉ số DA đo lường khả năng của mô hình trong việc dự báo đúng chiều hướng biến động (tăng hoặc giảm) của chuỗi dữ liệu so với bước thời gian trước đó.
Giá trị DA nằm trong khoảng [0, 1]; giá trị càng cao cho thấy mô hình dự đoán xu hướng càng chính xác.

$$DA = \frac{1}{n}\sum_{i=1}^{n} d_i, \quad \text{với } d_i = \begin{cases} 1 & \text{nếu } (y_i - y_{i-1})(\hat{y}_i - y_{i-1}) > 0 \\ 0 & \text{ngược lại} \end{cases} \tag{30}$$

Trong đó:

- y_i: giá trị thực tế tại quan sát thứ i;
- $\hat{y}_i$: giá trị dự đoán của mô hình tại quan sát thứ i;
- y_{i-1}: giá trị thực tế tại quan sát liền trước (bước $i-1$);
- n: số lượng quan sát.

Các chỉ số trên được sử dụng đồng thời để đảm bảo đánh giá toàn diện độ chính xác của từng mô hình dự báo trong nghiên cứu.

IV. Phân tích dữ liệu

4.1 Phân tích mô tả (Descriptive Statistics)

Phân tích mô tả cung cấp cái nhìn tổng quan về dữ liệu thông qua các thống kê: trung bình (Mean), trung vị (Median), giá trị cực đại (Max), độ lệch chuẩn (Std), độ lệch (Skewness) và độ nhọn (Kurtosis). Bảng 1 trình bày các đặc trưng mô tả cho từng biến trong tập dữ liệu.

Table 1: Thống kê mô tả các biến trong bộ dữ liệu giá xăng RON95

Biến	Mean	Median	Max	Std	Skew	Kurt
PRICE	21115.54	21100.00	32870.00	3579.13	0.1381	1.4321
CPI	100.26	100.20	101.52	0.46	-0.2282	2.6049
Brent	72.62	73.51	122.01	18.02	-0.0474	0.5445
WTI	68.13	69.29	120.67	17.98	0.0402	0.5526
VND_USD	23876.06	23355.00	26410.00	1059.38	1.0192	-0.3249

Các biến trong bộ dữ liệu nhìn chung có phân phối lệch phải nhẹ (Skewness > 0) và độ nhọn thấp (Kurtosis < 3). Biến *PRICE* có độ lệch chuẩn lớn nhất, thể hiện mức độ biến động cao hơn so với các biến vĩ mô còn lại.

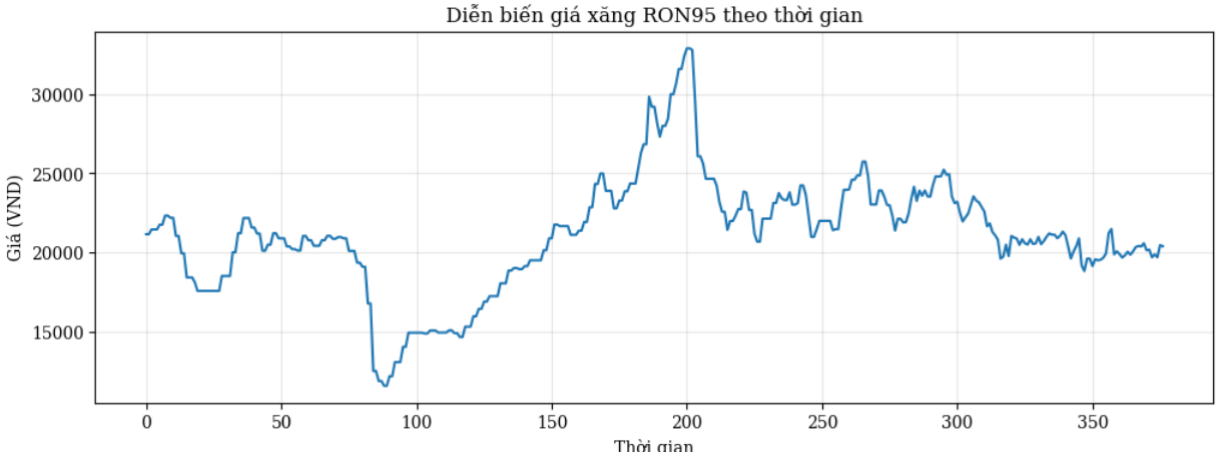

Figure 4: Biểu đồ giá Xăng theo tuần

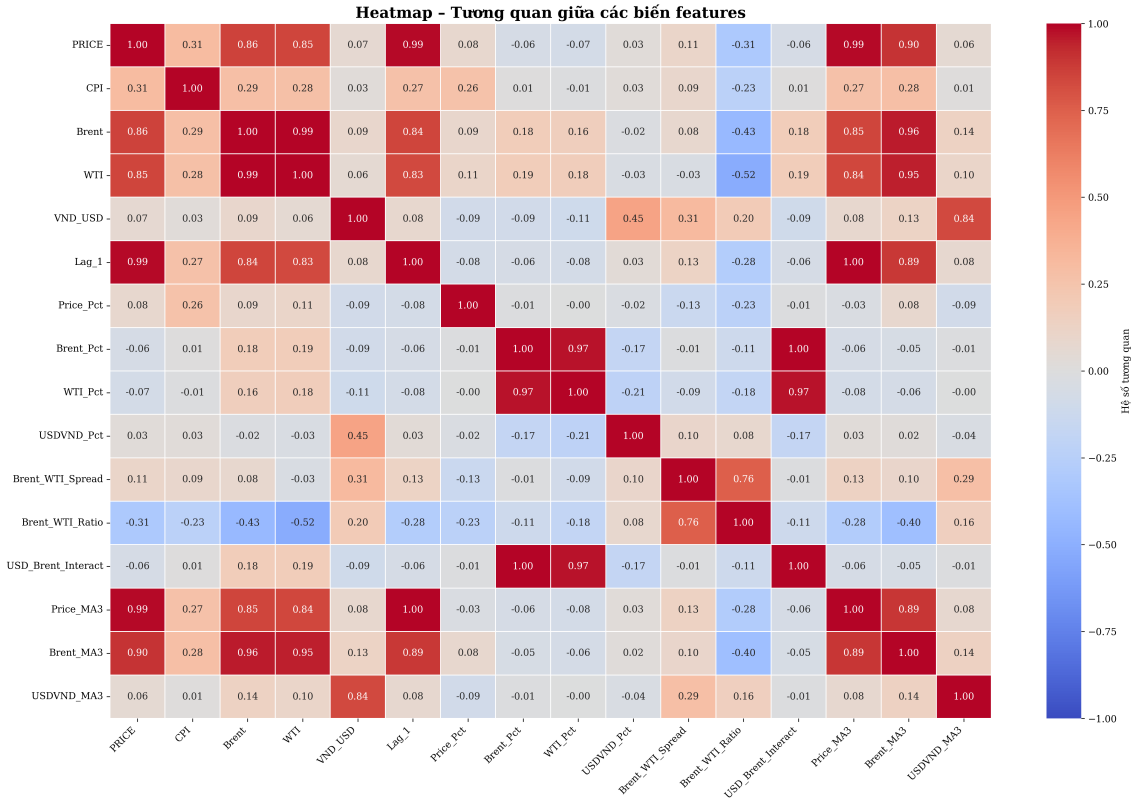

Figure 5: Biểu đồ thể hiện tương quan giữa các biến

Biểu đồ tương quan cho thấy **giá xăng RON95 (PRICE)** có tương quan dương mạnh với *giá dầu Brent*, *WTI* và *Lag_1*, phản ánh sự phụ thuộc của giá xăng trong nước vào thị trường dầu thô quốc tế và cơ chế điều hành theo kỳ trước. Tỷ giá *USD/VND* cũng có mức tương quan dương đáng kể, trong khi các biến dạng phần trăm (Pct) cho thấy tương quan thấp do đặc tính nhiễu ngắn hạn. Nhìn chung, Brent, WTI, Lag_1 và các biến MA3 là những biến đầu vào quan trọng nhất cho mô hình dự báo.

4.2 Kiểm định ý nghĩa thống kê (ANOVA và Chi-Square)

Nhằm kiểm tra xem các biến kinh tế có ảnh hưởng đáng kể đến giá xăng Việt Nam hay không, nhóm tiến hành hai phép kiểm định: (1) **ANOVA** – đánh giá sự khác biệt về trung bình giữa các biến, và (2) **Chi-Square** – kiểm tra mối quan hệ phụ thuộc giữa các nhóm giá xăng và các biến kinh tế.

Table 2: Kết quả kiểm định ANOVA cho các biến đầu vào

Feature	**F-statistic**	**p-value**	**Ý nghĩa**
CPI	12.4069	0.000007	Có ý nghĩa
Brent	169.4313	0.000000	Có ý nghĩa
WTI	160.1347	0.000000	Có ý nghĩa
VND_USD	4.0823	0.017974	Có ý nghĩa
Lag_1	313.5824	0.000000	Có ý nghĩa
Price_Pct	0.6695	0.512832	Không
Brent_Pct	0.2199	0.802771	Không
WTI_Pct	0.2852	0.752076	Không
USDVND_Pct	0.6516	0.522056	Không
Brent_WTI_Spread	4.6055	0.010836	Có ý nghĩa
Brent_WTI_Ratio	13.0560	0.000004	Có ý nghĩa
USD_Brent_Interact	0.2183	0.804059	Không
Price_MA3	295.9883	0.000000	Có ý nghĩa
Brent_MA3	208.5689	0.000000	Có ý nghĩa
USDVND_MA3	3.8142	0.023315	Có ý nghĩa

Table 3: Kết quả kiểm định Chi-Square giữa PRICE và các biến độc lập

Feature	χ^2	**p-value**	**Ý nghĩa**
CPI	26.5293	0.000025	Có ý nghĩa
Brent	150.9925	0.000000	Có ý nghĩa
WTI	139.0762	0.000000	Có ý nghĩa
VND_USD	41.7025	0.000000	Có ý nghĩa
Lag_1	435.3362	0.000000	Có ý nghĩa
Price_Pct	3.4451	0.486268	Không
Brent_Pct	2.0344	0.729423	Không
WTI_Pct	3.2027	0.524487	Không
USDVND_Pct	11.7362	0.019425	Có ý nghĩa
Brent_WTI_Spread	26.9138	0.000021	Có ý nghĩa
Brent_WTI_Ratio	24.5496	0.000062	Có ý nghĩa
USD_Brent_Interact	2.8688	0.580015	Không
Price_MA3	374.8274	0.000000	Có ý nghĩa
Brent_MA3	171.7432	0.000000	Có ý nghĩa
USDVND_MA3	37.4041	0.000000	Có ý nghĩa

Thực tế ANOVA và Chi-square không được thiết kế cho dữ liệu chuỗi thời gian, vì các quan sát trong time series không độc lập, trái với giả định của hai kiểm định. Do đó, kết quả ANOVA/Chi-square chỉ mang tính tham khảo mức độ liên hệ sơ bộ, chứ không phải tiêu chí quyết định giữ hay loại biến. [12]

Trong bài toán dự báo, đặc biệt khi mô hình ML/DL có thể học quan hệ phi tuyến và tương tác giữa các biến, việc một biến "không có ý nghĩa thống kê" không đồng nghĩa với việc nó không có giá trị dự báo.[13]

Vì lý do này, nhóm vẫn giữ toàn bộ các biến đầu vào để đảm bảo mô hình khai thác tối đa thông tin của chuỗi thời gian.

Kết luận: Vậy nên ở bài toán này sẽ giữ nguyên tất cả các biến đầu vào để không làm ảnh hưởng đến hiệu suất mô hình.

4.3 Mô hình Linear Regression

Mô hình hồi quy tuyến tính được sử dụng để dự đoán giá xăng tại Việt Nam dựa trên các biến kinh tế cũng như một số biến nội sinh.
Các hệ số này phản ánh mức độ ảnh hưởng tuyến tính của từng yếu tố kinh tế lên giá xăng trong giai đoạn 2018–2025.

Table 4: Tham số cấu hình của các mô hình Linear Regression

Mô hình	**Tham số**
OLS (Ordinary Least Squares)	Regularization: None; fit_intercept=True; normalize=False
Ridge Regression	Regularization: L2; Alpha tốt nhất: `best_ridge_alpha`; Tập alpha thử nghiệm: {0.001, 0.01, 0.1, 1, 10, 100}
Lasso Regression	Regularization: L1; Alpha tốt nhất: `best_lasso_alpha`; Tập alpha thử nghiệm: {0.001, 0.01, 0.1, 1, 10, 100}; max_iter=10000
ElasticNet	Regularization: L1 + L2; Alpha tốt nhất: `best_elastic_params[0]`; L1 Ratio tốt nhất: `best_elastic_params[1]`; Tập alpha thử nghiệm: {0.001, 0.01, 0.1, 1, 10}; Tập l1_ratio thử nghiệm: {0.1, 0.3, 0.5, 0.7, 0.9}; max_iter=10000
Polynomial Regression (degree = 2)	Degree: 2; include_bias=False; Số đặc trưng sau mở rộng: `X_train_poly.shape[1]`; Regularization: Ridge (alpha = 1.0)

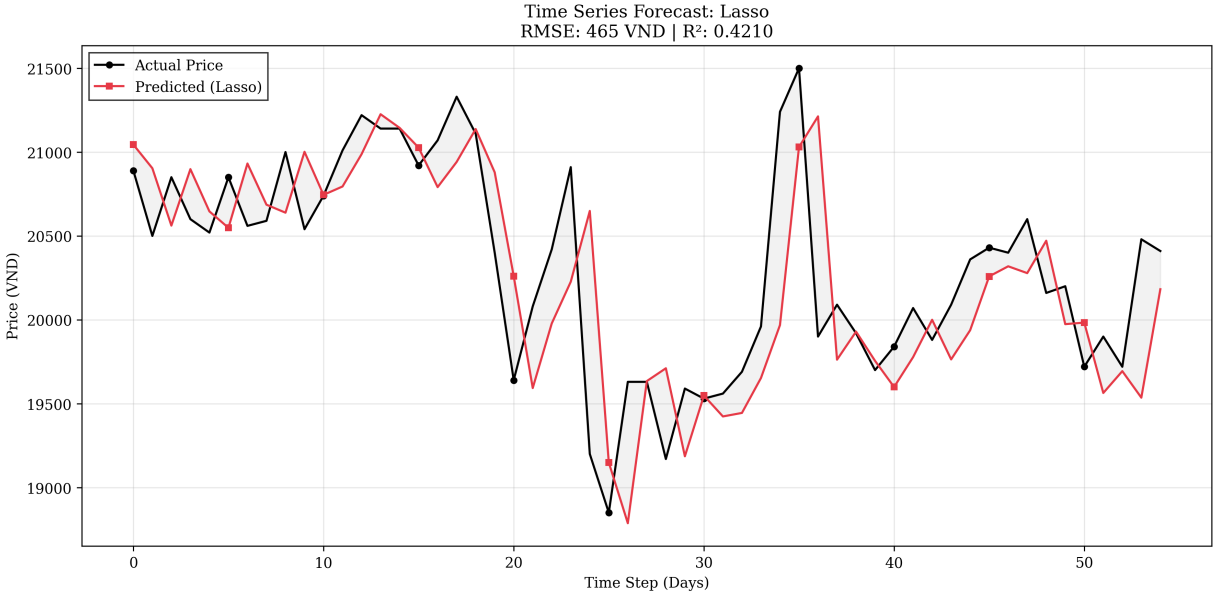

Figure 6: So sánh giá xăng thực tế và giá xăng dự đoán

Table 5: Đánh giá hiệu suất mô hình Linear Regression

Chỉ số đánh giá	**Giá trị**
MAE	340
MAPE	1.70%
RMSE	465
R^2	0.4210
DA	37.0%

Kết quả thực nghiệm cho thấy mô hình Linear Regression đạt mức sai số tương đối thấp với MAPE là 1.70% và RMSE là 465. Tuy nhiên, khả năng giải thích biến động dữ liệu còn hạn chế (R2=0.4210) và hiệu quả dự báo xu hướng chưa cao (DA=37.0%). Do đó, kết quả này được sử dụng làm ngưỡng cơ sở (*baseline*) để so sánh và làm nổi bật hiệu quả của các mô hình phi tuyến nâng cao trong phần tiếp theo.

4.4 Mô hình SVR (Support Vector Regression)

Mô hình **Support Vector Regression (SVR)** với hàm nhân *Radial Basis Function (RBF)* được áp dụng để dự đoán giá xăng tại Việt Nam. Phương pháp này có khả năng mô hình hóa quan hệ phi tuyến giữa các biến kinh tế, tuy nhiên kết quả trong thí nghiệm này cho thấy mô hình chưa đạt hiệu suất mong đợi.

Table 6: Các tham số cấu hình cho mô hình SVR

Tham số (Parameter)	**Giá trị / Thiết lập**
Kernel Functions	`rbf`, `linear`, `poly`
Degree (chỉ dùng cho `poly`)	3
C (Regularization)	1.0
Epsilon (ϵ)	0.1
Gamma	scale
Chuẩn hoá dữ liệu	StandardScaler
Hàm đánh giá	RMSE, MAE, R^2, Directional Accuracy
Tiêu chí chọn mô hình	Kernel có RMSE thấp nhất trên tập Validation

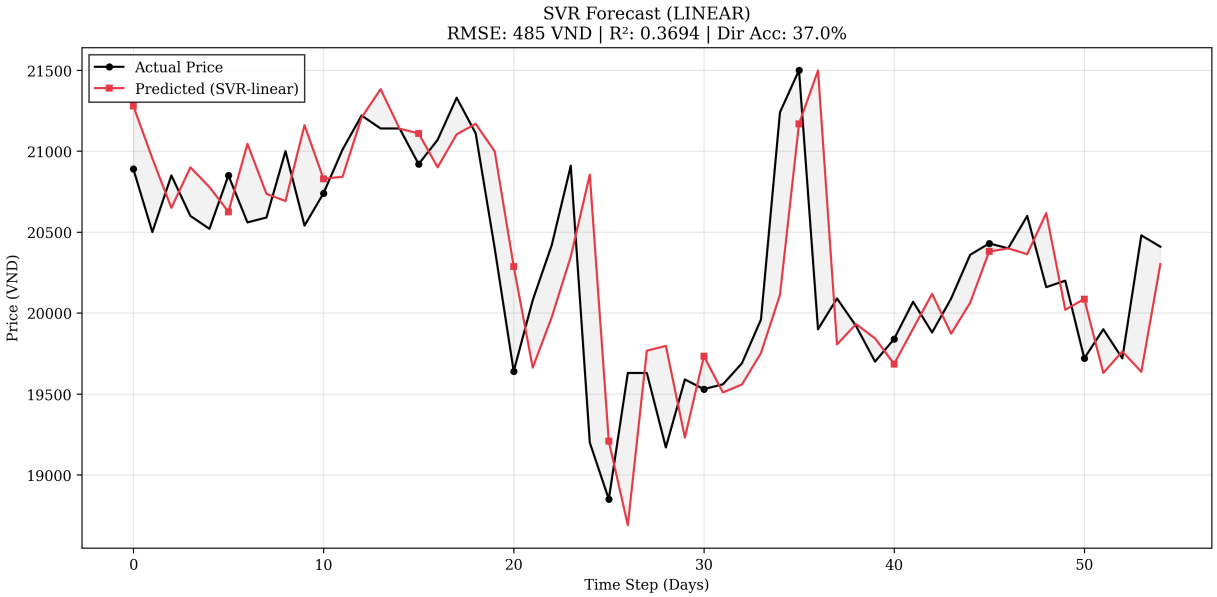

Figure 7: So sánh giá xăng thực tế và giá xăng dự đoán

Table 7: Đánh giá hiệu suất mô hình SVR

Chỉ số đánh giá	**Giá trị**
MAE	345.6
RMSE	484.9
R^2	0.3694
MAPE	1.71%
DA	37.04%

Mô hình SVR không mang lại sự cải thiện so với mô hình cơ sở, với các chỉ số sai số tăng nhẹ (MAE=345.6, RMSE=484.9) và mức độ giải thích biến thiên dữ liệu giảm xuống (R2=0.3694). Tương tự như Linear Regression,

khả năng dự báo xu hướng của SVR vẫn ở mức thấp (DA37%). Hạn chế này dẫn đến sự cần thiết phải thử nghiệm các mô hình Deep Learning chuyên biệt cho chuỗi thời gian như LSTM và GRU trong phần tiếp theo.

4.5 Mô hình ARIMAX

Mô hình **ARIMAX** được sử dụng để dự báo giá xăng Việt Nam theo chuỗi thời gian, với việc tự động lựa chọn các tham số tối ưu (p, d, q) nhằm giảm thiểu sai số dự đoán. Kết quả huấn luyện cho thấy mô hình đạt được bộ tham số tối ưu là:

$$(p, d, q) = (0, 1, 2)$$

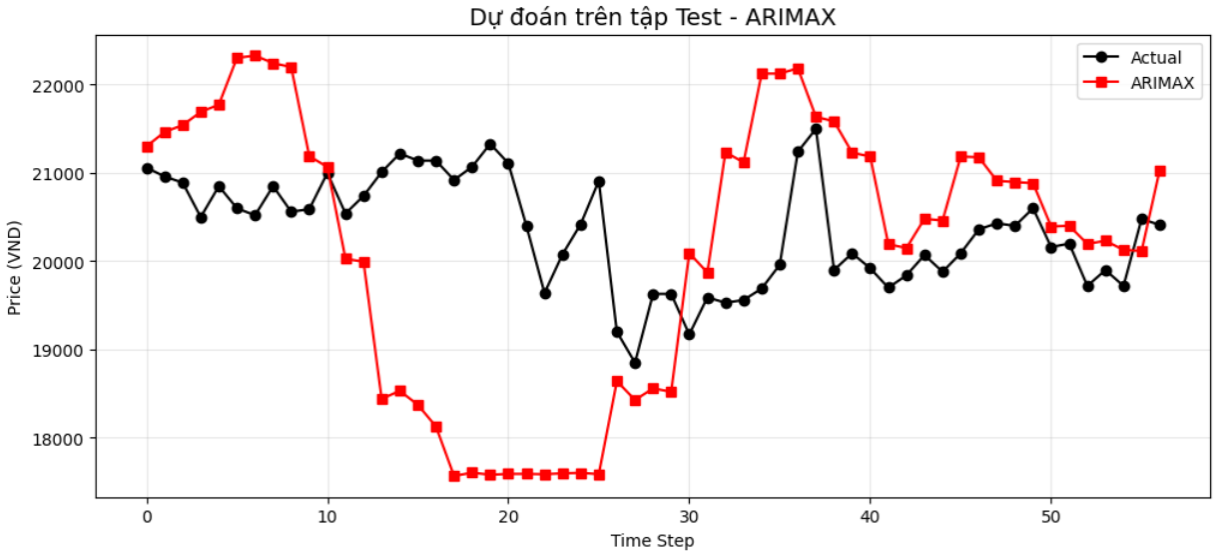

Figure 8: Giá xăng thực tế và giá xăng dự đoán

Table 8: Đánh giá hiệu suất mô hình Auto ARIMAX

Chỉ số đánh giá	**Giá trị**
MAE	1326.3
RMSE	1692.5
R^2	-6.616
MAPE	6.47%
DA	10.71%

Mô hình Auto ARIMAX cho thấy hiệu suất kém nhất trong các mô hình đã thử nghiệm. Sai số tăng vọt (MAPE=6.47%, RMSE=1692.5) đi kèm với hệ số R2 âm rất sâu (6.616), phản ánh việc mô hình hoàn toàn thất bại trong việc khớp dữ liệu. Đáng chú ý, độ chính xác hướng chỉ đạt 10.71%, thấp hơn mức ngẫu nhiên. Kết quả này khẳng định hạn chế của các phương pháp thống kê truyền thống đối với bộ dữ liệu này, củng cố tính cấp thiết của việc áp dụng các mô hình Deep Learning (LSTM, GRU) trong các bước tiếp theo.

4.6 Mô hình XGBOOST

4.6.1 Cấu hình mô hình và Tham số huấn luyện

Kiến trúc mô hình XGBOOST được thiết kế tối giản để phù hợp với quy mô dữ liệu. Chi tiết các tham số cấu hình được trình bày tại Bảng 9.

Table 9: Các tham số cấu hình cho mô hình XGBoost

Tham số (Parameter)	**Giá trị / Thiết lập**
Thuật toán (Algorithm)	Gradient Boosting (XGBoost)
Hàm mục tiêu (Objective)	reg:squarederror
Số lượng cây (Number of Trees)	200
Độ sâu tối đa (Max Depth)	4
Tốc độ học (Learning Rate / Eta)	0.05
Tỉ lệ mẫu con (Subsample)	0.8
Tỉ lệ đặc trưng (Colsample by Tree)	0.8
Trọng số con tối thiểu (Min Child Weight)	3
Gamma (Min Split Loss)	0.1
Regularization L1 (Alpha)	0.1
Regularization L2 (Lambda)	1.0
Cơ chế dừng sớm (Early Stopping)	Tự động theo validation
Random State (Seed)	42

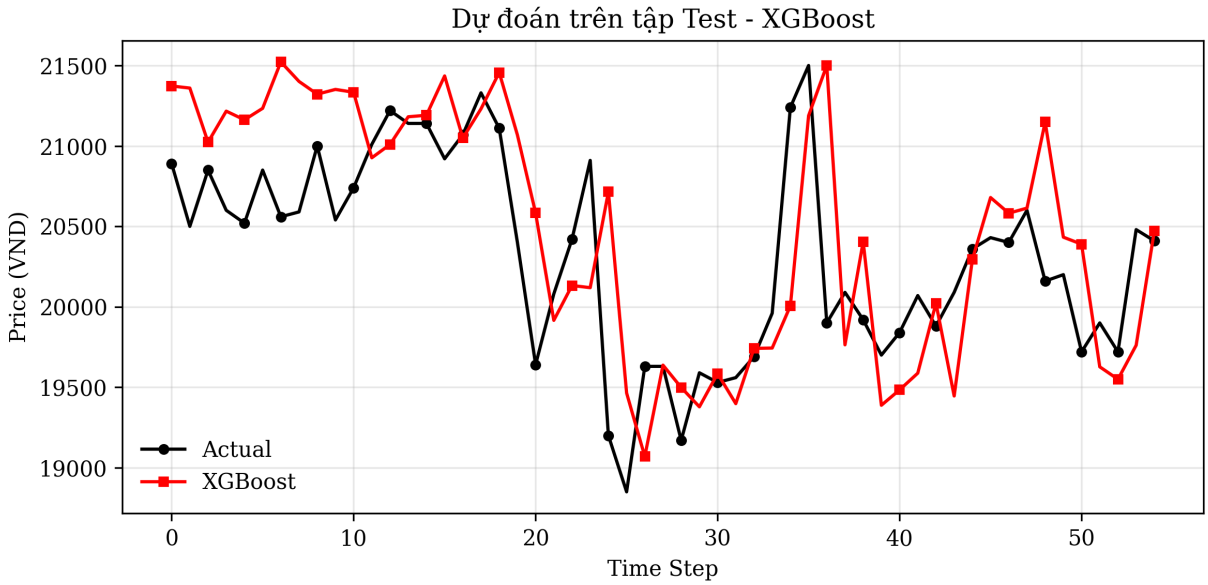

Figure 9: Giá xăng thực tế và giá xăng dự đoán

Table 10: Đánh giá hiệu suất mô hình XGBOOST

Chỉ số đánh giá	**Giá trị**
MAE	439
RMSE	572
R^2	0.1224
MAPE	2.17%
DA	44.4%

Kết quả tại Bảng 10 cho thấy mô hình hoạt động kém hiệu quả. Dù sai số lượng hóa thấp (MAPE=2.17%), nhưng hệ số R2 rất thấp (0.1224) và độ chính xác xu hướng (DA) chỉ đạt 44.4% (tệ hơn ngẫu nhiên). Điều này chứng tỏ mô hình chưa nắm bắt được quy luật biến động giá và không đủ tin cậy để dự báo.

4.7 Mô hình RNN (Recurrent Neural Network)

4.7.1 Cấu hình mô hình và Tham số huấn luyện

Kiến trúc mạng RNN được thiết kế tối giản để phù hợp với quy mô dữ liệu. Chi tiết các tham số cấu hình được trình bày tại Bảng 11.

Table 11: Các tham số cấu hình cho mô hình RNN

Tham số (Parameter)	**Giá trị / Thiết lập**
Độ dài chuỗi đầu vào (Window Size)	4 tuần ($T = 4$)
Số lượng đơn vị ẩn (LSTM Units)	16
Hàm kích hoạt (Activation)	Tanh
Lớp đầu ra (Output Layer)	Dense (1 unit)
Thuật toán tối ưu (Optimizer)	Adam
Tốc độ học (Learning Rate)	0.005
Hàm mất mát (Loss Function)	Mean Squared Error (MSE)
Kích thước lô (Batch Size)	8
Số epoch tối đa (Max Epochs)	150
Cơ chế dừng sớm (Early Stopping)	Patience = 30

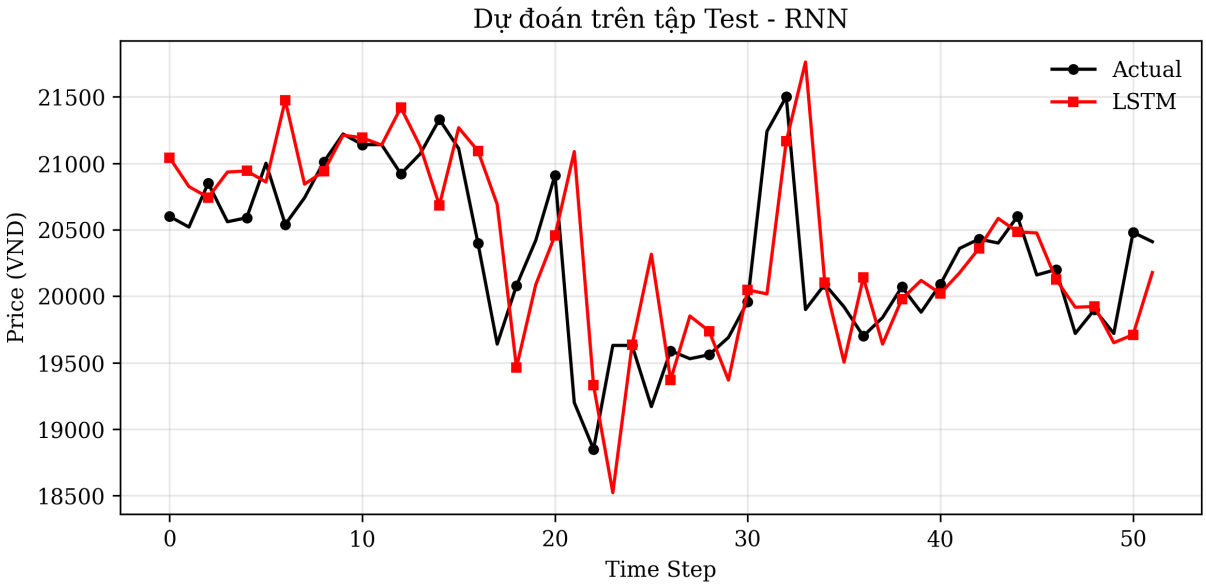

Figure 10: Giá xăng thực tế và giá xăng dự đoán

Table 12: Đánh giá hiệu suất mô hình RNN

Chỉ số đánh giá	**Giá trị**
MAE	389.80
RMSE	580.16
R^2	0.1174
MAPE	1.94%
DA	41.2%

Mô hình RNN đạt sai số lượng thấp (MAPE 1.94%) nhưng thất bại trong việc nắm bắt xu hướng ($R^2 = 0.1174$, DA 41.2%). Kết quả này phản ánh hiện tượng "trễ pha" (lagging), cho thấy mô hình thiên về sao chép dữ liệu quá khứ hơn là dự báo quy luật biến động thực tế.

4.8 Mô hình LSTM (Long Short-Term Memory)

4.8.1 Cấu hình mô hình LSTM

Mô hình **Long Short-Term Memory (LSTM)** được xây dựng nhằm khắc phục hạn chế về khả năng ghi nhớ dài hạn và vấn đề triệt tiêu đạo hàm (vanishing gradient) của RNN truyền thống. Chi tiết cấu hình được trình bày tại Bảng 13.

Table 13: Các tham số cấu hình cho mô hình LSTM

Tham số (Parameter)	**Giá trị / Thiết lập**
Độ dài chuỗi đầu vào (Window Size)	4 tuần ($T = 4$)
Số lượng đơn vị ẩn (LSTM Units)	16
Hàm kích hoạt (Activation)	Tanh
Lớp đầu ra (Output Layer)	Dense (1 unit)
Thuật toán tối ưu (Optimizer)	Adam
Tốc độ học (Learning Rate)	0.005
Hàm mất mát (Loss Function)	Mean Squared Error (MSE)
Kích thước lô (Batch Size)	8
Số epoch tối đa (Max Epochs)	150
Cơ chế dừng sớm (Early Stopping)	Patience = 30

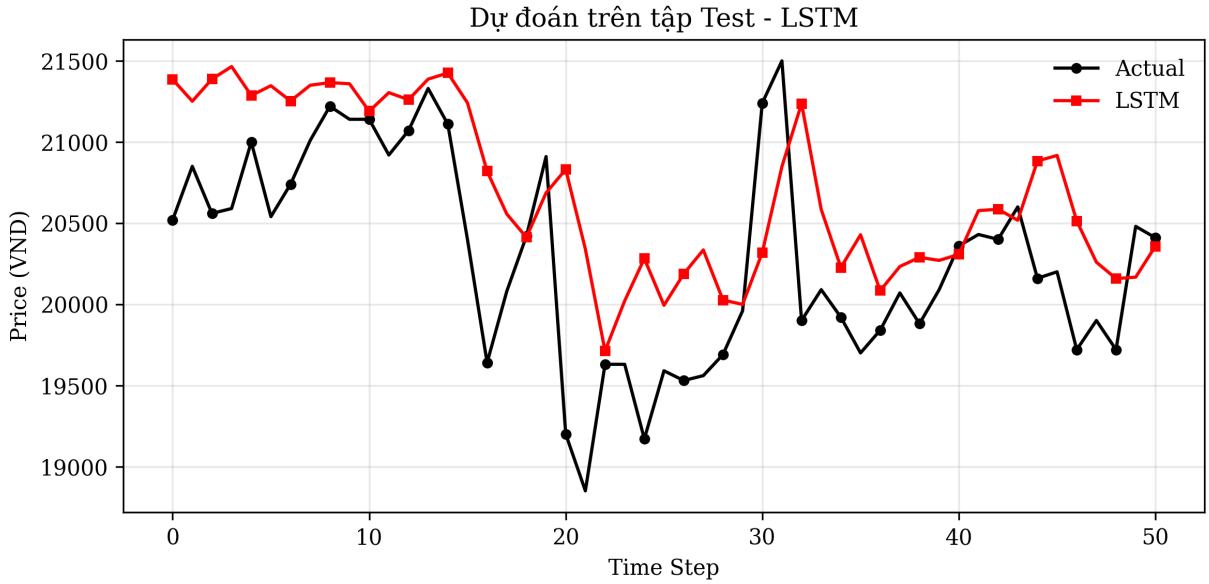

Figure 11: Giá xăng thực tế và giá xăng dự đoán

Table 14: Đánh giá hiệu suất mô hình LSTM

Chỉ số đánh giá	**Giá trị**
MAE	494.55
RMSE	628.72
R^2	-0.0215
MAPE	2.47%
DA	40.29%

Mô hình LSTM bám sát tốt biến động giá (MAPE 2.47%, $R^2 = -0.0215$) nhưng hạn chế trong việc dự báo xu hướng (DA 40.29%). Kết quả phản ánh mô hình không nắm bắt được quy luật dao động cơ bản và vẫn gặp hiện tượng trễ pha tại các điểm đảo chiều của thị trường.

4.9 Mô hình GRU (Gated Recurrent Unit)

Mô hình **Gated Recurrent Unit (GRU)** được thiết kế như một biến thể tinh gọn của LSTM, loại bỏ cổng đầu ra (output gate) giúp giảm số lượng tham số cần huấn luyện mà vẫn duy trì hiệu quả ghi nhớ chuỗi dữ liệu. Chi tiết cấu hình được trình bày tại Bảng 15.

Table 15: Các tham số cấu hình cho mô hình GRU

Tham số (Parameter)	**Giá trị / Thiết lập**
Độ dài chuỗi đầu vào (Window Size)	4 tuần ($T = 4$)
Số lượng đơn vị ẩn (GRU Units)	16
Hàm kích hoạt (Activation)	Tanh
Lớp đầu ra (Output Layer)	Dense (1 unit)
Thuật toán tối ưu (Optimizer)	Adam
Tốc độ học (Learning Rate)	0.004
Hàm mất mát (Loss Function)	Mean Squared Error (MSE)
Kích thước lô (Batch Size)	8
Số epoch tối đa (Max Epochs)	150
Cơ chế dừng sớm (Early Stopping)	Patience = 30

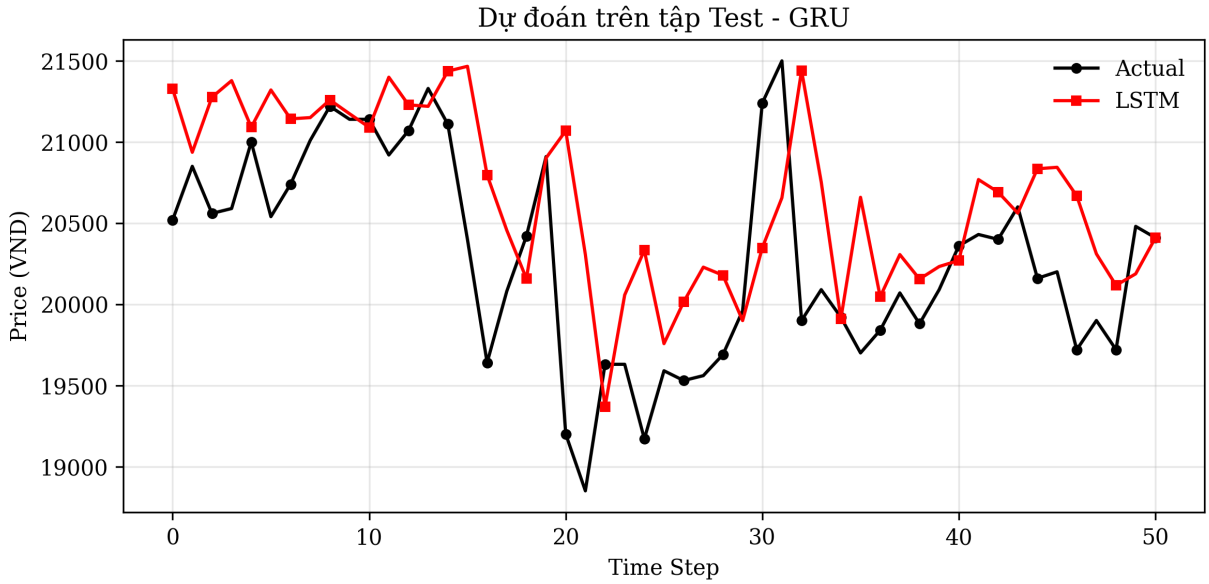

Figure 12: Giá xăng thực tế và giá xăng dự đoán

Table 16: Đánh giá hiệu suất mô hình GRU

Chỉ số đánh giá	**Giá trị**
MAE	486.69
RMSE	651.55
R^2	-0.0971
MAPE	2.43%
DA	42.00%

Kết quả tại Bảng 16 cho thấy mô hình GRU thất bại hoàn toàn. Với R2 âm (0.0971), mô hình hoạt động tệ hơn cả đường trung bình, kết hợp với DA chỉ 42% chứng tỏ mô hình không có khả năng dự báo giá trị lẫn xu hướng.

V. Kết quả và Thảo luận

5.1 Tổng hợp kết quả mô hình

Bảng 17 trình bày kết quả so sánh hiệu suất giữa các mô hình dự báo giá xăng Việt Nam. Trong đó, mô hình **Linear Regression** đạt hiệu suất có thể chấp nhận được với bài toán này.

Table 17: Tổng hợp so sánh hiệu suất giữa các mô hình

Mô hình	MAE	RMSE	R^2	MAPE (%)	DA (%)
Linear Regression	340.00	465.00	0.4210	1.70	37.00
SVR	345.60	484.90	0.3694	1.71	37.04
XGBOOST	439.00	572.00	0.1224	2.17	44.40
RNN	389.80	580.16	0.1174	1.94	41.20
LSTM	494.55	628.72	-0.0215	2.47	40.29
GRU	486.69	651.55	-0.0971	2.43	42.00
Auto ARIMAX	1,326.30	1,692.50	-6.6160	6.47	10.71

5.2 Dự đoán giá xăng tương lai

Nhóm sẽ sử dụng mô hình có kết quả khả quan nhất tức Linear Regression để dự đoán giá xăng 2 tuần tiếp theo.

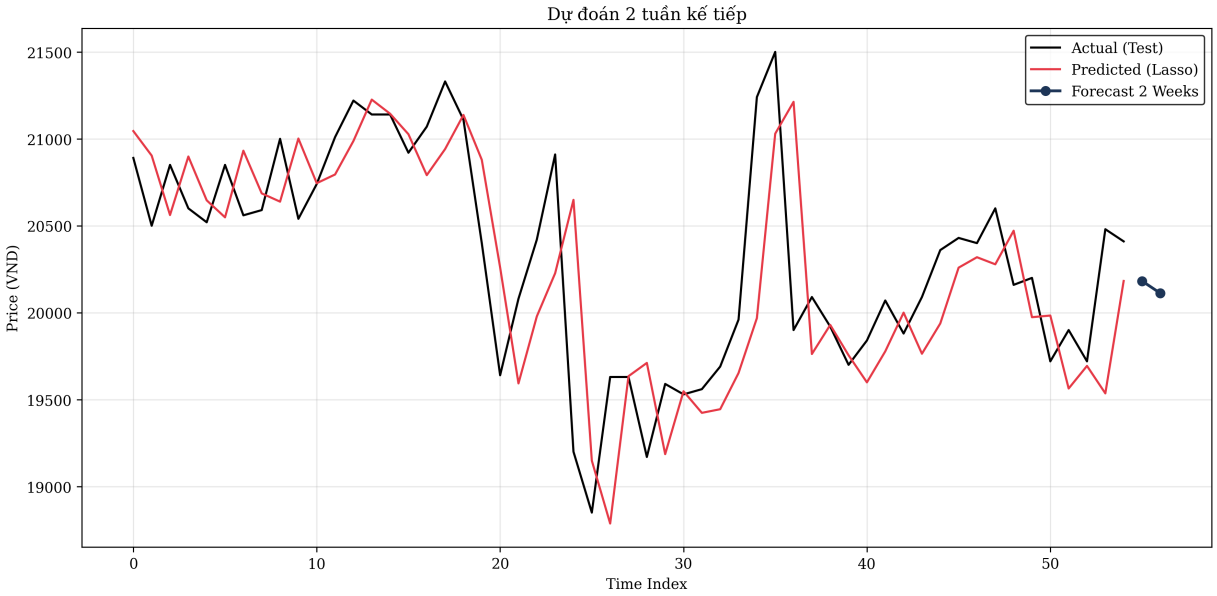

Figure 13: Dự đoán giá xăng 2 tuần kế tiếp

Table 18: Giá xăng dự báo cho 2 tuần kế tiếp (mô hình Linear Regression)

Tuần	Giá dự báo (VND/lít)
Tuần +1	20,182
Tuần +2	20,112

5.3 Đánh giá và thảo luận kết quả

Kết quả trong Bảng 17 cho thấy sự khác biệt rõ rệt giữa các mô hình, phản ánh đúng đặc trưng của dữ liệu giá xăng RON95 tại Việt Nam. Mô hình Linear Regression đạt hiệu suất tốt nhất nhờ giả định đơn giản, phù hợp với chuỗi

giá mang tính điều tiết và biến động dạng bậc thang. Quan hệ giữa giá xăng và các biến Brent, WTI, CPI và tỷ giá chủ yếu mang tính tuyến tính yếu, khiến mô hình tuyến tính hoạt động ổn định hơn các phương pháp phi tuyến.

Ngược lại, SVR và XGBoost không đạt kết quả như kỳ vọng. Dữ liệu ngắn, ít biến động và chịu ảnh hưởng lớn từ cơ chế điều hành khiến các mô hình này khó tìm được cấu trúc phi tuyến ý nghĩa. Mức tương quan thấp và sự gián đoạn trong chuỗi làm giảm khả năng phân tách của cây quyết định và bề mặt kernel, dẫn đến R^2 thấp và sai số cao hơn.

ARIMAX cho kết quả kém nhất khi không nắm bắt được các cú sốc chính sách và biến động gián đoạn. Giả định nhiễu ổn định và quan hệ tuyến tính theo thời gian của ARIMA hoàn toàn không phù hợp với chuỗi giá xăng nội địa vốn bị chi phối bởi quy định điều hành và độ trễ hành chính, dẫn tới hệ số R^2 âm sâu.

Các mô hình học sâu như RNN, LSTM và GRU cũng không đạt hiệu suất mong đợi. Chuỗi dữ liệu quá ngắn, thiếu tín hiệu dài hạn và có tính bậc thang khiến mô hình chỉ tái hiện mức giá gần nhất mà không học được xu hướng. Việc CPI được nội suy theo tháng và các biến ngoại sinh không biến động đồng bộ theo tuần càng làm suy giảm tín hiệu. Các mạng nơ-ron vì vậy không tận dụng được ưu thế ghi nhớ chuỗi và cho R^2 gần hoặc dưới 0.

Nhìn chung, hiệu suất thấp của các mô hình tiên tiến không đến từ thuật toán mà từ bản chất dữ liệu: ngắn, nhiễu chính sách, tín hiệu yếu và biến giải thích không đồng bộ. Trong bối cảnh đó, Linear Regression trở thành lựa chọn phù hợp nhất và phản ánh đúng mức độ phụ thuộc tương đối đơn giản giữa giá xăng và các biến vĩ mô.

VI. Kết luận và Insight

6.1 Insight rút ra từ kết quả phân tích

Kết quả nghiên cứu cho thấy giá xăng trong nước không phản ứng trực tiếp với biến động của Brent và WTI mà phụ thuộc chủ yếu vào cơ chế điều hành theo chu kỳ. Điều này làm yếu đi vai trò dự báo của các biến ngoại sinh. Tỷ giá USD/VND vẫn là yếu tố có ảnh hưởng rõ rệt nhất, nhưng mức độ biến động không đủ lớn để cải thiện đáng kể hiệu suất mô hình.

Dữ liệu theo tuần, có tính bậc thang và độ trễ chính sách, không thích hợp cho các mô hình học sâu vốn yêu cầu chuỗi dài và tín hiệu mạnh. Linear Regression do đó trở thành mô hình hiệu quả nhất trong điều kiện dữ liệu hiện tại.

6.2 Kết luận

Nghiên cứu cho thấy Linear Regression là mô hình phù hợp nhất để dự báo giá xăng RON95 trong bối cảnh dữ liệu hạn chế và tín hiệu thị trường yếu. Các mô hình phi tuyến và học sâu không mang lại cải thiện đáng kể do đặc trưng gián đoạn của chuỗi và sự chi phối của cơ chế điều hành giá. Trong tương lai, hiệu suất dự báo có thể được cải thiện nếu sử dụng dữ liệu tần suất cao hơn, mở rộng các biến chính sách và tăng độ dài chuỗi quan sát.

6.3 Hướng phát triển

Trong tương lai, nghiên cứu có thể tiếp tục được mở rộng theo những hướng linh hoạt và sâu hơn, đặc biệt tập trung vào việc cải thiện chất lượng dữ liệu và mở rộng bối cảnh kinh tế để mô hình phản ánh sát thực tế hơn. Việc khám phá thêm các dạng tín hiệu mới, các cấu trúc mô hình hiện đại cũng như những cách tiếp cận phù hợp hơn với đặc trưng gián đoạn của giá xăng có thể mang lại nhiều triển vọng. Ngoài ra, khả năng ứng dụng mô hình vào thực tiễn

dưới dạng hệ thống hỗ trợ quyết định hoặc công cụ giám sát biến động giá cũng là hướng đi tiềm năng, giúp biến kết quả nghiên cứu thành giá trị sử dụng thực tế. Những mở rộng này không chỉ góp phần khắc phục hạn chế của dữ liệu hiện tại mà còn tạo nền tảng để phát triển các phương pháp dự báo thích ứng tốt hơn với sự phức tạp của thị trường năng lượng Việt Nam.

VII. Tài liệu tham khảo

References

[1] B. C. T. V. Nam, "Báo cáo điều hành giá xăng dầu năm 2023," 2023.

[2] C. phủ Việt Nam, "Nghị định số 80/2023/nĐ-cp về cơ chế điều hành giá xăng dầu," 2023. Ban hành ngày 18/10/2023.

[3] A. Sagheer and M. Kotb, "Time series forecasting of petroleum production using lstm recurrent neural networks," *Energy*, vol. 201, pp. 117–128, 2020.

[4] A. Sagheer and M. Kotb, "Time series forecasting of petroleum production using deep lstm recurrent networks," *Neurocomputing*, vol. 323, pp. 203–213, 2019.

[5] X. J. He, "Forecasting gasoline price with time series models," *Communications of the IIMA*, vol. 21, no. 1, 2023.

[6] T. V. Nguyen, "The stable relationship between crude oil price and petrol price: Evidence from multivariate garch models," *Empirical Econometrics and Quantitative Economics Letters*, vol. 2, no. 2, pp. 27–40, 2013.

[7] J. Zhang, G. Hu, and L. Xiao, "Resampling strategies for irregular energy pricing data in short-term forecasting," *Energy Reports*, vol. 8, pp. 12765–12774, 2022.

[8] H. Wang, Z. Li, and J. Sun, "Hybrid deep learning model for crude oil price prediction," *Energy Economics*, vol. 129, 2023.

[9] M. Chen, R. Alvarez, and S. Gupta, "A step-wise temporal disaggregation method for transforming low-frequency economic indicators into weekly inputs for forecasting models," *Expert Systems with Applications*, vol. 233, 2024.

[10] A. Pirpanahi, M. Rahman, and K. Lee, "Temporal alignment of asynchronous economic indicators for energy price prediction," *Applied Energy*, vol. 350, 2024.

[11] T. Nguyen, J. Smith, and R. Patel, "Feature engineering in time series forecasting," *Preprint*, 2025. Available at ResearchGate.

[12] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*. OTexts, 2018.

[13] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*. Springer, 2013.